

OpenTP1 Version 7
Operation

3000-3-D53-30(E)

■ Relevant program products

Note: In the program products listed below, those marked with an asterisk (*) might be released later than the other program products.

For AIX 5L V5.1, AIX 5L V5.2, AIX 5L V5.3, and AIX V6.1

P-1M64-2131 uCosminexus TP1/Server Base 07-03*
P-1M64-2331 uCosminexus TP1/FS/Direct Access 07-03*
P-1M64-2431 uCosminexus TP1/FS/Table Access 07-03*
P-1M64-2531 uCosminexus TP1/Client/W 07-02
P-1M64-2631 uCosminexus TP1/Offline Tester 07-00
P-1M64-2731 uCosminexus TP1/Online Tester 07-00
P-1M64-2831 uCosminexus TP1/Multi 07-00
P-1M64-2931 uCosminexus TP1/High Availability 07-00
P-1M64-3131 uCosminexus TP1/Message Control 07-03
P-1M64-3231 uCosminexus TP1/NET/Library 07-04
P-1M64-8131 uCosminexus TP1/Shared Table Access 07-00
P-1M64-8331 uCosminexus TP1/Resource Manager Monitor 07-00
P-1M64-8531 uCosminexus TP1/Extension 1 07-00
P-1M64-C371 uCosminexus TP1/Message Queue 07-01
P-1M64-C771 uCosminexus TP1/Message Queue - Access 07-01
P-F1M64-31311 uCosminexus TP1/Message Control/Tester 07-00
P-F1M64-32311 uCosminexus TP1/NET/User Agent 07-00
P-F1M64-32312 uCosminexus TP1/NET/HDLC 07-00
P-F1M64-32313 uCosminexus TP1/NET/X25 07-00
P-F1M64-32314 uCosminexus TP1/NET/OSI-TP 07-00
P-F1M64-32315 uCosminexus TP1/NET/XMAP3 07-01
P-F1M64-32316 uCosminexus TP1/NET/HSC 07-00
P-F1M64-32317 uCosminexus TP1/NET/NCSB 07-00
P-F1M64-32318 uCosminexus TP1/NET/OSAS-NIF 07-01
P-F1M64-3231B uCosminexus TP1/NET/Secondary Logical Unit - TypeP2 07-00
P-F1M64-3231C uCosminexus TP1/NET/TCP/IP 07-02
P-F1M64-3231D uCosminexus TP1/NET/High Availability 07-00
P-F1M64-3231U uCosminexus TP1/NET/User Datagram Protocol 07-00
R-1M45F-31 uCosminexus TP1/Web 07-00

For AIX 5L V5.3 and AIX V6.1

P-1M64-1111 uCosminexus TP1/Server Base(64) 07-03*
P-1M64-1311 uCosminexus TP1/FS/Direct Access(64) 07-03*
P-1M64-1411 uCosminexus TP1/FS/Table Access(64) 07-03*
P-1M64-1911 uCosminexus TP1/High Availability(64) 07-00
P-1M64-1L11 uCosminexus TP1/Extension 1(64) 07-00
For HP-UX 11i V1 (PA-RISC) and HP-UX 11i V2 (PA-RISC)
P-1B64-3F31 uCosminexus TP1/NET/High Availability 07-00
P-1B64-8531 uCosminexus TP1/Extension 1 07-00
P-1B64-8931 uCosminexus TP1/High Availability 07-00
R-18451-41K uCosminexus TP1/Client/W 07-00
R-18452-41K uCosminexus TP1/Server Base 07-00

R-18453-41K uCosminexus TP1/FS/Direct Access 07-00
R-18454-41K uCosminexus TP1/FS/Table Access 07-00
R-18455-41K uCosminexus TP1/Message Control 07-03*
R-18456-41K uCosminexus TP1/NET/Library 07-04*
R-18459-41K uCosminexus TP1/Offline Tester 07-00
R-1845A-41K uCosminexus TP1/Online Tester 07-00
R-1845C-41K uCosminexus TP1/Shared Table Access 07-00
R-1845D-41K uCosminexus TP1/Resource Manager Monitor 07-00
R-1845E-41K uCosminexus TP1/Multi 07-00
R-1845F-41K uCosminexus TP1/Web 07-00
R-F18455-411K uCosminexus TP1/Message Control/Tester 07-00
R-F18456-411K uCosminexus TP1/NET/User Agent 07-00
R-F18456-415K uCosminexus TP1/NET/XMAP3 07-01*
R-F18456-41CK uCosminexus TP1/NET/TCP/IP 07-02*
For HP-UX 11i V2 (IPF) and HP-UX 11i V3 (IPF)
P-1J64-3F21 uCosminexus TP1/NET/High Availability 07-00
P-1J64-4F11 uCosminexus TP1/NET/High Availability(64) 07-00
P-1J64-8521 uCosminexus TP1/Extension 1 07-00
P-1J64-8611 uCosminexus TP1/Extension 1(64) 07-00
P-1J64-8921 uCosminexus TP1/High Availability 07-00
P-1J64-8A11 uCosminexus TP1/High Availability(64) 07-00
P-1J64-C371 uCosminexus TP1/Message Queue 07-01
P-1J64-C571 uCosminexus TP1/Message Queue(64) 07-01
P-1J64-C871 uCosminexus TP1/Message Queue - Access(64) 07-00
R-18451-21J uCosminexus TP1/Client/W 07-02
R-18452-21J uCosminexus TP1/Server Base 07-03*
R-18453-21J uCosminexus TP1/FS/Direct Access 07-03*
R-18454-21J uCosminexus TP1/FS/Table Access 07-03*
R-18455-21J uCosminexus TP1/Message Control 07-03*
R-18456-21J uCosminexus TP1/NET/Library 07-04*
R-18459-21J uCosminexus TP1/Offline Tester 07-00
R-1845A-21J uCosminexus TP1/Online Tester 07-00
R-1845C-21J uCosminexus TP1/Shared Table Access 07-00
R-1845D-21J uCosminexus TP1/Resource Manager Monitor 07-00
R-1845E-21J uCosminexus TP1/Multi 07-00
R-1845F-21J uCosminexus TP1/Web 07-00
R-1B451-11J uCosminexus TP1/Client/W(64) 07-02
R-1B452-11J uCosminexus TP1/Server Base(64) 07-03*
R-1B453-11J uCosminexus TP1/FS/Direct Access(64) 07-03*
R-1B454-11J uCosminexus TP1/FS/Table Access(64) 07-03*
R-1B455-11J uCosminexus TP1/Message Control(64) 07-03*
R-1B456-11J uCosminexus TP1/NET/Library(64) 07-04*
R-F18455-211J uCosminexus TP1/Message Control/Tester 07-00
R-F18456-215J uCosminexus TP1/NET/XMAP3 07-01*

R-F18456-21CJ uCosminexus TP1/NET/TCP/IP 07-02*
R-F1B456-11CJ uCosminexus TP1/NET/TCP/IP(64) 07-02*
For Solaris 8, Solaris 9, and Solaris 10
P-9D64-3F31 uCosminexus TP1/NET/High Availability 07-00
P-9D64-8531 uCosminexus TP1/Extension 1 07-00
P-9D64-8931 uCosminexus TP1/High Availability 07-00
R-19451-216 uCosminexus TP1/Client/W 07-00
R-19452-216 uCosminexus TP1/Server Base 07-00
R-19453-216 uCosminexus TP1/FS/Direct Access 07-00
R-19454-216 uCosminexus TP1/FS/Table Access 07-00
R-19455-216 uCosminexus TP1/Message Control 07-03*
R-19456-216 uCosminexus TP1/NET/Library 07-04*
R-19459-216 uCosminexus TP1/Offline Tester 07-00
R-1945A-216 uCosminexus TP1/Online Tester 07-00
R-1945C-216 uCosminexus TP1/Shared Table Access 07-00
R-1945D-216 uCosminexus TP1/Resource Manager Monitor 07-00
R-1945E-216 uCosminexus TP1/Multi 07-00
R-F19456-2156 uCosminexus TP1/NET/XMAP3 07-01*
R-F19456-21C6 uCosminexus TP1/NET/TCP/IP 07-02*
For Red Hat Enterprise Linux AS 4 (AMD64 & Intel EM64T), Red Hat Enterprise Linux AS 4 (x86), Red Hat Enterprise Linux ES 4 (AMD64 & Intel EM64T), and Red Hat Enterprise Linux ES 4 (x86)
P-9S64-2161 uCosminexus TP1/Server Base 07-00
P-9S64-2351 uCosminexus TP1/FS/Direct Access 07-00
P-9S64-2451 uCosminexus TP1/FS/Table Access 07-00
P-9S64-2551 uCosminexus TP1/Client/W 07-00
P-9S64-3151 uCosminexus TP1/Message Control 07-00
P-9S64-3251 uCosminexus TP1/NET/Library 07-00
P-9S64-C371 uCosminexus TP1/Message Queue 07-01
P-F9S64-3251C uCosminexus TP1/NET/TCP/IP 07-00
P-F9S64-3251U uCosminexus TP1/NET/User Datagram Protocol 07-00
R-1845F-A15 uCosminexus TP1/Web 07-00
For Red Hat Enterprise Linux AS 4 (AMD64 & Intel EM64T), Red Hat Enterprise Linux AS 4 (x86), Red Hat Enterprise Linux ES 4 (AMD64 & Intel EM64T), Red Hat Enterprise Linux ES 4 (x86), Red Hat Enterprise Linux 5 (AMD/Intel 64), Red Hat Enterprise Linux 5 (x86), Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64), and Red Hat Enterprise Linux 5 Advanced Platform (x86)
P-9S64-2951 uCosminexus TP1/High Availability 07-00
P-9S64-8551 uCosminexus TP1/Extension 1 07-00
P-9S64-C771 uCosminexus TP1/Message Queue - Access 07-01
P-F9S64-3251D uCosminexus TP1/NET/High Availability 07-00
For Red Hat Enterprise Linux 5 (AMD/Intel 64), Red Hat Enterprise Linux 5 (x86), Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64), and Red Hat Enterprise Linux 5 Advanced Platform (x86)
P-9S64-2171 uCosminexus TP1/Server Base 07-03
P-9S64-2361 uCosminexus TP1/FS/Direct Access 07-03
P-9S64-2461 uCosminexus TP1/FS/Table Access 07-03
P-9S64-2561 uCosminexus TP1/Client/W 07-02
P-9S64-3161 uCosminexus TP1/Message Control 07-03*

P-9S64-3261 uCosminexus TP1/NET/Library 07-04*

P-9S64-C571 uCosminexus TP1/Message Queue 07-01

P-F9S64-32611 uCosminexus TP1/NET/User Agent 07-00

P-F9S64-3261C uCosminexus TP1/NET/TCP/IP 07-02

P-F9S64-3261U uCosminexus TP1/NET/User Datagram Protocol 07-00

For Red Hat Enterprise Linux 5 (AMD/Intel 64) and Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64)

P-9W64-2111 uCosminexus TP1/Server Base(64) 07-03

P-9W64-2311 uCosminexus TP1/FS/Direct Access(64) 07-03

P-9W64-2411 uCosminexus TP1/FS/Table Access(64) 07-03

P-9W64-2911 uCosminexus TP1/High Availability(64) 07-02

P-9W64-8511 uCosminexus TP1/Extension 1(64) 07-02

For Red Hat Enterprise Linux AS 4 (IPF)

P-9V64-2121 uCosminexus TP1/Server Base 07-00

P-9V64-2321 uCosminexus TP1/FS/Direct Access 07-00

P-9V64-2421 uCosminexus TP1/FS/Table Access 07-00

P-9V64-2521 uCosminexus TP1/Client/W 07-00

P-9V64-3121 uCosminexus TP1/Message Control 07-00

P-9V64-3221 uCosminexus TP1/NET/Library 07-00

P-9V64-C371 uCosminexus TP1/Message Queue(64) 07-01

P-9V64-C771 uCosminexus TP1/Message Queue - Access(64) 07-00

P-F9V64-3221C uCosminexus TP1/NET/TCP/IP 07-00

P-F9V64-3221U uCosminexus TP1/NET/User Datagram Protocol 07-00

For Red Hat Enterprise Linux AS 4 (IPF), Red Hat Enterprise Linux 5 (Intel Itanium), and Red Hat Enterprise Linux 5 Advanced Platform (Intel Itanium)

P-9V64-2921 uCosminexus TP1/High Availability 07-00

P-9V64-8521 uCosminexus TP1/Extension 1 07-00

P-F9V64-3221D uCosminexus TP1/NET/High Availability 07-00

For Red Hat Enterprise Linux 5 (Intel Itanium) and Red Hat Enterprise Linux 5 Advanced Platform (Intel Itanium)

P-9V64-2131 uCosminexus TP1/Server Base 07-02

P-9V64-2331 uCosminexus TP1/FS/Direct Access 07-02

P-9V64-2431 uCosminexus TP1/FS/Table Access 07-02

P-9V64-2531 uCosminexus TP1/Client/W 07-02

P-9V64-3131 uCosminexus TP1/Message Control 07-03*

P-9V64-3231 uCosminexus TP1/NET/Library 07-04*

P-F9V64-3231C uCosminexus TP1/NET/TCP/IP 07-02*

P-F9V64-3231U uCosminexus TP1/NET/User Datagram Protocol 07-00

For Windows 2000, Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, Windows XP, Windows Vista, and Windows Vista x64

P-2464-2144 uCosminexus TP1/Client/P 07-02

For Windows 2000, Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, and Windows XP

R-1845F-8134 uCosminexus TP1/Web 07-00

For Windows 2000, Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, Windows XP, Windows Vista, Windows Vista x64, Windows Server 2008, and Windows Server 2008 x64

P-2464-7824 uCosminexus TP1/Client for .NET Framework 07-03

R-15451-21 uCosminexus TP1/Connector for .NET Framework 07-03

For Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, Windows XP, Windows Vista, Windows Vista x64, Windows Server 2008, and Windows Server 2008 x64

P-2464-2274 uCosminexus TP1/Server Base 07-03*

P-2464-2374 uCosminexus TP1/FS/Direct Access 07-03*

P-2464-2474 uCosminexus TP1/FS/Table Access 07-03*

P-2464-2544 uCosminexus TP1/Extension 1 07-00

P-2464-3154 uCosminexus TP1/Message Control 07-03*

P-2464-3254 uCosminexus TP1/NET/Library 07-04*

P-2464-3354 uCosminexus TP1/Messaging 07-00

P-2464-C374 uCosminexus TP1/Message Queue 07-01

P-2464-C774 uCosminexus TP1/Message Queue - Access 07-00

P-F2464-3254C uCosminexus TP1/NET/TCP/IP 07-02*

R-15452-21 uCosminexus TP1/Extension for .NET Framework 07-00

R-1945B-24 uCosminexus TP1/LiNK 07-02

For Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, and Windows XP

P-F2464-32545 uCosminexus TP1/NET/XMAP3 07-01*

For Windows Server 2003, Windows Server 2003 x64 Editions, Windows Server 2003 R2, Windows Server 2003 R2 x64 Editions, Windows Server 2008, and Windows Server 2008 x64

P-2464-2934 uCosminexus TP1/High Availability 07-00

P-F2464-3254D uCosminexus TP1/NET/High Availability 07-00

For Java VM

P-2464-7394 uCosminexus TP1/Client/J 07-02

P-2464-73A4 uCosminexus TP1/Client/J 07-02

This manual can be used for products other than the products shown above. For details, see the *Release Notes*.

These products were developed under a quality management system that has received ISO9001 and TickIT certification.

■ Trademarks

AIX is a trademark of International Business Machines Corporation in the United States, other countries, or both.

AIX 5L is a trademark of International Business Machines Corporation in the United States, other countries, or both.

BSD is a trademark of University of California, Berkeley in the U.S.A.

HP-UX is a product name of Hewlett-Packard Company.

Itanium is a trademark of Intel Corporation in the United States and other countries.

Java is a registered trademark of Oracle and/or its affiliates.

Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

MS-DOS is a registered trademark of Microsoft Corp. in the U.S. and other countries.

ORACLE is either a registered trademark or a trademark of Oracle and/or its affiliates.

Oracle is either a registered trademark or a trademark of Oracle Corporation and/or its affiliates.

Oracle and Oracle 10g are either registered trademarks or trademarks of Oracle and/or its affiliates.

Oracle and Oracle9i are either registered trademarks or trademarks of Oracle and/or its affiliates.

OSF is a trademark of the Open Software Foundation, Inc.

Red Hat is a trademark or a registered trademark of Red Hat Inc. in the United States and other countries.

Solaris is either a registered trademark or a trademark of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

Windows Server is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

Windows Vista is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

X/Open is a registered trademark of The Open Group in the U.K. and other countries.

Portions of this document are extracted from *X/Open CAE Specification System Interfaces and Headers, Issue 4*, (C202 ISBN 1 872630 47 2) Copyright (C) July 1992, X/Open Company Limited with the permission of X/Open; part of which is based on *IEEE Std 1003.1-1990*, (C) 1990 Institute of Electrical and Electronics Engineers, Inc., and *IEEE Std 1003.2/D12*, (C) 1992 Institute of Electrical and Electronics Engineers, Inc.

No further reproduction of this material is permitted without the prior permission of the copyright owners.

Portions of this document are copyrighted by Open Software Foundation, Inc.

This document and the software described herein are furnished under a license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the above copyright notice. Title to and ownership of the document and software remain with OSF or its licensors.

Other product and company names mentioned in this document may be the trademarks of their respective owners. Throughout this document Hitachi has attempted to distinguish trademarks from descriptive terms by writing the name with the capitalization used by the manufacturer, or by writing the name with initial capital letters. Hitachi cannot attest to the accuracy of this information. Use of a trademark in this document should not be regarded as affecting the validity of the trademark.

■ Restrictions

Information in this document is subject to change without notice and does not represent a commitment on the part of Hitachi. The software described in this manual is furnished according to a license agreement with Hitachi. The license agreement contains all of the terms and conditions governing your use of the software and documentation, including all warranty rights, limitations of liability, and disclaimers of warranty.

Material contained in this document may describe Hitachi products not available or features not available in your country.

No part of this material may be reproduced in any form or by any means without permission in writing from the publisher.

Printed in Japan.

■ Edition history

Edition 1 (3000-3-D53(E)): June 2006

Edition 3 (3000-3-D53-30(E)): October 2010

■ Copyright

All Rights Reserved. Copyright (C) 2006, 2010, Hitachi, Ltd.

Summary of amendments

The following table lists changes in this manual (3000-3-D53-30(E)) and product changes related to this manual for uCosminexus TP1/Server Base 07-03, uCosminexus TP1/Server Base(64) 07-03, uCosminexus TP1/Message Control 07-03, uCosminexus TP1/Message Control(64) 07-03, uCosminexus TP1/NET/Library 07-04, and uCosminexus TP1/NET/Library(64) 07-04.

Changes	Location
<p>Event trace information (FIL event traces) can now be output in OpenTP1 file access requests. With this change, a FIL event trace information file has been added. The argument <code>_f1</code> has also been added for the <code>-f</code> option of the <code>prfget</code> command.</p>	<p><i>1.3.6(3)(a), 3.8.5(1), 7.1.12(3), Table 10-2 in section 10.8, prfget in 13. Details of Operation Commands, Table L-1 and Table L-2 in subsection L(1)</i></p>
<p>The journal service is now able to output performance verification trace information (JNL performance verification traces). With this change, the JNL performance verification trace information file has been added. The argument <code>_j1</code> has also been added for the <code>-f</code> option of the <code>prfget</code> command.</p>	<p><i>1.3.6(3)(a), 3.8.5(1), 7.1.12(3), Table 10-2 in section 10.8, prfget in 13. Details of Operation Commands, Table L-1 and Table L-2 in subsection L(1)</i></p>
<p>Performance verification trace information for various lock events used by the lock service (LCK performance verification traces) can now be output. With this change, the LCK performance verification trace information file can now be output when the <code>-d</code> option of the <code>prfed</code> command is specified. The argument <code>_l1</code> has also been added for the <code>-f</code> option of the <code>prfget</code> command.</p>	<p><i>1.3.6(3)(a), 3.8.5(1), 7.1.12(3), Table 10-2 in section 10.8, prfed and prfget in 13. Details of Operation Commands, Table L-1 and Table L-2 in subsection L(1)</i></p>
<p>UAP trace information (UAP trace data file) can now be acquired without having to abort the process. With this change, the following files can now be acquired:</p> <ul style="list-style-type: none"> • UAP trace data file • Backup file for UAP trace data file 	<p><i>1.3.6(3)(a), 7.1.12(3)</i></p>
<p>OpenTP1 service startup and OpenTP1 service termination have been added to the audit events.</p>	<p><i>Table 3-13 in subsection 3.7.5(1), 3.7.5(2)(n), 3.7.5(2)(o), Table C-1 in Appendix C</i></p>
<p>A function for specifying whether to use service information of specific nodes in a prioritized fashion has been added (service information prioritizing function). With this change, an explanation has been added about the definition file for priority selection nodes. The explanation about the output format for the following command has also been changed:</p> <ul style="list-style-type: none"> • <code>namsvinf</code> 	<p><i>3.10.1(2) namsvinf in 13. Details of Operation Commands</i></p>

Changes	Location
A note has been added about use of the remote API facility.	3.12.7
A statement has been added explaining how to stop the realtime statistics service when an extension of the realtime statistics service (an RTSSPP) is running.	3.16.2(4)
Used and unused (free) areas can now be listed as user area information for the OpenTP1 file system. With this change, the following option has been added: <ul style="list-style-type: none"> • -S option of the <code>filstatfs</code> command. 	4.1.2, 4.1.9 <i>filstatfs</i> in 13. <i>Details of Operation Commands</i>
The following command has been added as one of the facilities that cannot be used in the journal fileless mode: <ul style="list-style-type: none"> • <code>jnlmkrf</code> 	Table 4-14 in subsection 4.3.19
An explanation about displaying the status of the MCF communication service has been added.	5.1
The status of application timer start requests can now be displayed. With this change, the following command has been added: <ul style="list-style-type: none"> • <code>mcfalstap</code> 	5.3(4), Table 12-1 in subsection 12.1.4 <i>mcfalstap</i> in 13. <i>Details of Operation Commands</i>
An explanation about the cause of a fallback operation along with its recovery procedure have been added.	10.2.8(3)
The status of user timer monitoring can now be displayed. With this change, the following command has been added: <ul style="list-style-type: none"> • <code>mcfslsum</code> 	Table 12-1 in subsection 12.1.4 <i>mcfslsum</i> in 13. <i>Details of Operation Commands</i>
The <code>prctee</code> process, used to redirect standard output and standard error output from OpenTP1, can now be stopped and restarted. With this change, the following command has been added: <ul style="list-style-type: none"> • <code>prctctrl</code> 	Table 12-1 in subsection 12.1.4 <i>prctctrl</i> in 13. <i>Details of Operation Commands</i>
The conditions under which the following commands are executed have been changed: <ul style="list-style-type: none"> • <code>damrstr</code> • <code>tamrstr</code> 	Table 12-1 in subsection 12.1.4
A note has been added about OpenTP1 behavior when <code>AUTO</code> is specified in the <code>mode_conf</code> operand of the system environment definition.	<i>dcreset</i> in 13. <i>Details of Operation Commands</i>
An explanation has been added about system behavior when the <code>-d</code> option is specified in the <code>dcsetup</code> command but neither the <code>-y</code> option nor the <code>-n</code> option is specified.	<i>dcsetup</i> in 13. <i>Details of Operation Commands</i>
A formula has been added for calculating the size of the shared memory pool, obtained in the <code>dcshmls</code> command execution results, that the operating system is requested to allocate.	<i>dcshmls</i> in 13. <i>Details of Operation Commands</i>

Changes	Location
A note has been added about the time displayed when the user clock is reset backward by the operating system's time synchronization function.	<i>jnlstts</i> in 13. <i>Details of Operation Commands</i>
Formulas have been added for estimating memory requirements for commands.	<i>jnlstts</i> , <i>tambkup</i> , <i>tamcre</i> , and <i>tamrstr</i> in 13. <i>Details of Operation Commands</i>
The limitation on the number of characters that can be used to specify the names of output files has been removed.	<i>jnlunlfg</i> in 13. <i>Details of Operation Commands</i>
The number of unsent messages that can be displayed in the output of the <code>mcftlsle</code> command has been changed.	<i>mcftlsle</i> in 13. <i>Details of Operation Commands</i>
The edited results of trace information files can now be output in CSV format. With this change, the following option has been added: <ul style="list-style-type: none"> -v option of the <code>prfed</code> command 	<i>prfed</i> in 13. <i>Details of Operation Commands</i>
Explanations have been added about the physical file statuses acquired in the execution results of the <code>stsls</code> command.	<i>stsls</i> in 13. <i>Details of Operation Commands</i>
Explanations been added about the information output for audit events (errors in accessing the OpenTP1 file system).	<i>Table C-1</i> in <i>Appendix C</i>
The formulas for estimating the following resources used by the UNIX message exchange functions have been changed: <ul style="list-style-type: none"> Number of message IDs Maximum number of queued message bytes per message ID Maximum number of queued OpenTP1 messages 	<i>Appendix J</i>
The trace data sizes of the following event IDs have been changed: <ul style="list-style-type: none"> From <code>0xf000</code> to <code>0xf104</code> From <code>0xf106</code> to <code>0xf204</code> From <code>0xf206</code> to <code>0xf218</code> 	<i>Table L-1</i> in subsection <i>L(1)</i>

The following table lists changes in this manual (3000-3-D53-30(E)) and product changes related to this manual for uCosminexus TP1/Message Control 07-02 and uCosminexus TP1/NET/Library 07-03

Changes	Location
The state of processing during queue monitoring and while the system is waiting to receive response messages from a remote system can now be tracked by means of log messages that are output during queue monitoring and when processing finishes.	<i>2.2.1(1)</i> , <i>2.2.1(3)</i>
MHPs can now use the facility for dynamic loading of service functions.	<i>Table 3-9</i> in subsection <i>3.2.8</i>

Changes	Location
<p>The following operations can now be performed using library functions:</p> <ul style="list-style-type: none"> • Display of connection status, and connection establishment and release. • Display of the status of server-type connection establishment requests, and start and stop of acceptance of server-type connection establishment requests. • Deletion of application timer start requests. • Display of the status of logical terminals, logical terminal shutdown, release of logical terminal shutdown status, and deletion of logical terminal output queues. • Acquisition of the status of the MCF communication service. 	<p>5.2(1), 5.2(2), 5.3(5), 5.4(1), 5.4(2), 5.4(6) <i>mcfidlqle</i>, <i>mcfidmpqu</i>, and <i>mcfispqle</i> in 13. <i>Details of Operation Commands</i>, Table L-6 in subsection L(2)</p>
<p>The network status of messages exchanged with a remote system can now be displayed. With this change, the following command has been added:</p> <ul style="list-style-type: none"> • <code>mcfplsln</code> 	<p>5.2(4), Table 12-1 in subsection 12.1.4 <i>mcfplsln</i> in 13. <i>Details of Operation Commands</i></p>
<p>The maximum number of unprocessed receive messages can now be displayed. With this change, the following option has been added:</p> <ul style="list-style-type: none"> • <code>-m</code> option of the <code>mcfplssg</code> command <p>A note has been added about the maximum number of unprocessed receive messages.</p>	<p><i>mcfplssg</i> in 13. <i>Details of Operation Commands</i></p>

The following table lists changes in this manual (3000-3-D53-30(E)) and product changes related to this manual for uCosminexus TP1/Message Control 07-01 and uCosminexus TP1/NET/Library 07-01

Changes	Location
<p>Performance verification information (MCF performance verification traces) can now be output for important events during message exchange processing. With this change, an MCF performance verification trace information file has been added. The argument <code>_mc</code> has also been added for the <code>-f</code> option of the <code>prfget</code> command.</p>	<p>1.3.6(3)(a), 3.8.5(1), 7.1.12(3), Table 10-2 in section 10.8, <i>prfget</i> in 13. <i>Details of Operation Commands</i>, subsection L(2)</p>
<p>Starting and stopping of reception of server-type connection establishment requests can now be performed manually. With this change, the following commands have been added:</p> <ul style="list-style-type: none"> • <code>mcftofln</code> • <code>mcftonln</code> 	<p>5.2(4), Table 12-1 in subsection 12.1.4, <i>mcftofln</i> and <i>mcftonln</i> in 13. <i>Details of Operation Commands</i></p>

Changes	Location
<p>MCF information can now be acquired as realtime statistics acquisition items. With this change, explanations have been added about option combinations for the following commands:</p> <ul style="list-style-type: none"> • <code>rtsedit</code> • <code>rtsls</code> • <code>rtsstats</code> 	<p><i>rtsedit</i>, <i>rtsls</i>, and <i>rtsstats</i> in 13. <i>Details of Operation Commands</i>, Table E-6 in subsection E.5(1), Table E-8 in subsection E.5(1), Table E-9 in subsection E.5(2), Table E-10 in subsection E.5(2), Table E-11 in subsection E.5(2)</p>

In addition to the above changes, minor editorial corrections have been made.

The following table lists changes in the manual (3000-3-D53-20(E)) and product changes related to that manual for uCosminexus TP1/Server Base 07-02, uCosminexus TP1/Message Control 07-01, and uCosminexus TP1/NET/Library 07-01.

Changes
<p>A caution regarding the registration of an OpenTP1 system administrator has been added.</p>
<p>An event trace can now be output by a process service. With this addition, the process service event trace information file has been added. The argument <code>_pr</code> has also been added to the <code>-f</code> option of the <code>prfget</code> command.</p>
<p>A function that enables backup of PRF trace files to be suppressed has been added. With this addition, new operating modes and backup files relating to performance verification trace files have been added.</p>
<p>A performance verification trace (PRF trace) can now be output by the XA resource service. With this addition, the XAR performance verification trace information file has been added.</p>
<p>An event trace can now be output by the name service. With this addition, the NAM event trace has been added. The argument <code>_nm</code> has also been added to the <code>-f</code> option of the <code>prfget</code> command.</p>
<p>The file extension of files output when the following commands are executed in Linux has been changed to <code>.gz</code>.</p> <ul style="list-style-type: none"> • <code>dcrasget -c</code> • <code>usmdump</code>
<p>A function that enables service functions to be loaded dynamically has been added. With this addition, the description of the refresh function for the user server process has been changed. Notes for the following items have also been added:</p> <ul style="list-style-type: none"> • <code>rtsstats</code> command • Acquiring real-time statistics • Changing the settings of targets and the items to be acquired in real time
<p>An audit log output function has been added. With this addition, the operating modes for audit logs have been added, and the <code>dcauditsetup</code> command has also been added.</p>

Changes
The description of the remote API facility has been changed.
The following directory has been added to the description of OpenTP1's disk space requirements: <ul style="list-style-type: none"> • \$DCDIR/spool/dcjnlinf/unload/*
The description of changing the system time has been changed.
A function that allows the system to operate without using system journal files (journal fileless mode) has been added. With this addition, the following content has been added: <ul style="list-style-type: none"> • Descriptions of system operation with the journal fileless mode • Messages output by the operation commands • -j option of dcsetup command
A function that allows the real-time statistics service to backup RTS log files has been added. With this addition, the description of creating RTS log files has been changed.
A function that allows parallel access to system journal files has been added. With this addition, the description of using system journal files has been changed.
The description of inhibiting unload checks has been changed.
A function that monitors the backlog of messages in the input queue has been added. With this addition, an overview of the function and an illustration of the flow of processing have been added.
Whether or not command logs are output has been changed for the following commands: <ul style="list-style-type: none"> • trnstics • trndlinf • lckrminf • damadd • damrm • damhold • damrles • damchdef • tamadd • tamrm • tamhold • tamrles • tamload • tamunload • prfget • rtsstats
The cautions for the <code>filmkfs</code> command have been changed.
KFCA10266-W has been added as an output message for the <code>mcftswptr</code> command.
Cautions have been added to the descriptions of the <code>namchgfl</code> and <code>namndchg</code> commands regarding changing the node count.

Changes
A performance verification trace (PRF trace) can now be output by the XA resource service. With this addition, the XAR performance verification trace information file has been added. The argument <code>_xr</code> has also been added to the <code>-f</code> option of the <code>prfget</code> command.
An explanation that the <code>rpcmgr</code> command may fail to output any information or output information that is incomplete when a message spans more than one RPC trace file has been added.
A description of a situation where <code>----</code> appears in the command output of the <code>rtsedit</code> or <code>rtsls</code> commands has been added.
A description of the servers that cannot be specified as command targets has been added.
The schedule service can now be implemented on a service-by-service basis. With this addition, the following options have been added to the <code>scdlis</code> command to acquire information for specific services: <ul style="list-style-type: none"> • <code>-ae</code> • <code>-e</code> • <code>-t</code>
The following two descriptions have been added: <ul style="list-style-type: none"> • An initial state has been added to the transaction status display in the output format. • A description of situations where <code>***</code> appears in the command output has been added to the cautions.
The long attributes in the formatting of the structure <code>dc_mcf_dump_info</code> have been changed to <code>int</code> attributes.
Further items that the real-time statistics service can acquire have been added.

The following table lists changes in the manual (3000-3-D53-20(E)) and product changes related to that manual for uCosminexus TP1/Server Base 07-01

Changes
A detailed check can now be performed to make sure that no values are specified incorrectly in the system definitions. With this addition, the following options have been added to the <code>dcdefchk</code> command: <ul style="list-style-type: none"> • <code>-l</code> • <code>-c</code> • <code>-w</code> • <code>-e</code>
Some commands have been modified to allow the acquisition of command logs.
A function (MSDTC linkage) that enables transactional linkage by means of two-phase commit between OpenTP1 and an application that runs on .NET Framework has been added. With this addition, the <code>-s</code> option has been added to the <code>xarinit</code> command, and the <code>-r</code> option has been added to the <code>xarls</code> command.
Descriptions of what information to acquire when an error occurs in OpenTP1 and how to go about investigating the error have been added.

Changes
A function has been added for displaying the product name, version number, and other information about products operating in environments set up in the OpenTP1 directory. With this addition, the <code>dcpplist</code> command has been added.
The range of journal information that can be acquired as real-time statistics has been expanded.

Preface

This manual explains the administration and operation of the OpenTP1 Distributed Transaction Processing Facility.

Products described in this manual, other than those for which the manual is released, may not work with OpenTP1 Version 7 products. You need to confirm that the products you want to use work with OpenTP1 Version 7 products.

Intended readers

This manual is intended for system administrators and system designers.

This manual assumes that the user has already read the manual *OpenTP1 Description*.

Organization of this manual

This manual is organized into the following parts, chapters, and appendixes:

Part 1: OpenTP1 Environment Setup

1. *Environment Setup*

This chapter explains OpenTP1 environment setup and the setup procedure.

Part 2: OpenTP1 Operations

2. *OpenTP1 Startup and Termination*

This chapter explains the OpenTP1 startup and termination procedures.

3. *OpenTP1 Online Operation*

This chapter explains the normal online operation for OpenTP1.

4. *OpenTP1 File Operations*

This chapter explains the operations of files used for OpenTP1.

5. *Sending and Receiving Messages*

This chapter explains sending and receiving messages.

6. *OpenTP1 Optional Facility Operation*

This chapter explains the OpenTP1 optional facilities.

7. *Operations Under the Multi-node Facility*

This chapter explains operations required when the multinode facility is used: environment setup, OpenTP1 startup and termination procedures, OpenTP1 node

status display, and file operation and recovery.

8. MultiOpenTP1 Operation

This chapter explains the differences between multi OpenTP1 and normal OpenTP1 operation in units of nodes.

9. Operations Under the System Switchover Facility

This chapter explains operations required when the system switchover facility is used, more specifically, the differences in OpenTP1 operation from when the system switchover facility is not used.

Part 3: OpenTP1 Error Handling Procedures

10. Error Handling Procedures

This chapter classifies OpenTP1 errors and explains actions to be taken for different types of errors. It also explains information to be collected when errors occur and information to be inherited during full recovery.

11. Troubleshooting

This chapter explains what information to acquire when an error occurs in OpenTP1, and how to investigate the problem.

Part 4: OpenTP1 Operation Commands

12. Operation Commands

This chapter explains the operation commands used with OpenTP1 in terms of the input, specification, and details of these commands.

13. Details of Operation Commands

This chapter explains the operation commands in alphabetical order.

Appendixes

A. Dump File Format of Input/Output Queues

This appendix explains the dump file format of input/output queues.

B. OpenTP1 Events

This appendix explains the OpenTP1 events that can be registered with the JP1 Event Service.

C. Information Output for Audited Events

This appendix explains the information output for audited events.

D. Journal Information Collected by the Message Control Facility

This appendix explains journal information collected by the message control facility.

E. Details of Statistics

This appendix provides details of each type of statistics.

F. List of Files Output by OpenTP1

This appendix provides tables that list the files output by OpenTP1.

G. Estimating the Size of the Physical File for a Message Queue

This appendix provides formulae for estimating the size of physical files for a message queue.

H. Estimating the Size of OpenTP1 Files

This appendix explains the formulas for estimating the size of OpenTP1 files.

I. Estimating the Number of Record Locks

This appendix explains the formulae for estimating the number of record locks to be registered by OpenTP1.

J. Estimating the Resources Used by UNIX Message Exchange Functions

This appendix explains the formulae for estimating the resources used by the message exchange functions of UNIX.

K. Estimating the Size of an OpenTP1 File System

This appendix explains the formulae for estimating the size of an OpenTP1 file system.

L. Acquiring Performance Verification Traces

This appendix provides the timing to acquire the performance verification trace and describes contents to be acquired.

M. Details About Scenario Templates

This appendix explains the function of scenario templates and the details about scenario templates such as input scenario variables.

Related publications

This manual is part of a related set of manuals. The manuals in the set are listed below (with the manual numbers):

OpenTP1 products

- *OpenTP1 Version 7 Description* (3000-3-D50(E))
- *OpenTP1 Version 7 Programming Guide* (3000-3-D51(E))
- *OpenTP1 Version 7 System Definition* (3000-3-D52(E))
- *OpenTP1 Version 7 Operation* (3000-3-D53(E))

- *OpenTP1 Version 7 Programming Reference C Language* (3000-3-D54(E))
- *OpenTP1 Version 7 Programming Reference COBOL Language* (3000-3-D55(E))
- *OpenTP1 Version 7 Messages* (3000-3-D56(E))
- *OpenTP1 Version 7 Tester and UAP Trace User's Guide* (3000-3-D57(E))
- *OpenTP1 Version 7 TP1/Client User's Guide TP1/Client/W, TP1/Client/P* (3000-3-D58(E))
- *OpenTP1 Version 7 TP1/Client User's Guide TP1/Client/J* (3000-3-D59(E))
- *OpenTP1 Version 7 TP1/LiNK User's Guide* (3000-3-D60(E))[#]
- *OpenTP1 Version 7 Protocol TP1/NET/TCP/IP* (3000-3-D70(E))
- *OpenTP1 Version 7 TP1/Message Queue User's Guide* (3000-3-D90(E))[#]
- *OpenTP1 Version 7 TP1/Message Queue Messages* (3000-3-D91(E))[#]
- *OpenTP1 Version 7 TP1/Message Queue Application Programming Guide* (3000-3-D92(E))[#]
- *OpenTP1 Version 7 TP1/Message Queue Application Programming Reference* (3000-3-D93(E))[#]

Other OpenTP1 products

- *TP1/Web User's Guide and Reference* (3000-3-D62(E))[#]

Other related products

- *Indexed Sequential Access Method ISAM* (3000-3-046(E))
- *XP/W* (3000-3-047(E))
- *Extended Mapping Service 2/Workstation XMAP2/W DESCRIPTION/USER'S GUIDE* (3000-7-421(E))
- *SEWB 3 General Information* (3000-7-450(E))
- *Job Management Partner I/Base User's Guide* (3020-3-K06(E))
- *Job Management Partner I/Base Messages* (3020-3-K07(E))
- *Job Management Partner I/Base Software Developer's Guide* (3020-3-K08(E))

For OpenTP1 protocol manuals, please check whether English versions are available.

#

If you want to use this manual, confirm that it has been published. (Some of these manuals might not have been published yet.)

Conventions: Abbreviations for product names

This manual uses the following abbreviations for product names:

Abbreviation		Full name or meaning	
AIX		AIX 5L V5.1	
		AIX 5L V5.2	
		AIX 5L V5.3	
		AIX V6.1	
Client .NET	TP1/Client for .NET Framework	uCosminexus TP1/Client for .NET Framework	
Connector .NET	TP1/Connector for .NET Framework	uCosminexus TP1/Connector for .NET Framework	
DPM		JP1/ServerConductor/Deployment Manager	
HI-UX/WE2		HI-UX/workstation Extended Version 2	
HP-UX	HP-UX (IPF)		HP-UX 11i V2 (IPF)
			HP-UX 11i V3 (IPF)
	HP-UX (PA-RISC)		HP-UX 11i V1 (PA-RISC)
			HP-UX 11i V2 (PA-RISC)
IPF		Itanium(R) Processor Family	
Java		Java™	
JP1	JP1/AJS2	JP1/AJS2 - Agent	JP1/Automatic Job Management System 2 - Agent
		JP1/AJS2 - Manager	JP1/Automatic Job Management System 2 - Manager
		JP1/AJS2 - View	JP1/Automatic Job Management System 2 - View
	JP1/AJS2 - Scenario Operation	JP1/AJS2 - Scenario Operation Manager	JP1/Automatic Job Management System 2 - Scenario Operation Manager
		JP1/AJS2 - Scenario Operation View	JP1/Automatic Job Management System 2 - Scenario Operation View
		JP1/NETM/Audit	JP1/NETM/Audit - Manager
Linux		Linux(R)	
Linux (AMD64/Intel EM64T/x86)		Red Hat Enterprise Linux AS 4 (AMD64 & Intel EM64T)	

Abbreviation		Full name or meaning
		Red Hat Enterprise Linux AS 4 (x86)
		Red Hat Enterprise Linux ES 4 (AMD64 & Intel EM64T)
		Red Hat Enterprise Linux ES 4 (x86)
		Red Hat Enterprise Linux 5 (AMD/Intel 64)
		Red Hat Enterprise Linux 5 (x86)
		Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64)
		Red Hat Enterprise Linux 5 Advanced Platform (x86)
Linux (IPF)		Red Hat Enterprise Linux AS 4 (IPF)
		Red Hat Enterprise Linux 5 (Intel Itanium)
		Red Hat Enterprise Linux 5 Advanced Platform (Intel Itanium)
MS-DOS		Microsoft ^(R) MS-DOS ^(R)
NETM/DM		JP1/NETM/DM Client
		JP1/NETM/DM Manager
		JP1/NETM/DM SubManager
Oracle		Oracle 10g
		Oracle9i
Solaris		Solaris 8
		Solaris 9
		Solaris 10
TP1/Client	TP1/Client/J	uCosminexus TP1/Client/J
	TP1/Client/P	uCosminexus TP1/Client/P
	TP1/Client/W	uCosminexus TP1/Client/W
		uCosminexus TP1/Client/W(64)
TP1/EE		uCosminexus TP1/Server Base Enterprise Option
		uCosminexus TP1/Server Base Enterprise Option(64)

Abbreviation	Full name or meaning
TP1/Extension 1	uCosminexus TP1/Extension 1
	uCosminexus TP1/Extension 1(64)
TP1/FS/Direct Access	uCosminexus TP1/FS/Direct Access
	uCosminexus TP1/FS/Direct Access(64)
TP1/FS/Table Access	uCosminexus TP1/FS/Table Access
	uCosminexus TP1/FS/Table Access(64)
TP1/High Availability	uCosminexus TP1/High Availability
	uCosminexus TP1/High Availability(64)
TP1/LiNK	uCosminexus TP1/LiNK
TP1/Message Control	uCosminexus TP1/Message Control
	uCosminexus TP1/Message Control(64)
TP1/Message Control/Tester	uCosminexus TP1/Message Control/Tester
TP1/Message Queue	uCosminexus TP1/Message Queue
	uCosminexus TP1/Message Queue(64)
TP1/Message Queue - Access	uCosminexus TP1/Message Queue - Access
	uCosminexus TP1/Message Queue - Access(64)
TP1/Messaging	uCosminexus TP1/Messaging
TP1/Multi	uCosminexus TP1/Multi
TP1/NET/HDLC	uCosminexus TP1/NET/HDLC
TP1/NET/High Availability	uCosminexus TP1/NET/High Availability
	uCosminexus TP1/NET/High Availability(64)
TP1/NET/HSC	uCosminexus TP1/NET/HSC
TP1/NET/Library	uCosminexus TP1/NET/Library
	uCosminexus TP1/NET/Library(64)
TP1/NET/NCSB	uCosminexus TP1/NET/NCSB
TP1/NET/OSAS-NIF	uCosminexus TP1/NET/OSAS-NIF
TP1/NET/OSI-TP	uCosminexus TP1/NET/OSI-TP

Abbreviation		Full name or meaning
TP1/NET/SLU - TypeP2	TP1/NET/ Secondary Logical Unit - TypeP2	uCosminexus TP1/NET/Secondary Logical Unit - TypeP2
TP1/NET/TCP/IP		uCosminexus TP1/NET/TCP/IP
		uCosminexus TP1/NET/TCP/IP(64)
TP1/NET/UDP		uCosminexus TP1/NET/User Datagram Protocol
TP1/NET/User Agent		uCosminexus TP1/NET/User Agent
TP1/NET/X25		uCosminexus TP1/NET/X25
TP1/NET/X25-Extended		uCosminexus TP1/NET/X25-Extended
TP1/NET/XMAP3		uCosminexus TP1/NET/XMAP3
TP1/Offline Tester		uCosminexus TP1/Offline Tester
TP1/Online Tester		uCosminexus TP1/Online Tester
TP1/Resource Manager Monitor		uCosminexus TP1/Resource Manager Monitor
TP1/Server Base		uCosminexus TP1/Server Base
		uCosminexus TP1/Server Base(64)
TP1/Shared Table Access		uCosminexus TP1/Shared Table Access
TP1/Web		uCosminexus TP1/Web
Windows 2000		Microsoft ^(R) Windows ^(R) 2000 Advanced Server Operating System
		Microsoft ^(R) Windows ^(R) 2000 Datacenter Server Operating System
		Microsoft ^(R) Windows ^(R) 2000 Professional Operating System
		Microsoft ^(R) Windows ^(R) 2000 Server Operating System
Windows Server 2003		Microsoft ^(R) Windows Server ^(R) 2003, Datacenter Edition
		Microsoft ^(R) Windows Server ^(R) 2003, Enterprise Edition
		Microsoft ^(R) Windows Server ^(R) 2003, Standard Edition
Windows Server 2003 R2		Microsoft ^(R) Windows Server ^(R) 2003 R2, Enterprise Edition

Abbreviation	Full name or meaning
	Microsoft ^(R) Windows Server ^(R) 2003 R2, Standard Edition
Windows Server 2003 x64 Editions	Microsoft ^(R) Windows Server ^(R) 2003, Datacenter x64 Edition
	Microsoft ^(R) Windows Server ^(R) 2003, Enterprise x64 Edition
	Microsoft ^(R) Windows Server ^(R) 2003, Standard x64 Edition
Windows Server 2003 R2 x64 Editions	Microsoft ^(R) Windows Server ^(R) 2003 R2, Enterprise x64 Edition
	Microsoft ^(R) Windows Server ^(R) 2003 R2, Standard x64 Edition
Windows Server 2008	Microsoft ^(R) Windows Server ^(R) 2008 Datacenter (x86)
	Microsoft ^(R) Windows Server ^(R) 2008 Enterprise (x86)
	Microsoft ^(R) Windows Server ^(R) 2008 Standard (x86)
Windows Server 2008 x64 Editions	Microsoft ^(R) Windows Server ^(R) 2008 Datacenter (x64)
	Microsoft ^(R) Windows Server ^(R) 2008 Enterprise (x64)
	Microsoft ^(R) Windows Server ^(R) 2008 Standard (x64)
Windows Vista	Microsoft ^(R) Windows Vista ^(R) Business (x86)
	Microsoft ^(R) Windows Vista ^(R) Enterprise (x86)
	Microsoft ^(R) Windows Vista ^(R) Ultimate (x86)
Windows Vista x64 Editions	Microsoft ^(R) Windows Vista ^(R) Business (x64)
	Microsoft ^(R) Windows Vista ^(R) Enterprise (x64)
	Microsoft ^(R) Windows Vista ^(R) Ultimate (x64)
Windows XP	Microsoft ^(R) Windows ^(R) XP Professional Operating System

- The term Windows is used to indicate Windows Server 2003, Windows XP and Windows Vista if the difference in functions among them need not be considered.
- The term UNIX is used to indicate AIX, HP-UX, Linux, and Solaris.

Conventions: Acronyms

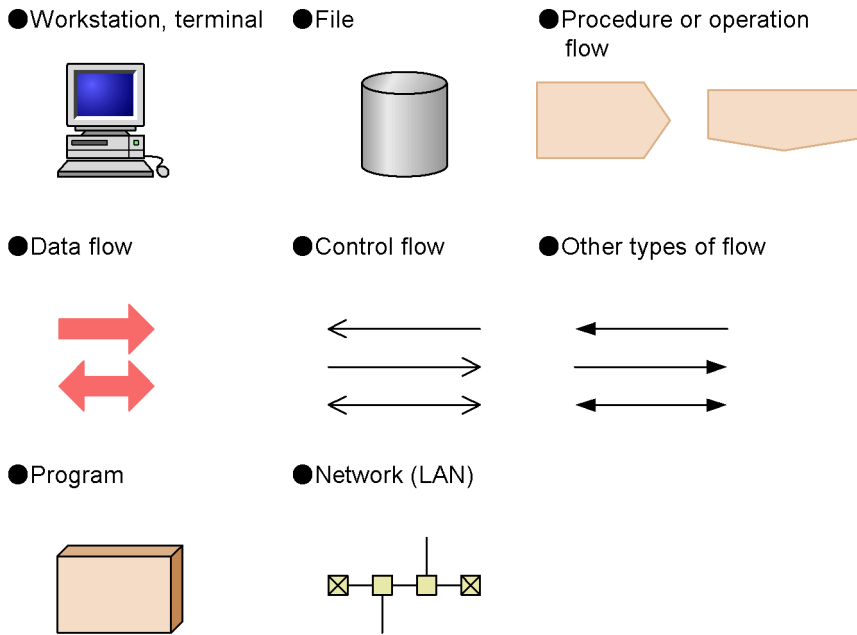
This manual also uses the following acronyms:

Acronym	Full name or meaning
ANSI	American National Standards Institute
AP	Application Program
CPU	Central Processing Unit
CUP	Client User Program
DAM	Direct Access Method
DAM FRC	DAM File Recovery
DHCP	Dynamic Host Configuration Protocol
DID	Distributed Identifier
DML	Data Manipulation Language
DNS	Domain Name System
HA	High Availability
I/O	Input/Output
IP	Internet Protocol
ISAM	Indexed Sequential Access Method
IST	Internode Shared Table
LAN	Local Area Network
MCF	Message Control Facility
MHP	Message Handling Program
MQA	Message Queue Access
MSDTC	Microsoft Distributed Transaction Coordinator
NIS	Network Information Service
OS	Operating System
OSI	Open Systems Interconnection
OSI TP	Open Systems Interconnection Transaction Processing
PRF	Performance

Acronym	Full name or meaning
RI	Recovery Information
RMM	Resource Manager Monitor
RPC	Remote Procedure Call
RTS	Real Time Statistic
SPP	Service Providing Program
TAM	Table Access Method
TAM FRC	TAM File Recovery
UAP	User Application Program
UID/GID	User Identifier/Group Identifier
UOC	User Own Coding
UTC	Coordinated Universal Time
VM	Virtual Machine
XA	Extended Architecture
XAR	Extended Architecture Resource

Conventions: Diagrams

This manual uses the following conventions in diagrams:



Conventions: Differences between JIS and ASCII keyboards

The JIS code and ASCII code keyboards are different in the input characters represented by the following codes. In this manual, the use of a JIS keyboard is assumed for these characters.

Code	JIS keyboard	ASCII keyboard
(5c) ₁₆	¥ (yen symbol)	\ (backslash)
(7e) ₁₆	— (overline)	~ (tilde)

Conventions: Fonts and symbols

The following table explains the fonts used in this manual:

Font	Convention
Bold	<p>Bold type indicates text on a window, other than the window title. Such text includes menus, menu options, buttons, radio box options, or explanatory labels. For example:</p> <ul style="list-style-type: none"> • From the File menu, choose Open. • Click the Cancel button. • In the Enter name entry box, type your name.

Font	Convention
<i>Italics</i>	<p><i>Italics</i> are used to indicate a placeholder for some actual text to be provided by the user or system. For example:</p> <ul style="list-style-type: none"> Write the command as follows: <code>copy source-file target-file</code> The following message appears: A file was not found. (file = <i>file-name</i>) <p><i>Italics</i> are also used for emphasis. For example:</p> <ul style="list-style-type: none"> Do <i>not</i> delete the configuration file.
Code font	<p>A code font indicates text that the user enters without change, or text (such as messages) output by the system. For example:</p> <ul style="list-style-type: none"> At the prompt, enter <code>dir</code>. Use the <code>send</code> command to send mail. The following message is displayed: The password is incorrect.

The following table explains the symbols used in this manual:

Symbol	Convention
	In syntax explanations, a vertical bar separates multiple items, and has the meaning of OR. For example: A B C means A, or B, or C.
{ }	In syntax explanations, curly brackets indicate that only one of the enclosed items is to be selected. For example: {A B C} means only one of A, or B, or C.
[]	In syntax explanations, square brackets indicate that the enclosed item or items are optional. For example: [A] means that you can specify A or nothing. [B C] means that you can specify B, or C, or nothing.
...	In coding, an ellipsis (...) indicates that one or more lines of coding are not shown for purposes of brevity. In syntax explanations, an ellipsis indicates that the immediately preceding item can be repeated as many times as necessary. For example: A, B, B, ... means that, after you specify A, B, you can specify B as many times as necessary.
~	The value for the preceding item must be specified in accordance with the attributes listed immediately following the swung dash.
<<>>	Default value assumed when specification is omitted.
<>	Syntax element (see below).
(())	Range of specifiable values.

Conventions for permitted characters

In most cases, only the following characters are permitted as syntax elements (if other characters are permitted, the manual will state this explicitly):

Type	Definition
Upper-case alphabetic characters	A to Z and underscore (_)
Lower-case alphabetic characters	a to z
Alphabetic characters	A to Z, a to z
Alphabetic symbols	Alphabetic characters (A to Z, a to z), #, @, and \$
Numeric characters	0 to 9
Alphanumeric characters	A to Z, a to z, 0 to 9
Special characters	*
Unsigned integers	Numeric values (0 to 9)
Hexadecimals	Numeric values (0 to 9), A to F, and a to f (However, if the manual specifies numeric characters (0 to 9) and alphabetic characters a to f, the hexadecimal value you specify cannot include the characters A to F.)
Identifiers	Alphanumeric character string beginning with an alphabetic character (A to Z, a to z)
Symbolic names	String of alphanumeric symbols beginning with an alphabetic symbol
Character strings	Array of any characters
Path names	Symbolic name, slash (/), and period (.) (The path name depends on the OS being used.)
OpenTP1 file names	Character string consisting of alphabetic characters (A to Z, a to z), numeric characters (0 to 9), period (.), underscore (_), and @ (maximum length of 14 characters)

Conventions: KB, MB, GB, and TB

This manual uses the following conventions:

- 1 KB (kilobyte) is 1,024 bytes.
- 1 MB (megabyte) is 1,024² bytes.
- 1 GB (gigabyte) is 1,024³ bytes.
- 1 TB (terabyte) is 1,024⁴ bytes.

Conventions: Platform-specific notational differences

For the Windows version of OpenTP1, there are some notational differences from the description in the manual. The following table describes these differences.

Item	Description in the manual	Change to:
Environment variable	<i>\$aaaaaa</i> Example: \$DCDIR	<i>%aaaaaa%</i> Example: %DCDIR%
Path name separator	Colon (:)	Semicolon (;)
Directory name separator	Slash (/)	Backslash (\)
Absolute path name	A path from the root directory Example: /tmp	A path name from a drive letter and the root directory Example: C:\tmp
Executable file name	File name only (without an extension) Example: mcfmngrd	File name with an extension Example: mcfmngrd.exe
make command	make	nmake

Conventions: Version numbers

The version numbers of Hitachi program products are usually written as two sets of two digits each, separated by a hyphen. For example:

- Version 1.00 (or 1.0) is written as 01-00.
- Version 2.05 is written as 02-05.
- Version 2.50 (or 2.5) is written as 02-50.
- Version 12.25 is written as 12-25.

The version number might be shown on the spine of a manual as *Ver: 2.00*, but the same version number would be written in the program as *02-00*.

Important note

Please check the availability of the products and manuals for HAmonitor, ServerConductor/DeploymentManager, Cosminexus, and Job Management Partner 1/ Automatic Job Management System 2.

Contents

Preface	i
Intended readers	i
Organization of this manual	i
Related publications	iii
Conventions: Abbreviations for product names	v
Conventions: Acronyms	x
Conventions: Diagrams	xi
Conventions: Differences between JIS and ASCII keyboards	xii
Conventions: Fonts and symbols	xii
Conventions: KB, MB, GB, and TB	xiv
Conventions: Platform-specific notational differences	xv
Conventions: Version numbers	xv
Important note	xv

PART 1: OpenTP1 Environment Setup

1. Environment Setup	1
1.1 Overview	2
1.1.1 Overview of the environment setup procedure	2
1.2 Environment setup by superuser	5
1.2.1 Cataloging of OpenTP1 administrator	5
1.2.2 Definition of OpenTP1 group	5
1.2.3 Installation of OpenTP1	5
1.2.4 Creation of OpenTP1 directory	6
1.2.5 Cataloging of OpenTP1 into the OS and deletion of OpenTP1	6
1.2.6 Creation of OpenTP1 file system area	8
1.2.7 Setting up environment for domain communication	8
1.2.8 Changing common definition for the system	10
1.3 Environment setup by OpenTP1 administrator	11
1.3.1 Creation and checking of the system definition	11
1.3.2 OpenTP1 administrator environment setup	14
1.3.3 Reservation of resources for OpenTP1 internal control	15
1.3.4 Initialization of OpenTP1 file systems	15
1.3.5 Creation of OpenTP1 files	15
1.3.6 Creation of non-OpenTP1 files	17
1.3.7 Cataloging of resource manager	30
1.3.8 Creation of transaction control object files	30

1.3.9	Changing system common definition.....	31
1.4	Preparations for TP1/Message Control execution.....	32
1.4.1	Creation of the MCF main function of the MCF communication service....	32
1.4.2	Creation of the MCF main function of the application startup service.....	34
1.4.3	Installation of the MCF main function into the directory.....	36
1.4.4	Cataloging of MCF service name.....	37
1.4.5	Creation of system service information definition file.....	37
1.4.6	Creation of definition object files.....	38
1.4.7	Preparation for acquiring command logs.....	41

PART 2: OpenTP1 Operations

2. OpenTP1 Startup and Termination 43

2.1	Startup.....	44
2.1.1	Startup methods.....	44
2.1.2	Startup modes.....	44
2.1.3	Selection of startup method.....	44
2.1.4	Selection of startup form.....	44
2.2	Termination.....	47
2.2.1	Termination modes.....	47
2.2.2	Termination methods.....	49
2.2.3	Notes.....	49

3. OpenTP1 Online Operation 51

3.1	Server operations.....	52
3.1.1	User server startup.....	52
3.1.2	User server termination.....	52
3.1.3	Display of server status.....	53
3.1.4	Search path for user server and command activated from user server.....	54
3.1.5	Replacement of user server.....	54
3.1.6	User server process.....	55
3.2	Scheduling operations.....	56
3.2.1	Display of scheduling status.....	56
3.2.2	Shutdown and restart of scheduling.....	56
3.2.3	Automatic shutdown of scheduling.....	57
3.2.4	Inter-node load balance.....	59
3.2.5	Changing the number of processes.....	64
3.2.6	Monitoring the schedule queue.....	64
3.2.7	Schedule queue congestion monitoring.....	68
3.2.8	Refreshing user server processes.....	74
3.3	Transaction operations.....	76
3.3.1	Display of transaction status.....	76
3.3.2	Forcible determination of transaction.....	76

3.3.3	Forcible termination of transaction	76
3.3.4	Deleting undecided-transaction information files	77
3.3.5	Startup and termination of collecting transaction statistics	77
3.3.6	Acquisition of transaction statistics when the XA resource service is used	78
3.4	Lock operations	79
3.4.1	Display of lock information	79
3.4.2	Display of lock control table pool information	79
3.4.3	Deletion of deadlock information files and timeout information files	79
3.5	Standard output file operations	80
3.6	Logging	81
3.6.1	Message log	81
3.6.2	Command log	82
3.7	Audit log operations	85
3.7.1	Setting up the environment for audit log acquisition	85
3.7.2	Audit log output format	87
3.7.3	Example of audit log file estimation	89
3.7.4	Information output to audit logs	90
3.7.5	List of audited events and their triggers	95
3.7.6	Examples of audit log operation	111
3.8	Trace operations	117
3.8.1	Output of UAP trace information	117
3.8.2	RPC trace operations	117
3.8.3	Swapping MCF trace information	118
3.8.4	Temporarily outputting MCF trace information	118
3.8.5	Performance verification trace operations	119
3.9	Shared memory operations	124
3.9.1	Display of shared memory utilization status	124
3.9.2	Buffer pool for storing messages	124
3.10	OpenTP1 domain operations	126
3.10.1	Changing the domain configuration	126
3.10.2	Domain representative schedule service	129
3.10.3	Invalidating the startup notification information	130
3.10.4	Verifying OpenTP1 startup and clearing the cache	130
3.11	Operations using TP1/NET/OSI-TP-Extended	131
3.12	Operations using the remote API facility	132
3.12.1	Preparations for using the remote API facility	132
3.12.2	Displaying the status of a RAP-processing listener or a RAP-processing server	133
3.12.3	Performance improvement of the remote API facility	133
3.12.4	Starting and stopping the RAP-processing listener and RAP-processing server	134
3.12.5	Starting and stopping the RAP-processing client manager	135
3.12.6	Establishing connection with RAP-processing listener	136
3.12.7	Cautions when using the remote API facility	136

3.13	Nonstop operation of OpenTP1	138
3.13.1	Handling resources	138
3.13.2	Notes on changing the configuration.....	140
3.13.3	Replacing part of an MCF communication service.....	141
3.13.4	Notes on changing the time	143
3.14	Operations using the XA resource service.....	145
3.14.1	Preparations for using the XA resource service	145
3.14.2	Transaction management using the XA resource service.....	145
3.14.3	Starting and stopping the XA resource service.....	150
3.14.4	Action to take if an error occurs in the XAR file	152
3.14.5	Trace information acquired by the XA resource service	162
3.15	Operations using Job Management Partner 1	163
3.15.1	System operations using scenario templates	163
3.15.2	Registering a scenario.....	164
3.15.3	Using Scale Out.....	166
3.15.4	Using Scale In.....	173
3.15.5	Using Rolling Update	175
3.15.6	Using sample scenario templates.....	178
3.16	Operations for using the real-time statistics service	186
3.16.1	Preparation for using the real-time statistics service.....	186
3.16.2	Starting and stopping the real-time statistics service.....	186
3.16.3	Acquiring real-time statistics.....	190
3.16.4	Outputting real-time statistics.....	194
3.16.5	Changing the settings of targets and the items to be acquired in real time.....	196

4. OpenTP1 File Operations 199

4.1	OpenTP1 file system operations	200
4.1.1	Creation of OpenTP1 file system	200
4.1.2	Display of OpenTP1 file system status.....	200
4.1.3	Display of OpenTP1 file system contents	200
4.1.4	Backup of OpenTP1 file system.....	201
4.1.5	Restoration of OpenTP1 file system.....	201
4.1.6	Change of OpenTP1 file owner.....	201
4.1.7	Change of OpenTP1 file group.....	201
4.1.8	Change of OpenTP1 file access authorization mode.....	201
4.1.9	OpenTP1 file system garbage collection	202
4.1.10	Change of OpenTP1 file system attributes	202
4.1.11	Re-creation of OpenTP1 files	203
4.2	Status file operations.....	204
4.2.1	Creation and definition of status files.....	204
4.2.2	Use of status files.....	204
4.2.3	Display of status file status.....	206
4.2.4	Display of status file contents.....	206

4.2.5	Status file open and close	206
4.2.6	Deletion of status file	207
4.2.7	Status file space shortage	207
4.2.8	Status changes for status files.....	207
4.3	System journal file operations.....	210
4.3.1	OpenTP1 journals.....	210
4.3.2	Creating and defining system journal files.....	211
4.3.3	Using system journal files	213
4.3.4	Unloading of system journal files	216
4.3.5	Reuse of system journal files	222
4.3.6	Display of system journal file information	223
4.3.7	System journal file open and close.....	223
4.3.8	Change of system journal file status	224
4.3.9	Swapping of system journal files	224
4.3.10	Nonswappable file group	224
4.3.11	Status changes for system journal files	225
4.3.12	Output of system statistics to journal	227
4.3.13	Chronological sorting and merging of unload journal files	227
4.3.14	Copying of unload journal files.....	227
4.3.15	Editing and output of unload journal files.....	227
4.3.16	Output of records of unload journal files	228
4.3.17	Output of uptime statistics.....	228
4.3.18	Journal integration for file recovery.....	234
4.3.19	Journal fileless mode.....	234
4.4	Recovery journal file operations.....	239
4.4.1	Transaction recovery journal file operations.....	239
4.4.2	Server recovery journal file operations	239
4.5	Checkpoint dump file operations.....	240
4.5.1	Creation and definition of checkpoint dump files	240
4.5.2	Use of checkpoint dump files.....	241
4.5.3	Deletion of checkpoint dump files	242
4.5.4	Automatic opening of checkpoint dump files	243
4.5.5	No checkpoint dump collection destination	243
4.5.6	Display of checkpoint dump file information	243
4.5.7	Checkpoint dump file open and close	243
4.5.8	Duplication of checkpoint dump files	243
4.5.9	Status changes for checkpoint dump files	245
4.6	DAM file operations.....	249
4.6.1	Creation of DAM files.....	249
4.6.2	DAM file status management.....	249
4.6.3	Display of DAM file status	249
4.6.4	DAM file addition and deletion	250
4.6.5	Logical file logical shutdown and release of shutdown status.....	250
4.6.6	DAM file backup and restore	251

4.6.7	DAM file recovery.....	251
4.6.8	DAM file lock.....	252
4.6.9	DAM file addition in online mode	253
4.6.10	Setting the number of cache blocks for a DAM file.....	253
4.6.11	Extending the block length for a DAM file	254
4.6.12	Extracting user data from a DAM file	255
4.7	TAM file operations.....	256
4.7.1	Creation of TAM files.....	256
4.7.2	TAM table status management	257
4.7.3	Display of TAM table status	257
4.7.4	TAM table addition and removal	257
4.7.5	TAM table logical shutdown and release of shutdown status.....	258
4.7.6	TAM table loading and unloading	258
4.7.7	TAM file backup and restoration	259
4.7.8	Creation of a TAM data file from a TAM file	259
4.7.9	TAM file deletion	260
4.7.10	TAM file recovery	260
4.7.11	TAM file lock.....	260
4.7.12	TAM file addition in online mode	261
4.7.13	Displaying the synonym information after creating a TAM file	261
4.7.14	Increasing the number of records in a TAM file.....	261
5.	Sending and Receiving Messages	263
5.1	MCF communication service operations	264
5.2	Connection operations	265
5.3	Application operations	267
5.4	Logical terminal operations	269
5.5	Service group operations	274
5.6	Service operations.....	278
5.7	Protocol-specific operations	279
5.7.1	Startup and termination of a session.....	279
5.7.2	Display of buffer group utilization status	279
5.7.3	Map file.....	279
5.8	Monitoring message queues.....	280
5.9	Queue operations	288
6.	OpenTP1 Optional Facility Operation	289
6.1	Resource manager operations	290
6.1.1	Display of information about resource manager	290
6.1.2	Cataloging and deletion of resource manager	290
6.1.3	Creation of transaction control object files.....	290
6.1.4	Resource manager monitor operations	291
6.1.5	Suspending OpenTP1 startup until the startup of a resource manager.....	296
6.1.6	Suspending transaction recovery (when Oracle9i RAC is used).....	298

6.1.7	Pre-online transaction recovery.....	298
7.	Operations Under the Multi-node Facility	301
7.1	OpenTP1 environment setup.....	302
7.1.1	Cataloging of OpenTP1 administrator.....	302
7.1.2	Definition of OpenTP1 group.....	302
7.1.3	Installation of OpenTP1.....	302
7.1.4	Creation of OpenTP1 directory.....	302
7.1.5	Creation of system definition.....	303
7.1.6	Cataloging of OpenTP1 into OS.....	303
7.1.7	Creation of OpenTP1 file system areas.....	303
7.1.8	Reservation of resources for OpenTP1 internal control.....	304
7.1.9	OpenTP1 administrator environment setup.....	304
7.1.10	Initialization of OpenTP1 file systems.....	305
7.1.11	Creation of OpenTP1 files.....	305
7.1.12	Creation of non-OpenTP1 files.....	306
7.2	OpenTP1 startup and termination.....	316
7.2.1	Startup.....	316
7.2.2	Termination.....	316
7.3	Display of OpenTP1 node status.....	319
7.4	Global journal operations.....	320
7.4.1	Configuration, creation, and definition of archive journal files.....	320
7.4.2	Use of archive journal files.....	321
7.4.3	Unloading of archive journal files.....	323
7.4.4	Reuse of archive journal files.....	324
7.4.5	Display of archive journal file information.....	325
7.4.6	Display of archive status.....	325
7.4.7	Archive journal file open and close.....	325
7.4.8	Change of archive journal file status.....	326
7.4.9	Swapping of archive journal files.....	326
7.4.10	Nonswappable target file group.....	326
7.4.11	Status changes of archive journal files.....	327
7.4.12	Chronological sorting and merging of global archive unload journal files and unload journal files.....	328
7.4.13	Editing and output of global archive unload journal files.....	329
7.4.14	Output of records of global archive unload journal files.....	329
7.4.15	Output of uptime statistics.....	329
7.4.16	Journal integration for file recovery.....	329
7.4.17	Unloading of system journal files at archive-journal source node.....	330
7.4.18	File recovery at archive-journal source node.....	330
7.4.19	Recovery of recovery journal files.....	332
8.	MultiOpenTP1 Operation	333
8.1	MultiOpenTP1 environment setup.....	334

8.1.1	OpenTP1 administrator setup	334
8.1.2	Creation of OpenTP1 directories	334
8.1.3	Cataloging of OpenTP1s into OS	334
8.1.4	Initialization of OpenTP1 file systems	334
8.1.5	User environment setup	334
8.1.6	Changing a shared library	335
8.1.7	Creation of OpenTP1 files	336
8.1.8	Creation of other files	337
8.2	Operation command execution environment	338
9.	Operations Under the System Switchover Facility	339
9.1	Preparations for using system switchover facility	340
9.2	Startup and termination	342
9.2.1	Startup	342
9.2.2	Termination	342
9.3	System switchover procedure	344
9.3.1	Automatic system switchover	344
9.3.2	Planned system switchover	344
9.3.3	Grouped system switchover	345
9.4	Standby user server	346
9.5	Operation commands	347
PART 3: OpenTP1 Error Handling Procedures		
10.	Error Handling Procedures	349
10.1	Error events and causes	350
10.2	OpenTP1 file errors	353
10.2.1	Status files	353
10.2.2	System journal files	356
10.2.3	Archive journal files	358
10.2.4	Recovery journal files	359
10.2.5	Checkpoint dump files	360
10.2.6	DAM file	362
10.2.7	TAM file	363
10.2.8	Message queue files	365
10.2.9	XAR file	366
10.2.10	MCF definition file	366
10.3	File errors	367
10.3.1	Message log file	367
10.4	Communication errors	368
10.5	UAP errors	369
10.5.1	UAP cannot be started	369
10.5.2	UAP does not terminate	369

10.5.3	UAP terminates abnormally	369
10.5.4	UAP deadlocks	369
10.6	OpenTP1 errors	370
10.6.1	OpenTP1 cannot be started	370
10.6.2	OpenTP1 cannot be terminated	372
10.6.3	OpenTP1 terminates abnormally	373
10.6.4	OpenTP1 operation command does not terminate normally	374
10.6.5	Response wait overtime occurs during OpenTP1 operation command execution	374
10.6.6	OpenTP1 errors under multinode facility	374
10.7	CPU errors	376
10.8	Information collection when error occurs	377
10.9	Information inherited during full recovery	382

11. Troubleshooting 383

11.1	Information to acquire and items to check	384
11.1.1	Information to acquire	384
11.1.2	Items to check	388
11.1.3	Method of checking traces	388
11.2	Troubleshooting procedures	391
11.2.1	When the message KFCA00307-E is output	391
11.2.2	When the message KFCA00327-W is output	395
11.2.3	When the message KFCA00328-W is output	401
11.2.4	When the message KFCA00502-I is output	404
11.2.5	When the message KFCA00837-I is output	406
11.2.6	When the message KFCA00854-E is output	409
11.2.7	When the message KFCA00906-E or KFCA00907-E is output	413
11.2.8	When the message KFCA01803-I is output	415
11.2.9	When the message KFCA01820-E is output	418
11.2.10	When the message KFCA01864-E is output	419
11.2.11	When the error -902: DCTRNER_ROLLBACK is returned	422
11.2.12	When a user server fails to start	424

PART 4: OpenTP1 Operation Commands

12. Operation Commands 427

12.1	Overview of operation commands	428
12.1.1	Operation command input methods	428
12.1.2	Operation command specification format	428
12.1.3	Displaying how to use operation commands	430
12.1.4	List of operation commands	431
12.1.5	Notes on operation command entry	459

13. Details of Operation Commands

461

damadd (Add a logical file)	462
dambkup (Back up a physical file)	465
damchdef (Set a threshold for the number of cache blocks).....	467
damchinf (Acquire the number of cache blocks).....	468
damdel (Delete a physical file)	471
damfrc (Recover logical files)	472
damhold (Shut down a logical file logically).....	477
damload (Initialize a physical file)	479
daml (Display status of logical files).....	481
damlr (Release logical file from shutdown)	484
damm (Remove a logical file)	486
dammr (Restore a physical file).....	488
dcauditsetup (Set up the environment for the audit log facility)	491
dccspool (Delete troubleshooting information)	494
dcdefchk (Check system definitions).....	497
dcjchconf (Set the value of an operand in the system definition).....	501
dcjcmdex (Execute an OpenTP1 command from a scenario template).....	504
dcjnamch (Update the domain definition file).....	506
dcmakeup (Reserve or release resources for OpenTP1 internal control)	508
dcmapchg (Change map file path name)	509
dcmapl (Display loaded resources of map file).....	511
dcmstart (Start multinode area or subarea)	517
dcmstop (Terminate multinode area or subarea).....	521
dcndls (Display status of OpenTP1 node)	524
dcplist (Display product information).....	527
dcrasget (Acquire maintenance data).....	529
dcreport (Edit and output system statistics to standard output in real time).....	531
dcreset (Restart process service and reflect definitions).....	535
dcsetup (Catalog OpenTP1 into OS, delete OpenTP1 from OS).....	536
dcshtml (Display utilization status of common memory)	539
dcstart (Start OpenTP1)	544
dcstats (Start and terminate collecting system statistics).....	547
dcstop (Terminate OpenTP1).....	550
dcsvstart (Start servers).....	553
dcsvstop (Terminate server).....	555
filbkup (Back up an OpenTP1 file system)	557
filchgrp (Change an OpenTP1 file group)	559
filchmod (Change an OpenTP1 file access authorization mode)	561
filchown (Change an OpenTP1 file owner).....	564
fills (Display contents of an OpenTP1 file system).....	566
filmkfs (Initialize an OpenTP1 file system)	571
filrstr (Restore an OpenTP1 file system)	573
filstatfs (Display status of an OpenTP1 file system)	576

jnladdpf (Allocate journal physical files)	580
jnlardis (Forces to release connection to resource group)	582
jnlaris (Display archive status)	583
jnlautnl (Control the automatic unloading facility)	586
jnlchgfg (Change status of journal files)	590
jnlclsfg (Close journal files)	593
jnlcolc (Integrate file recovery journals)	596
jnlcopy (Copy unload journal files)	601
jnldelpf (Delete journal physical files)	606
jnledit (Edit and output unload journal files or global archive unload journal files)	607
jnlinit (Initialize journal files)	618
jnlis (Display journal files information)	620
jnlmest (Output MCF uptime statistics)	633
jnlmkrf (Recover journal files)	640
jnlpnfg (Open journal files)	641
jnlrinf (Display journal information that is read during restart)	644
jnlrm (Delete journal files)	646
jnlrput (Output records of unload journal files or global archive unload journal files)	647
jnlsort (Sort and merge unload journal files or global archive unload journal files chronologically)	662
jnlstts (Output uptime statistics)	665
jnlswpfg (Swap journal files)	683
jnlunlfg (Unload journal files)	685
lckls (Display lock information)	691
lckpool (Display lock control table pool information)	693
lckrminf (Delete deadlock information file and timeout information file)	695
logcat (Display message log file)	696
logcon (Switch message log realtime output facility)	699
mcfactap (Release shutdown of applications)	701
mcfaclep (Initialize abnormal terminations counts for applications)	704
mcfadctap (Shut down applications)	707
mcfadltap (Delete timer activation requests for applications)	710
mcfalsap (Display status of applications)	713
mcfalstap (Display application timer startup requests)	716
mcfreport (Edit MCF statistics)	720
mcfstats (Output MCF statistics)	726
mctactcn (Establish connections)	730
mctactle (Release shutdown of logical terminals)	734
mctactmj (Start message journal collection for logical terminals)	737
mctactsg (Release service groups from shutdown)	739
mctactss (Start a session)	742
mctactsv (Release services from shutdown)	745
mctfchcn (Switch connections)	747
mctfdctn (Release connections)	749

mcftdctle (Shut down logical terminals).....	753
mcftdctmj (Terminate message journal collection for logical terminals)	756
mcftdctsg (Shut down service groups).....	759
mcftdctss (Terminate a session).....	762
mcftdctsv (Shut down services).....	765
mcftdlql (Delete output queues for logical terminals)	767
mcftdlqsg (Delete input queues of service groups)	770
mcftdmpqu (Copy contents of input/output queues)	773
mcftedalt (Terminate alternate sending)	776
mcftendct (Terminate forcibly continuous inquiry-response processing for logical terminals)	778
mcfthldiq (Hold process of input queue for service groups)	781
mcfthldoq (Hold process of a logical terminal output queue)	784
mcftlsbuf (Display utilization status of buffer groups).....	787
mcftlscn (Display status of connections).....	789
mcftlscm (Reference MCF communication service status).....	793
mcftlsle (Display status of logical terminals)	795
mcftlsln (Display network status).....	800
mcftlssg (Display status of service groups)	802
mcftlssv (Display status of services)	806
mcftlstrd (Display the status of multi-processing of messages)	809
mcftlsutm (Display user timer monitoring status)	812
mcftofln (Stop accepting server-type connection establishment requests).....	817
mcftonln (Start accepting server-type connection establishment requests).....	819
mcftrlsiq (Release held process of a service group input queue).....	821
mcftrlsoq (Release held process of a logical terminal output queue).....	824
mcftspql (Skip first message in a logical terminal message queue)	827
mcftstalt (Start alternate sending).....	830
mcftstart (Partial start of an MCF communication service)	832
mcftstop (Partial stop of an MCF communication service).....	834
mcftstptr (End acquiring an MCF trace).....	837
mcftstrtr (Start acquiring an MCF trace)	839
mcftswptr (Swap MCF trace files forcibly).....	840
mcfuevt (Start application programs)	842
namalivechk (Verify OpenTP1 startup and clear cache).....	844
namblad (Manipulate RPC suppression list of OpenTP1 nodes)	846
namchgfl (Change domain configuration (by using the domain definition files))	848
namdomainsetup (Register and delete domain representative schedule service).....	853
namndchg (Change domain configuration (by using the system common definition))	854
namsvinf (Display OpenTP1 server information)	857
namunavl (Forcibly invalidate the OpenTP1 start notification information)	862
prckill (Forcibly terminate OpenTP1 process)	864
prcls (Display status of servers).....	865

prepath (Change search path names for user server and for command activated from user server)	868
prepaths (Display search path names for user server and for command activated from user server)	870
prectrl (Stop or restart the prttee process).....	871
prttee (Redirect standard output and standard error output of OpenTP1)	873
prfed (Edit and output trace information files).....	875
prfget (Get trace information files).....	892
queinit (Allocate physical file for message queue)	894
quels (Display status of queue groups).....	895
querm (Delete physical file for message queue)	898
rapdfgen (Automatically generate definitions for using the remote API facility).....	899
rapls (Display the status of a RAP-processing listener or a RAP-processing server) ...	901
rapsetup (Set up execution environment for the remote API facility).....	904
rpcdump (Output RPC trace information).....	905
rpcmrg (Merge RPC traces).....	916
rpstat (Display execution status of temporary closing)	918
rtsedit (Edit and output RTS log files).....	920
rtsls (Output real-time statistics to standard output).....	931
rtsssetup (Set execution environment for real-time statistics service).....	942
rtssstats (Change targets and items for real-time statistics acquisition).....	944
scdchprc (Change the number of processes)	950
scdhold (Shut down scheduling)	954
scdls (Display scheduling status).....	956
scdrles (Release scheduling shutdown)	966
scdrsprc (Stop and restart processes).....	968
stsclose (Close a status file).....	970
stsfills (Display contents of a status file).....	972
stsinic (Create and initialize a status file)	975
stsls (Display status of status files).....	977
stsoopen (Open a status file).....	981
stsrms (Delete a status file).....	983
stsswap (Swap status files)	984
tamadd (Add a TAM table).....	985
tambkup (Back up a TAM file).....	989
tamcre (Initialize a TAM file).....	992
tamdel (Delete a TAM file)	995
tamfrc (Recover a TAM file).....	996
tamhold (Shut down a TAM table logically).....	1000
tamhsls (Display the synonym information about a hash-format TAM file and TAM table)	1001
tamlckls (Convert TAM locked resource names).....	1004
tamload (Load a TAM table)	1006
tamls (Display status of TAM tables).....	1008

tamrles (Release TAM table from shutdown).....	1012
tamrm (Remove a TAM table).....	1015
tamrstr (Restore a TAM file).....	1017
tamunload (Unload a TAM table).....	1018
tptrnls (Display undecided transaction information for OSI TP communication).....	1020
trncmt (Commit transactions).....	1023
trndlinf (Delete undecided-transaction information files).....	1025
trnfgt (Terminate transactions forcibly).....	1026
trnlkrm (Catalog and delete resource managers).....	1028
trnls (Display status of transactions).....	1033
trnlsrm (Display information about resource manager).....	1041
trnmkobj (Create a transaction control object file).....	1043
trnrbk (Roll back transactions).....	1046
trnstics (Start and terminate collecting transaction statistics).....	1048
usmdump (Output memory dump).....	1050
xarevtr (Display XAR event trace information).....	1052
xarfills (Display XAR file status).....	1056
xarforce (Change the status of an XAR transaction).....	1059
xarhold (Shut down the XA resource service).....	1062
xarinit (Create an XAR file).....	1063
xarls (Display XAR transaction information).....	1065
xarrles (Release XA resource service from shutdown status).....	1071
xarrm (Delete an XAR file).....	1072

Appendixes 1075

A. Dump File Format of Input/Output Queues.....	1076
B. OpenTP1 Events.....	1080
B.1 Cataloging events.....	1080
B.2 OpenTP1 events that can be cataloged.....	1080
C. Information Output for Audited Events.....	1082
D. Journal Information Collected by the Message Control Facility.....	1100
D.1 AJ record format.....	1100
D.2 GJ record format.....	1101
D.3 IJ record format.....	1102
D.4 MJ record format.....	1103
D.5 OJ record format.....	1104
D.6 Conditions for collecting journals by the message control facility.....	1106
D.7 Calculation formula of the necessary amount of journal collected by the message control facility.....	1107
E. Details of Statistics.....	1109
E.1 System statistics.....	1109
E.2 Statistics about transaction.....	1130
E.3 Statistics about response.....	1131
E.4 Statistics about communication delay.....	1146

E.5 Real-time statistics	1146
F. List of Files Output by OpenTP1	1190
G. Estimating the Size of the Physical File for a Message Queue	1269
G.1 Estimation formula of the record length	1269
G.2 Estimation formula of the number of records	1269
G.3 Estimation example	1269
H. Estimating the Size of OpenTP1 Files	1271
H.1 Estimating the size of a status file	1271
H.2 Estimating the size of a system journal file	1276
H.3 Estimating the size of a server recovery journal file	1286
H.4 Estimating the size of a checkpoint dump file	1286
H.5 Estimating the size of an archive journal file	1287
H.6 Estimating the size of a DAM file	1289
H.7 Estimating the size of a TAM file	1289
I. Estimating the Number of Record Locks	1291
J. Estimating the Resources Used by UNIX Message Exchange Functions	1293
K. Estimating the Size of an OpenTP1 File System	1295
L. Acquiring Performance Verification Traces	1297
M. Details About Scenario Templates	1325
M.1 OpenTP1_AddNode (Add an OpenTP1 node to the domain definition file)	1325
M.2 OpenTP1_ChangeNodeID (Set the node ID of an OpenTP1 node)	1326
M.3 OpenTP1_Deploy (Register OpenTP1)	1327
M.4 OpenTP1_ScenarioAddNode (Add a new OpenTP1 node to the domain)	1328
M.5 OpenTP1_Start (Start OpenTP1)	1329
M.6 OpenTP1_StartUAP (Start a UAP)	1330
M.7 OpenTP1_Stop (Stop OpenTP1)	1331
M.8 OpenTP1_StopUAP (Stop a UAP)	1332
M.9 OpenTP1_Undeploy (Delete OpenTP1)	1333
M.10 OpenTP1_UpdateDomain (Update the domain configuration)	1333
M.11 OpenTP1_ScenarioScaleout (Scale Out sample scenario template)	1334

Index	1337
--------------------	-------------

List of figures

Figure 1-1: OpenTP1 environment setup procedure	3
Figure 1-2: Environment setup procedure for TP1/Message Control	4
Figure 1-3: Environment setup procedure for TP1/Message Queue	4
Figure 1-4: Overview of how to code the MCF main function that defines a protocol and user exit routines (for ANSI C and C++)	32
Figure 1-5: Overview of how to code the MCF main function that defines a protocol and user exit routines (for K&R C)	33
Figure 1-6: Overview of how to code the MCF main function of the application startup service (for ANSI C and C++)	35
Figure 1-7: Overview of how to code the MCF main function of the application startup service (for K&R C)	36
Figure 1-8: Overview of how to install the MCF main function into the directory	37
Figure 1-9: Overview of definition object file creation procedure	39
Figure 3-1: User server startup	52
Figure 3-2: Example of monitoring the schedule queue (schedule_delay_limit=15, schedule_delay_abort=Y)	66
Figure 3-3: Service requests remaining in the schedule queue	69
Figure 3-4: Example of schedule queue congestion monitoring	72
Figure 3-5: Audit log acquisition by file rotation	88
Figure 3-6: Point of audit log output (OpenTP1 startup)	96
Figure 3-7: Point of audit log output (OpenTP1 standby)	97
Figure 3-8: Point of audit log output (normal termination of OpenTP1)	98
Figure 3-9: Point of audit log output (abnormal termination of OpenTP1 or critical error in process service)	99
Figure 3-10: Point of audit log output (user server startup)	100
Figure 3-11: Point of audit log output (normal termination of user server)	101
Figure 3-12: Point of audit log output (abnormal termination of user server)	102
Figure 3-13: Point of audit log output (user server shutdown or service shutdown on user server)	103
Figure 3-14: Point of audit log output (successful or unsuccessful client user authentication)	104
Figure 3-15: Point of audit log output (synchronous-response RPC)	105
Figure 3-16: Point of audit log output (asynchronous-response RPC)	105
Figure 3-17: Point of audit log output (non-response RPC)	106
Figure 3-18: Point of audit log output (invalid message discarded)	107
Figure 3-19: Point of audit log output (error accessing the OpenTP1 file system)	108
Figure 3-20: Point of audit log output (command execution)	109
Figure 3-21: Point of audit log output (OpenTP1 service startup)	110
Figure 3-22: Point of audit log output (OpenTP1 service termination)	111
Figure 3-23: Flow of automatic audit log backup	112

Figure 3-24: Flow of automatic system shutdown when audit log output fails.....	113
Figure 3-25: Example of collecting and centrally managing audit logs using JP1/NETM/Audit	114
Figure 3-26: Overview of temporarily outputting MCF trace information	119
Figure 3-27: Flow of performance verification trace file operation	121
Figure 3-28: Partially replacing MCF communication service files.....	142
Figure 3-29: Operation of the XA resource service.....	146
Figure 3-30: Swapping XAR files at failure of the online XAR file.....	155
Figure 3-31: Swapping XAR files at failure of the backup XAR file	157
Figure 3-32: Swapping XAR files at failure of the online XAR file with OpenTP1 not running	159
Figure 3-33: Swapping XAR files at failure of both the online and backup XAR files with OpenTP1 not running.....	161
Figure 3-34: Operating model for Scale Out (when the domain configuration management node is used).....	166
Figure 3-35: Operating model for Scale Out (when the domain configuration management node is not used)	168
Figure 3-36: Sequence of scenario templates executed by the scenario for adding a new OpenTP1 node.....	170
Figure 3-37: Sequence of scenario templates executed by the scenario for updating the domain configuration	172
Figure 3-38: Scenario for stopping OpenTP1 and releasing resources	174
Figure 3-39: Scenario for stopping a UAP and releasing resources	175
Figure 3-40: Operating model for Rolling Update	176
Figure 3-41: Sequence of scenario templates executed by Rolling Update	177
Figure 3-42: OpenTP1 directory configuration after the sample scenario templates are executed.....	182
Figure 3-43: Acquiring real-time statistics	190
Figure 3-44: Interval for acquiring real-time statistics	192
Figure 3-45: Switching RTS log files	195
Figure 3-46: Example of adding items to be acquired.....	196
Figure 4-1: Garbage collection in OpenTP1 file system	202
Figure 4-2: Status changes for status file in online mode.....	205
Figure 4-3: Relationship between records and journal blocks.....	211
Figure 4-4: Execution results of jnlis -j sys -d.....	211
Figure 4-5: Using the unload journal file storage directories	219
Figure 4-6: Status changes for system journal file in online mode.....	223
Figure 4-7: Status changes caused by open and close commands (system journal file).....	224
Figure 4-8: Sequence of using checkpoint dump files.....	241
Figure 4-9: Status changes by open and close commands (checkpoint dump file)	243
Figure 4-10: Relationship between TAM data file format and record format of data section of TAM file (-s option not specified).....	256
Figure 4-11: Relationship between TAM data file format and record format of data section of TAM file (-s option specified).....	257

Figure 5-1: Overview of message queue monitoring facility.....	280
Figure 5-2: Example of the message queue monitoring facility	285
Figure 6-1: Process ID output format.....	293
Figure 6-2: Format of standard output of process ID acquisition command	294
Figure 7-1: Status changes of online archive journal file.....	325
Figure 7-2: Status changes by open and close commands (archive journal file).....	326
Figure 11-1: Troubleshooting procedure when message KFCA00307-E is output	392
Figure 11-2: Troubleshooting procedure when message KFCA00327-W is output.....	396
Figure 11-3: Troubleshooting procedure when message KFCA00328-W is output.....	403
Figure 11-4: Troubleshooting procedure when message KFCA00502-I is output	405
Figure 11-5: Troubleshooting procedure when message KFCA00837-I is output	408
Figure 11-6: Troubleshooting procedure when message KFCA00854-E is output	411
Figure 11-7: Troubleshooting procedure when message KFCA00906-E is output	414
Figure 11-8: Troubleshooting procedure when message KFCA00907-E is output	415
Figure 11-9: Troubleshooting procedure when message KFCA01803-I is output	417
Figure 11-10: Troubleshooting procedure when message KFCA01820-E is output	419
Figure 11-11: Troubleshooting procedure when message KFCA01864-E is output	421
Figure 11-12: Troubleshooting procedure when the error -902: DCTRNER_ROLLBACK is returned.....	423
Figure 11-13: Troubleshooting procedure when a user server does not start.....	425
Figure 13-1: Relationship between RTS log files and specified options	930
Figure A-1: Dump file format.....	1076
Figure A-2: Format of structure dc_mcf_dump_info.....	1078
Figure E-1: When the target service exists in a node different from the node that requests the service.....	1133
Figure E-2: When a service is executed in the local node	1139
Figure E-3: When a service is executed from a client	1143

List of tables

Table 1-1: OpenTP1 file system area owner and access rights.....	8
Table 1-2: Flow of checks performed by the dcddefchk command.....	13
Table 1-3: OpenTP1 files and operation commands for their creation and initialization.....	16
Table 1-4: User-created files and directories	17
Table 1-5: Files and directories created at installation or cataloging into OS	20
Table 1-6: Files and directories created during normal operation	23
Table 1-7: Files and directories created when an error occurs.....	28
Table 2-1: Startup forms	45
Table 2-2: Handling of service requests in scheduling queue and messages in input and output queues.....	49
Table 3-1: Message handling at abnormal termination.....	57
Table 3-2: Handling of disk queue messages.....	58
Table 3-3: Handling of memory queue messages	58
Table 3-4: Handling of messages at application shutdown.....	58
Table 3-5: Handling of messages at service shutdown	59
Table 3-6: Definitions for using inter-node load balancing	60
Table 3-7: Combinations of scd_announce_server_status and scd_this_node_first operands ..	62
Table 3-8: Operations of inter-node load balancing facility used with other facilities.....	63
Table 3-9: Scope of changes permitted to service functions on user servers.....	75
Table 3-10: Directories and files created when the dcauditsetup command is executed.....	85
Table 3-11: Examples of audit log file estimation	89
Table 3-12: Items output to audit log file.....	91
Table 3-13: List of audited events.....	95
Table 3-14: Correspondence between output point and message ID for an audit log entry associated with OpenTP1 startup	97
Table 3-15: Correspondence between output points and message IDs for an audit log entry associated with OpenTP1 standby.....	97
Table 3-16: Correspondence between output points and message IDs for an audit log entry associated with normal termination of OpenTP1	98
Table 3-17: Correspondence between output points and message IDs for an audit log entry associated with abnormal termination of OpenTP1 and critical errors in the process service.....	99
Table 3-18: Correspondence between output points and message IDs for an audit log entry associated with user server startup	100
Table 3-19: Correspondence between output points and message IDs for an audit log entry associated with normal termination of user server.....	101
Table 3-20: Correspondence between output points and message IDs for an audit log entry associated with abnormal termination of user server	102
Table 3-21: Correspondence between output points and message IDs for an audit log entry associated with user server shutdown or service shutdown on user server.....	103

Table 3-22: Correspondence between output points and message IDs for an audit log entry associated with successful or unsuccessful client user authentication	104
Table 3-23: Correspondence between output points and message IDs for an audit log entry associated with RPC execution	106
Table 3-24: Correspondence between output points and message IDs for an audit log entry associated with discarding invalid messages.....	107
Table 3-25: Correspondence between output points and message IDs for an audit log entry associated with OpenTP1 file system access errors	108
Table 3-26: Correspondence between output points and message IDs for an audit log entry associated with command execution	109
Table 3-27: Correspondence between output point and message ID for an audit log entry associated with OpenTP1 service startup	110
Table 3-28: Correspondence between output point and message ID for an audit log entry associated with the OpenTP1 service service.....	111
Table 3-29: JP1 series products required for linkage with JP1/NETM/Audit.....	115
Table 3-30: Disk space used by OpenTP1	138
Table 3-31: Transaction information management (when linked with a J2EE application server).....	147
Table 3-32: Transaction information management (when linked with a .NET Framework application).....	148
Table 3-33: Transaction branch recovery processing at OpenTP1 restart.....	152
Table 3-34: Environment variables whose values are determined from input scenario variables.....	165
Table 3-35: Scenario templates in the scenario for adding a new OpenTP1 node.....	171
Table 3-36: Scenario templates in the scenario for updating the domain configuration.....	172
Table 3-37: Scenario template in the scenario for stopping OpenTP1 and releasing resources	174
Table 3-38: Scenario template in the scenario for stopping a UAP and releasing resources ..	175
Table 3-39: Scenario templates in Rolling Update.....	177
Table 3-40: Processing executed by the sample scenario templates	178
Table 3-41: Paths for the shell files for each OS.....	181
Table 3-42: Information to be set in the sample scenario templates	181
Table 3-43: Settings in the sample OpenTP1 environment setup shell file	183
Table 3-44: Values set for character string @DCxxxxxx@ in the sample scenario template ..	184
Table 3-45: Files created in the OpenTP1 file systems set for the input scenario variables BETRAN_FILE1 and BETRAN_FILE2	185
Table 3-46: Directories and files created when the rtsetup command is executed.....	186
Table 3-47: Status of RTSSUP at startup and the operation of the real-time statistics service.....	189
Table 3-48: Values specified for the options in the rtspout definition command and acquisition targets.....	191
Table 4-1: Command options for re-creating an OpenTP1 file.....	203
Table 4-2: Status changes for status files in online mode	207
Table 4-3: Relationship between element file status and physical file status	213

Table 4-4: Relationship between file statuses when an element file error occurs	214
Table 4-5: First unload journal file storage directory to be used	218
Table 4-6: Commands that input unload journal files.....	222
Table 4-7: Data that can be output by jnlrput command.....	228
Table 4-8: Edited contents of transaction statistics.....	229
Table 4-9: Edited contents of statistics about response	230
Table 4-10: Edited contents of statistics about communication delay.....	231
Table 4-11: Edited contents of MCF uptime statistics.....	231
Table 4-12: Types of MCF statistics	233
Table 4-13: Edited contents of the MCF statistics (mcfreport command).....	234
Table 4-14: Facilities unavailable in journal fileless mode	235
Table 4-15: Files unavailable in journal fileless mode	238
Table 4-16: Difference in file group status at error between when one-system operation is available and when unavailable.....	244
Table 4-17: Difference in online operation for only one system	245
Table 4-18: Status changes for checkpoint dump files in online mode	246
Table 4-19: Status changes for checkpoint dump files in online mode when one-system operation is available.....	246
Table 4-20: Status changes for checkpoint dump files in online mode when one-system operation is unavailable.....	248
Table 5-1: Operation when the mcftdctl command is entered while a message is being received or sent	269
Table 5-2: Relationship between judgment conditions during message queue monitoring and MCF behavior	282
Table 5-3: Queued message counts and judgment results	286
Table 7-1: OpenTP1 file system area owners and access permissions	304
Table 7-2: OpenTP1 files to be created and initialized, and operation commands used for creation and initialization.....	305
Table 7-3: Files and directories created by the user.....	306
Table 7-4: Files and directories created at installation or cataloging into OS	307
Table 7-5: Files and directories created at execution of OpenTP1 (when using the multi-node facility)	308
Table 7-6: Relationship between file group status and physical file status	321
Table 7-7: Commands that input unload journal files.....	324
Table 7-8: Status changes of online archive journal file.....	327
Table 8-1: Initialization of OpenTP1 files	336
Table 9-1: Combinations of host names OpenTP1 uses in running and standby systems.....	341
Table 10-1: Error events and causes	350
Table 10-2: Information collection when error occurs	377
Table 10-3: Information inherited during full recovery.....	382
Table 11-1: Information required for troubleshooting (for TP1/Server Base, TP1/LiNK, TP1/FS/Direct Access, TP1/FS/Table Access, etc.).....	384
Table 11-2: Information required for troubleshooting (for TP1/Client/P).....	385
Table 11-3: Information required for troubleshooting (for TP1/Client/W)	385

Table 11-4: Information required for troubleshooting (for TP1/Client/J).....	386
Table 11-5: Information required for troubleshooting (for TP1/Client for .NET Framework)	386
Table 11-6: Information required for troubleshooting (for TP1/Message Control and TP1/NET/***)	387
Table 11-7: Values specified for max_socket_descriptors operand	393
Table 11-8: Values specified for max_open_fds operand	394
Table 11-9: Operands that require modification when the parallel_count operand value is increased	411
Table 11-10: Operands that require modification when the message_store_bufLen operand value is increased	413
Table 12-1: List of OpenTP1 operation commands	432
Table 13-1: Files and directories created by the dcauditsetup command.....	491
Table 13-2: Execution result of the dcauditsetup command when the audit log output directory already exists	492
Table 13-3: Execution result of the dcauditsetup command when the audit log file already exists	492
Table 13-4: Files deleted by using the dccspool command.....	495
Table 13-5: Changes you can make to the all_node operand and the status of the all_node and all_node_ex operands after the namndchg command is executed.....	856
Table 13-6: Node startup statuses displayed by the namsvinf command.....	861
Table 13-7: Combinations of rtsedit command options and the statistics to be edited	922
Table 13-8: Examples of valid intervals for outputting edited real-time statistics to a CSV file.....	923
Table 13-9: Combinations of rtsls command options and the statistics to be output	932
Table 13-10: Arguments of the -o and -b options in the rtsstats command, and the object determined from the specified arguments	946
Table B-1: OpenTP1 events that can be cataloged in the JP1 event service facility.....	1080
Table C-1: Information output for audited events.....	1082
Table D-1: Attribute symbols.....	1100
Table E-1: Details of the system's statistics	1109
Table E-2: Items to be edited in system statistics.....	1122
Table E-3: Details of statistics about transaction	1131
Table E-4: Details of statistics about response.....	1131
Table E-5: Details of statistics about communication delay	1146
Table E-6: Details about real-time statistics.....	1147
Table E-7: Process termination and applicable item counts.....	1167
Table E-8: Edited real-time statistics information.....	1168
Table E-9: Details about real-time statistics (when a target is specified)	1183
Table E-10: Format for specifying a target for acquiring real-time statistics	1185
Table E-11: Edited real-time statistics information (when a target is specified)	1186
Table F-1: List of files output by TP1/Server Base (common to both UNIX and Windows versions)	1190

Table F-2: Descriptions of the files output by TP1/Server Base (common to both UNIX and Windows versions)	1205
Table F-3: List of files output by TP1/Server Base (applicable to UNIX version only)	1231
Table F-4: Descriptions of the files output by TP1/Server Base (applicable to UNIX version only)	1237
Table F-5: List of files output by TP1/Server Base (applicable to Windows version only) ..	1244
Table F-6: Descriptions of the files output by TP1/Server Base (applicable to Windows version only)	1247
Table F-7: List of files output by TP1/FS/Direct Access.....	1251
Table F-8: Descriptions of the files output by TP1/FS/Direct Access.....	1252
Table F-9: List of files output by TP1/FS/Table Access.....	1253
Table F-10: Descriptions of the files output by TP1/FS/Table Access.....	1254
Table F-11: List of files output by TP1/Online Tester	1255
Table F-12: Descriptions of the files output by TP1/Online Tester	1257
Table F-13: List of files output by TP1/Message Control	1261
Table F-14: Descriptions of the files output by TP1/Message Control	1264
Table H-1: Number of bytes in the status file used by each key of each service and the number of keys (for an archive source node).....	1272
Table H-2: Number of bytes in the status file used by each key of each service and the number of keys (for an archive node).....	1274
Table H-3: Number of bytes in the status file used by each key and the number of keys (for an MCF service).....	1275
Table H-4: Number of journals per transaction	1279
Table H-5: Number of journals for statistics to be acquired by using the dcstats command	1285
Table H-6: Number of journals per RPC	1286
Table H-7: Size of a checkpoint dump file	1287
Table K-1: Formulas for estimating the sizes of OpenTP1 files.....	1295
Table L-1: Timing to acquire the performance verification Traces	1297
Table L-2: Information acquired for each event ID (performance verification traces)	1308
Table L-3: Timing for acquisition of MCF performance verification trace information.....	1317
Table L-4: Information acquired for each event ID (MCF performance verification traces)	1319
Table L-5: Dump output information for each event ID (MCF performance verification trace).....	1321
Table L-6: API name that is output for each MCF function	1322
Table L-7: User exit routine name that is output for each MCF user exit routine.....	1324

Chapter

1. Environment Setup

This chapter explains OpenTP1 environment setup and the setup procedure.

This chapter contains the following sections:

- 1.1 Overview
- 1.2 Environment setup by superuser
- 1.3 Environment setup by OpenTP1 administrator
- 1.4 Preparations for TP1/Message Control execution

1.1 Overview

This section explains the environment setup procedure up to the point immediately before OpenTP1 operation begins. The OS administrator (the superuser) begins setup of the OpenTP1 environment. The OpenTP1 system administrator registered by the superuser continues the setup procedure.

Use of the following OpenTP1 functions requires a special procedure for each product in addition to the normal environment setup procedure:

- TP1/Message Control
- TP1/Messaging
- TP1/Message Queue
- TP1/NET/OSI-TP-ExtendedTP1/Message Control

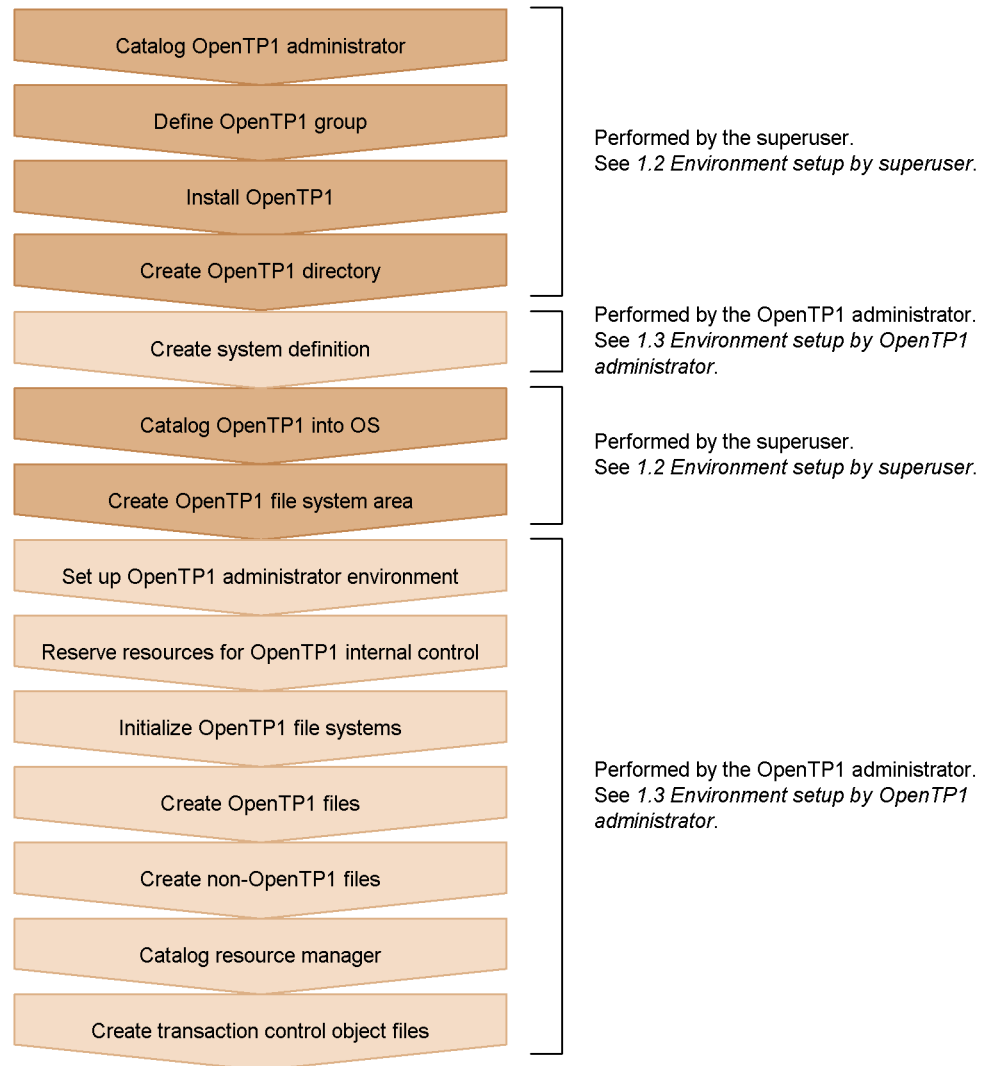
1.1.1 Overview of the environment setup procedure

The procedure for setting up the OpenTP1 environment is explained below.

(1) *OpenTP1 environment setup*

The following shows the OpenTP1 environment setup procedure.

Figure 1-1: OpenTP1 environment setup procedure



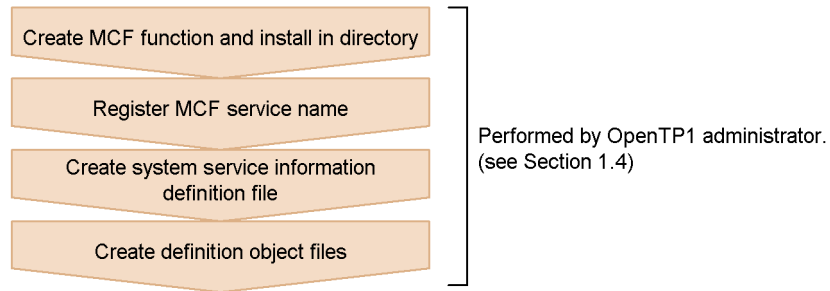
When you link JP1/Base, JP1/AJS2, and JP1/AJS2 - Scenario Operation, and use the scenario templates of the Scale Out scenario, you can partially automate the OpenTP1 environment setup procedure. For details about environment setup using scenario templates, see 3.15.3(3) *Using Scale Out and DPM for the OpenTP1 environment setup procedure.*

(2) Setup for TP1/Message Control

The following shows the environment setup procedure for TP1/Message Control.

1. Environment Setup

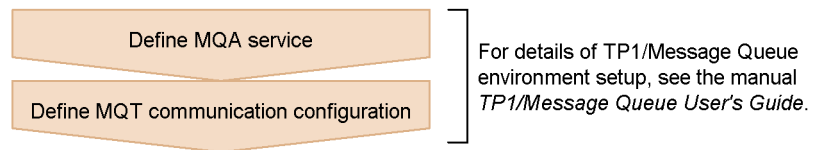
Figure 1-2: Environment setup procedure for TP1/Message Control



(3) Setup for TP1/Message Queue

The following shows the environment setup procedure for TP1/Message Queue.

Figure 1-3: Environment setup procedure for TP1/Message Queue



1.2 Environment setup by superuser

This section explains the environment setup that is performed by the superuser in order to run OpenTP1. For details about the OpenTP1 operation commands, see *13. Details of Operation Commands*.

For the setup details that depend on the OS, see the manual for the appropriate OS.

1.2.1 Cataloging of OpenTP1 administrator

Before OpenTP1 can be installed, the user ID of the OpenTP1 administrator must be cataloged into the OS, and a password must be assigned to the user ID. Do not register the OpenTP1 administrator under a large user ID, as this type of user ID is not supported by the OpenTP1 system.

Login name: Any name

User ID: Any ID

Group ID: Any ID

Home directory: Any directory

Login shell: Any shell

The OpenTP1 administrator is granted the following authorities:

- Authority to access the various OpenTP1 system files and directories as their owner (this authority enables the OpenTP1 administrator to prohibit write operations by other users)
- Authority to execute the operation commands that change the OpenTP1 configuration

Once the OpenTP1 administrator's user ID has been cataloged, a password must be defined.

1.2.2 Definition of OpenTP1 group

A group specifically in charge of OpenTP1 must be defined.

Once an OpenTP1 group has been defined, access to files by any user other than a member of the OpenTP1 group can be restricted, thereby enhancing the security of OpenTP1 usage.

1.2.3 Installation of OpenTP1

The OpenTP1 program product can now be installed. The directory in which you install OpenTP1 depends on the OS you are using.

1.2.4 Creation of OpenTP1 directory

The OpenTP1 directory must be created. The length of the name of the OpenTP1 directory depends on your OS.

You cannot use symbolic links for the OpenTP1 directory.

Assign one partition for the OpenTP1 directory. Do not use this partition for any other programs. If other programs create files in the partition for the OpenTP1 directory, thereby consuming free space on the disk or the number of i nodes, operation of OpenTP1 may be affected.

Specify the owner, group, and mode as follows:

Owner: OpenTP1 administrator

Group: OpenTP1 group

Mode: 0755

1.2.5 Cataloging of OpenTP1 into the OS and deletion of OpenTP1

This subsection describes how to catalog OpenTP1 into and delete OpenTP1 from the OS.

(1) *Cataloging OpenTP1 into the OS*

The OpenTP1 administrator must create OpenTP1 system definition information and then the superuser catalogs OpenTP1 into the OS by using the OpenTP1 `dcsetup` command.

If the OpenTP1 home directory specified in the `dcsetup` command does not contain the directories and files required for executing OpenTP1, the command creates the directories, copies the files, and adds the resource managers required for executing OpenTP1 when the command catalogs OpenTP1 into the OS. When you execute the `dcsetup` command for the directory for which the necessary environment has already been created, OpenTP1 is only cataloged into the OS.

If you do not use multiOpenTP1, specify `/BeTRAN` as the OpenTP1 home directory. If you do this, the `dcsetup` command does not need to copy the necessary files when you catalog OpenTP1 into the OS for the first time. The `/BeTRAN` owner and group are changed to the ID for root when you install the program. If you want to execute the `dcsetup` command after you install OpenTP1, change the `/BeTRAN` owner and group to the ID of the OpenTP1 administrator.

When you use multiOpenTP1, create a new directory for OpenTP1, change the directory owner and group to the ID for the OpenTP1 administrator, and specify it in an argument of `dcsetup`. When you use multiOpenTP1, specify a directory other than the root partition so that the root partition is not adversely affected.

Note the following when you catalog OpenTP1 into the OS:

- As the OpenTP1 home directory, do not specify a directory on a remote file system or a directory that is symbolically linked. If you do, operation is not guaranteed.
- When you specify the OpenTP1 home directory in the `dcsetup` command, specify a directory that is specifically created for OpenTP1 or `/BeTRAN`. If you do not do this, the environment will be damaged.
- When you execute the `dcsetup` command, the owner and group of the directories in the OpenTP1 home directory (`aplib`, `bin`, `conf`, `etc`, `examples`, `include`, `lib`, `spool`, `tmp`) and the files in those directories are changed along with the owner and group of the OpenTP1 home directory.

The owner and group are changed when a new directory, other than `examples`, is created.

(2) Deleting OpenTP1 from the OS

If you want to delete OpenTP1 from the OS, terminate OpenTP1. After this operation, the superuser must use the `dcsetup` command for OpenTP1 for the deletion.

When you delete OpenTP1 from the OS, you need to select whether to delete the environment in the OpenTP1 home directory. If you want to delete the environment in the OpenTP1 home directory, processing differs as follows depending on the location of the OpenTP1 home directory:

When the OpenTP1 home directory is `/BeTRAN`

- The `spool` and `tmp` directories are deleted.
- RM connection information is deleted.
- The message object file is deleted.

When the OpenTP1 home directory is not `/BeTRAN`

- The `bin`, `etc`, `include`, `lib`, `spool`, and `tmp` directories are deleted.
- RM connection information is deleted.
- The message object file is deleted.

If OpenTP1 has already been deleted from the OS when you execute the `dcsetup` command, the `dcsetup` command deletes the environment in the OpenTP1 home directory.

If you do not delete the environment in the OpenTP1 home directory, you can execute the `dcsetup` command whenever necessary to catalog OpenTP1 into the OS. If you want to change the environment under the OpenTP1 home directory, use the `dcsetup` command to delete OpenTP1 from the OS and then execute `dcsetup` again to catalog OpenTP1 into the OS.

1.2.6 Creation of OpenTP1 file system area

The OpenTP1 file systems can be created on character special files or ordinary files.

When a character special file is used, a disk partition must be allocated for the OpenTP1 file system. Do not mount this disk partition. When an ordinary file is used, there is no need to allocate a disk partition for the OpenTP1 file system.

Regardless of the type of files used for the OpenTP1 file systems, the following two OpenTP1 file systems must be created:

- OpenTP1 file system for the system
- OpenTP1 file system for users

If character special files are to be used, two disk partitions must be allocated, one for the system and one for users.

The files in which the OpenTP1 file systems are created are called OpenTP1 file system areas. To prevent unauthorized users from accessing an OpenTP1 file system area, an owner of the OpenTP1 file system area and access rights must be defined as shown in the following table; the owner and access rights are defined with OS commands.

Table 1-1: OpenTP1 file system area owner and access rights

Type of OpenTP1 file system area	Owner		Access permissions		
	User ID	Group ID	Owner	Group	Other
For system	OpenTP1 administrator	OpenTP1 group	rw (can read and write)	r- (can read only)	r- (can read only)
For user	OpenTP1 administrator	OpenTP1 group	rw (can read and write)	rw (can read and write)	r- (can read only)

Note

The OpenTP1 file system for the system and OpenTP1 file systems for users can all be allocated in the same OpenTP1 file system area. When this is done, the access rights must be for users.

1.2.7 Setting up environment for domain communication

Registering the domain representative schedule service in the domain data file makes it possible to perform domain qualification using `dc_rpc_call()`. In inter-domain communication, all the messages are sent to the destination schedule service via the domain representative schedule service.

A domain is composed of multiple OpenTP1 servers. Determine the OpenTP1 server

that represents the domain. The schedule service of the OpenTP1 server becomes the domain representative schedule service. The user registers the domain representative schedule service with the host name in the domain data file. The domain data file refers to the hosts information file of DNS.

To perform domain-specified communication, issue an `dc_rpc_call()`, specifying the domain. On receiving a message, the domain representative schedule service passes it to the final destination schedule service in the domain.

Domain-specified communication requires the following settings:

(1) Registering the domain representative schedule services

In a domain data file, three domain representative schedule services can be registered. Use the `namdomainsetup` command to register the host name of a domain representative schedule service. Only the superuser can register domain representative schedule services.

(2) Registering the port number of the domain representative schedule services

Register the port number of the domain representative schedule services in `/etc/services`. The port number of each domain must be the same and must be specified in the following format:

```
OpenTP1scd port-number/tcp alias-name
```

- `OpenTP1scd` is a constant.
- Specify `tcp` as the protocol name.
- Any alias name can be used.

Register the port number of the domain representative schedule service in the `/etc/services` files of all the hosts where OpenTP1 is activated. However, if operated by NIS, register the port number in `/etc/services` of only the NIS server.

If issuing a domain-specified `dc_rpc_call()` when no port number is registered in `/etc/services`, error is returned with `DCRPCER_NO_PORT`.

(3) System definitions

1. Schedule service definition

Define the port number of the domain representative schedule service by specifying it in the `scd_port` operand of the schedule service definition. As the port number, specify the port number registered in `/etc/services`.

2. System common definition

Set the nodes that configure the local domain. Specify the node names by the `all_node` operand in the system common definition.

(4) When there is more than one schedule service in the same host

When starting more than one schedule service (OpenTP1) in the same host for multiOpenTP1 operation or system switchover, the schedule services have different port numbers. If assigning more than one schedule service in the same host to the domain representative schedule service, it is impossible to perform domain communication between the specified schedule services by issuing a domain-specified `dc_rpc_call()`. To start more than one schedule service in the same host, assign only one of them to the domain representative schedule service. The following shows the required environment settings.

1. Registering the domain representative schedule service

Register only one schedule service in the domain data file as the domain representative schedule service.

2. Registering the port number

Only one port number in `/etc/services` is registered. Register the port number for the schedule service registered in the domain data file.

3. System definition

Specify the port number for the domain representative schedule service by the `scd_port` operand in the schedule service definition. This is same as the port number registered in `/etc/services`.

Specify the names of the nodes that configure the local domain by the `all_node` operand.

1.2.8 Changing common definition for the system

OpenTP1 must be re-cataloged if the common definition for the system is changed after cataloging OpenTP1 into OS. There are two ways of re-cataloging:

(1) When a superuser changes definition

After changing the definition, terminate OpenTP1 normally and then delete OpenTP1 from the OS. Then, execute the `dcsetup` command with its `-d` option specified to re-catalog OpenTP1 to apply the new definition.

(2) When the OpenTP1 administrator changes definition

The OpenTP1 administrator can change the system common definition by using the `dcreset` command after normal termination of OpenTP1. The system will go down if OpenTP1 is online and execution of the `dcreset` command is attempted.

1.3 Environment setup by OpenTP1 administrator

This section explains the environment setup that is performed by the OpenTP1 administrator in order to run OpenTP1. For details of the OpenTP1 operation commands, see *13. Details of Operation Commands*.

For the setup details that depend on the OS, see the manual for the appropriate OS.

1.3.1 Creation and checking of the system definition

The OpenTP1 administrator must create the system definition before the superuser catalogs OpenTP1 into the OS. By running the `dcdefchk` command after creating the system definition, you can make sure that no values have been specified incorrectly before starting OpenTP1. For details about the system definition and how check it, see the manual *OpenTP1 System Definition*.

(1) Checks performed by the `dcdefchk` command

The `dcdefchk` command performs the checks described below.

(a) Checking the syntax of the system definition

The syntax of the system definition is checked as follows:

- Whether valid characters are specified in the values for each operand
- Whether the values specified for each operand exceed the minimum or maximum specifiable values
- Whether invalid options are specified for any definition commands
- Whether valid characters are specified in the command and flag arguments for definition commands
- Whether the values specified for the command and flag arguments for definition commands exceed the minimum or maximum specifiable values

(b) Checking the logic of the system definition

The logic of the system definition is checked to determine whether any of the values specified in an operand or a definition command may cause problems with OpenTP1 operation. The logic is checked as follows:

- Whether any incompatibility exists between the values specified for different operands or definition commands
- Whether the values specified for operands or definition commands are the recommended values
- Whether the appropriate access permissions are in place to access the files and directories specified in operands and definition commands

(c) Checking OpenTP1 files

OpenTP1 files are checked to determine whether the correct files are specified in the system definition.

(2) Flow of checking with the `dcdefchk` command

The `dcdefchk` command performs checks in the following sequence:

1. Checks the syntax of operands.

The command checks the syntax of operands in the system definition files located in the definition directories. The check sequence is as follows:

1. System definition files under the directory specified by the `$DCCONFPATH` environment variable

2. User service definition files under the directory specified by the `$DCUAPCONFPATH` environment variable

2. Outputs the KFCA00258-I message.

A list of definition files checked for syntax in step 1 is output in a KFCA00258-I message. At this time, only the files that are checked as the system service definition are output. The files that are checked as the user service definition are not output.

3. Outputs the KFCA00254-R message.

When the `-r` option is specified in the `dcdefchk` command and an error is found during the syntax check in step 1, a KFCA00254-R message is output. The KFCA00254-R message asks you whether to continue or stop command processing.

4. Checks the syntax (and some logic aspects) of definition commands specified in the system definition (excluding the user service definition).

The command checks the syntax (and some aspects of logic) of definition commands in the system definition files located in the definition directories (the directories specified by the environment variables `$DCCONFPATH` and `$DCUAPCONFPATH`).

After each of the checks described in 4 to 7, the command outputs a message with one of the following indicating the message type labels. Note that some output messages may not be assigned a category.

Labels indicating message type:

ERROR: This type of message is output on detection of a problem that prevents OpenTP1 from starting or stopping, or impairs some aspects of OpenTP1 operation.

WARNG: This type of message is output when a specified value is not the

recommended value.

CHECK: This type of message is output to prompt the user to check the validity of a specified definition.

5. Checks the logic of system definitions (excluding the user service definition).

The command checks the logic of the definitions specified in the system definitions located in the definition directories.

6. Checks the syntax (and some logic aspects) of the definition commands specified in the user service definition).

The command checks the syntax (and some aspects of logic) of the definition commands specified in the user service definitions located in the definition directories.

7. Checks the logic of user service definitions.

The command checks the logic of the definitions specified in the user service definitions located in the definition directories.

The following table shows the order in which the various checks take place. The numbers in the table correspond to the steps listed above. Note that checking of OpenTP1 files takes place in steps 4. and 5. of the flow of checking with the `dcdefchk` command.

Table 1-2: Flow of checks performed by the `dcdefchk` command

Type	Operand (set format)	Definition commands (command format)	
		Other than user service definition	Use service definition
Syntax check	1.	4.	6.
Logic check	5. and 7.	4. and 5.	6. and 7.

(3) Cautionary notes

- You can execute the `dcdefchk` command while OpenTP1 is operating. If you do so, the command checks the values specified in the system definition before the command is executed, not the values that are valid for the current operation of OpenTP1. For example, if you use the `prcpath` command to change the value for `prcsvpath` specified in the process service definition, the `dcdefchk` command checks the original value specified in the process service definition.
- The `dcdefchk` command uses the directory specified in the `$DCDIR` environment variable as the OpenTP1 directory. The `$DCDIR` environment variable is set in the environment where the `dcdefchk` command is to be executed. Therefore, if the directory specified in the `$DCDIR` environment variable differs from the path

name for the OpenTP1 directory set in the environment where the `dstart` command is to be executed, the `dcdefchk` command cannot correctly check definitions.

- If the directory paths specified in `$DCCONFPATH` and `$DCUAPCONFPATH` are longer than 50 bytes, only part of a definition file name is output in the message (KFCA00242-E) that is output if an error is found during the syntax check. To avoid this, copy the target definition file to a directory whose path name is 50 bytes or less, specify the copy destination directory in the `DCCONFPATH` environment variable that is specified in `putenv` format in `$DCDIR/conf/env`, and execute the `dcdefchk` command.
- In the directory for storing definitions, store only the files for which you want to perform a definition check. If you store files other than system definition files and user service definition files under the directories for storing system definitions (`$DCCONFPATH` and `$DCUAPCONFPATH`), the `dcdefchk` command cannot correctly check the definitions. This precaution does not include the directories for storing domain definition files and domain definition files.
- For files other than system service definition files under the directories for storing system definitions (`$DCCONFPATH` and `$DCUAPCONFPATH`), the `dcdefchk` command does not check files whose names begin with periods (.) and underscores (_) and names that are 9 or more bytes.
- The `dcdefchk` command checks only the definitions of the operands that are listed in the manual *OpenTP1 System Definition*.
- Do not execute the command for checking definitions (`dcdefchk`) at the same time as a command for creating OpenTP1 files (such as `jnlinit`).
- If a syntax check detects a problem with a value specified for an operand, the logic check will proceed as if the default value is specified for that operand.
- The `dcdefchk` command checks file and directory access permissions based on the UID/GID of the user who executed the command.
- A message may be output during a logic check if the OpenTP1 file system is inaccessible for some reason, such as being in a standby state.
- Some of the messages output by the `dcdefchk` command during the checking process do not belong to any of the message types associated with the `dcdefchk` command. For this reason, if you run the `dcdefchk` command with the `-e` option specified, messages with the message type `E` may still be output.
- The `dcdefchk` command may be unable to correctly check the logic of environment variables (definitions in `putenv` or `dputenv` format).

1.3.2 OpenTP1 administrator environment setup

The following environment variables must be defined in the login environment in

order to execute OpenTP1 commands:

DCDIR

Specifies the full path name of the OpenTP1 directory. The directory name to be specified in `DCDIR` must be 50 bytes or less. Do not use a symbolic link.

DCCONFPATH

Specifies the full path name of the directory for storing OpenTP1 definition files using 246 bytes or less.

DCUAPCONFPATH

Specifies the full path name of a directory for storing OpenTP1 user service definition files, using 246 bytes or less, if you want to store the files in a directory different from the directory set in the `DCCONFPATH` environment variable.

PATH

Adds `$DCDIR/bin` to `PATH`.

1.3.3 Reservation of resources for OpenTP1 internal control

The `dcmakeup` command must be used in order to reserve the OS resources that OpenTP1 uses to perform its internal control. When the `dcmakeup` command is executed, as many resources as necessary are reserved and stored under the OpenTP1 directory. How many resources to be reserved is analyzed on the basis of the system definition information under the OpenTP1 directory.

If the value of the `prc_process_count` operand in the process service definition is changed, the `dcsetup` command must be executed. Before executing the `dcstart` command, the `dcmakeup` command must be executed.

If the `dcmakeup` command is not executed before the `dcstart` command, the process by this command will be run when OpenTP1 begins but it may take much time to reserve resources many enough to perform complete internal control.

1.3.4 Initialization of OpenTP1 file systems

The OpenTP1 administrator must initialize files (character special files or ordinary files) with the `filmkfs` command so that the files can be used as OpenTP1 file systems.

1.3.5 Creation of OpenTP1 files

The OpenTP1 files listed in the following table must be created and initialized in the OpenTP1 file systems that have been initialized by the `filmkfs` command.

Table 1-3: OpenTP1 files and operation commands for their creation and initialization

OpenTP1 files		Operation command for initialization	OpenTP1 file system area for OpenTP1 file creation	Remarks
Status files		<code>stsinit</code>	OpenTP1 file system area for system	Required in a node
Journal files	System journal files	<code>jnlinit</code>		
	Checkpoint dump files			
XAR files		<code>xarinit</code>	Optional ^{#1}	
Message queue files		<code>queinit</code>	OpenTP1 file system area for users	Optional ^{#2}
TP1/Message Queue queue file		<code>mqainit</code>		Optional ^{#3}
DAM files		<code>damload</code>		Optional
TAM files		<code>tamcre</code>		Optional

#1

Required when the XA resource service is used.

#2

Required when TP1/Message Control is used.

#3

Required when TP1/Message Queue is used. For details on the operation command, see the manual *TP1/Message Queue User's Guide*.

Status files, system journal files, checkpoint dump files, and XAR files must be created in the OpenTP1 file system area for the system.

The following files must be created in the OpenTP1 file system area for users:

- Message queue file
- TP1/Message Queue queue file
- DAM file
- TAM file

For ISAM files, see the manual *Indexed Sequential Access Method ISAM*.

1.3.6 Creation of non-OpenTP1 files

In addition to the OpenTP1 files, the OS files listed below are required in order to run OpenTP1.

(1) Files created by users

The following files must be created by users:

- User program files
Such files store execution form UAPs.
- MCF communication process program file and MCF application startup process program file
These files store the executable programs of the MCF communication process and the MCF application startup process that you create to use the message control facility.
- Various definition files
Such files store various types of OpenTP1 definition information. The OS text editor must be used to create a definition file as a text file.

The files list above are created under user-created directories. The `$DCDIR/aplib` and `$DCDIR/conf` directories are created during OpenTP1 installation.

The following table lists the files and directories that are created by users.

Table 1-4: User-created files and directories

Name		Directory	File name	File type
User program file		<code>\$DCDIR/aplib/#1</code>	Name of execution form program ^{#2}	Execution form file
MCF communication process program file and MCF application startup process program file		<code>\$DCDIR/lib/servers/</code>	Name of the executable program	Executable file
Definition files	System environment definition	<code>\$DCDIR/conf/</code>	env	Text file
	Definitions other than system environment definition	<code>\$DCCONFPATH/</code>	Name of definition file	Text file
	System service information definition file	<code>\$DCDIR/lib/sysconf/</code>	Name of system service information definition file	Text file

1. Environment Setup

	Name	Directory	File name	File type
	System service common information definition ^{#3}	\$DCDIR/lib/sysconf/	mcf	Text file

#1

Can be changed by the process service definition.

#2

Can be changed by the user service definition.

#3

This file is created when OpenTP1 is installed or registered into the OS. However, you need to change the contents of the definition depending on the operation environment.

(2) Files and directories created at installation or cataloging into OS

The files listed below must be created during installation.

- OpenTP1 program files

These files store OpenTP1 programs.

The two types of files are OpenTP1 execution form files and the files used to create UAPs.

- OpenTP1 execution form files
System services and commands
- Files used to create UAPs
Header files and archive files

- Definition analysis file

This file is used internally by OpenTP1 to analyze definitions.

- Message object file

This file stores message texts.

The files listed above are created under directories created by OpenTP1.

The directories that are created when OpenTP1 is installed or cataloged into the OS are:

- User program file directory

This directory stores user program files.

- Definition file directory
This directory stores various OpenTP1 definition information files.
- System management information directory
This directory stores system management information.
- Core file directory
This directory stores core files.
- Command log directory
This directory stores command log files.
- Deadlock information file directory
This directory stores deadlock information and timeout information files.
- Transaction information directory
This directory stores transaction information.
- Transaction journal directory
This directory stores transaction journal information.
- Process service information directory
This directory stores process service information.
- Journal information directory
This directory stores journal information.
- Server recovery journal directory
This directory stores server recovery journal information.
- Directory for storing transaction control objects
This directory stores transaction control object files.
- Directory for online testers
This directory stores files that are used by the online tester.
- Directory for system internal lock control
This directory stores information for OpenTP1 internal lock control.
- Directory for system internal synchronization control
This directory stores information for OpenTP1 internal synchronization control.

The following table lists the files and directories that are created when OpenTP1 is installed or cataloged into the OS.

Table 1-5: Files and directories created at installation or cataloging into OS

Name		Directory	File name
OpenTP1 program file	OpenTP1 servers	\$DCDIR/lib/servers/	..#
	Commands	\$DCDIR/bin/	--
	Header file	\$DCDIR/include/	--
	Archive file	\$DCDIR/lib/	--
Definition analysis file		\$DCDIR/lib/sysconf/	..#
		\$DCDIR/lib/sysdef/	--
Message object file		\$DCDIR/lib/	emsgtxt, jmsgtxt
User program file directory		\$DCDIR/aplib/	None
Definition file directory		\$DCDIR/conf/	None
System management information directory		\$DCDIR/etc	--
Core file directory		\$DCDIR/spool/save/	None
Command log directory		\$DCDIR/spool/cmdlog/	--
Deadlock information file directory		\$DCDIR/spool/dclckinf/	None
Transaction information directory		\$DCDIR/spool/dctrninf/	None
Journal information directory		\$DCDIR/spool/dcjnlinf/errinf/	None
Process service information directory		\$DCDIR/spool/dcprcinf/	None
Transaction journal directory		\$DCDIR/spool/dctjlinf/	--
Server recovery journal directory		\$DCDIR/spool/dcsjl/	--
Directory for storing transaction control objects		\$DCDIR/spool/trnrmcmd/userobj/	None
Directory for online testers		\$DCDIR/spool/uto/	None
Directory for system internal lock control		\$DCDIR/spool/olkfifs/	--
Directory for system internal synchronization control		\$DCDIR/spool/olkrfsf/	--

Legend:

--: Need not be specified by the user.

#

The file needs to be created when you use the MCF.

(3) Files created during OpenTP1 execution

The files listed below are created dynamically by OpenTP1 during online execution.

(a) Files created during normal operation

The files listed below are created during normal OpenTP1 operation.

- Message log file
This file stores system messages issued by OpenTP1.
- MCF trace file
This file stores TP1/Message Control trace information.
- Scheduling queue information file
This file is used internally by OpenTP1 to store scheduling queue information.
- RPC trace file
This file stores RPC trace information.
- Trace information dump file
This file stores OpenTP1 internal trace information.
- Performance verification trace information file
The file stores trace information for verification.
- Backup file for performance verification trace information file
This file serves as a backup of the performance verification trace information file.
- XAR performance verification trace information file
This file stores trace information for events linked to transactions that use the XA resource service (transaction requests from application servers and transaction processing by OpenTP1).
- Backup file for XAR performance verification trace information file
This file serves as a backup of the XAR performance verification trace information file.
- JNL performance verification trace information file
This file stores trace information for the journal service.
- Backup file for JNL performance verification trace information file

- This file serves as a backup of the JNL performance verification trace information file.
- LCK performance verification trace information file
This file stores trace information for various lock events that use the lock service.
- Backup file for LCK performance verification trace information file
This file serves as a backup of the LCK performance verification trace information file.
- MCF performance verification trace information file
This file stores trace information for MCF performance verification.
- Backup file for MCF performance verification trace information file
This file serves as a backup of the MCF performance verification trace information file.
- TRN event trace information file
This file stores traces of various events of the XA functions and transaction services that are called in transaction branches (transaction management service, transaction recovery service, and resource manager monitoring service).
- Backup file for TRN event trace information file
This file serves as a backup of the TRN event trace information file.
- NAM event trace information file
This file stores trace information for events relating to the name service, including communication invoked by the name service and the registration and deletion of service information from the cache.
- Backup file for NAM event trace information file
This file serves as a backup of the NAM event trace information file.
- Process service event trace information file
This file stores process service trace information.
- Backup file for process service event trace information file
This file serves as a backup of the process service event trace information file.
- FIL event trace information file
This file stores event information for when an OpenTP1 file access request required more processing time than the specified value.
- Backup file for FIL event trace information file

This file serves as a backup of the FIL event trace information file.

- RTS log file

The file stores real-time statistics.

- MCF statistics file

This file stores MCF statistics.

- UAP trace data file

This file stores UAP trace information. It is created only when Y is specified in the `uap_trace_file_put` operand. You specify the `uap_trace_file_put` operand in the following definitions:

- System common definition
- User service default definition
- User service definition
- Backup file for UAP trace data file

This file serves as a backup file of UAP trace data file.

The current working directory for the system service and user server is also created during OpenTP1 execution.

During execution, OpenTP1 creates files in directories it has created. The following table lists the files and directories created in the course of OpenTP1 operation.

Table 1-6: Files and directories created during normal operation

Name	Directory	File name
Message log file	\$DCDIR/spool/	dclog1 and dclog2
MCF trace file ^{#1}	\$DCDIR/spool/	mcf t AXXZZ ^{#2}
Scheduling queue information file	\$DCDIR/spool/	scdqid1 and scdqid2
RPC trace file	\$DCDIR/spool/	rpctr1 and rpctr2 ^{#3}
Trace information dump file	\$DCDIR/spool/save/	dcmtrdp1 and dcmtrdp2
Performance verification trace information file	\$DCDIR/spool/dcprfinf/	prf_ <i>nnn</i> ^{#4}
Backup file for performance verification trace information file	\$DCDIR/spool/save/	prf_ <i>nnn</i> .bk1 ^{#5} , and prf_ <i>nnn</i> .bk2 ^{#5}

1. Environment Setup

Name	Directory	File name
XAR performance verification trace information file	\$DCDIR/spool/dcxarinf/	_xr_nnn#4
Backup file for XAR performance verification trace information file	\$DCDIR/spool/save/	_xr_nnn.bk1#5, and _xr_nnn.bk2#5
JNL performance verification trace information file	\$DCDIR/spool/dcjnlinf/ prfinf/	_jl_nnn#4
Backup file for JNL performance verification trace information file	\$DCDIR/spool/save/	_jl_nnn.bk1#5, and _jl_nnn.bk2#5
LCK Performance verification trace information file	\$DCDIR/spool/dclckinf/prf/	_lk_nnn#4
Backup file for LCK performance verification trace information file	\$DCDIR/spool/save/	_lk_nnn.bk1#5, and _lk_nnn.bk2#5
MCF performance verification trace information file	\$DCDIR/spool/dcmcfinf/	_mc_nnn#4
Backup file for MCF performance verification trace information file	\$DCDIR/spool/save/	_mc_nnn.bk1#5, and _mc_nnn.bk2#5
TRN event trace information file	\$DCDIR/spool/dctrninf/trace/ prf/	_tr_nnn#4
Backup file for TRN event trace information file	\$DCDIR/spool/save/	_tr_nnn.bk1#5, and _tr_nnn.bk2#5
NAM event trace information file	\$DCDIR/spool/dcnaminf/	_nm_001, _nm_002, and _nm_003
Backup file for NAM event trace information file	\$DCDIR/spool/save/	_nm_nnn.bk1#5, and _nm_nnn.bk2#5
Process service event trace information file	\$DCDIR/spool/dcprcinf/	_pr_001, _pr_002, and _pr_003
Backup file for process service event trace information file	\$DCDIR/spool/save/	_pr_nnn.bk1#5, and _pr_nnn.bk2#5
FIL event trace information file	\$DCDIR/spool/dfilinf/	_fl_001, _fl_002, and _fl_003
Backup file for FIL event trace information file	\$DCDIR/spool/save/	_fl_nnn.bk1#5, and _fl_nnn.bk2#5

Name	Directory	File name
RTS log file	\$DCDIR/spool/dcrtsinf/#6	rtslog[1] to rtslog[10] #6, #7
MCF statistics file	\$DCDIR/spool/#8	mcfstc #8
Current working directory #9	\$DCDIR/tmp/home/ server-name.ID #10/	--
UAP trace data file	\$DCDIR/tmp/home/ server-name.ID	dcuat.map
Backup file for UAP trace data file	\$DCDIR/spool/save/trc/#11	server-namen.uatmap #12, and server-name_n.uatmap #12

Legend:

--: Need not be specified by the user.

#1

Required only when TP1/Message Control is used.

#2

AXXZZ:

AXX: MCF identifier:

A: MCF manager process identifier specified in the *id* operand of *mcfmenv -m* in the MCF manager definition

XX: MCF communication process identifier specified in *mcfteenv -s* in the MCF communication configuration definition

ZZ: Trace swap file identifier.

#3

The file name can be changed in the system common definition.

#4

Value ranging from 001 to the value specified in the *prf_file_count* operand in the following definition:

- Performance verification trace: Performance verification trace definition
- XAR performance verification trace: XAR performance verification trace definition
- JNL performance verification trace: JNL performance verification trace definition

1. Environment Setup

- LCK performance verification trace: LCK performance verification trace definition
- MCF performance verification trace: MCF performance verification trace definition
- TRN event trace: TRN event trace definition

#5

nnn: Value corresponding to the name of the backup source file.

#6

If you do not specify the `rts_log_file_name` operand in the real-time statistics service definition, RTS log files are created in this directory.

#7

The number of created files is the number specified in the `rts_log_file_count` operand in the real-time statistics service definition.

#8

The `mcfstc` file is created in the `$DCDIR/spool/` directory if you do not specify the output file name in the operation command (`mcfstats`).

#9

You can change the current working directory by setting the `prc_current_work_path` operand in the system common definition.

#10

For user environment setting commands, `_usrcmd` is displayed as the server name.

"ID" is a serial number not less than 1. ID is not added to operation commands (`dcstart`, `dcstop`, `dcsvstart`, and `dcsvstop`) and user environment setting commands.

#11

If the `prc_coresave_path` operand is specified in the process service definition, the directory will be `prc_coresave_path-value/trc/`.

#12

n: File serial number (1-3).

(b) Files created in the event of an error

The files listed below are created when errors occur.

- Shared memory dump file

This file stores the contents of the shared memory output by OpenTP1.

- Core file
This file saves the core file of an abnormally terminated process.
- Deadlock and timeout information file
This file stores deadlock and timeout information.
Delete unnecessary files.
- MCF dump file
This file stores a dump of TP1/Message Control.
- MCF shared memory dump file
This file stores the contents of the shared memory pool that were output by TP1/Message Control when errors occurred in the system service associated with sending and receiving messages.
- Undecided-transaction information file
This file stores undecided transaction information in the event of an error.
Delete unnecessary files.
- Incorrect journal information file
This file stores incorrect data contained in an incorrect journal detected during reading of journals.
- Copy file of input/output queue contents
This file stores the contents of input/output queues when the mcftdmpqu command (copy contents of input/output queues) is executed.
- UAP trace edit/output file
The file stores UAP trace information that is edited and output automatically when a UAP terminated abnormally.
- OpenTP1 debug information file
The file stores OpenTP1 debug information when a UAP terminated abnormally.

All files created during OpenTP1 execution are placed under an OpenTP1-created directory. The following table lists the files and directories that are created when an error occurs.

Table 1-7: Files and directories created when an error occurs

Name	Directory	File name
Shared memory dump file	\$DCDIR/spool/	Linux: shmdump[1 to 3].gz and shmdump.XXX ^{#1} [1 to 3].gz Other operating systems: shmdump[1 to 3].Z and shmdump.XXX ^{#1} [1 to 3].Z
Core file	\$DCDIR/spool/save/	<i>server-namen</i> ^{#2}
Deadlock and timeout information file	\$DCDIR/spool/ dclckinf/	<i>file-name</i> ^{#3}
MCF dump file ^{#4}	\$DCDIR/spool/	mcf dKAXXNN ^{#5}
MCF shared memory dump file ^{#4}	\$DCDIR/spool/	mcf sAXXY ^{#6}
Undecided-transaction information file	\$DCDIR/spool/ dctrninf/	<i>file-name</i> ^{#7}
Incorrect journal information file	\$DCDIR/spool/ dcjnlinf/ errinf/	<i>r_n</i> ^{#8}
Copy file of input/output queue contents	\$DCDIR/spool/	<i>file-name</i> ^{#9}
UAP trace edit output file	\$DCDIR/spool/save	<i>server-namen</i> ^{#10} .uat
OpenTP1 debug information file	\$DCDIR/spool/save	<i>server-namen</i> ^{#10} .deb

#1

XXX: Resource manager name (dam, tam, ist, ism).

#2

n

Core file serial number (1-3).

No serial number is assigned to a core file that is output when OpenTP1 is forced to terminate (that is, when the `dcsvstop -df` command or `prkill` command is executed or real time monitoring is timed out). A process service core file may be acquired in the directory under the name *core*. If a user environment setup command terminates abnormally, the core file is saved under the name `_usrcmd` followed by a sequential number (1-3).

#3

The file name is determined on the basis of the deadlock detection date and time. The length of the file name depends on whether the date is one digit or two digits.

Example

6: 29: 56 on October 3: Oct3062956

18: 06: 00 on October 10: Oct10180600

#4

Required only when TP1/Message Control is used.

#5

KAXXNN:

K: Process type.

AXX: MCF identifier:

A: MCF manager process identifier specified in the *id* operand of *mcfmenv -m* in the MCF manager definition

XX: MCF communication process identifier specified in *mcfteenv -s* in the MCF communication configuration definition

NN: Dump serial number (01-99).

#6

AXXY:

AXX: MCF identifier:

A: MCF manager process identifier specified in the *id* operand of *mcfmenv -m* in the MCF manager definition

XX: MCF communication process identifier specified in *mcfteenv -s* in the MCF communication configuration definition

Y: File serial number (1-3).

#7

The file name is *rl* + transaction service start time (eight hexadecimal digits).

#8

Generation number of a journal in which incorrect data was detected (hexadecimal)

#9

File name specified by the *mcfteumpqu* command (copy file of input/output queue contents).

#10

n: File serial number (1-3).

(c) Files for OpenTP1 internal processing

In addition to the files listed in (a) and (b) above, files for OpenTP1 internal processing are placed under `$DCDIR/spool/` and `$DCDIR/tmp/`. Users must not change these files without authorization. All files placed under `$DCDIR/tmp/` for internal processing are cleared when OpenTP1 is started.

(d) OS files that OpenTP1 uses

OpenTP1 uses `/dev/console` and `/dev/null` for standard input and output. Do not change the permissions for `/dev/console` and `/dev/null`. The operation may not be assured if the permissions are changed.

1.3.7 Cataloging of resource manager

XA interface object files provided by resource managers have not been linked with a transaction service control executable-form program and a client service executable-form program that are used to install an OpenTP1 program product. When the `dcsetup` command is executed, the system determines the OpenTP1 program product installed and automatically links XA interface object files of a resource manager (DAM, TAM, MCF, ISAM, and MQA) provided by OpenTP1. To execute a transaction using a non-OpenTP1 resource manager under OpenTP1, execute the `dcsetup` command and then execute the `trlnkrm` command to catalog the non-OpenTP1 resource manager before starting OpenTP1. Executing the `trlnkrm` command re-creates a transaction service control execution-form program and a client service executable-form program and a standard transaction control object file. For information needed when non-Hitachi databases are used, see the manual *OpenTP1 Programming Guide*.

1.3.8 Creation of transaction control object files

In order for a UAP running under OpenTP1 to access a resource manager within a transaction, a transaction control object file must be linked with that UAP.

For a UAP that accesses all resource managers registered in OpenTP1, the standard transaction control object file named `dc_trn_allrm.o` (a file under `$DCDIR/spool/trnrmcmd/userobj` that is created when the `dcsetup` command is executed, and re-created when the `trlnkrm` command is executed) must be linked with that UAP.

For a UAP that does not access all of the source managers registered in OpenTP1, a transaction control object file must be created by executing the `trnmkobj` command and linked with that UAP.

For a UAP that does not access a resource manager within a transaction, no transaction control object file needs to be linked with that UAP. However, if the `trlnkrm`

command is executed to change the resource manager registration status, the UAP with which a standard transaction control object file has been linked must be linked again.

Note that linking the same resource manager with all UAPs that make up a global transaction optimizes the commit process (by suppressing process-to-process communication) to enhance transaction performance.

Before starting OpenTP1, the transaction control object file created by the `trnmkobj` command and the XA interface object file provided by a resource manager must be linked with the UAP.

1.3.9 Changing system common definition

The OpenTP1 administrator can use the `dcreset` command to change the system common definition. This command applies the new definition to OpenTP1. Execute the `dcreset` command after normal termination of OpenTP1. The system will go down if OpenTP1 is online and execution of this command is attempted.

When using the `dcreset` command, there is no need to execute the `dcsetup -d` and `dcsetup` commands to delete and catalog the definition. A superuser who changed the system common definition must execute these commands to reflect the change into OpenTP1.

1.4 Preparations for TP1/Message Control execution

The OpenTP1 process service starts TP1/Message Control. To start the communication service of TP1/Message Control, the user needs to code the MCF main function.

This section describes what the user needs to prepare to execute TP1/Message Control. Note that TP1/Message Control will be referred to as MCF hereafter.

1.4.1 Creation of the MCF main function of the MCF communication service

To use the message transmission facility provided by a protocol, you must code, compile, and link the MCF main function of the MCF communication service.

Issue the start function (`dc_mcf_svstart`) from the MCF main function.

When you use a user exit routine, specify the function address of the desired user exit routine in the MCF main function. Create user exit routines using the same language that is used to create the MCF main function (K&R C, ANSI C, or C++).

When you use the TP1/NET/XMAP3 protocol, you need to start the mapping service that is used in the user exit routine beforehand in the MCF main function. For details about starting the mapping service, see the manual *OpenTP1 Protocol TP1/NET/XMAP3*.

Figures 1-4 and 1-5 give an overview of how to code the MCF main function that defines a protocol and user exit routines.

Figure 1-4: Overview of how to code the MCF main function that defines a protocol and user exit routines (for ANSI C and C++)

```

#include <dcmxxxx.h>      /* Header file of protocol XXXX          */ 1.
#include <dcmcfuoc.h>

extern long msgrcv01(dcmcf_uoc_min_n *);
                        /* Declaring extern for the function of the user exit routine for editing input messages */ 2.
extern long msgsend01(dcmcf_uoc_mout_n *);
                        /* Declaring extern for the function of the user exit routine for editing output messages */ 2.
extern dcmcf_uoc_t      dcmcf_uoctbl;
                        /* Declaring extern for the user exit routine table          */ 3.

main()
{
dcmcf_uoctbl.msgrcv = (dcmcf_uocfunc)msgrcv01;
                        /* Setting the address of the user exit routine for editing input messages          */ 4.
dcmcf_uoctbl.msgsend = (dcmcf_uocfunc)msgsend01;
                        /* Setting the address of the user exit routine for editing output messages          */ 4.
dc_mcf_svstart();     /* Issuing the start function                */ 5.
}

```


1. Include the header file provided by the protocol.
XXXX of `dcmXXXX.h` changes depending on the protocol. See the manual for the protocol.
2. Declare extern for the functions of the user exit routines to be used. The return values of the user exit routines must be the long type. You only need to perform this coding if you use user exit routines.
3. Declare extern for the user exit routine table.
 You only need to perform this coding if you use the user exit routines specified in step 2.
4. Set the function address of each user exit routine in the following variables provided by the system:

```

dcmcf_uoctbl.msgrcv /* Address of the user exit
                    routine for editing input
                    messages */
dcmcf_uoctbl.msgsend /* Address of the user exit
                     routine for editing output
                     messages */

```

You only need to perform this coding if you use those user exit routines.

5. Issue the start function. This coding is always required.

Figure 1-5: Overview of how to code the MCF main function that defines a protocol and user exit routines (for K&R C)

<code>#include <dcmxxxx.h></code>	<code>/* Header file of protocol XXXX</code>	<code>*/</code>	1.
<code>extern long msgrcv01();</code>	<code>/* Declaring extern for the function of the user exit routine for editing input messages</code>	<code>*/</code>	2.
<code>extern long msgsend01();</code>	<code>/* Declaring extern for the function of the user exit routine for editing output messages</code>	<code>*/</code>	2.
<code>extern dcmcf_uoc_t dcmcf_uoctbl;</code>	<code>/* Declaring extern for the user exit routine table</code>	<code>*/</code>	3.
<code>main()</code>			
<code>{</code>			
<code> dcmcf_uoctbl.msgrcv = msgrcv01;</code>	<code>/* Setting the address of the user exit routine for editing input messages</code>	<code>*/</code>	4.
<code> dcmcf_uoctbl.msgsend = msgsend01;</code>	<code>/* Setting the address of the user exit routine for editing output messages</code>	<code>*/</code>	4.
<code> dc_mcf_svstart();</code>	<code>/* Issuing the start function</code>	<code>*/</code>	5.
<code>}</code>			

1. Include the header file provided by the protocol.
XXXX of `dcmXXXX.h` changes depending on the protocol. See the manual for the protocol.

1. Environment Setup

2. Declare extern for the functions of the user exit routines to be used. The return values of the user exit routines must be the long type. You only need to perform this coding if you use user exit routines.
3. Declare extern for the user exit routine table.
You only need to perform this coding if you use the user exit routines specified in step 2.
4. Set the function address of each user exit routine in the following variables provided by the system:

```
dcmcf_uoctbl.msgrcv /* Address of the user exit
                    routine for editing input
                    messages */
dcmcf_uoctbl.msgsend /* Address of the user exit
                     routine for editing output
                     messages */
```

You only need to perform this coding if you use those user exit routines.

5. Issue the start function. This coding is always required.

1.4.2 Creation of the MCF main function of the application startup service

To use the application startup facility and the MHP for processing MCF events, code, compile, and link the MCF main function of the application startup service in addition to the MCF main function of the protocol.

Issue the start function (`dc_mcf_svstart`) from the MCF main function.

To use the user exit routine for determining the inheritance of the timer startup, specify the address of the user exit routine for determining the inheritance of the timer startup in the main function of the application startup service. Create the user exit routine for determining the inheritance of the timer startup using the same language that is used to create the MCF main function (K&R C, ANSI C, or C++). For details about the user exit routine for determining the inheritance of the timer startup, see the manual *OpenTPI Programming Reference C Language*.

Figures 1-6 and 1-7 give an overview of how to code the MCF main function of the application startup service.

Figure 1-6: Overview of how to code the MCF main function of the application startup service (for ANSI C and C++)

```

#include <dcmpsvr.h> /* Header file of the application startup service */ 1.
#include <dcmpsv.h>

extern long rrntime01(dcmpsv_uoc_rtime *); /* Declaring extern for the function of the user exit routine
for determining the inheritance of the timer startup */ 2.

extern dcmcf_uoc_t dcmcf_uoctbl; /* Declaring extern for the user exit routine table */ 3.

main()
{
    dcmcf_uoctbl.rrntime = (dcmcf_uocfunc)rrntime01;
/* Declaring the address of the user exit routine for
determining the inheritance of the timer startup */ 4.

    dc_mcf_svstart(); /* Issuing the start function */ 5.
}

```

1. Include the header file provided by the application startup service.
2. Declare extern for the function of the user exit routine to be used. The return values of the user exit routine must be the long type. You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.
3. Declare extern for the user exit routine table. You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.
4. Set the function address of the user exit routine for determining the inheritance of the timer startup in the following variable provided by the system:

```

dcmcf_uoctbl.rrntime /* Address of the user exit
routine for determining the
inheritance of the timer
startup */

```

You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.

5. Issue the start function. This coding is always required.

Figure 1-7: Overview of how to code the MCF main function of the application startup service (for K&R C)

<code>#include <dcmpsvr.h></code>	<code>/* Header file of the application startup service</code>	<code>*/</code>	1.
<code>extern long rrntime01();</code>	<code>/* Declaring extern for the function of the user exit routine for determining the inheritance of the timer startup</code>	<code>*/</code>	2.
<code>extern dcmcf_uoc_t dcmcf_uoctbl;</code>	<code>/* Declaring extern for the user exit routine table</code>	<code>*/</code>	3.
<code>main()</code> <code>{</code>			
<code>dcmcf_uoctbl.rrntime = rrntime01;</code>	<code>/* Declaring the address of the user exit routine for determining the inheritance of the timer startup</code>	<code>*/</code>	4.
<code>dc_mcf_svstart();</code>	<code>/* Issuing the start function</code>	<code>*/</code>	5.
<code>}</code>			

1. Include the header file provided by the application startup service.
2. Declare extern for the function of the user exit routine to be used. The return values of the user exit routine must be the long type. You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.
3. Declare extern for the user exit routine table. You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.
4. Set the function address of the user exit routine for determining the inheritance of the timer startup in the following variable provided by the system:

<code>dcmcf_uoctbl.rrntime</code>	<code>/* Address of the user exit routine for determining the inheritance of the timer startup */</code>
-----------------------------------	--

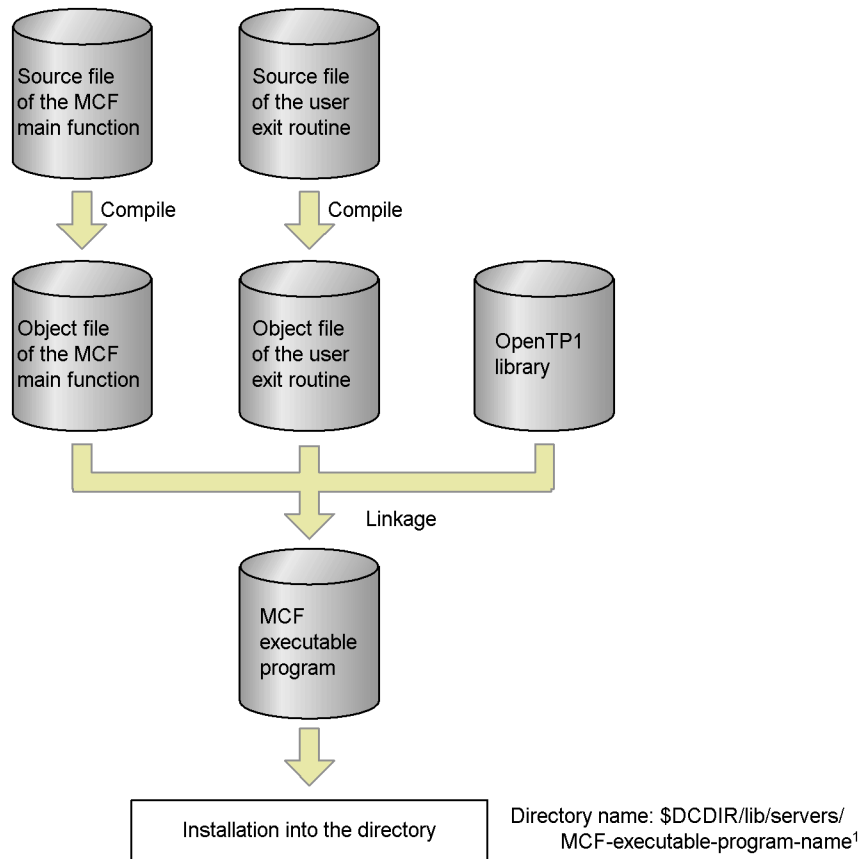
You only need to perform this coding if you use the user exit routine for determining the inheritance of the timer startup.

5. Issue the start function. This coding is always required.

1.4.3 Installation of the MCF main function into the directory

The following gives an overview of how to install the MCF main function into the directory.

Figure 1-8: Overview of how to install the MCF main function into the directory



¹ The name of the MCF executable program must consist of 8 characters beginning with mcfu.

1.4.4 Cataloging of MCF service name

The MCF service name must be defined in the system service configuration definition in order to run MCF, and must be the same as the name of the MCF manager definition object file.

1.4.5 Creation of system service information definition file

A system service information definition file must be created using the OS text editor. The path name for the file must be `$DCDIR/lib/sysconf/system-service-information-definition-file-name`.

The following shows the definition format:

```
set module="MCF-execution-form-program-name"  
set fixpriority=52
```

`module`

Specifies the MCF execution-form program name for the user-created MCF communication service or application startup service. The MCF execution-form program name must be specified as 1-8 characters beginning with `mcfu`.

`fixpriority`

Specifies 52 as OS fixed process execution priority.

1.4.6 Creation of definition object files

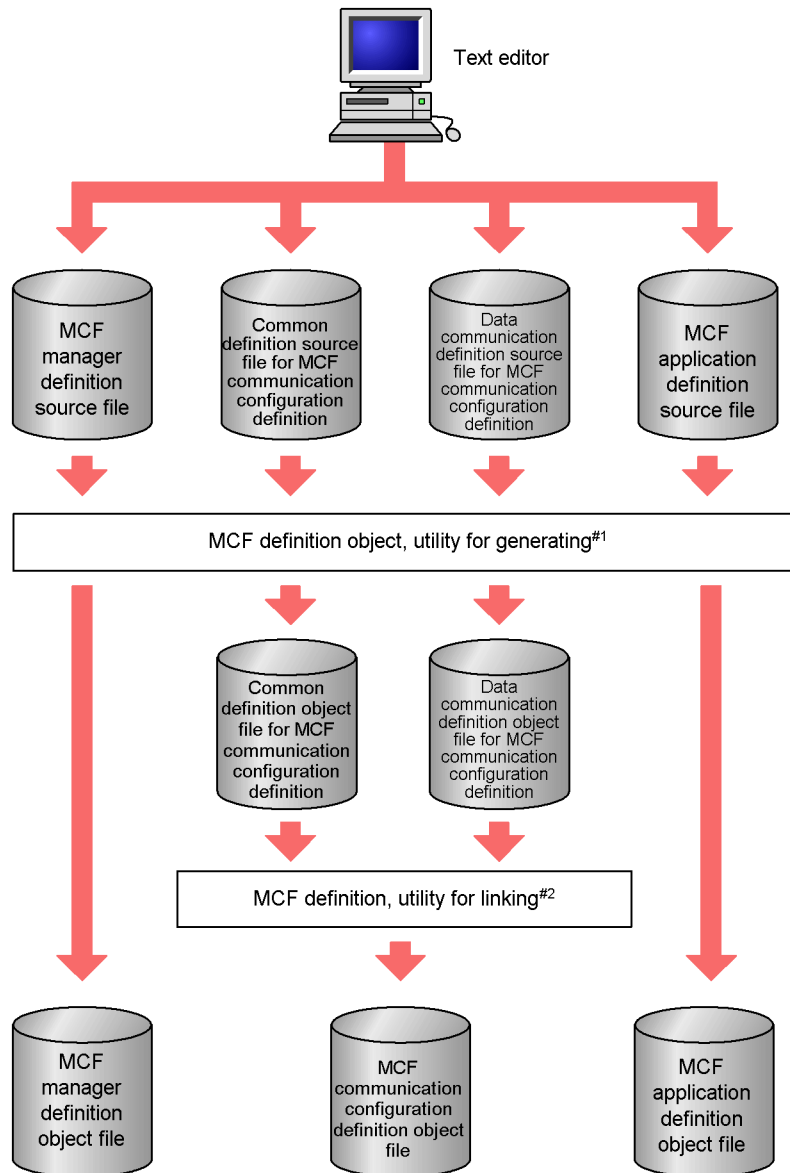
The procedure for creating definition object files is described below.

Definition object files must not be changed between startup and a restart. If they are changed, the restart operation cannot be guaranteed.

1. Use the OS text editor to create the following definition source files from the MCF definition files:
 - MCF manager definition source file
 - Common definition source file for MCF communication configuration definition
 - Data communication definition source file for MCF communication configuration definition
 - MCF application definition source file
2. Use the MCF definition object generation utility to generate the following object files from the definition source files:
 - MCF manager definition object file
 - Common definition object file for MCF communication configuration definition
 - Data communication definition object file for MCF communication configuration definition
 - MCF application definition object file
3. Use the definition linkage utility to link the common definition object file and the data communication definition object file for the MCF communication configuration definition.

The following shows an overview of the definition object file creation procedure.

Figure 1-9: Overview of definition object file creation procedure



#1

The following command is used:

```
mcfXXXX Δ -i Δ [path-name] input-file-name Δ -o Δ [path-name] output-object-file-name
```

mcfXXXX depends on the source file:

mcfmng

MCF manager definition source file

mcfcom

MCF communication configuration definition source file

mcfTCP

TP1/NET/TCP/IP protocol-specific definition source file for MCF communication configuration definition

mcfostp

TP1/NET/OSI-TP protocol-specific definition source file for MCF communication configuration definition

mcfxp

TP1/NET/XMAP3 protocol-specific definition source file for MCF communication configuration definition

mcfhna560

TP1/NET/HNA-560/20 protocol-specified definition source file for MCF communication configuration definition

mcfapli

MCF application definition source file

mcfhdlcs

TP1/NET/HDLC protocol-specific definition source file for MCF communication configuration definition

mcfhnanf

TP1/NET/HNA-NIF protocol-specific definition source file for MCF communication configuration definition

mcfx25

TP1/NET/X25 protocol-specific definition source file for MCF communication configuration definition

#2

The following command is used to link two object files for the MCF

communication configuration definition:

```
mcf1ink Δ -i Δ common-definition-object-file-name Δ data-communication-definition-object-file-name Δ -o Δ  
output-object-file-name
```

1.4.7 Preparation for acquiring command logs

Before you can acquire logs for MCF-related operation commands, you need to set the relevant environment variables. For details about setting environment variables, see *3.6.2(2) Acquiring command logs*.

Chapter

2. OpenTP1 Startup and Termination

This chapter explains the OpenTP1 startup and termination procedures.

This chapter contains the following sections:

- 2.1 Startup
- 2.2 Termination

2.1 Startup

This section explains the OpenTP1 startup methods and modes.

2.1.1 Startup methods

There are two methods of starting OpenTP1:

- Automatic startup

This method starts OpenTP1 automatically.

- Manual startup

This method starts OpenTP1 when the startup command (dcstart) is entered.

2.1.2 Startup modes

There are two OpenTP1 startup modes:

- Normal startup

This mode is used when there is no information to be inherited because the previous online operation terminated normally or when OpenTP1 is newly started.

- Restart following full recovery

This mode is used when the previous online termination status is to be inherited (called a restart).

2.1.3 Selection of startup method

The startup method is determined by a specification in the system environment definition (`mode_conf`).

(1) *AUTO specified in system environment definition*

Automatic startup is used as the startup method when AUTO is specified.

(2) *MANUAL1 specified in system environment definition*

Manual startup is used as the startup method when MANUAL1 is specified, unless OpenTP1 previously terminated normally, in which case it is restarted automatically.

(3) *MANUAL2 specified in system environment definition*

Manual startup is always used as the startup method when MANUAL2 is specified.

2.1.4 Selection of startup form

The combination of a startup method and a startup mode is called the startup form.

The startup form is determined from the previous termination mode and a specification in the system environment definition (`mode_conf`).

The following table explains the startup forms.

Table 2-1: Startup forms

Startup form selection conditions		Startup form	
Previous Termination Mode	Specification in <code>mode_conf</code>	Startup Method	Startup Mode
Normal termination	AUTO	Manual ^{#1}	Normal startup
	MANUAL1	Manual	
	MANUAL2		
Forced normal termination	AUTO	Manual ^{#1}	Normal startup
	MANUAL1	Manual	
	MANUAL2		
Planned termination A	AUTO	Manual ^{#1}	Restart
	MANUAL1	Manual	Restart ^{#3}
	MANUAL2		
Planned termination B	AUTO	Manual ^{#1}	Restart
	MANUAL1	Manual	Restart ^{#3}
	MANUAL2		
Forced termination	AUTO	Manual ^{#1}	Restart
	MANUAL1	Manual	Restart ^{#3}
	MANUAL2		
Abnormal termination	AUTO	Automatic	Restart
	MANUAL1	Automatic ^{#2}	
	MANUAL2	Manual	Restart ^{#3}

#1

Automatic startup will be performed during OS startup.

2. OpenTP1 Startup and Termination

#2

Manual startup will be performed during OS startup.

#3

It is also possible to forcibly perform normal startup by entering the `dcstart -n` command. In such a case, information being used during the previous operation will be lost.

2.2 Termination

This section explains the OpenTP1 termination modes and methods.

2.2.1 Termination modes

There are six OpenTP1 termination modes, as explained below.

(1) **Normal termination**

Under normal termination, no new service requests are accepted, and OpenTP1 terminates after completion of processing of all service requests currently in the scheduling queue.

If MCF is being used, OpenTP1 terminates after completion of processing of all messages currently in the input and output queues. Therefore, under normal termination, MCF monitors the input queue and then the output queue until no more messages remain. You specify the monitoring time in the timer definition (`mcftim`) of the MCF communication configuration definition.

When the queues are being monitored, log messages (KFCA16532-I through KFCA16537-I) are output at specific intervals. Output of these log messages can be suppressed by `mcfmsmsg` (definition for suppressing the output of log messages) in the MCF manager definition.

Note that if new messages occur successively (or intermittently) in the input and output queues during normal termination, events might occur, such as *normal termination does not end*, *the OpenTP1 system shuts down*, or *ERREVT A is activated*.

(2) **Forced normal termination**

Under forced normal termination, OpenTP1 terminates the same way as for normal termination in (1) above even if a server has terminated abnormally.

This termination mode allows the system to terminate normally even if a UAP server has terminated abnormally.

Since the system starts in the normal startup mode after forced normal termination, the last online status is not inherited.

(3) **Planned termination A**

Under planned termination A, no new service requests are accepted, and OpenTP1 terminates after completion of processing of all service requests currently in the scheduling queue.

If MCF is being used, all unprocessed messages in the input queue are processed. Therefore, under planned termination A, MCF monitors the input queue until no more messages remain. You specify the monitoring time in the timer definition (`mcftim`) of

the MCF communication configuration definition.

When the queue is being monitored, log messages (KFCA16532-I, KFCA16533-I, KFCA16536-I, and KFCA16537-I) are output at specific intervals. Output of these log messages can be suppressed by `mcfmsmsg` (definition for suppressing the output of log messages) in the MCF manager definition.

Note that if new messages occur successively (or intermittently) in the input queue during planned termination A, events might occur, such as *planned termination A does not end* or *the OpenTP1 system shuts down*.

Any unsent messages in the output queue are retained.

If processing of all messages in the output queue has not been completed at the time of planned online termination, OpenTP1 terminates under planned termination A.

(4) Planned termination B

Under planned termination B, OpenTP1 terminates after completion of the service that is being processed currently.

The scheduler processes only the service currently being processed without accepting any new service requests. Any other service requests are discarded.

If MCF is being used, all messages in the memory input and output queues are discarded. Any unprocessed messages in the disk input queue and any unsent messages in the output queue are retained.

(5) Forced termination

Under forced termination, OpenTP1 terminates immediately without waiting for completion of the service being processed currently.

All service requests in the scheduling queue are discarded.

If MCF is being used, all messages in the memory input and output queues are discarded. Any unprocessed messages in the disk input queue and any unsent messages in the output queue are retained.

(6) Abnormal termination

If an error occurs, OpenTP1 shuts down completely, which constitutes abnormal termination.

All service requests in the scheduling queue are discarded.

If MCF is being used, all messages in the memory input and output queues are discarded. Any unprocessed messages in the disk input queue and any unsent messages in the output queue are retained.

The following table explains the handling of service requests in the scheduling queue and messages in the input and output queues under the various termination modes.

Table 2-2: Handling of service requests in scheduling queue and messages in input and output queues

Termination mode	Service requests in scheduling queue	Messages in input queue		Messages in output queue	
		Disk	Memory	Disk	Memory
Normal termination	P	P	P	P	P
Forced normal termination	P	P	P	P	P
Planned termination A	P	P	P	R	R
Planned termination B	D	R	D	R	D
Forced termination	D	R	D	R	D

Legend:

P: Processed entirely.

R: Retained.

D: Discarded entirely.

2.2.2 Termination methods

Operation commands are used to issue termination requests to OpenTP1.

Following are the termination modes and the corresponding operation commands:

- Normal termination: `dcstop` command
- Forced normal termination: `dcstop -n` command
- Planned termination A: `dcstop -a` command
- Planned termination B: `dcstop -b` command
- Forced termination: `dcstop -f` command

In the case of forced termination when the operating status needs to be inherited, the restart mode must be used the next time OpenTP1 is started. If there is no need to inherit the operating status, the normal startup mode can be used.

2.2.3 Notes

If the OpenTP1 system service is directly stopped with the kill command, OpenTP1 is abnormally terminated.

Chapter

3. OpenTP1 Online Operation

This chapter explains the procedures for normal operation of OpenTP1.

This chapter contains the following sections:

- 3.1 Server operations
- 3.2 Scheduling operations
- 3.3 Transaction operations
- 3.4 Lock operations
- 3.5 Standard output file operations
- 3.6 Logging
- 3.7 Audit log operations
- 3.8 Trace operations
- 3.9 Shared memory operations
- 3.10 OpenTP1 domain operations
- 3.11 Operations using TP1/NET/OSI-TP-Extended
- 3.12 Operations using the remote API facility
- 3.13 Nonstop operation of OpenTP1
- 3.14 Operations using the XA resource service
- 3.15 Operations using Job Management Partner 1
- 3.16 Operations for using the real-time statistics service

3.1 Server operations

3.1.1 User server startup

User servers can be started and terminated individually. They can also be started in the online mode.

There are two methods of starting a user server:

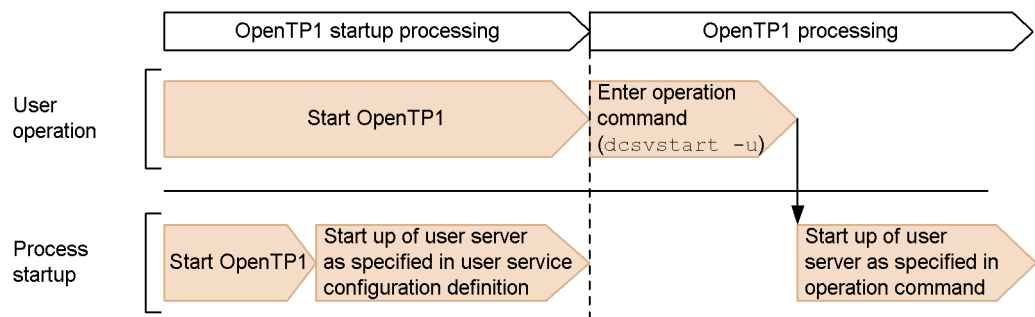
- User server startup based on user service configuration definition when OpenTP1 is started

In this case, `uap_conf` must have been specified in the system service configuration definition.

- User server startup by entering operation command (`dcsvstart -u`) in online mode

The following figure shows the user server startup procedures.

Figure 3-1: User server startup



3.1.2 User server termination

(1) Termination modes

There are two termination modes for a user server:

- Normal termination

Under normal termination, no new messages are accepted, and the user server terminates after completion of processing of all service requests currently in the scheduling queue.

- Forced termination

Under forced termination, the user server terminates immediately, all service requests in the scheduling queue are discarded, and control returns as an error to

the client in response to wait status.

(2) Termination methods

Operation commands are used to issue user server termination requests to OpenTP1.

Following are the termination modes and the corresponding operation commands:

- Normal termination: `dcsvstop` command
- Forced termination: `dcsvstop -f` command

(3) Relationship between user server's termination form and OpenTP1's termination form

If a user server is terminated forcibly, OpenTP1 cannot terminate normally. OpenTP1 must be terminated normally after restarting and then terminating the user server normally. Alternatively, OpenTP1 must be terminated by other than normal termination (i.e., by forced normal termination, planned termination A, planned termination B, or forced termination).

(4) Handling the application startup requests when the user server (MHP) is terminated

The startup requests for an application are handled as follows if such requests occur when the corresponding MHP is not running, is normally terminated, or is forcibly stopped:

- Startup requests from messages received:
ERREVT2 is started.
- Startup requests from functions issued from user application programs:
ERREVT2 is started (however, ERREVT4 is started for the timer startup)
- An MCF event occurs that causes a startup request to be issued for an application that handles MCF events:
Requests are discarded.

(5) Note

If the user server is directly stopped with the kill command, OpenTP1 may be abnormally terminated.

3.1.3 Display of server status

The `prcls` command displays the status of all user servers and the system service. The information displayed for a server includes the server status, process ID, etc.

3.1.4 Search path for user server and command activated from user server

(1) *Displaying and changing search path names*

The `prcpaths` command displays the search path names to be used when OpenTP1 starts a user server.

Search paths can be changed with the `prcpath` command.

(2) *Inheriting a search path*

When `prc_take_over_svpath=Y` is specified in the process service definition, the search path can be inherited at restart. In the following cases, however, the path is not inherited even when `prc_take_over_svpath=Y` is specified.

- When a system switchover has occurred
- When the search path could not be saved during the previous online processing
- When search path recovery from the file failed

(3) *Notes*

Ensure that a given command name does not refer to more than one command in the directories specified in the search paths. If there is such an overlap, the system may inadvertently execute the wrong command. Also ensure that the command names in the search paths do not overlap with the commands provided by OpenTP1 (under `$DCDIR/bin`).

3.1.5 Replacement of user server

A user server can be replaced by another user server during OpenTP1 execution.

The following procedure must be used to replace a user server:

(1) *Service Providing Program (SPP)*

1. Terminate the user server with the `dcsvstop` command.
2. Replace the user server (if necessary, also replace the user service definition)
3. Start the new server with the `dcsvstart` command.

It is also possible to create a new user server under a different directory from the one containing the currently active user server. Following is the procedure:

1. Terminate the user server with the `dcsvstop` command.
2. Define a new user server under another directory
3. Change the path to the new directory with the `prcpath` command
4. Start the new server with the `dcsvstart` command.

If OpenTP1 is restarted with the `dcstart` command, the previous user server will become effective again.

(2) Message Handling Program (MHP)

A currently active user server can be replaced by another user server for each service group.

Use the following procedure to replace the user server:

1. Shut down the scheduling of the currently active user server (service group) with the `mcftdctsg` command. Or, hold the input queue with the `mcfthldiq` command.
2. Terminate the user server with the `dcsvstop` command.
3. Replace the user server.
4. Restart the new user server with the `dcsvstart` command.
5. Release the scheduling of the user server (service group) shutdown with the `mcftactsg` command. Or, release the hold of the input queue with the `mcfrlsiq` command.

In the case of a disk queue, any start request occurring during user server (service group) shutdown will be re-scheduled after release of the shutdown. For a memory queue, all such start requests, including those made to already scheduled user servers, will be an error event.

When the scheduling is shut down (with `mcftdctsg`) or the input queue is held (with `mcfthldiq`) specifying the `-r` option, the shutdown or hold status is not inherited after full recovery.

3.1.6 User server process

(1) Termination of process

To forcibly terminate user server processes, execute the `dcsvstop -f` command to forcibly terminate the user server. This forcibly terminates all the processes of the user server.

Use the `prckill` command to forcibly terminate the specified user server process. This command forcibly terminates the process with the signal number specified in the `prc_abort_signal` operand in the user service definition or user service default definition.

3.2 Scheduling operations

3.2.1 Display of scheduling status

The scheduling status or service status of servers operating under the scheduler can be displayed with the `scdls` command for the following purposes:

- To determine the status of the servers
- To determine the status of the current scheduling queue
- To determine the number of queued service requests
- To determine the shutdown status of each service (with `service_hold=Y` specified in the user service definition)

The displayed information includes the number of servers operating under the scheduler, the server names, the current number of queued service requests, etc.

3.2.2 Shutdown and restart of scheduling

If an error occurs in a user server or scheduling of a user server needs to be terminated in order to replace execution form files, scheduling can be shut down with the `scdhold` command. When `service_hold=Y` is specified in the user service definition, the scheduling can be held in units of services. Scheduling for MHP and for the system service cannot be shut down with this command (the `mcfidctsv` command must be used to shut down MHP scheduling).

Once scheduling is shut down, any subsequently issued service request and any service request in the scheduling queue results in an error return.

Note that, if the `schhold` command with the `-p` option specified is executed, the service request in a scheduling queue is held until scheduling is restarted. In addition, any service request that may occur after the shutdown of scheduling is accepted.

Scheduling that was shut down with the `scdhold` command can be restarted with the `scdrls` command.

At restart, the shutdown status of scheduling can be inherited. To inherit the shutdown status, specify the following in the user service definition:

- `receive_from=queue`
- `hold_recover=Y`

Scheduling can also be shut down on the basis of a specification in the user service definition. The following lists the specifications and the conditions that result in scheduling shutdown:

- `hold=Y` specified:

User server terminates abnormally once

- `term_watch_time` specified:

User server terminates abnormally three times in a row within the specified time

3.2.3 Automatic shutdown of scheduling

(1) Message at abnormal termination

If MHP terminates abnormally, the receive message of MHP that terminated abnormally can be re-scheduled at the beginning of the schedule queue by shutting down the scheduling of the service group automatically. However, the receive message of MHP cannot be re-queued at full recovery.

In the application attribute definition, specify the following:

```
mcfhaalcap -g srvghold = s
            recvmsg = r
            -d holdlimit = 1
```

The handling of the message of the abnormally terminated MHP differs depending on the specification of the service group automatic shutdown. The following table shows the message handling at abnormal termination.

Table 3-1: Message handling at abnormal termination

Specification of application attribute definition			Memory queue	Disk queue
Automatic shutdown of service group scheduling specified (<code>srvghold=s</code>)	Rescheduling specified (<code>recvmsg=r</code>)	Maximum error count reached	ERREVT2 ^{#1}	Scheduling wait status ^{#2}
		Maximum error count not reached	ERREVT2 or ERREVT3	ERREVT2 or ERREVT3
	Rescheduling not specified (<code>recvmsg=e</code> or omitted)		ERREVT2 or ERREVT3	ERREVT2 or ERREVT3
Automatic shutdown of service group scheduling not specified (<code>srvghold=m</code> or omitted)			ERREVT2 or ERREVT3	ERREVT2 or ERREVT3

#1

The memory queue does not store messages.

#2

Scheduled at the beginning of the schedule queue.

(2) Messages arrived after the service group automatic shutdown

Tables 3-2 and 3-3 show the handling of the messages arrived before and after the

service group scheduling is automatically shut down.

Table 3-2: Handling of disk queue messages

Shutdown type	Timing	
	Messages arrived before shutdown and put into the input queue	Messages arrived after shutdown
Shutdown of both input and scheduling	Placed in scheduling wait status. After release of the shutdown, MHP is started.	Put into the input queue for ERREVT2.
Shutdown of input	MHP is started.	Put into the input queue for ERREVT2.
Shutdown of scheduling	Placed in scheduling wait status. After release of the shutdown, MHP is started.	Put into the input queue and placed in scheduling wait status.

Table 3-3: Handling of memory queue messages

Shutdown type	Timing	
	Messages arrived before shutdown and put into the input queue	Messages arrived after shutdown
Shutdown of both input and scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.
Shutdown of input	MHP is started.	Put into the input queue for ERREVT2.
Shutdown of scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.

(3) Messages at application shutdown and messages at service shutdown

Tables 3-4 and 3-5 show the handling of the messages that shut down an application or service.

Table 3-4: Handling of messages at application shutdown

Shutdown type	Timing	
	Messages arrived before shutdown and put into the input queue	Messages arrived after shutdown
Shutdown of both input and scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.
Shutdown of input	MHP is started.	Put into the input queue for ERREVT2.

Shutdown type	Timing	
	Messages arrived before shutdown and put into the input queue	Messages arrived after shutdown
Shutdown of scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.

Table 3-5: Handling of messages at service shutdown

Shutdown type	Timing	
	Messages arrived before shutdown and put into the input queue	Messages arrived after shutdown
Shutdown of both input and scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.
Shutdown of input	MHP is started.	Put into the input queue for ERREVT2.
Shutdown of scheduling	Put into the input queue for ERREVT2 again.	Put into the input queue for ERREVT2.

3.2.4 Inter-node load balance

OpenTP1 distributes the load of service request processing to nodes. This method is called *inter-node load balancing*.

(1) Prerequisites for inter-node load balancing

To use the inter-node load balancing facility, the following conditions must be satisfied:

- A user server providing the same service to multiple nodes must be active.
- Each OpenTP1 node must have the other nodes defined for the `all_node` operand of the system common definition so that the OpenTP1 nodes can exchange the information (name information) of the user servers operating on the nodes among them.

Note

Inter-node load balancing can be achieved when the user server operates similarly for all nodes. If the conditions listed below vary greatly among the selected nodes, inter-node load balancing may be difficult. For this reason, do not place service groups with the same name on multiple nodes.

- Communication costs such as the line fees for public lines
- Line speed

- Line quality
- Node performance

(2) Operation modes of inter-node load balancing

There are the following two operation modes for using the inter-node load balancing facility:

Mode where the server side determines how to perform load balancing

The scheduler on the server side (TP1/Server Base) transfers a request to the least loaded node for processing.

Mode where the client side determines how to perform load balancing based on the load information from the server

In this mode, the processing varies depending on the program used for the client.

- TP1/Client/P or TP1/Client/W is used for the client.

The client side (TP1/Client/P or TP1/Client/W) determines to which OpenTP1 node the service request should be directed based on the information of the server load level acquired from the server, and then performs an RPC.

- TP1/Server Base is used for the client.

The client side (TP1/Server Base) already knows the load level of the server which is going to issue a request. Therefore it performs an RPC to the least loaded node from the start. Upon receiving a request, the scheduler does not perform a transfer based on the load level. The scheduler instead performs the processing within its local node if the request can be processed within the local node. The scheduler transfers the request to another node only if the server is blocked, or if the server of the local node has a load level of LEVEL2 and another node has a server with a lower load level.

To use the inter-node load balancing facility, specify the following definitions:

Table 3-6: Definitions for using inter-node load balancing

Operation mode for inter-node load balancing	Type	Definition
Load balancing determined on the server side	Server (TP1/Server Base)	Schedule service definition: set scd_this_node_first=N (default)set scd_announce_server_status=Y (default)
	Client (TP1/Client/P, TP1/Client/W)	Client environment definition: dcscddirect=Y (for TP1/Client/P) dchostselect=Y (only when you want to change the schedule request randomly)

Operation mode for inter-node load balancing	Type	Definition
Load balancing determined on the client side (when using TP1/Client/P, TP1/Client/W for the client)	Server (TP1/Server Base)	Schedule service definition: scd_this_node_first=N (default) scd_announce_server_status=Y (default)
	Client (TP1/Client/P, TP1/Client/W)	Client environment definition: dccltloadbalance=Ydccltcachetim=xx (seconds)
Load balancing determined on the client side (when using TP1/Server Base for the client)	Server (TP1/Server Base)	Schedule service definition: scd_this_node_first=N (default) scd_announce_server_status=Y (default)
	Client (TP1/Server Base)	

(3) Referencing the load status

When `scd_announce_server_status=Y` is specified in the schedule service definition, OpenTP1 notifies all the nodes of the server's load status at specified intervals (of 30 seconds or more). OpenTP1 selects a node according to the server's load status and schedules the execution of the service request.

When `scd_announce_server_status=N` is specified, the server's load status is not reported to the other nodes. OpenTP1 randomly selects a node and schedules the server.

For all the nodes in the distributed environment, the specification of the `scd_announce_server_status` operand must be same. If the specification differs, the load is concentrated upon particular nodes.

When both of the following conditions apply, specify `scd_announce_server_status=N`:

- The same service group does not exist on multiple nodes.
- Less communication is preferred because a high cost communication line is being used or for some other reason.

In this case, inter-node load balance is unnecessary. Therefore, the server's load status does not need to be reported to the other nodes. When `scd_announce_server_status=Y` is specified, a line is used to report the server's load status. When a server is activated at only one node, specify `scd_announce_server_status=N` to suppress reporting the server's load status.

(4) Precedence of the local node

The inter-node balancing does not always schedule the processing to the local node

even when the server is placed at the local node.

When the requested server is placed at the local node, to give that server the precedence of scheduling, specify `scd_this_node_first=Y` in the schedule service definition. If the requested server is not placed at the local node, if the request cannot be scheduled to the server at the local node due to overload or shutdown, the request is scheduled to another node.

This operation is effective when less inter-node communication is required.

The following table shows the combinations of the `scd_announce_server_status` and `scd_this_node_first` operands.

Table 3-7: Combinations of `scd_announce_server_status` and `scd_this_node_first` operands

Operand		scd_announce_server_status	
		Y	N
scd_this_node_first	Y	Reports the server's load status to the other nodes. The load between nodes is balanced but the server at the local node has precedence.	Does not report the server's load status to the other nodes. The request is scheduled to the server of a randomly selected node. When the server is at the local node and the request can be scheduled to it, the request is scheduled to the local node.
	N	Reports the server's load status to the other nodes. The load between nodes is balanced. The request is scheduled to the node of the least-loaded schedulable server.	Does not report the server's load status to the other nodes. The request is scheduled to the server of a randomly selected node. The request may not be scheduled to the server of the local node even when the requested server is at the local node.

(5) Operations when inter-node load balancing facility is used with other facilities

The following table shows the operations when the inter-node load balancing facility is used with other facilities.

Table 3-8: Operations of inter-node load balancing facility used with other facilities

When using	Operation mode of inter-node load balancing facility	Operation
Permanent connection by TP1/Client	<ul style="list-style-type: none"> The server side determines how to perform load balancing. The client side determines how to perform load balancing (TP1/Client/P or TP1/Client/W is used for the client). 	The CUP execution process of TP1/Server Base performs an RPC in the node that established the permanent connection. This is the same operation as in the case when the client side determines how to perform load balancing (TP1/Server Base is used for the client).
Transaction control API by TP1/Client	<ul style="list-style-type: none"> The server side determines how to perform load balancing. The client side determines how to perform load balancing (TP1/Client/P or TP1/Client/W is used for the client). 	The transaction delegated execution process of TP1/Server Base performs an RPC. This is the same operation as in the case when the client side determines how to perform load balancing (TP1/Server Base is used for the client).
Remote API facility	<ul style="list-style-type: none"> The server side determines how to perform load balancing. The client side determines how to perform load balancing (TP1/Client/P or TP1/Client/W is used for the client). The client side determines how to perform load balancing (TP1/Server Base is used for the client). 	The RAP-processing server of TP1/Server Base actually performs an RPC. This is the same operation as in the case when the client side determines how to perform load balancing (TP1/Server Base is used for the client).

(6) **Extended inter-node load-balancing facilities**

You can specify the following as the extended facilities for distributing the load between nodes:

- Specifying the rate of schedules made to LEVEL0 nodes

By specifying the `schedule_rate` operand in the schedule service definition, you can make schedules to LEVEL0 load level nodes with priorities defined.

- Specifying the load check interval

By specifying the `loadcheck_interval` operand in the user service definition and the user service default definition, you can specify how often to check the load for each service group.

- Specifying the threshold of load levels

By specifying the `levelup_queue_count` operand and the `leveldown_queue_count` operand in the user service definition and the user

service default definition, you can specify the threshold (number of remaining server requests) that determines the load level for each service group.

- Specifying the number of retries to be made if a communication error occurs

By specifying the `scd_retry_of_comm_error` operand in the schedule service definition, you can retry scheduling to nodes other than the failed node even if a communication error occurs while service requests are scheduled.

If you do not make this specification, no re-scheduling is performed and an error is returned if a communication error occurs.

To use this facility, TP1/Extension 1 needs to be installed beforehand. If TP1/Extension 1 is not installed, the operation is not guaranteed.

3.2.5 Changing the number of processes

You can use the `scdchprc` command to change the number of resident processes running on a server or the maximum number of processes that can run on a server while the server is operating. This command can be used for user servers and some system servers. As a result of the change, new processes are generated or excess processes are terminated. The number of processes changed by this command is effective until the server is terminated (including when the server is forcibly terminated). The system does not inherit the change after full system recovery.

You must install TP1/Extension 1 to use this facility. Without TP1/Extension 1 installed, this facility may not operate correctly.

3.2.6 Monitoring the schedule queue

If the service requests from clients start to be delayed in the service processing stage, they may stay too long in the schedule queue because they cannot be fetched. By specifying a schedule delay limit in the `schedule_delay_limit` operand in the user service definition, you can monitor how long a service request stays in the schedule queue. If the service requests registered in the schedule queue remain in the schedule queue beyond the schedule delay limit, OpenTP1 outputs message KFCA00838-W for each server affected. In other words, a message is output if the interval from the time of the last queue operation to the status check by the scheduler exceeds the time specified in the `schedule_delay_limit` operand.

The time of the last queue operation is updated when:

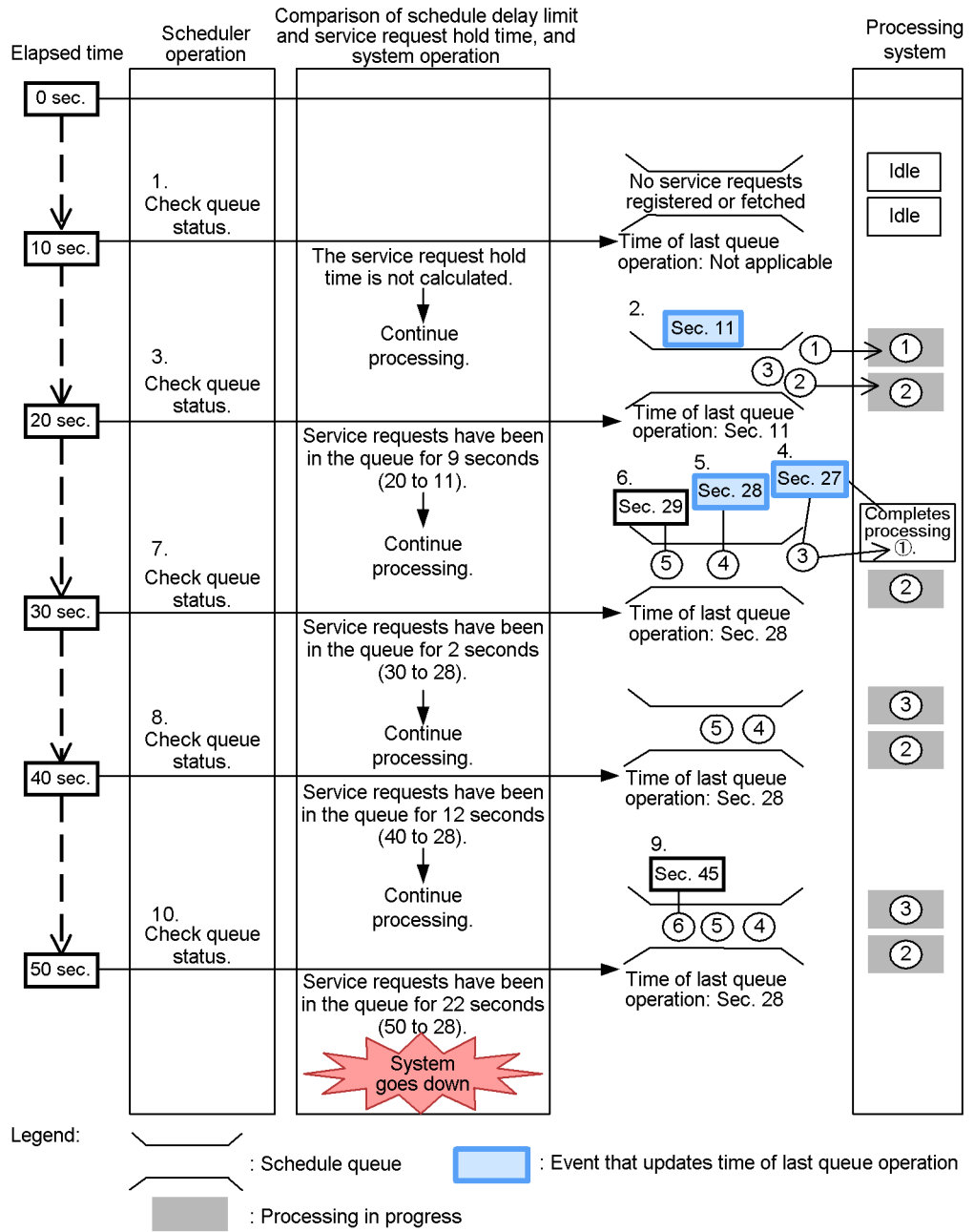
- A service request is fetched from the schedule queue.
- A service request is registered in the schedule queue when there are no service requests remaining in the schedule queue.

If you specify `schedule_delay_abort=Y` in the user service definition, the servers output message KFCA00839-E along with KFCA00838-W, causing the scheduler daemon to forcibly terminate and OpenTP1 to shut down. When the OpenTP1 system

runs in a hot standby configuration, OpenTP1 detects the delayed status of service requests in the schedule queue and executes a system switchover.

The following figure shows an example of monitoring the schedule queue when 15 is specified in the `schedule_delay_limit` operand and `Y` is specified in the `schedule_delay_abort` operand.

Figure 3-2: Example of monitoring the schedule queue
(schedule_delay_limit=15, schedule_delay_abort=Y)



■ Explanation

1. The scheduler checks the status of the queue.
No service requests are registered in the schedule queue or have been fetched from the schedule queue. Since the service request hold time is not calculated, the schedule delay limit and the service request hold time are not compared. The scheduler continues processing.
2. Three service requests are registered.
The first and second service requests are fetched, and processing of them begins. The third service request remains in the queue because the processing systems are busy. At this time, the last queue operation time is second 11.
3. The scheduler checks the status of the queue.
At this time, the last queue operation time is second 11. The service request hold time^{#1} is 9 seconds, which does not exceed the schedule delay limit. The scheduler continues processing.
4. Processing of the first service request is completed.
The third service request is fetched by the system that processed the first service request, and processing of the third service request begins^{#2}. At this time, the last queue operation time is second 27.
5. The fourth service request is registered.
The second and third service requests are being processed, and the processing systems are busy. Therefore, the fourth service request remains in the schedule queue. At this time, the last queue operation time is second 28.
6. The fifth service request is registered.
The second and third service requests are being processed, and the processing systems are busy. Therefore, the fifth service request remains in the schedule queue. At this time, the last queue operation time is second 28.
7. The scheduler checks the status of the queue.
At this time, the last queue operation time is second 28. The service request hold time^{#1} is 2 seconds, which does not exceed the schedule delay limit. The scheduler continues processing.
8. The scheduler checks the status of the queue.
At this time, the last queue operation time is second 28. The service request hold time^{#1} is 12 seconds, which does not exceed the schedule delay limit. The scheduler continues processing.

9. The sixth service request is registered.

The second and third service requests are being processed, and the processing systems are busy. Therefore, the sixth service request remains in the schedule queue. At this time, the last queue operation time is second 28.

10. The scheduler checks the status of the queue.

At this time, the last queue operation time is second 28. The service request hold time^{#1} is 22 seconds, which exceeds the specified schedule delay limit. The OpenTP1 system goes down.

#1

The service request hold time is determined by subtracting the last queue operation time from the time at which the scheduler checks the queue.

#2

The third service request actually has been in the schedule queue for 16 seconds (second 27 - second 11). This value exceeds the specified schedule delay limit.

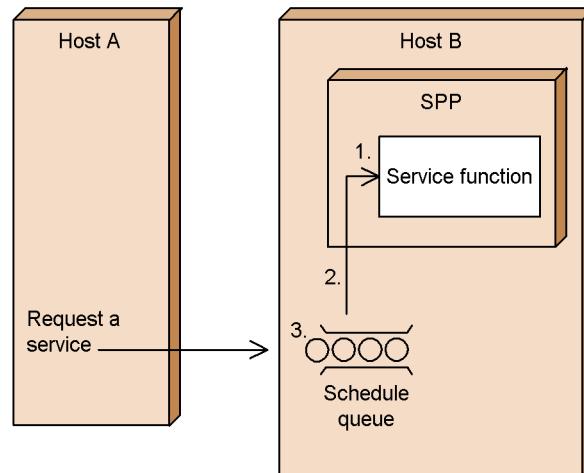
In actuality, however, OpenTP1 determines whether the specified schedule delay limit is exceeded by checking the value calculated by subtracting the last queue operation time from the time at which the scheduler checks the queue status (second 30 - second 28). Therefore, the scheduler continues processing.

3.2.7 Schedule queue congestion monitoring

If the service requests from clients start to be delayed in the service processing stage, they may stay too long in the schedule queue because they cannot be fetched.

Therefore, OpenTP1 checks (monitors) the service requests remaining in the schedule queue at a specified interval for each user server. This function is called *schedule queue congestion monitoring*, and is available only for user servers (SPPs). The following figure shows an example of service requests remaining in the schedule queue.

Figure 3-3: Service requests remaining in the schedule queue



1. The processing of service requests is delayed for some reason.
2. Fetching of service requests from the schedule queue is delayed.
3. Service requests accumulate in the schedule queue.

If the number of service requests remaining in the schedule queue exceeds the value specified for the applicable operand in the system definition during schedule queue congestion monitoring, OpenTP1 outputs message KFCA00833-W. Depending on the specification of the operand, OpenTP1 system also outputs message KFCA00834-E and goes down (forced termination).

(1) Operands to be specified

To use schedule queue congestion monitoring, specify the following operands in the user service definition or the user service default definition. For details about each operand, see the manual *OpenTP1 System Definition*.

1. `stay_watch_queue_count`
Use this operand to specify the number of service requests in the schedule queue that triggers judgment of whether the schedule queue is congested.
2. `stay_watch_check_rate`
Use this operand to specify the processing rate at which service requests can be processed by the server. The processing rate is used in the judgment of whether the schedule queue is congested.
3. `stay_watch_abort`
Use this operand to specify whether to allow shutdown of OpenTP1 when the

expression for judging congestion of the schedule queue evaluates to true.

4. `stay_watch_start_interval`

Use this operand to specify the interval for monitoring the number of service requests remaining in the schedule queue.

5. `stay_watch_check_interval`

Use this operand to specify the interval for judging whether the schedule queue is congested based on the expression for judging congestion of the schedule queue.

You can specify the above operands only on SPP user servers. If you specify these operands on a RAP-processing server or an MHP server, schedule queue congestion monitoring will not be enabled. If you omit the `stay_watch_queue_count` operand or specify 0 for it, the values specified for operands 2 to 5 have no effect.

(2) Processing sequence

This subsection describes the processing sequence of schedule queue congestion monitoring. Note that the user server (SPP) has already been started.

1. Start the monitoring of the schedule queue at the interval specified in the `stay_watch_start_interval` operand.
2. When the number of service requests remaining in the schedule queue exceeds the value specified in the `stay_watch_queue_count` operand, OpenTP1 enters a period in which it judges whether the schedule queue is congested and starts the judgment process.

When judgment of congestion starts, the judgment of whether the schedule queue is congested is based on the expression for judging congestion of the schedule queue.

Expression for judging congestion of the schedule queue:

$$\begin{array}{l} \text{Number of service requests} \\ \text{that are processed} \end{array} < \begin{array}{l} \text{Processing rate of service requests} \\ \text{(value specified in the} \\ \text{stay_watch_check_rate operand)} \end{array} \times \begin{array}{l} \text{Number of service} \\ \text{requests remaining in} \\ \text{the schedule queue} \end{array}$$

After OpenTP1 makes its judgment, operation is as follows:

- If the expression for judging congestion does not evaluate to true:
The scheduler continues processing.
- If the expression for judging congestion evaluates to true and N is specified in the `stay_watch_abort` operand:
OpenTP1 outputs message KFCA00833-W, and the scheduler continues processing.

- If the expression for judging congestion evaluates to true and Y is specified in the `stay_watch_abort` operand:

OpenTP1 system outputs message KFCA00833-W and message KFCA00834-E, and goes down.

3. Judgment of whether the schedule queue is congested is performed at the interval specified in the `stay_watch_check_interval` operand for each user server.

If the number of service requests remaining in the schedule queue falls below the value specified for the `stay_watch_queue_count` operand, OpenTP1 returns to step 1.

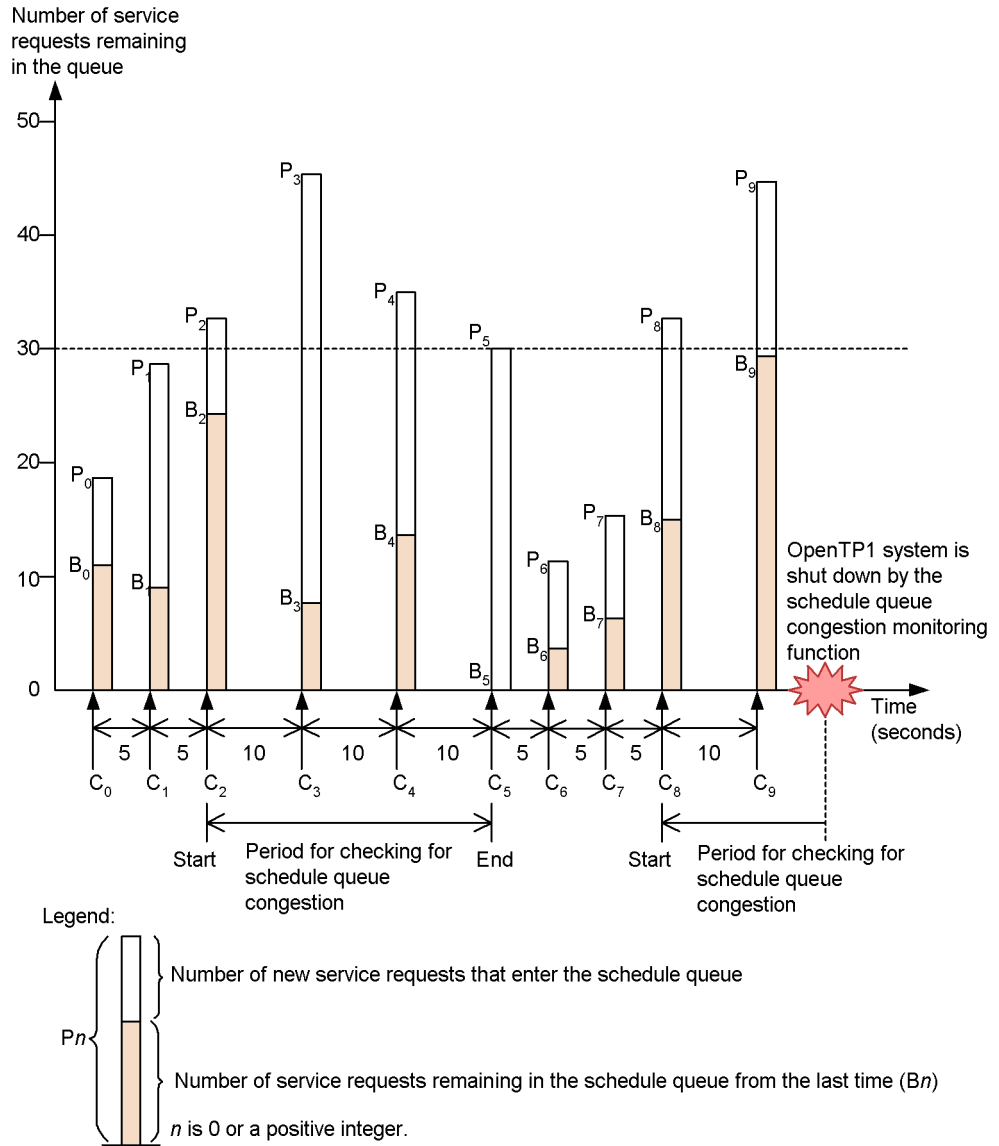
(3) Example of the processing sequence

This subsection describes an example of schedule queue congestion monitoring when the following values are specified for the operands in the user service definition:

User service definition

```
set stay_watch_queue_count=30 (number of service requests)
set stay_watch_check_rate=70 (%)
set stay_watch_abort=Y
set stay_watch_start_interval=5 (seconds)
set stay_watch_check_interval=10 (seconds)
```

Figure 3-4: Example of schedule queue congestion monitoring



In Figure 3-4, judgment of whether the schedule queue is congested occurs in the period from C_2 to C_5 and in the period starting at C_8 . Judgment starts because of the specification `stay_watch_queue_count=30`. During each period, judgment of queue congestion is performed at the interval specified in the `stay_watch_check_interval` operand. At this interval, OpenTP1 judges whether the

schedule queue is congested according to the expression for judging congestion, which is based on the number of service requests that are processed and the processing rate of service requests.

Expression for judging congestion of the schedule queue:

$$\begin{matrix} \text{Number of service requests} \\ \text{that are processed} \end{matrix} < \begin{matrix} \text{Processing rate of service requests} \\ \text{(value specified in the} \\ \text{stay_watch_check_rate operand)} \end{matrix} \times \begin{matrix} \text{Number of service} \\ \text{requests remaining in} \\ \text{the schedule queue} \end{matrix}$$

The expression for the example in Figure 3-4 is as follows:

$$P_{n-1} - B_n < m_1 \times P_{n-1}$$

n : 0 or a positive integer

$P_{n-1} - B_n$: Number of service requests that are processed during the applicable period

m_1 : Processing rate of service requests (value specified in the set stay_watch_check_rate operand)

P_{n-1} : Number of service requests remaining in the schedule queue during the applicable period

The following table shows the number of service requests that are processed in Figure 3-4, and the results of evaluating the expression for judging congestion of the schedule queue.

n	Number of processed service requests ($P_{n-1} - B_n$)	Number of service requests expected to be processed ($m_1 \times P_{n-1}$)	Result of evaluating the expression for judging congestion of the schedule queue ($P_{n-1} - B_n < m_1 \times P_{n-1}$)
0	--	--	Not applicable.
1	18-9=9	0.7 x 18=12.6	Not applicable.
2	28-25=3	0.7 x 28=19.2	Judgment of whether the schedule queue is congested starts.
3	32-8=24	0.7 x 32=22.4	OpenTP1 continues online operation.
4	45-13=32	0.7 x 45=31.5	OpenTP1 continues online operation.
5	35-0=35	0.7 x 35=24.5	OpenTP1 continues online operation.
6	30-3=27	0.7 x 30=21	Not applicable.
7	11-5=6	0.7 x 11=7.7	Not applicable.

n	Number of processed service requests ($P_{n-1} - B_n$)	Number of service requests expected to be processed ($m_1 \times P_{n-1}$)	Result of evaluating the expression for judging congestion of the schedule queue ($P_{n-1} - B_n < m_1 \times P_{n-1}$)
8	17-15=2	0.7 × 17=11.9	Judgment of whether the schedule queue is congested starts.
9	32-29=3	0.7 × 32=22.4	The OpenTP1 system goes down.

Legend:

n : 0 or a positive integer

--: Not applicable.

When n equals 1 or 7 in the above table, the expression for judging congestion of the schedule queue evaluates to true. However, the OpenTP1 system does not go down because judgment of whether the schedule queue is congested is not being performed. When n equals 2 or 8, the OpenTP1 system does not go down because the judgment process is only beginning.

When n equals 9, the OpenTP1 system goes down because judgment of whether the schedule queue is congested is being performed, the expression for judging congestion of the schedule queue evaluates to true, and `stay_watch_abort=Y` is specified.

3.2.8 Refreshing user server processes

You can use the `scdrsprc` command to stop and restart a user server process without taking the user server offline. User server processes that are executing service requests when you execute the `scdrsprc` command are stopped and then restarted at completion of those service requests.

You can also use this command to change a user server's load module without taking the user server offline.

To change a load module:

1. Create the directory where the user server's new load module will be installed.
2. Use the `prcpathls` command to check the user server's search paths.
3. Use the `prcpath` command to specify the directory created in step 1 ahead of the search paths specified for the server.
4. Execute the `scdrsprc` command.

If you are using the dynamic loading of service functions, you can replace the service functions for a user server without taking the user server offline.

To replace service functions:

1. Change the name of the UAP shared library specified in the `service` operand in the user service definition for the applicable user server.
2. Execute the `scdrsprc` command.

To check the startup of a new load module or service function, use the system commands provided by your OS to check the process start time on the user server, and compare it with the execution time of the `scdrsprc` command. When replacing service functions using the `scdrsprc` command, you cannot add or delete a service function in that user server. To add or delete a service function in a user server, first stop the user server by the `dcsvstop` command, and then start it again by using the `dcsvstart` command.

The following table shows what changes you can make to service functions on user servers when you use the `scdrsprc` command, and when you use the `dcsvstop` and `dcsvstart` commands.

Table 3-9: Scope of changes permitted to service functions on user servers

Execution method	Replacement of service functions	Addition of service functions	Deletion of service functions
<code>scdrsprc</code> command	Y	N	N
<code>dcsvstop</code> and <code>dcsvstart</code> commands	Y	Y#	Y

Legend:

Y: Can be changed.

N: Cannot be changed.

#

If you add service functions to an MHP, you must change the MCF application attribute definitions (`mcfaalcap`).

You must install TP1/Extension 1 to use this facility. Without TP1/Extension 1 installed, this facility may not operate correctly.

3.3 Transaction operations

3.3.1 Display of transaction status

The `trnls` command displays the transactions managed by the transaction manager and resource manager information.

The displayed information includes process IDs, transaction global IDs, resource manager names, etc.

3.3.2 Forcible determination of transaction

When transaction status is displayed by the `trnls` command and a transaction's first, second, and third statuses are all `READY(p, n)` for a prolonged period of time, it may mean that for some reason (such as a communication error) no transaction branch comprising the global transaction is on a status in which it can determine the transaction. OpenTP1 will reattempt to determine the transaction, but if the user cannot wait for the OpenTP1 retry, the affected transaction branch can be terminated forcibly (heuristic decision) by entering an operation command. If the root transaction branch is committed, the transaction branch can be committed by executing the `trncmt` command. If the root transaction branch is rolled back, the transaction branch can be rolled back by executing the `trnrbk` command. Whether the root transaction branch is committed or rolled back can be verified either by displaying transaction status with the `trnls` command or by referencing the message log file with the `logcat` command.

If transaction branches in the same global transaction are distributed on separate computers, the operation commands must be entered individually. In such a case, all the transaction branches must be determined by the same method (either commitment or rollback).

If the `trncmt` command executes, the first transaction status displayed by the `trnls` command is placed on `HEURISTIC_COMMIT` status; if the `trnrbk` command is executed, it is placed on `HEURISTIC_ROLLBACK` status. The status then changes temporarily to `HEURISTIC_FORGETTING` before the `trncmt` or `trnrbk` command's processing terminates.

3.3.3 Forcible termination of transaction

A transaction that has been determined forcibly as a result of a communication error cannot be terminated until all communication is completed between the transaction branches. If the communication error cannot be recovered for a long time, it is possible to terminate the transaction forcibly. The `trncmt` or `trnrbk` command with the `-f` option specified or the `trnfgt` command is used in such a case.

When the `trncmt` or `trnrbk` command with the `-f` option specified executes, the transaction whose first, second, and third status are all `READY(p, n)` is committed or rolled back forcibly. The determination result of the corresponding transaction branch

cannot be reported to any other transaction branch in the same global transaction. It may, therefore, not be possible using this method to determine all the transaction branches in the same global transaction.

The `trnfgt` command terminates forcibly the transaction after executing the `trncmt` or `trnrbk` command without the `-f` option specified. When the `trnfgt` command executes, the transaction whose first, second, and third status are all `HEURISTIC_FORGETTING(p, n)` is terminated forcibly. In this case also, the determination result of the corresponding transaction branch cannot be reported to any other transaction branch in the same global transaction. It may, therefore, not be possible using this method to determine all the transaction branches in the same global transaction.

Once the transaction has been terminated forcibly, the resources previously being used by the resource manager are released and become available to other UAPs.

3.3.4 Deleting undecided-transaction information files

When `trn_tran_recovery_list=Y` is specified in the transaction service definition, an undecided-transaction information file is created under `$DCDIR/spool/dctrninf`. This file is created each time a full recovery is performed. Delete unnecessary files. Many unnecessary files may slow the file system.

Files can be deleted automatically at start of OpenTP1. Also, you can delete files at any time.

To delete files automatically at start of OpenTP1, specify `normal` or `force` by the `trn_recovery_list_remove` operand in the transaction service definition.

To delete files any time you want to delete, execute the `trndlinf` command.

3.3.5 Startup and termination of collecting transaction statistics

The `trnstics` command makes it possible to collect transaction statistics into a journal file.

Collection of transaction statistics can be started with the `trnstics` command with the `-s` option specified. Executing the `trnstics` command with the `-s` option specified does not collect transaction statistics from transactions that are to start after normal termination of the `trnstics` command. No transaction statistics can be collected from the transactions that had started before normal termination of the command.

Transaction statistics can be collected only from transactions that have been executed by the user service for which 1-4 is specified in the `trn_statistics_type` option of the OpenTP1 system definition.

Collection of transaction statistics can be terminated with the `trnstics` command with the `-e` option specified. Executing the `trnstics` command with the `-e` option specified does not let the system collect transaction statistics from transactions that are

to start after normal termination of the `trnstics` command.

The greater the number of kinds of transaction statistics are collected, the lower the transaction performance. For this reason, it is recommended to collect statistics only when transaction performance is not a problem.

At restart of OpenTP1, the specification of the `trnstics` command cannot be inherited and that of `trn_tran_statistics` (to choose whether statistics for each transaction branch are to be inherited) of the transaction service definition predominates.

3.3.6 Acquisition of transaction statistics when the XA resource service is used

Using the `trnstics` command, you can acquire transaction statistics for the XA resource service. However, no distinction is made between transaction branches that originate from an application server and transaction branches started from OpenTP1. All transaction branches are treated as if they were started from OpenTP1. Also, you cannot acquire information on CPU time for transaction branches on a RAP-processing server.

3.4 Lock operations

3.4.1 Display of lock information

The lckls command displays lock information.

The displayed information includes the deadlock priority value, resource name, wait time, etc.

3.4.2 Display of lock control table pool information

The lckpool command monitors the utilization of the lock control table pool.

The lckpool command displays the lock request count, usage rate of the table pool for locks, and other information for the following server and services:

- User server
- DAM service
- TAM service
- MQA service

3.4.3 Deletion of deadlock information files and timeout information files

The output deadlock information and timeout information can be deleted with the lckrminf command. Executing this command deletes the deadlock information files and timeout information files that were created before "24 hours x (number of days specified by the lckrminf command)" counted from the time when the command was executed.

3.5 Standard output file operations

The standard output and standard error output for the processes under OpenTP1 are usually /tmp/betran.log (default). This file may unlimitedly increase and occupy the disk unless the machine is provided with a mechanism to delete the file when it starts up. To prevent this, you can perform generation management of the /tmp/betran.log file.

To perform generation management of the /tmp/betran.log file, make the following settings:

1. If OpenTP1 is operating under \$DCDIR, shut down OpenTP1.
\$ dcstop<CR>
2. If OpenTP1 is cataloged in the OS (/etc/inittab), delete OpenTP1 from the OS.
/BeTRAN/bin/dcsetup -d \$DCDIR<CR>
KFCA01836-R specify whether to delete the files necessary for execution from the specified OpenTP1 directory.
[y: Yes, n: No]
n<CR>
3. Open the \$DCDIR/bin/prcout file and change the definition of the prctee command (on the underlined line). For details of the prctee command, see *13. Details of Operation Commands*.

Example for HP-UX

```

1 #!/bin/sh
2 #!ALL RIGHTS RESERVED,COPYRIGHT (C)1995,HITACHI,LTD.
3 #!LICENSED MATERIAL OF HITACHI,LTD.
4 #!@(#) prcout(96/12/18 14:36:42)-1.7
5 PATH=/bin:/usr/bin
6 export PATH
7
8 if [ ! -d "$1" ] ; then
9     echo "usage : prcout OpenTP1_directory" 1>&2
10    exit 1
11 fi
12 DCDIR="$1"
13 export DCDIR
14 $DCDIR/bin/prctee 0 /tmp/betran.log

```

4. Catalog OpenTP1 in the OS (/etc/inittab).
/BeTRAN/bin/dcsetup \$DCDIR<CR>

3.6 Logging

3.6.1 Message log

The following two facilities are used to monitor the online status:

- Realtime output
- Output to message log file for editing and output

(1) *Realtime output facility*

Message logs output to the message log file can be output in realtime to the standard output. To do this, `log_msg_console=Y` must be specified in the log service definition. The user can also specify whether or not the message serial numbers, request source process IDs, output request dates and times, etc., are to be output.

The `log_msg_console` specification can be changed in the online mode with the `logcon` command.

(2) *Output to message log file and editing*

(a) **Output to message log file**

OpenTP1 outputs a message log to the message log file upon receipt of an output request from each system service, MCF, or UAP.

The two message log files are `$DCDIR/spool/dcllog1` and `$DCDIR/spool/dcllog2`. These two files are used with the round-robin scheduling method, and the immediately preceding generation of the message information is guaranteed. When the files are switched, a message to that effect is displayed. If a file needs to be saved, the user must back it up before new information is overwritten into it.

(b) **Message log editing and output**

The messages in the message log files can be output to the standard output by executing the `logcat` command. The messages in the two message log files, `$DCDIR/spool/dcllog1` and `$DCDIR/spool/dcllog2`, are merged and then output in chronological order.

(c) **Cautions about the amount of log output**

If there is a large amount of log output, any of the following failures may occur. When operating OpenTP1, care should be taken to prevent the amount of output messages from extremely increasing if a failure occurs.

- The `dc_logprint` function and `CBLDCLOG('PRINT')` returns more slowly.

This means that the transaction processing takes longer and therefore all the processing may be slowed down.

- A large capacity of CPU is used for the log output processing.

According to the specification in the system definitions, messages are passed to syslogfile of the OS or Job Management Partner 1 in addition to log output, and therefore the CPU usage tends to increase.

- Operation monitoring may be affected.

The program collecting messages for operation monitoring may not be able to handle the processing because of the large amount of messages output.

3.6.2 Command log

(1) Command log output format

When you execute an OpenTP1 operation command, information such as the times of command execution and termination are output to `cmdlog1` and `cmdlog2` in `$DCDIR/spool/cmdlog`, `cmdlog1` and `cmdlog2` wrap around when the size of data exceeds 1 MB.

You can use an editor such as `vi` to view `cmdlog1` and `cmdlog2`. Because the times when a command starts and ends are output in a command log, you can measure the length of time required to execute the command.

Information is output in the following format:

```
> AAAA BBBB C DDDD/DD/DD EE:EE:EE.EEEEEE FFFFFFFF : GGGGGGGG : HHHHHHH
```

>

Command log record start identifier. This symbol indicates the beginning of a line in the command log.

AAAA

Process ID of the executed command, in one-byte numerics

BBBB

User ID of the person who executed the command

C

Message sequence number in the process (sequence number of the command log information that is output by the executed command process). This is a one-byte number from 0 to 65,535. If the number exceeds 65,535, numbering starts again at 0.

DDDD/DD/DD

Date when information is output to the command log in the *YYYY/MM/DD* format, in one-byte numerics

EE:EE:EE.EEEEE

Time when information is output to the command log in the *HH:MM:SS.mmmmmm* format, in one-byte numerics. *mmmmmm* indicates a value in microseconds.

FFFFFFFF

Command line information at the time the command was executed. A command executed internally in OpenTP1 may be output here.

GGGGGGGG

Identification information, which is one of the following:

start

Information on starting the command.

end

Information on termination of the command.

info

Maintenance information generated during command execution

HHHHHHH

Maintenance information for the command

(2) Acquiring command logs

To acquire command logs for MCF-related operation commands, you need to assign a value to an environment variable. For details of which commands require the environment variable to be set, see *Table 12-1*. To set the environment variable:

- When executing operation commands manually

While logged in as an OpenTP1 system administrator, set the following environment variable:

Variable: DCMCFCMDLOG

Value: Y

If you do not assign a value to this environment variable, or assign a value other than Y, command logs will not be acquired.

- When executing operation commands automatically

In environments where operation commands are executed by issuing the `dc_adm_call_command` function from a UAP running on OpenTP1, add the following line to the user service definition of the relevant UAP, or to the user service default definition.

3. OpenTP1 Online Operation

```
putenv DCMCFCMDLOG Y
```

If you do not assign a value to the environment variable, or assign a value other than `Y`, command logs will not be acquired. The same applies to commands executed through COBOL and DML interfaces.

(3) Notes

Note the following regarding the acquisition of command logs for MCF-related operation commands:

- If a UAP or shell file executes a large number of commands concurrently, a bottleneck might develop in the command log output, reducing system performance. Do not set the `DCMCFCMDLOG` environment variable if this is likely to occur.
- If an operation command results in an error, a command log may be acquired for that command regardless of the setting of the `DCMCFCMDLOG` environment variable.

3.7 Audit log operations

An audit log is a file containing historical information about the operations performed on OpenTP1 programs by system developers, operators, and users, together with the program behavior triggered by those operations.

An audit log includes information about who executed what operation, at what time. This information can be used for a variety of auditing tasks, including evaluating system usage and discovering unauthorized access.

3.7.1 Setting up the environment for audit log acquisition

(1) Creating definitions

To output audit logs, specify the following definitions:

- Specify `Y` for the `log_audit_out` operand in the log service definition.
- Specify the message IDs to be output to the audit log in the `log_audit_message` operand in the log service definition.

Store the log service definition you created in the directory specified by the `DCCONFPATH` environment variable in the system environment definition (`$DCDIR/conf/env`). If no value is assigned to the `DCCONFPATH` environment variable, system definitions are stored in the directory `$DCDIR/conf`.

For details about the log service definition, see the manual *OpenTP1 System Definition*.

(2) Creating the execution environment

When you execute the `dcauditsetup` command after creating the log service definition, the directories and files required by the audit log function are created. The following table lists the directories and files that are created.

Table 3-10: Directories and files created when the `dcauditsetup` command is executed

File or directory	User ID	Group ID	Access permission	Description
<code>\$DCDIR/auditlog[#]</code>	User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0777	Directory for storing audit log files
<code>\$DCDIR/auditlog/audit.log[#]</code>	User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0666	Audit log file

[#]

This directory is used by default when the `log_audit_path` operand is unspecified in the log service definition. When the `log_audit_path` operand is specified, the specified directory is created and the audit log file is created in that directory.

Note that the `dcauditsetup` command only creates the lowest-level directory. Any higher-level directories must already be in place. If you also create the lowest-level directory in advance of executing the `dcauditsetup` command, assign the directory the access permission shown in the table above.

To change the definitions relating to audit logs after the execution environment is set up, perform the following steps:

1. Stop OpenTP1.
2. Change the definitions.
3. Execute the `dcauditsetup` command (as a superuser).
4. Execute the `dcreset` command.
5. Start OpenTP1.

If you change the `DCCONFPATH` environment variable in the system environment definition, perform the same procedure to apply the settings in the log service definition stored in the new directory specified by `DCCONFPATH`.

(3) Specifying which items to acquire as audit log information

Use the `log_audit_message` operand in the log service definition or user service definition to specify the types of information to be acquired as audit log information.

For details about the correspondence between the definitions and the message IDs of items to be acquired as audit log information, see the manual *OpenTP1 System Definition*.

If you change the value assigned to the `log_audit_message` operand in the log service definition, you must then execute the `dcauditsetup` command to apply the new setting.

(4) Examples of items to acquire as audit log information

The acquisition of large quantities of audit log entries can impair system performance. For this reason, acquire only those items you feel are necessary. We recommend that you acquire the following items as audit log information:

- Audit log information relating to OpenTP1 startup and shutdown (KFCA33400-I to KFCA33404-E)
- Audit log information relating to command execution (KFCA33419-I)
- User-defined audit log information acquired from a user server (KFCA34000-x to

KFCA34999-*x*)#

If you intend to acquire audit log information for user servers, first identify the user servers for which you want to keep an operation history. If you are unable to amend a particular user server program, you can keep an access record for the program by acquiring the messages that indicate execution of the service function has begun (KFCA33412-I) or completed (KFCA33413-I) as audit log information.

#

The message IDs from KFCA34000-*x* to KFCA34999-*x* are allocated to audit log information acquired from a UAP. The letter *x* is replaced by the message type (E, W, or I) as specified by the `dc_log_audit_print` function.

The following examples show how to specify the `log_audit_message` operand in a log service definition and user service definition to acquire the recommended audit log information.

Specification of `log_audit_message` operand in the log service definition

```
set log_audit_message=33400,33401,33402,33403,33404,33419
```

Specification in user service definition for a user server that will output audit log information

```
set log_audit_message=34000
```

(5) Caution when acquiring audit log information

The acquisition of large quantities of audit log entries can impair system performance. For this reason, only those items that are necessary for the auditing process should be acquired. We recommend that you evaluate your system's performance before using this feature.

3.7.2 Audit log output format

(1) Output destination

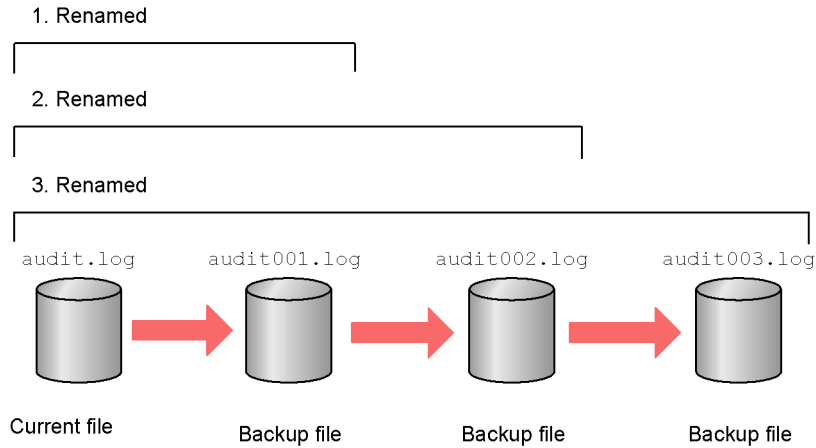
Audit logs are output to the destination specified by the `log_audit_path` in the log service definition, using a file rotation system. The files are output to the following destinations by default:

- In Windows
%DCDIR%\auditlog\audit.log
- In UNIX
\$DCDIR/auditlog/audit.log

Audit log information is output to the audit log file until the file reaches the size specified in the `log_audit_size` operand of the log service definition. The number of generations of audit log files that are created is determined by the `log_audit_count` operand in the log service definition.

The following figure shows the flow of audit log acquisition under a file rotation system. In this example, the value of the `log_audit_count` operand is 4.

Figure 3-5: Audit log acquisition by file rotation



1. When the size of the current file (`audit.log`) with the accumulated audit log information reaches the value specified for the `log_audit_size` operand, the current file is renamed `audit001.log`.
2. When the new current file again reaches the size specified for the `log_audit_size` operand, the file `audit001.log`, which is a backup of the old current file, is renamed `audit002.log`. The new current file is then renamed `audit001.log`.
3. When yet another current file reaches the size specified for the `log_audit_size` operand, the backup file `audit002.log` is renamed `audit003.log`, and `audit001.log` is renamed `audit002.log`. The current file is then renamed `audit001.log`.

When the number of files exceeds the value specified for the `log_audit_count` operand, the oldest backup file is deleted.

(2) Errors in audit log acquisition

When an error occurs that causes audit log output to fail, a message reporting this fact is output to standard error output and `syslog`.

(3) Cautions when rotating audit log files

If a process executed by a user who is not root or an OpenTP1 system administrator results in the creation of a new audit log file, the user who executed the process becomes the owner of the new file.

3.7.3 Example of audit log file estimation

The more log file generations there are, the more overhead will be associated with the rotation of the destination file. You can keep the number of generations relatively small by specifying a large size for the audit log file, to reduce how often file rotation takes place.

The following table shows examples of estimating the file size for an audit log file intended to remain in use for seven days.

Table 3-11: Examples of audit log file estimation

Example of audited system	Acquired audit logs	Size added to audit log file per day	Maximum size of audit log file value specified in the log_audit_size operand
For auditing primarily of operations related to starting and stopping OpenTP1 and user servers: <ul style="list-style-type: none"> OpenTP1 startup and shutdown: Once per day User server startup and shutdown: Once per day 100 servers 	<ul style="list-style-type: none"> KFCA33400-I (startup of OpenTP1) KFCA33402-I (normal termination of OpenTP1) KFCA33405-I (startup of user server) KFCA33406-I (normal termination of user server) 	60 KB (approx.)	1 MB
For auditing primarily of operations related to executing services (on a subset of the servers in the OpenTP1 system) <ul style="list-style-type: none"> Number of user servers to be monitored: 5 Number of service functions executed per server: 60/hour Daily uptime: 8 hours Number of times audit logs are acquired: Once per service 	KFCA34000-I (a size of 400 bytes is assumed for each optional log and log message issued by UAPs)	1 MB (approx.)	10 MB

Example of audited system	Acquired audit logs	Size added to audit log file per day	Maximum size of audit log file value specified in the log_audit_size operand)
Auditing primarily for operations related to executing services (on all the servers in the OpenTP1 system) <ul style="list-style-type: none"> • Number of active user servers: 100 • Number of service functions executed per server: 60/hour • Daily uptime: 8 hours • Number of times audit logs are acquired: Once per service 	KFCA34000-I (a size of 600 bytes is assumed for each optional log and log message issued by UAPs)	28 MB (approx.)	200 MB

Note:

- Estimate the size of the audit log file and the number of generations after estimating the quantity of audit log information produced per unit of time.
- For details of the quantity of audit log information output for each event, see *C. Information Output for Audited Events*.

3.7.4 Information output to audit logs

(1) Audit log output format

Entries are output to an audit log in the following format:

```
CALFHM 1.0, output-item-1=value-1, output-item-2=value-2, ... output-item-n=value-n
```

The string CALFHM 1.0 serves as header information, and is output for all audit log entries.

(2) Output example and output items

The following is an example of audit log output:

```
CALFHM 1.0, seqnum=1, msgid=KFCA33400-I,
date=2007-10-30T16:09:59.884+09:00, progid=OpenTP1, compid=adm, pid=11600,
ocp:ipv4=192.112.100.10, ctgry=StartStop, result=Success,
subj:uid="tp1user", obj="smpl", op=Start, loc="/OpenTP1", msg="User tp1user
started OpenTP1(smpl)."
```

The following table lists the items entered in an audit log file.

Table 3-12: Items output to audit log file

Item name	Meaning	Content	Common or program-specific ^{#1}
seqnum	Sequence number	A process-specific sequence number assigned to audit logs	Common information
msgid	Message ID	The message ID	
date	Date and time	The date and time when the message was output, in the following format: <i>YYYY-MM-DDT</i> <i>hh:mm:ss</i> <i>ZZ</i> <i>YYYY</i> : Year <i>MM</i> : Month <i>DD</i> : Day <i>T</i> : delimiter between date and time <i>hh</i> : Hours <i>mm</i> : Minutes <i>ss</i> : Seconds <i>sss</i> : Milliseconds <i>TZD</i> : Timezone ^{#2}	
progid	Source program	The character string <code>OpenTP1</code>	
compid	Source component	The name of the component where the event occurred. Audit logs acquired from a UAP by an API which outputs audit logs have the format <i>*AA</i> , where <i>AA</i> is the value specified in the API. Audit logs that do not begin with <i>*</i> are output by OpenTP1.	
pid	Process ID	The ID of the process associated with the event	
ocp:host	Source location	The host name or IP address of the server where the event occurred	
ocp:ipv4			

3. OpenTP1 Online Operation

Item name	Meaning	Content	Common or program-specific ^{#1}
ctgry	Event category	<p>The event category, as one of the following:</p> <ul style="list-style-type: none"> • StartStop: Indicates that a server or service has started or stopped. • Authentication: Indicates that user authentication was attempted. • ConfigurationAccess: Indicates that a setting or aspect of system configuration has been changed. • AccessControl: Indicates that a user has attempted to access a managed resource, and the attempt failed or was successful. • Failure: Indicates a software error. • LinkStatus^{#3}: Indicates the link status between devices. • ExternalService^{#3}: Indicates the result of communication with an external service. • ContentAccess: Indicates that a user has attempted to access critical data, and whether the attempt failed. • Maintenance: Indicates that a maintenance-related operation was executed, and whether the operation failed. • AnomalyEvent: Indicates that a communication error occurred. • ManagementAction^{#3}: Indicates execution of a critical action by a program, or an action triggered by another event. 	
result	Event result	<p>The result of the event, as one of the following:</p> <ul style="list-style-type: none"> • Success: The event was successful. • Failure: The event was a failure. • Occurrence: There is no distinction between success or failure for the event. 	
subj:eid	Subject ID information	<p>The user or process that caused the event, as one of the following:</p> <ul style="list-style-type: none"> • User name (the user ID of the OS account) • Process ID 	
subj:pid			

Item name	Meaning	Content	Common or program-specific ^{#1}
obj	Object information	Information identifying the target of the operation that generated the event	Program-specific information
op	Action information	<p>The type of action that generated the event, as one of the following:</p> <ul style="list-style-type: none"> • Start: A program started. • Stop: A program stopped. • Login: Login occurred. • Logout^{#3}: Logout occurred. • Logon^{#3}: Logon occurred. • Logoff^{#3}: Logoff occurred. • Refer: A setting was referenced. • Add^{#3}: A setting was added. • Update^{#3}: A setting was updated. • Delete: A setting was deleted. • Occur: An error or the like occurred. • Enforce: Processing was enforced. • Up^{#3}: A link became active. • Down^{#3}: A link became inactive. • Request^{#3}: A request was issued. • Response^{#3}: A response was issued. • Send^{#3}: Information was sent. • Receive^{#3}: Information was received. • Install^{#3}: A program was installed. • Uninstall^{#3}: A program was uninstalled. • Backup^{#3}: A backup was taken. • Maintain: A maintenance task was performed. • Invoke^{#3}: A system administrator or the like called a function. • Notify^{#3}: A system administrator or the like was issued a notification. 	
objloc	Object location information	Information about the location of the object	
from:host	Request source host	When the event involves multiple programs, the host name or IP address where the request originated	
from:ipv4			

Item name	Meaning	Content	Common or program-specific ^{#1}
from:port	Request source port	When the event involves multiple programs, the port number where the request originated	
to:host	Request destination host	When the event involves multiple programs, the host name or IP address where the request was directed	
to:ipv4			
to:port	Request destination port	When the event involves multiple programs, the port number where the request was directed	
loc	Location information	The information set in the DCDIR environment variable	
msg	Message	A message describing the nature of the event	

#1

All the output items categorized as common information are output to the audit log. Items categorized as program-specific information may or may not be output depending on the particular circumstances.

#2

The time zone is expressed as an offset from UTC. The following explains how to interpret the time zone:

+hh:mm

Indicates a time zone *hh:mm* ahead of UTC.

-hh:mm

Indicates a time zone *hh:mm* behind UTC.

Z

Indicates a time zone equivalent to UTC.

Japan Standard Time appears as +09:00.

#3

This information is output only when an API that outputs audit logs is used to acquire audit log information from a UAP.

For details about which items are output for each type of event, see *C. Information Output for Audited Events*.

3.7.5 List of audited events and their triggers

This section gives a list of the events that can result in the output of audit log information (called *audited events*), and the point during the event when the audit log information is output.

(1) List of audited events

The following table lists the events that can be entered in an audit log file.

Table 3-13: List of audited events

Audited event	Message IDs
OpenTP1 startup	KFCA33400-I
OpenTP1 standby	KFCA33401-I
Normal termination of OpenTP1	KFCA33402-I
Abnormal termination of OpenTP1	KFCA33403-E
Critical error in process service	KFCA33404-E
User server startup	KFCA33405-I
Normal termination of user server	KFCA33406-I
Abnormal termination of user server	KFCA33407-E
User server shutdown	KFCA33408-I
Service shutdown on user server	KFCA33409-I
Successful client user authentication	KFCA33410-I
Unsuccessful client user authentication	KFCA33411-W
Service function started execution	KFCA33412-I
Service function completed execution	KFCA33413-I
Invalid message discarded	KFCA33414-W
RPC call completed	KFCA33415-I
RPC response received (when using the <code>dc_rpc_poll_any_replies</code> function)	KFCA33416-I
Invalid RAP message discarded	KFCA33417-W
Error accessing the OpenTP1 file system	KFCA33418-W
Command execution	KFCA33419-I

Audited event	Message IDs
Startup of OpenTP1 service	KFCA33420-I
Termination of OpenTP1 service	KFCA33421-I
User-specific information acquired from a UAP	KFCA34000-x to KFCA34999-x [#]

For details about these messages, see the manual *OpenTP1 Messages*.

#

The message IDs from KFCA34000-x to KFCA34999-x are allocated to optional audit log information acquired from a UAP. x substitutes for the message type (E, W, or I) as specified by the `dc_log_audit_print` function.

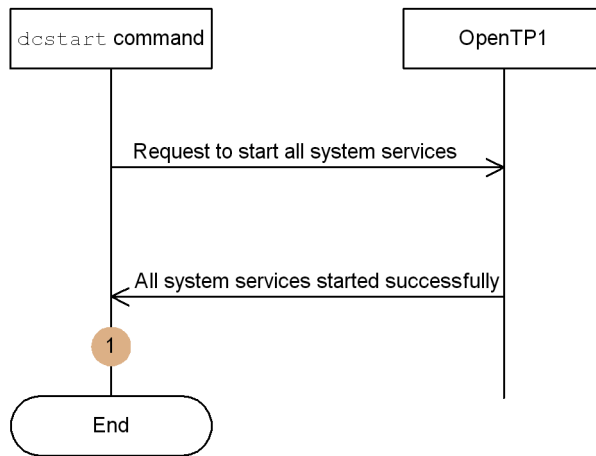
(2) Output points for audit log information


The following describes the particular stage during each event at which the information is output to the audit log.

(a) OpenTP1 startup

The following figure shows the point at which an entry is output to the audit log during the OpenTP1 startup process.

Figure 3-6: Point of audit log output (OpenTP1 startup)



Legend:
 Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

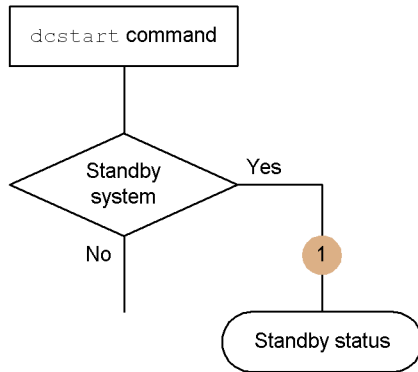
Table 3-14: Correspondence between output point and message ID for an audit log entry associated with OpenTP1 startup

Audited event	Number in figure	Message ID of output audit log information
OpenTP1 startup	1	KFCA33400-I


(b) OpenTP1 standby

The following figure shows the point at which an entry is output to the audit log when OpenTP1 enters standby status.

Figure 3-7: Point of audit log output (OpenTP1 standby)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

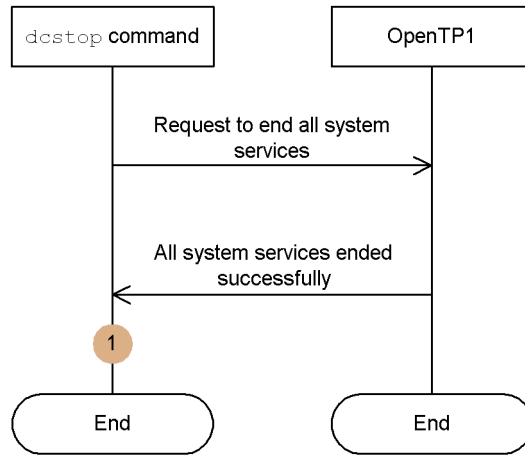
Table 3-15: Correspondence between output points and message IDs for an audit log entry associated with OpenTP1 standby

Audited event	Number in figure	Message ID of output audit log information
OpenTP1 standby	1	KFCA33401-I


(c) Normal termination of OpenTP1

The following figure shows the point at which an entry is output to the audit log when OpenTP1 terminates normally.

Figure 3-8: Point of audit log output (normal termination of OpenTP1)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

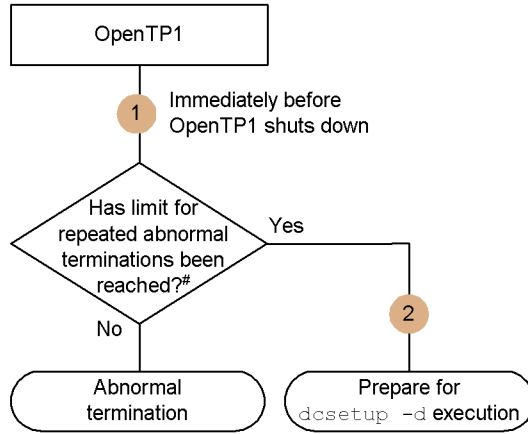
Table 3-16: Correspondence between output points and message IDs for an audit log entry associated with normal termination of OpenTP1

Audited event	Number in figure	Message ID of output audit log information
Normal termination of OpenTP1	1	KFCA33402-I

(d) Abnormal termination of OpenTP1 or critical error in process service

The following figure shows the point at which an entry is output to the audit log when OpenTP1 terminates abnormally or a critical error occurs that affects the process service.

Figure 3-9: Point of audit log output (abnormal termination of OpenTP1 or critical error in process service)



Legend:

Point of audit log output

#: For information about the limit for repeated abnormal terminations, see the description of the `term_watch_count` operand in the process service definition in the manual *OpenTP1 System Definition*.

The following table lists the points of audit log output and the corresponding message IDs.

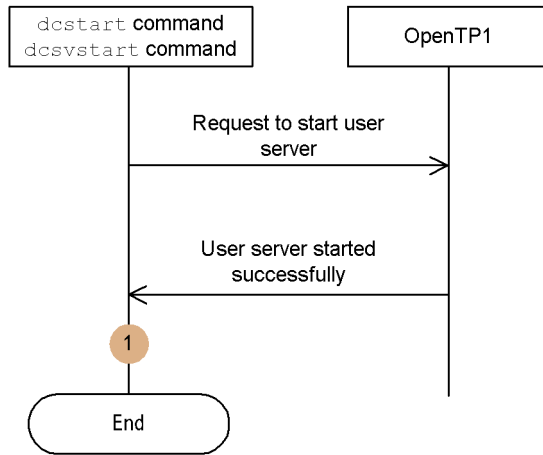
Table 3-17: Correspondence between output points and message IDs for an audit log entry associated with abnormal termination of OpenTP1 and critical errors in the process service

Audited event	Number in figure	Message ID of output audit log information
Abnormal termination occurred OpenTP1	1	KFCA33403-E
Critical error in process service	2	KFCA33404-E


(e) User server startup

The following figure shows the point at which an entry is output to the audit log during startup of a user server.

Figure 3-10: Point of audit log output (user server startup)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

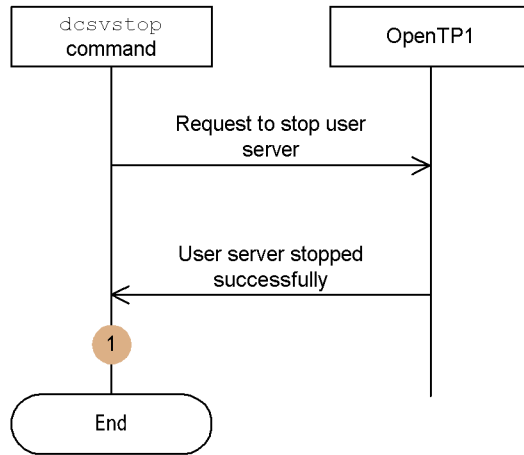
Table 3-18: Correspondence between output points and message IDs for an audit log entry associated with user server startup

Audited event	Number in figure	Message ID of output audit log information
User server startup	1	KFCA33405-I


(f) Normal termination of user server

The following figure shows the point at which an entry is output to the audit log when a user server terminates normally.

Figure 3-11: Point of audit log output (normal termination of user server)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

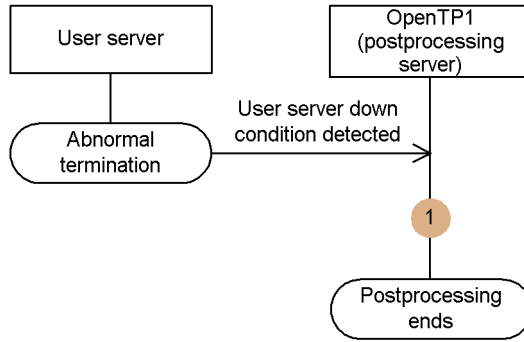
Table 3-19: Correspondence between output points and message IDs for an audit log entry associated with normal termination of user server

Audited event	Number in figure	Message ID of output audit log information
Normal termination of user server	1	KFCA33406-I


(g) Abnormal termination of user server

The following figure shows the point at which an entry is output to the audit log when a user server terminates abnormally.

Figure 3-12: Point of audit log output (abnormal termination of user server)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

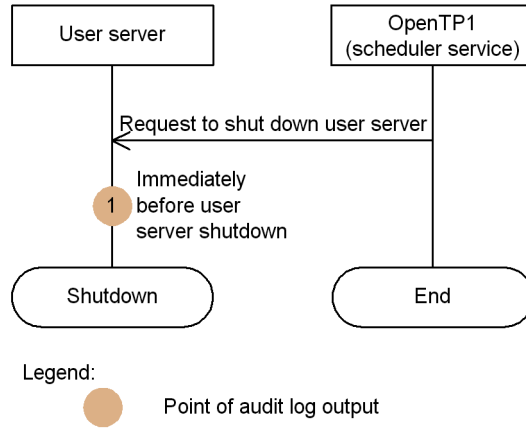
Table 3-20: Correspondence between output points and message IDs for an audit log entry associated with abnormal termination of user server

Audited event	Number in figure	Message ID of output audit log information
Abnormal termination of user server	1	KFCA33407-E

(h) User server shutdown or service shutdown on user server

The following figure shows the points at which an entry is output to the audit log when a user server is shut down, or a service on a user server is shut down.

Figure 3-13: Point of audit log output (user server shutdown or service shutdown on user server)



The following table lists the points of audit log output and the corresponding message IDs.

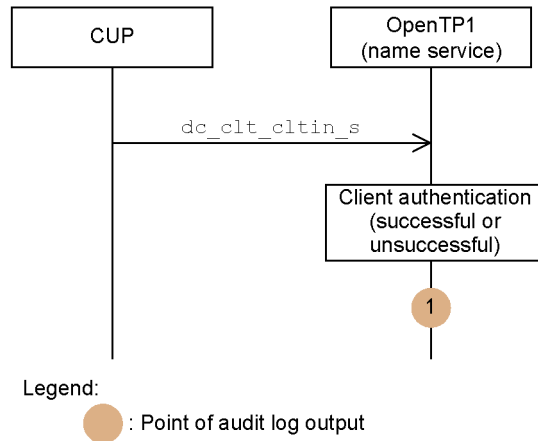
Table 3-21: Correspondence between output points and message IDs for an audit log entry associated with user server shutdown or service shutdown on user server

Audited event	Number in figure	Message ID of output audit log information
User server shutdown	1	KFCA33408-I
Service shutdown on user server	1	KFCA33409-I

(i) Successful or unsuccessful client user authentication

The following figure shows the points at which an entry is output to the audit log when a client user performs authentication and succeeds, or attempts authentication but fails.

Figure 3-14: Point of audit log output (successful or unsuccessful client user authentication)



The following table lists the points of audit log output and the corresponding message IDs.

Table 3-22: Correspondence between output points and message IDs for an audit log entry associated with successful or unsuccessful client user authentication

Audited event	Number in figure	Message ID of output audit log information
Successful client user authentication	1	KFCA33410-I
Unsuccessful client user authentication	1	KFCA33411-W

(j) RPC execution

The points at which an entry is output to the audit log for synchronous-response RPC, asynchronous-response RPC, and non-response RPC are shown in Figures 3-15, 3-16, and 3-17, respectively.

Figure 3-15: Point of audit log output (synchronous-response RPC)

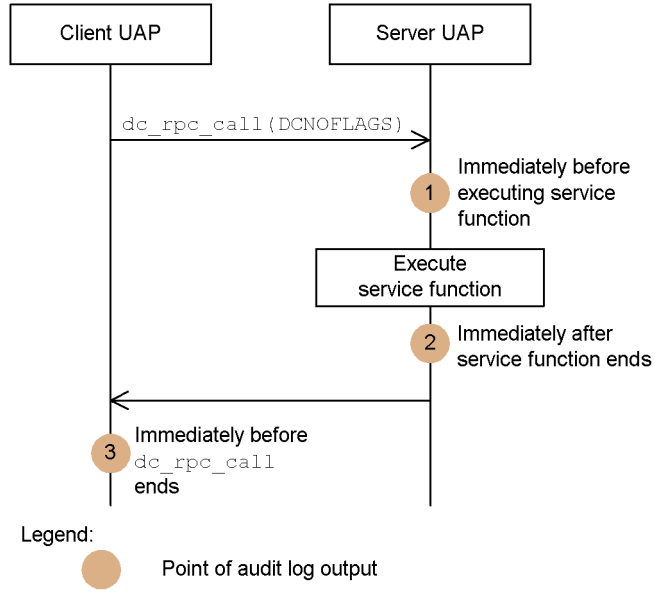


Figure 3-16: Point of audit log output (asynchronous-response RPC)

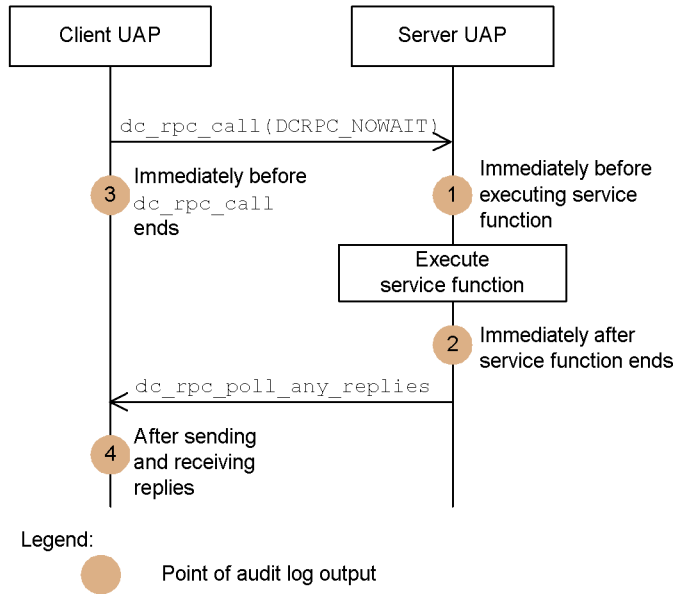
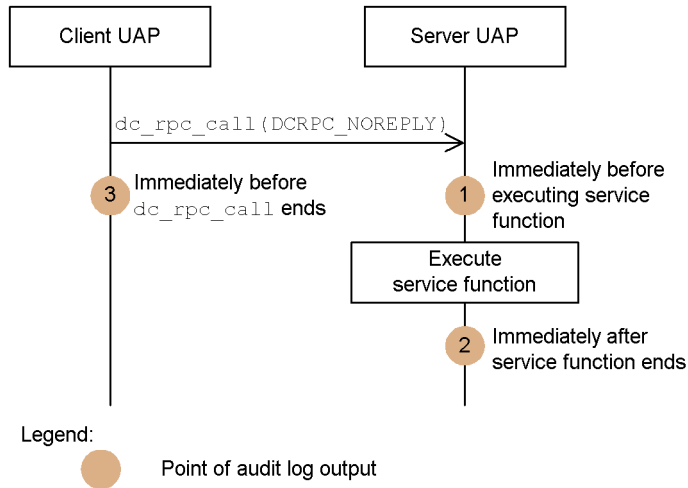


Figure 3-17: Point of audit log output (non-response RPC)



The following table lists the points of audit log output and the corresponding message IDs.

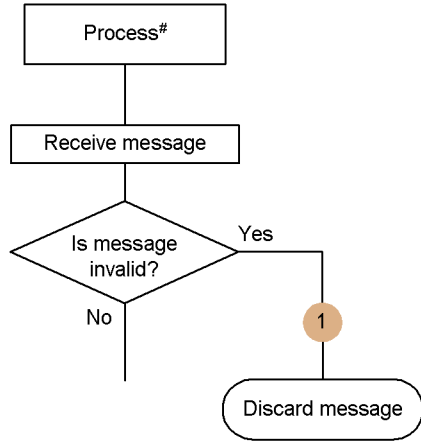
Table 3-23: Correspondence between output points and message IDs for an audit log entry associated with RPC execution

Audited event	Number in figure	Message ID of output audit log information
Service function started execution	1	KFCA33412-I
Service function completed execution	2	KFCA33413-I
RPC call completed	3	KFCA33415-I
RPC response received (when using the <code>dc_rpc_poll_any_replies</code> function)	4	KFCA33416-I

(k) Invalid message discarded

The following figure shows the point at which an entry is output to the audit log when an invalid message is discarded.

Figure 3-18: Point of audit log output (invalid message discarded)



Legend:
 Point of audit log output

#: User server, RAP-processing server, or system service

The following table lists the points of audit log output and the corresponding message IDs.

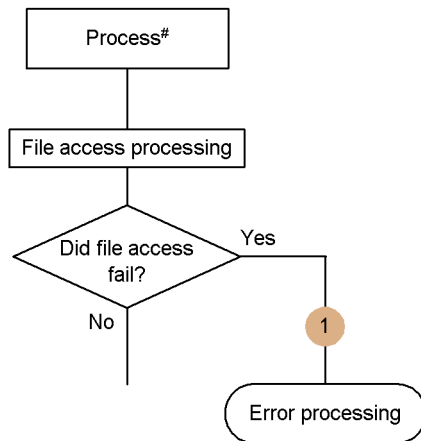
Table 3-24: Correspondence between output points and message IDs for an audit log entry associated with discarding invalid messages

Audited event	Number in figure	Message ID of output audit log information
Invalid message discarded	1	KFCA33414-W
Invalid RAP message discarded	1	KFCA33417-W


(I) Error accessing the OpenTP1 file system

The following figure shows the point at which an entry is output to the audit log when an error occurs while accessing the OpenTP1 file system.

Figure 3-19: Point of audit log output (error accessing the OpenTP1 file system)



Legend:

 Point of audit log output

#: User server, system service, or command

The following table lists the point of audit log output and the corresponding message ID.

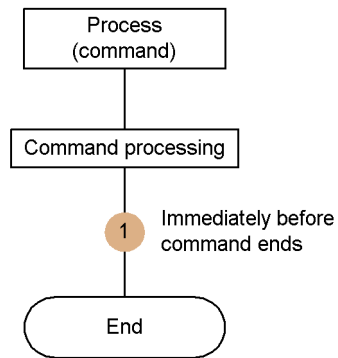
Table 3-25: Correspondence between output points and message IDs for an audit log entry associated with OpenTP1 file system access errors

Audited event	Number in figure	Message ID of output audit log information
Error accessing the OpenTP1 file system	1	KFCA33418-W


(m) Command execution

The following figure shows the point at which an entry is output to the audit log when a command is executed.

Figure 3-20: Point of audit log output (command execution)



Legend:

 : Point of audit log output

The following table lists the point of audit log output and the corresponding message ID.

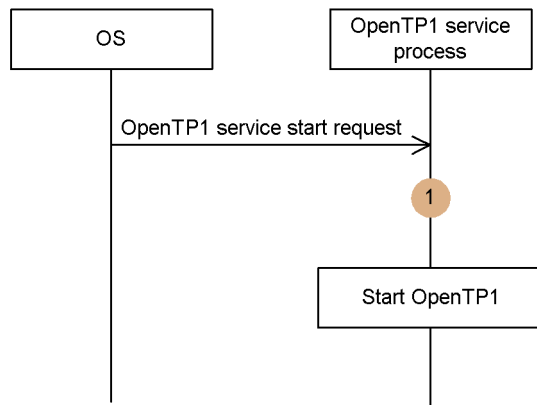
Table 3-26: Correspondence between output points and message IDs for an audit log entry associated with command execution


Audited event	Number in figure	Message ID of output audit log information
Command execution	1	KFCA33419-I

(n) OpenTP1 service startup

The following figure shows the point at which an entry is output to the audit log when the OpenTP1 service starts.

Figure 3-21: Point of audit log output (OpenTP1 service startup)



Legend:
 Point of audit log output

The following table shows the point of audit log output and the corresponding message ID.

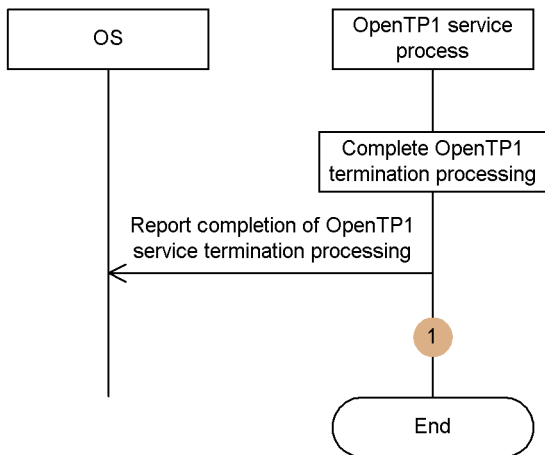
Table 3-27: Correspondence between output point and message ID for an audit log entry associated with OpenTP1 service startup


Audited event	Number in figure	Message ID of output audit log information
Startup of OpenTP1 service	1	KFCA33420-I

(o) OpenTP1 service termination

The following figure shows the point at which an entry is output to the audit log when the OpenTP1 service stops.

Figure 3-22: Point of audit log output (OpenTP1 service termination)



Legend:
 Point of audit log output

The following table shows the point of audit log output and the corresponding message ID.

Table 3-28: Correspondence between output point and message ID for an audit log entry associated with the OpenTP1 service service

Audited event	Number in figure	Message ID of output audit log information
Termination of OpenTP1 service	1	KFCA33421-I

3.7.6 Examples of audit log operation

An auditing policy that utilizes audit logs requires that you retain all the audit logs produced by the system. For this reason, backups must be taken of any audit log files deleted in the course of file rotation, and measures such as stopping the system must be put in place in case audit log acquisition fails. We recommend that these tasks be carried out automatically by using an operation management program such as JP1. By using JP1/NETM/Audit, audit logs located on multiple hosts can be automatically collected and centrally managed.

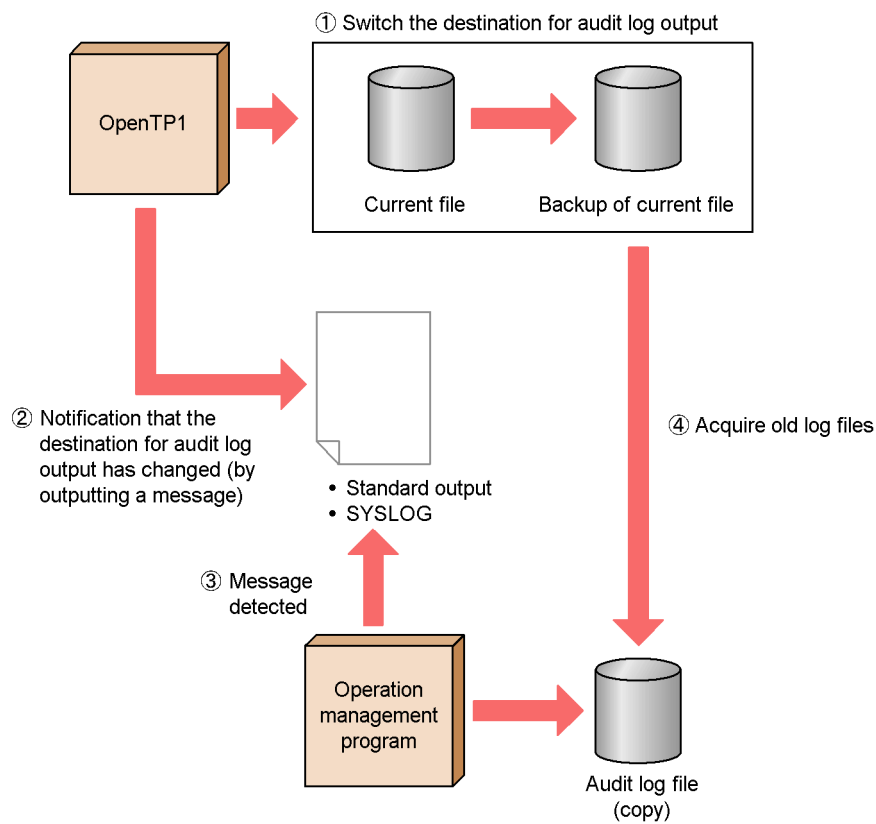
This section describes examples of working with audit logs by linking with an operation management program such as JP1.

(1) Example of operation in which audit logs are backed up automatically

When the destination file for audit log output is changed, a message reporting this fact is output to standard output and syslog. An operation management program such as

JP1 can monitor for this message and back up files automatically when it appears. The following figure shows the process of automatic audit log backup using an operation management program.

Figure 3-23: Flow of automatic audit log backup



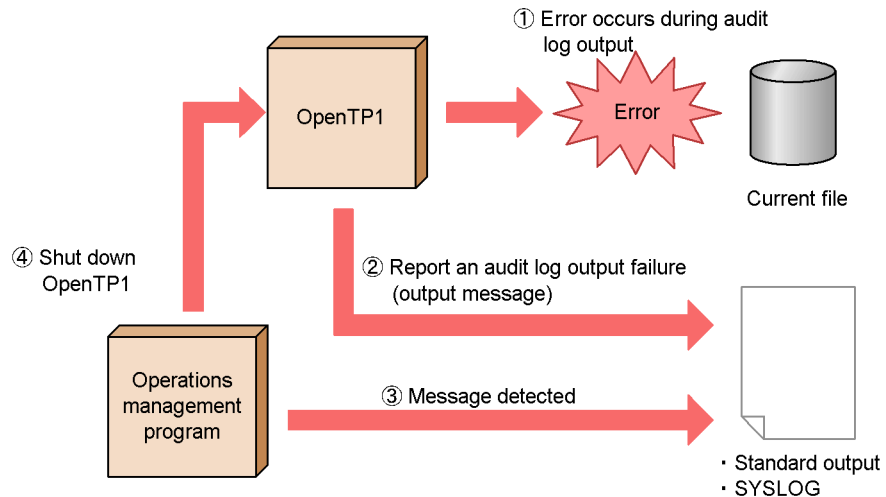
1. The file serving as the output destination for audit logs (current file) is swapped out and becomes a backup file.
2. OpenTP1 outputs a message (KFCA01925-I) reporting that the output destination for audit logs has changed.
3. JP1 or another operation management program detects that the message has been output.
4. JP1 or another operation management program acquires the old audit log file.

(2) Example of operation in which the system is stopped when audit log output fails

An operation management program such as JP1 can monitor for messages that report failed attempts to output audit logs, and stop the system automatically if they appear.

The following figure shows the process of shutting down the system automatically when audit log output fails.

Figure 3-24: Flow of automatic system shutdown when audit log output fails

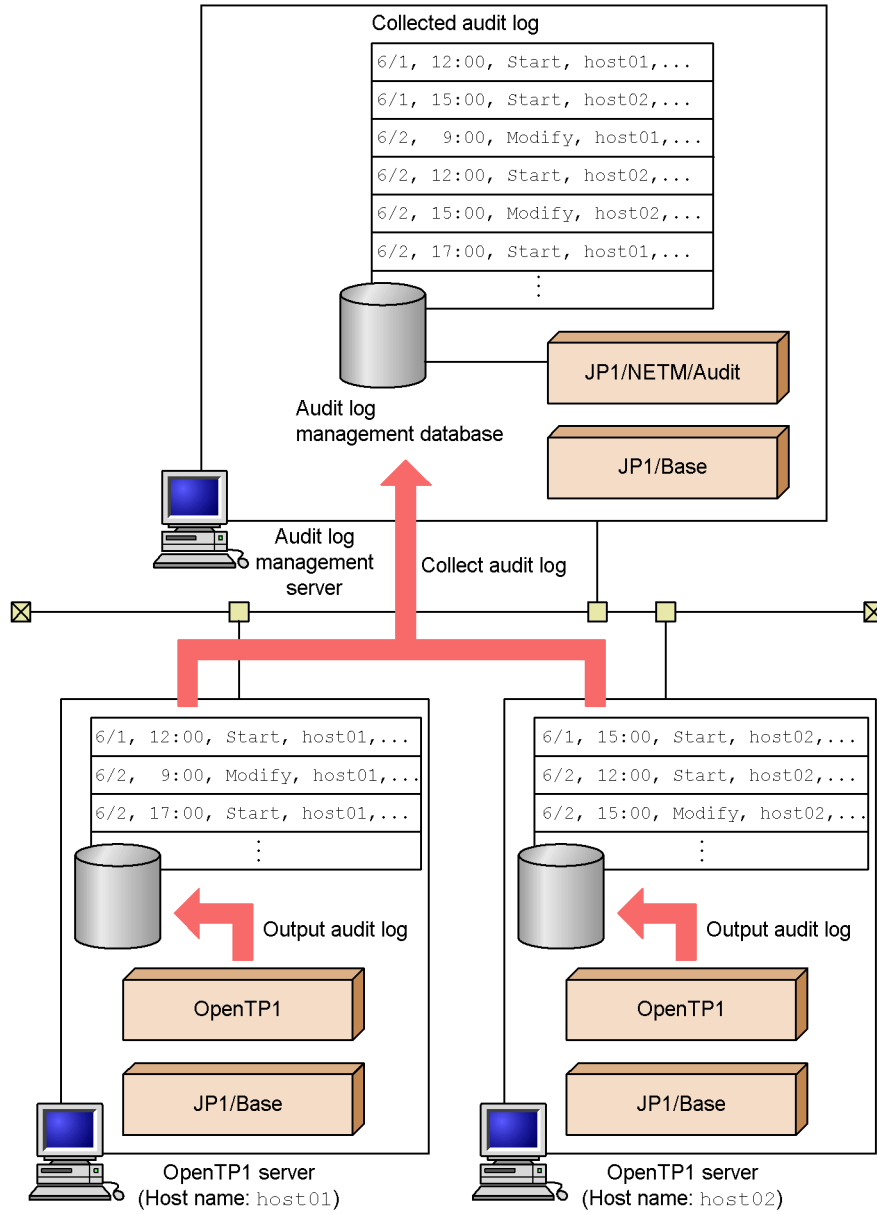


1. OpenTP1 fails in an attempt to output an audit log entry, due to an error.
2. OpenTP1 outputs an error message (KFCA01921-E to KFCA01924-E) reporting that audit log output has failed.
3. JP1 or another operation management program detects that the message has been output.
4. JP1 or another operation management program shuts down OpenTP1.

(3) Example of operation in which audit logs are collected automatically and centrally managed

By using JP1/NETM/Audit as an operation management program, you can collect audit logs automatically, and manage them from a centralized viewpoint. This provides a simple way to retrieve and summarize audit logs gathered from multiple servers, and output the results. The following figure shows an example of the process of collecting and centrally managing audit logs using JP1/NETM/Audit.

Figure 3-25: Example of collecting and centrally managing audit logs using JP1/NETM/Audit



Legend:

... ⋮ These symbols indicate omission of other audit log entries.

In this example, the audit logs output by a number of OpenTP1 servers to their respective disks are collected automatically by an audit log management server. The audit logs collected by the audit log management server are managed centrally as an audit log management database.

The following table shows the JP1 series products required to link with JP1/NETM/Audit.

Table 3-29: JP1 series products required for linkage with JP1/NETM/Audit

JP1 series product	Function	Location
JP1/NETM/Audit	Links with JP1/Base to collect the audit logs output by OpenTP1 servers. Also centrally manages the audit logs it collects as a database on an audit log management server.	Audit log management server
JP1/Base	Sends and receives the output audit logs as JP1 events.	<ul style="list-style-type: none"> • Audit log management server • OpenTP1 servers

The automatic collection of audit logs can take place at the following times:

- On a regular basis at a specified date and time
- Whenever the event database (of JP1/Base) begins a new file
- Immediately

The setup required on the OpenTP1 servers and the audit log management server is as follows:

Setup required on OpenTP1 servers

Perform the following setup on the OpenTP1 servers:

1. Set up JP1/Base.
2. Copy the JP1/Base adapter command file, and the definition files required for command execution, from the audit log management server to each OpenTP1 server where you intend to collect audit logs.
3. Start the event service.

Setup required on audit log management server

JP1/NETM/Audit support is provided as standard in OpenTP1, allowing you to set OpenTP1 as the target for collection of audit information by JP1/NETM/Audit.

The product definition file and operation definition file which JP1/NETM/Audit uses to collect log information from OpenTP1 are stored in the following locations:

3. OpenTP1 Online Operation

- Location of product definition file:

installation-directory/jpl_template/JP1_NETM_Audit/
OpenTP1.conf

- Location of operation definition file:

installation-directory/jpl_template/JP1_NETM_Audit/
admjevlog_OpenTP1.conf

For details about the collection and centralized management of audit logs by JP1/NETM/Audit, see the manual for JP1/NETM/Audit.

3.8 Trace operations

3.8.1 Output of UAP trace information

If a UAP terminated abnormally, UAP trace information is automatically edited and output to the file. The `uatdump` command outputs UAP trace information from a core file.

The output information includes the name of the service group to which the active service belongs, the UAP trace information collection dates and times, etc.

For details of UAP traces and the `uatdump` command, see the manual *OpenTP1 Tester and UAP Trace User's Guide*.

3.8.2 RPC trace operations

(1) Merging of RPC traces

The `rpcmrg` command merges trace information of more than one RPC trace file. Executing this command lists traces chronologically and outputs them in RPC trace file format.

(2) Output of RPC traces

The `rpcdump` command edits and outputs RPC traces from an RPC trace file.

Only as many traces as the user needs can be output by specifying appropriate options.

The output contents include the name of the service group to which the active service belongs, and the process ID for the RPC trace output source.

(3) Notes

When RPC traces are acquired, processing speed may decrease and the RPC may return an error due to a timeout. In such a case, perform either of the following actions:

- Issue the `dc_rpc_set_watch_time` function and increase the maximum time for waiting for a response (180 seconds when the function is not issued) to an appropriate value.
- Increase whichever of the maximum times for waiting for a response (default: 180 seconds) described in 1 or 2 below applies to the situation to an appropriate value:
 1. Maximum time for waiting for a response specified in the `watch_time` operand (specified in the system common definition, user service definition, or user service default definition)
 2. Maximum time for waiting for a response passed from the client to the RAP-processing server

You can use the `DCWATCHTIMINHERIT` operand (for TP1/Client/W and TP1/

Client/P) in the client environment definition or the `dcwatchtiminherit` operand (for TP1/Client/J) in the TP1/Client/J environment definition to specify whether to pass the maximum time for waiting for a response from the client to the RAP-processing server.

When Y is specified in the `DCWATCHTIMINHERIT` operand or the `dcwatchtiminherit` operand, use the maximum time for waiting for a response described in 2 above.

When N is specified in the `DCWATCHTIMINHERIT` operand or the `dcwatchtiminherit` operand, or the specification is omitted, use the maximum time for waiting for a response described in 1 above.

3.8.3 Swapping MCF trace information

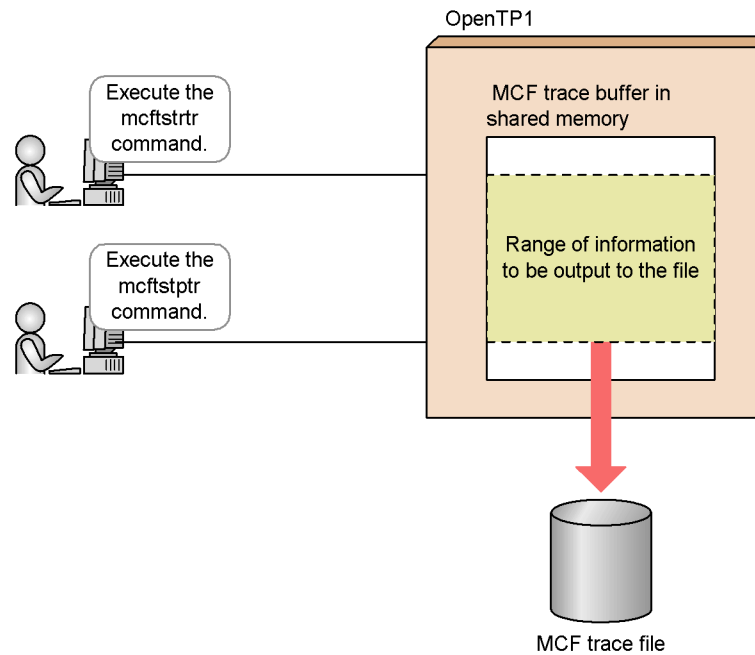
To output the latest MCF trace to the file, the MCF trace files must be swapped by the `mcftswptr` command.

3.8.4 Temporarily outputting MCF trace information

You can output the MCF trace information acquired in the shared memory to the MCF trace file at a desired time. To start acquiring MCF trace information, use the `mcftstrtr` command. To end the acquisition of MCF trace information and output the information to the MCF trace file, use the `mcftstprr` command. Use these commands mainly to temporarily acquire MCF trace information when the disk output feature for MCF traces is defined not to be used (`mcfttrc -t "disk=no"`).

The following figure gives an overview of outputting MCF trace information to the MCF trace file.

Figure 3-26: Overview of temporarily outputting MCF trace information



3.8.5 Performance verification trace operations

OpenTP1 acquires the performance verification trace for events of services operating on OpenTP1. The performance verification trace is used to improve efficiency of performance verification and troubleshooting.

The `prfget` command acquires the trace. The `prfed` command edits and outputs the acquired trace. The `prfget` command outputs the performance verification trace in binary format to a UNIX file. The `prfed` command converts the trace from binary format to character format and then edits and outputs the trace. The `prfed` command is used to view the details of the trace.

The features of the facility for acquiring the performance verification trace are as follows:

- You can trace events across nodes and processes.
- You can verify performance of processing by tracing each internal event, not by tracing each API.

You can use an output event ID to determine when the trace information is acquired. For the event IDs and the timing of acquiring the trace, see *L. Acquiring Performance Verification Traces*.

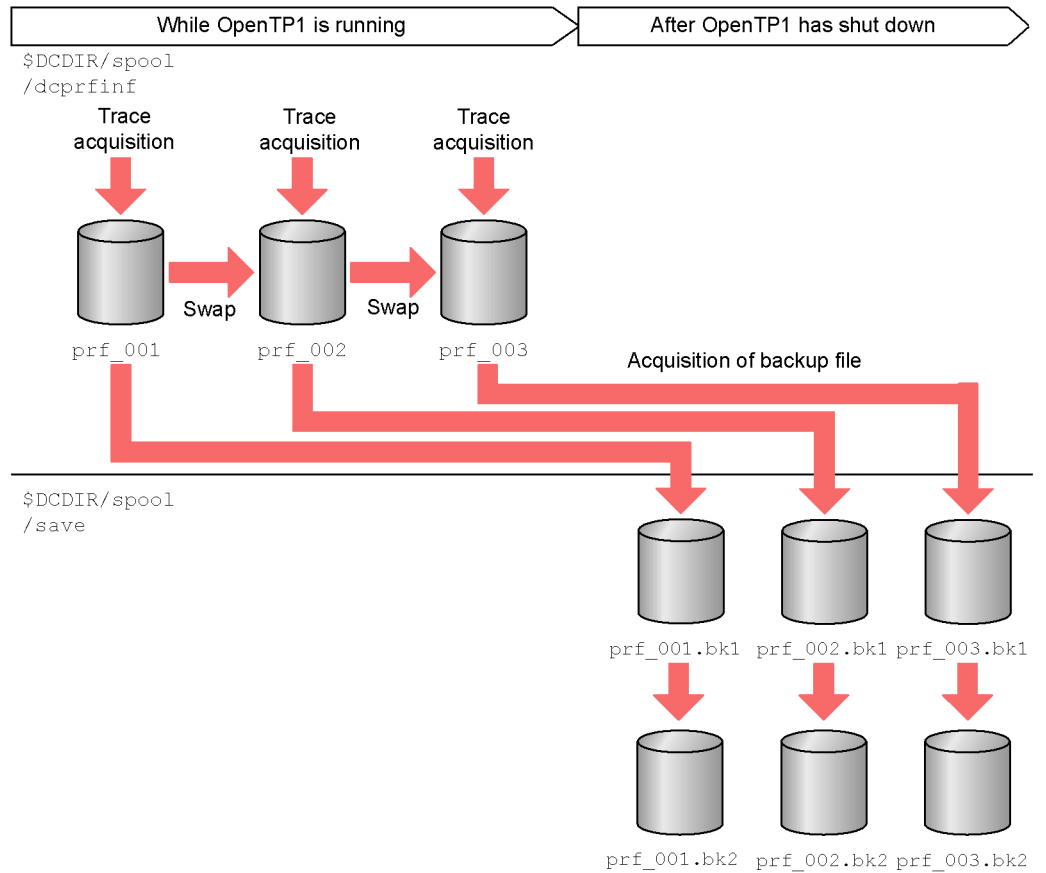
You must install TP1/Extension 1 to use this facility. Without TP1/Extension 1 installed, this facility may not operate correctly.

Notes

- Trace information is not passed for normal restart and hot standby.
- The facility for acquiring the performance verification trace does not perform locking when acquiring trace information so that online performance is not affected. Accordingly, if contention for trace acquisition occurs in a multi-processor environment, some trace information may be lost or invalid trace information may be acquired. When you use the `prfed` command to edit trace information, an error code is displayed for invalid trace information.

The following figure shows the flow of performance verification trace file operation by using the function that acquires the performance verification trace.

Figure 3-27: Flow of performance verification trace file operation



1. PRF trace data output during online operation is stored as files with the file name `prf_####1` under the directory `$DCDIR/spool/dcprfinf`. The `prf_file_size` operand specifies the maximum size of each file, and the `prf_file_count` operand specifies the number of files. When a file reaches its capacity during online operation, the file is swapped and subsequent traces are acquired to the new file.
2. When OpenTP1 shuts down, the files `prf_###.bk2#2` in `$DCDIR/spool/save` are moved to `prf_###.bk1.#2`
3. The files `prf_####1` in `$DCDIR/spool/dcprfinf` are copied to `$DCDIR/spool/save` under the name `prf_###.bk2.#2`

#1

nnn: A number, starting from 001 and having the value specified for the `prf_file_count` operand as its upper limit.

#2

nnn: Value corresponding to the file name at the backup source.

(1) Performance verification traces acquired by OpenTP1

The table below lists the trace information that OpenTP1 acquires for purposes of performance verification, collectively referred to as *performance verification traces*. The files acquired as performance verification trace data are called performance verification trace files.

Trace name	Performance verification trace file name
Performance verification trace	<code>prf_#nnn</code>
XAR performance verification trace	<code>_xr_#nnn</code>
JNL performance verification trace	<code>_jl_#nnn</code>
LCK performance verification trace	<code>_lk_#nnn</code>
MCF performance verification trace	<code>_mc_#nnn</code>
NAM event trace	<code>_nm_001, _nm_002, _nm_003</code>
Process service event trace	<code>_pr_001, _pr_002, _pr_003</code>
FIL event trace	<code>_fl_001, _fl_002, _fl_003</code>
TRN event trace	<code>_tr_#nnn</code>

#

nnn: Value ranging from 001 to the value specified in the `prf_file_count` operand in the following definition:

- Performance verification trace: Performance verification trace definition
- XAR performance verification trace: XAR performance verification trace definition
- JNL performance verification trace: JNL performance verification trace definition
- LCK performance verification trace: LCK performance verification trace definition

- MCF performance verification trace: MCF performance verification trace definition
- TRN event trace: TRN event trace definition

(2) Backing up performance verification trace files

The OpenTP1 functionality for acquiring performance trace data backs up the acquired trace files at OpenTP1 shutdown, to prevent the auditing process from being compromised in cases where trace data is deleted inadvertently. The advantages and disadvantages of taking such a backup are as follows:

- Advantages

Takes a backup of the acquired trace data when OpenTP1 shuts down, preventing the auditing process from being compromised by missing data.

- Disadvantages

- The data takes up disk space.
- OpenTP1 takes longer to shut down.

(3) Disabling backup of performance verification trace files

In some system configurations, the disadvantages of backing up performance verification trace files may outweigh the advantages. In this case, you can prevent OpenTP1 from backing up these files. However, as this means that performance verification trace files will not be backed up automatically, you must back up these files yourself when necessary.

Before you disable backup of performance verification trace files, we recommend that you take due care to ensure that sufficient data is available for troubleshooting and other activities.

3.9 Shared memory operations

3.9.1 Display of shared memory utilization status

The `dcshmls` command lets the user know the utilization status of the shared memory during OpenTP1 operation. Executing this command displays the total and utilization rate of shared memory in current use. The command also displays the utilization status of static or dynamic shared memory depending on the specification of the appropriate option.

3.9.2 Buffer pool for storing messages

OpenTP1 stores service requests from the client to the *buffer pool for storing messages*. Since this buffer pool is created in the shared memory for a user server, increase of the number of user servers for service processing may tax the system resources.

Specify the `scdbufgrp` operand in the schedule service definition to share one buffer pool for storing messages by more than one user server. This reduces the amount of shared memory usage. User servers which share one buffer pool for storing messages are called a *scheduling buffer group*.

Specify the `-b` option for the `scdls` command to display the status of the buffer pool for storing messages shared by the scheduling buffer group. The displayed information includes:

- Size of the buffer pool that remains available for storing messages
- Size of the currently used buffer pool for storing messages
- Maximum size of the buffer pool for storing messages
- User server names sharing the buffer pool for storing messages

The above information is displayed for each scheduling buffer group. The following information is displayed for each one of user servers sharing the buffer pool for storing messages:

- User server status
- Number of remaining service requests
- Maximum number of services and remaining service requests
- Size of the buffer pool for storing messages used by the user server
- Maximum size of the buffer pool for storing messages available for the user server

The system goes down if the amount of shared memory specified by the schedule service definition cannot be allocated at startup of OpenTP1. In this case, re-estimate

the value of memory.

3.10 OpenTP1 domain operations

A group of nodes specified by the `all_node` operand in the system common definition is called an *OpenTP1 domain*.

3.10.1 Changing the domain configuration

Use the `namndchg` command or the `namchgfl` command to change the nodes composing a domain while OpenTP1 is active. The following shows the procedure for changing the domain configuration for each command.

(1) Using the `namndchg` command to change the domain configuration

Change the `all_node` operand in the system common definition and execute the `namndchg` command to change the domain configuration. The procedure for changing the domain configuration is as follows:

1. Change the node configuration specified by the `all_node` operand in the system common definition.
2. Execute the `namndchg` command. When the command terminates normally, the new nodes are applied to compose the domain.

Specify the `-l` option for the `namndchg` command to display the names and port numbers of the current nodes of the domain (specification of the `all_node` operand).

(2) Using the `namchgfl` command to change the domain configuration

Change the domain configuration by using the domain definition file. You can use the `namchgfl` command to change the domain configuration without changing the system common definition. To enable the `namchgfl` command, specify `Y` in the `name_domain_file_use` operand in the system common definition. For details about the `name_domain_file_use` operand, see the manual *OpenTP1 System Definition*. The procedure for changing the domain configuration is as follows:

1. Create a domain definition file.
2. Execute the `namchgfl` command. When the command ends normally, the domain consists of new nodes. The global cache is initialized and registers nothing. To change the domain definition file for specifying the value of `all_node_ex`, specify the `-e` option in the `namchgfl` command.

If the same node is specified for both the old and new domain configurations, the following information about the node in the old configuration is inherited by the new domain configuration.

- Registration status of the RPC suppression list
- Time when connections are established

■ Domain definition files

You need *domain definition files* to change the domain configuration without changing the system common definition. There are three domain definition files, one for `all_node`, one for `all_node_ex`, and one for priority selection nodes. The method for specifying the definition file for priority selection nodes is the same as for the definition files for `all_node` and `all_node_ex`.

You can use any names for the files, but the length of a file name must not exceed 40 bytes. Store the domain definition files in the following directories:

Domain definition file for `all_node`:

`$DCCONFPATH/dcnamnd` directory

Domain definition file for `all_node_ex`:

`$DCCONFPATH/dcnamndex` directory

Domain definition file for priority selection nodes:

`$DCCONFPATH/dcnampr` directory

If a node name or port number is invalid in one of these files, the `namchgfl` command issues the message KFCA00656-E (invalid node name), KFCA00657-E (invalid port number), or KFCA00666-E (host undefined) and then returns control with an error, in which case the specified domain configuration change is ignored.

If a node specified in the definition file for priority selection nodes is not defined in the domain definition file for `all_node`, the command issues the KFCA00603-W message (undefined in `all_node`), in which case the specification of the priority selection node is ignored. If a priority selection node is defined correctly on another line in the same file, the specification as a priority selection node is effective.

If you store multiple domain definition files in each directory, one domain (OpenTP1 system) is created by combining all the domain definition files stored in the same directory.

In the cases below, the OpenTP1 system is created as a domain consisting only of the local node. The definition file for priority selection nodes depends on the information specified in the domain definition file.

- The domain definition files are not at the correct location.
- There is no directory containing domain definition files.
- No node is specified in a domain definition file.

The format for coding a domain definition file is as follows:

<code>node-name [, [port-number]] [,]</code>

node-name <identifier of 1 to 255 characters>

Specify the names of all the nodes in the OpenTP1 system. An identifier can consist of alphanumeric characters, the period, and the hyphen. The host names defined in `/etc/hosts` become node names.

port-number <unsigned integer>((5001 - 65535))

Specify the port number used by the name server as a well-known port number. If you omit a port number, OpenTP1 sets the port number of the name service specified in the `name_port` operand in the system common definition. If the `name_port` operand is not specified, OpenTP1 assumes 10000.

- You can specify a maximum of 272 bytes for one line.
- You can specify only one pair of node name and port number on one line.
- To enter a comment, add # at the beginning of the comment.

Beginning a line with # marks the entire line a comment.

- The continuation sign (\) used to indicate a continued line in the system definition is not required here.
- Characters after the second comma (,) are not checked for errors.

Cautions

- Specify a unique node identifier for a node specified in the domain definition file.
- If you delete a node from the old domain configuration, the following functions may not operate normally:
 - IST service
 - Multinode facility
 - Remote MCF facility
- If you are using a service that uses transactional RPCs, do not add a node that has the same node identifier as a node that is being deleted when the domain configuration is changed. If you specify a node that is being deleted, the following operations are not guaranteed:
 - Recovery processing
 - IST service
 - Multinode facility
 - Remote MCF facility
- If you use the `namchgfl` command in the system switchover configuration, store the same domain definition files in the running and standby systems in case a restart is required.

- Do not forcibly terminate the `namchgf1` command process by using the `kill` command or some other means while the `namchgf1` command is being executed. If you do, the name service facility may not operate normally.
- Operation is not guaranteed if the `name_domain_file_use` operand is changed before or after a restart.

The following is an example of coding a domain definition file:

```

0-----1-----2-----3-----4-----5-----6-----7-----+
10.209.117.30,11111,                                     #comment
10.209.117.40,33333,                                     #comment
                (linefeed)#
#zzz#
10.209.117.50,11111
10.209.117.60,33333
10.209.117.70
10.209.117.80,
10.209.117.90,,
                (blank line)#

```

#

OpenTP1 determines that specification continues even when it encounters a line with only a linefeed character, a blank line, or a line with only a comment.

(3) **Changing the number of nodes to be added**

Define an area of shared memory that can be used by the name service when OpenTP1 starts. Because you cannot change the size of this area while OpenTP1 is running, you need to secure the required size (determined by the maximum number of nodes) when OpenTP1 starts. You can specify the maximum number of nodes in the `all_node_extend_number` operand or the `all_node_ex_extend_number` operand in the system common definition. For details about these operands, see the manual *OpenTP1 System Definition*.

3.10.2 Domain representative schedule service

When OpenTP1 is used for construction of a large-scale distributed system, the OpenTP1 system can be divided into smaller domains. When dividing the OpenTP1 system into domains, only define the OpenTP1 nodes of the local domain in the `all_node` operand in the system common definition. There is no need to define all OpenTP1 nodes in the OpenTP1 system.

A schedule service used for an access point to the domain is called a *domain representative schedule service*. It must be registered to the domain data file (hosts information file of DNS). Use the `namdomainssetup` command to register or delete the service from the domain data file.

To perform inter-domain communication, use the `dc_rpc_call()` specifying domain qualification for the service group name. In inter-domain communication,

OpenTP1 sends a query to DNS about the IP address of the specified domain representative schedule service, and passes the service request to that domain representative schedule service. The domain representative schedule service that accepts the service request schedules it for the server UAP in the local domain.

Specify the `domain_masters_addr` operand in the system common definition to directly communicate with the destination domain representative schedule service without sending a query about the IP address to DNS. This reduces the amount of communication. For how to define the domain representative schedule service, see the manual *OpenTP1 System Definition*.

3.10.3 Invalidating the startup notification information

To perform a forced invalidation of the startup notification to the communication target OpenTP1 that is specified using the `name_notify` operand in the system common definition, use the `namunavl` command. When the `namunavl` command is executed, a new connection is established. When a connection is established, OpenTP1 can communicate normally even if it does not receive the startup notification information from the communication target.

3.10.4 Verifying OpenTP1 startup and clearing the cache

Using the `namalivechk` command, you can verify the status of the other OpenTP1 nodes in the OpenTP1 system at the time the command was executed.

If the command detects that an OpenTP1 node has failed, it deletes all of the service information for that node from the cache. Subsequent remote procedure calls will not communicate with the failed node, thereby preventing unnecessary communication attempts. By executing the `namalivechk` command whenever an OpenTP1 node in the OpenTP1 system fails, you can ensure that remote procedure calls are performed efficiently.

3.11 Operations using TP1/NET/OSI-TP-Extended

You can use TP1/NET/OSI-TP-Extended to achieve client/server communications using the OSI communication protocol.

For details about the environment settings and commands for using TP1/NET/OSI-TP-Extended, see the manual *OpenTP1 Protocol TP1/NET/OSI-TP-Extended*.

3.12 Operations using the remote API facility

3.12.1 Preparations for using the remote API facility

The remote API facility needs the following preparations.

(1) Setup of the execution environment

- On the RAP-processing server
Use the `rapsetup` command to set up the execution environment of the remote API facility.
- On the RAP-processing client
When using the RAP-processing client manager function, also execute the `rapsetup` command on the RAP-processing client to set up the execution environment of the remote API facility.

(2) Creating definitions

- On the RAP-processing server
Create the RAP-processing listener service definition.
- On the RAP-processing client
Create the user service definition and the user service network definition. In the user service network definition, include the `dcsvgdef` definition command with the `-w` option specified to indicate that communication is to take place via the remote API facility.

Example definition:

```
dcsvgdef -g service-group-name -h host-name -p port-number -w
```

If you are using the RAP-processing client manager facility, also create the RAP-processing client manager service definition.

For details about the definitions, see the manual *OpenTPI System Definition*.

(3) Automatic generation of definitions for using the remote API facility

- On the RAP-processing server
Execute the `rapdfgen` command, specifying the RAP-processing listener service definition you created in (2) above as the input file. This creates the RAP-processing listener user service definition and the RAP-processing server user service definition.

The command outputs generated definitions to be under the `$DCCONFPATH` directory.

Example of command execution:

```
> rapdfgen RAP-processing-listener-service-definition-file-name
```

- On the RAP-processing client

If you are using the RAP-processing client manager facility, execute the `rapdfgen` command specifying the RAP-processing client manager service definition you created in (2) above as the input file. This creates the RAP-processing client manager user service definition under the `$DCCONFPATH` directory.

Example of command execution:

```
> rapdfgen -m
RAP-processing-client-manager-service-definition-file-name
```

3.12.2 Displaying the status of a RAP-processing listener or a RAP-processing server

You can use the `rapls` command to display the status of a RAP-processing listener or a RAP-processing server.

3.12.3 Performance improvement of the remote API facility

When sending or receiving messages using the remote API facility, the response may be deteriorated and the performance may be degraded depending on the length of the messages to be exchanged. The remote API facility provides the parameters for tuning the performance to improve the response. By using these parameters, increase the socket window size and make the send and receive retry interval shorter. By increasing the window size and shortening the interval you can improve the response.

The users must determine whether or not to apply the parameters for tuning the performance according to the system requirements. To set the performance tuning parameters, perform the following procedure:

1. Specify the following operands of the RAP-processing listener service definition:
 - `rap_max_buff_size` operand
 - `rap_io_retry_interval` operand
 - `rap_sock_count` operand
 - `rap_sock_interval` operand
 - `rap_connect_retry_count` operand
 - `rap_connect_retry_interval` operand
 - `rap_listen_backlog` operand

For details about the operands, see the manual *OpenTP1 System Definition*.

2. After specifying the operands, execute the `rapdfgen` command.

3.12.4 Starting and stopping the RAP-processing listener and RAP-processing server

To start the RAP-processing listener and RAP-processing server, execute the `dcsvstart` command. Alternatively, you can start the RAP-processing listener and RAP-processing server by starting OpenTP1 with the `dcsvstart` definition command specified in the user service structure definition.

If you execute the `dcsvstart` command with the name of the RAP-processing listener specified as a command argument, the command starts both the RAP-processing listener and the RAP-processing server.

Example of command execution:

```
> dcsvstart -u RAP-processing-listener-name
```

To stop the RAP-processing listener and RAP-processing server, execute the `dcsvstop` command. You can also stop the RAP-processing listener and RAP-processing server by using the `dcstop` command to shut down OpenTP1.

If you execute the `dcsvstop` command with the name of the RAP-processing listener specified as a command argument, the command stops both the RAP-processing listener and the RAP-processing server.

Example of command execution:

```
> dcsvstop RAP-processing-listener-name
```

Start up the RAP-processing listener and RAP-processing server using the user identifier of the system administrator. The RAP-processing listener and RAP-processing server create files under `$DCDIR/spool` and therefore a startup without the user identifier of the system administrator results in an error. This error happens because of the permission of the directory.

When the RAP-processing listener starts up, the message `KFCA00327-W` may appear. This is an error message to an RPC that the RAP-processing listener issues to the RAP-processing server when it executes a service function. The RAP-processing listener is set to retry even if an error occurs in the RPC. Since it is set to retry, there is no problem in the program even if the error message appears.

Do not shut down (by the `dcsvstop` command) or start up (by the `dcsvstart` command) the RAP-processing server solely. While the RAP-processing server is online, do not execute the `scdhold` command for the RAP-processing server. In the following cases, however, start up the RAP-processing server by using the `dcsvstart` command.

- When an error occurs due to a definition error
The RAP-processing listener automatically starts up the RAP-processing server.

The message KFCA00244-E does not indicate the location of the error in the system definition. If the cause code in the message KFCA01812-E is CONFIGURATION, check the definitions of the RAP-processing server. The RAP-processing listener has no way of knowing that the RAP-processing server cannot start up because of a definition error. Therefore, it outputs the message KFCA26950-I and waits for the remote API service to become ready. In this case, correct the operand of the RAP-processing server and start up the RAP-processing server by using the `dcsvstart` command.

- When the RAP-processing listener goes down while shutting down the RAP-processing listener or server

When the RAP-processing listener goes down while shutting down the RAP-processing listener or server, the RAP-processing server may output the message KFCA26950-I and wait for the remote API service to become ready, even if the next `dcsvstart` command starts up the RAP-processing listener. If the RAP-processing server is not started up, execute the `dcsvstart` command to start up the RAP-processing server manually.

3.12.5 Starting and stopping the RAP-processing client manager

To start the RAP-processing client manager, execute the `dcsvstart` command. Alternatively, you can start the RAP-processing client manager by starting OpenTP1 with the `dcsvstart` definition command specified in the user service structure definition.

Example of command execution:

```
> dcsvstart -u RAP-processing-client-manager-name
```

To stop the RAP-processing client manager, execute the `dcsvstop` command. You can also stop the RAP-processing client manager by using the `dcstop` command to shut down OpenTP1.

Example of command execution:

```
> dcsvstop RAP-processing-client-manager-name
```

To use the RAP-processing client manager facility, start the components in the following order:

1. Start the RAP-processing client manager.

Start the RAP-processing client manager on the RAP-processing client node.

2. Start the RAP-processing listener.

On the RAP-processing server node, start a RAP-processing listener for which `y` is specified for the `rap_notify` operand in the RAP-processing listener service definition. The RAP-processing listener will notify the RAP-processing client manager. Information about the particular RAP-processing listener must be

defined in the `rap_listen_inf` operand in the RAP-processing client manager service definition.

3. Start the RAP-processing client.

On the RAP-processing client node, start the RAP-processing client and invoke the `dc_rpc_call` function.

Start the RAP-processing client manager using the user identifier of the system administrator. For details about the definitions, see the manual *OpenTP1 System Definition*.

3.12.6 Establishing connection with RAP-processing listener

If the listen queue of the RAP-processing listener overflows because a concentrated number of `connect ()` is issued, the error code `ECONNREFUSED` returns. When this code returns, issue `connect ()` again. Specify the retry count and retry interval for `connect ()` in the `DCEPL_CONNECT_RETRY_COUNT` and `DCFPL_CONNECT_RETRY_INTERVAL` operands of the user service definition. In the RAP-processing listener service definition, specify the `rap_connect_retry_count` and `rap_connect_retry_interval` operands and then execute the `rapdfgen` command.

3.12.7 Cautions when using the remote API facility

- When the remote API facility is used and the RAP-processing client's node is TP1/Server Base or TP1/LiNK, do not issue an API substitution execution request to a RAP-processing server on the same node as the RAP-processing client. If such a request is issued, operation cannot be guaranteed.
- When you use the remote API facility, you cannot use the feature for compressing RPC user data if TP1/Server Base serves as the RAP-processing client.
- When a chain RPC is performed using the remote API facility (specifying `DCRPC_CHAINED` in the flags argument of the `dc_rpc_call` function) and the RPC terminates abnormally to stop the UAP issuing the `dc_rpc_close` function, the RAP-processing server outputs the message `KFCA26921-E` reason code=401. This completely clears the resources of the chain RPC and terminates abnormally with `Vrext03`. Even if the chain RPC is performed normally, the RAP-processing server also goes down if the terminating RPC is not performed for the chain RPC (specifying `DCNOFLAGS` in the flags argument of the `dc_rpc_call` function).
- The user service definitions for the RAP-processing listener, RAP-processing server, and RAP-processing client manager created by the `rapdfgen` command are assumed to be under `$DCCONFPATH`. Therefore, do not move them from under `DCCONFPATH` to under `$DCUAPCONFPATH`. If moved, the operation cannot be assured.

- Do not execute the following commands on a RAP-processing server; otherwise, correct system operation cannot be assured.
 - `scdchprc` command
 - `scdhold` command
 - `scdrls` command
 - `scdrsprc` command
- When you use the remote API facility between OpenTP1 systems by specifying the `-w` option in the `dcsvgdef` definition command in the user service network definition (for example, to perform an RPC via a gateway such as the application gateway firewall), the `dc_rpc_call` function will not become a transaction even though you specify the transaction attribute. Therefore, when the remote API facility is used, starting a chained RPC from within a transaction and ending the chained RPC in synchronous point processing does not work correctly. You will need to end the chained RPC explicitly by using the `dc_rpc_call` function in which `DCNOFLAGS` has been specified in the `flags` parameter.
- If you monitor messages, check for the KFCA26952-I message to determine whether or not the remote API facility has started. Do not use the KFCA01813-I message for this purpose because it is also output when an error occurs during the RAP-processing listener start processing.

For details about these messages, see the manual *OpenTPI Messages*.

3.13 Nonstop operation of OpenTP1

3.13.1 Handling resources

To run OpenTP1 without stopping, you need to keep the number of resources to be used from increasing. OpenTP1 controls resources as follows:

Serial numbers

If the serial number of journal blocks and other items reaches the maximum, it is wrapped around to 0 or 1. Therefore, there is no problem in nonstop operation.

Memory requirements

The memory requirements do not increase during nonstop operation.

Disk space

The following table shows how each file uses disk space.

Table 3-30: Disk space used by OpenTP1

Directory	Description
/tmp	Stores the standard output and standard error output file (betran.log) for the processes under OpenTP1. This file may increase unlimitedly and therefore we recommend that you create a mechanism for deleting the files when the machine starts up, or to perform generation management of betran.log. See 3.5 <i>Standard output file operations</i> .
\$DCDIR/bin	The required disk space does not increase.
\$DCDIR/lib	The required disk space does not increase.
\$DCDIR/include	The required disk space does not increase.
\$DCDIR/etc	The required disk space does not increase.
\$DCDIR/aplib	The required disk space does not increase.
\$DCDIR/tmp	Used as the home directory of every process. If a process goes down, a core file is created in this directory and the file is moved to \$DCDIR/spool/save. Therefore, the required disk space increases temporarily but it will soon return to the original size. The directory is cleared when online operations resume.

Directory	Description
\$DCDIR/spool/save/*	A core file is saved. Since only three generations of the core file are created, the required disk space does not exceed the limit.
\$DCDIR/spool/dclckinf/*	This directory stores deadlock information and timeout information. You should delete the data at regular intervals ^{#1} .
\$DCDIR/spool/dctrninf/*	This directory stores the undecided-transaction information when you specify <code>trn_tran_recovery_list=Y</code> in the transaction service definition. ^{#2} Although <code>\$DCDIR/spool/dctrninf/trace</code> stores trace data for the XA interface, a wraparound occurs when its size reaches the preset limit. Therefore, the required disk space does not exceed the limit.
\$DCDIR/spool/errlog/*	This directory stores OpenTP1 error log information. Since a wraparound occurs when the size of the information reaches the preset limit, the required disk space does not exceed the limit.
\$DCDIR/spool/cmdlog/*	This directory stores the OpenTP1 command log. Since a wraparound occurs when the size of the log reaches the preset limit, the required disk space does not exceed the limit.
\$DCDIR/spool/dcjnlinf/*	This directory stores invalid journal block information. No file is usually created in this directory.
\$DCDIR/spool/dcjnlinf/unload/*	This directory stores the unload journal files generated by the automatic unloading facility when <code>jnl_auto_unload=Y</code> is specified in the system journal service definition and no value is specified for the <code>jnl_auto_unload_path</code> operand. We recommend that you delete the files you no longer need at regular intervals.
\$DCDIR/spool/dctjlinf/* \$DCDIR/spool/dcsjl/*	Do not delete the files created in these directories.
\$DCDIR/spool/trnrmcmd/*	This directory stores the object file created by the <code>trnmkobj</code> command.
\$DCDIR/spool/olkfifs/* \$DCDIR/spool/olkrsfs/*	The file size is fixed and the number of files corresponds to the definition. Therefore, the number of files does not exceed the limit.

Directory	Description
\$DCDIR/spool/dcprfinf/*	The file size and the number of files are fixed. In addition, a wraparound occurs when its size reaches the preset limit. Therefore, the required disk space does not exceed the limit.
\$DCDIR/spool/	This directory stores traces such as dlog1 and dlog2. Since a wraparound occurs with the OpenTP1 trace when its size reaches the preset limit, the required disk space does not exceed the limit. Shared memory dumps are also output to this directory. Since the size of the dump depends on the definition, the required disk space does not exceed the limit.

#1

When you specify `lck_deadlock_info_remove_level`, you can automatically delete the information at regular intervals.

#2

Since the information accumulates at each rerun, delete it at regular intervals. When you specify `trn_recovery_list_remove_level`, you can automatically delete the information at regular intervals.

3.13.2 Notes on changing the configuration

Note the following when you change the environment or the configuration of OpenTP1:

- Changing the system definition

Some changes in the system definition are reflected when OpenTP1 is restarted and others are reflected when OpenTP1 starts normally. Therefore, stop OpenTP1 when necessary.
- Upgrading product versions

Stop OpenTP1 normally and then upgrade the version.
- Adding application programs

You can add application programs during online operation.
- Switching application programs

Stop the application program you want to replace and then exchange it with the new one.
- Replacing part of an MCF communication service

You can replace MCF communication service files during online processing when

TP1/NET/TCP/IP, TP1/NET/Secondary Logical Unit - TypeP2, or TP1/NET/OSAS-NIF is used.

For details about partial replacement of MCF communication services, see 3.13.3 *Replacing part of an MCF communication service*.

3.13.3 Replacing part of an MCF communication service

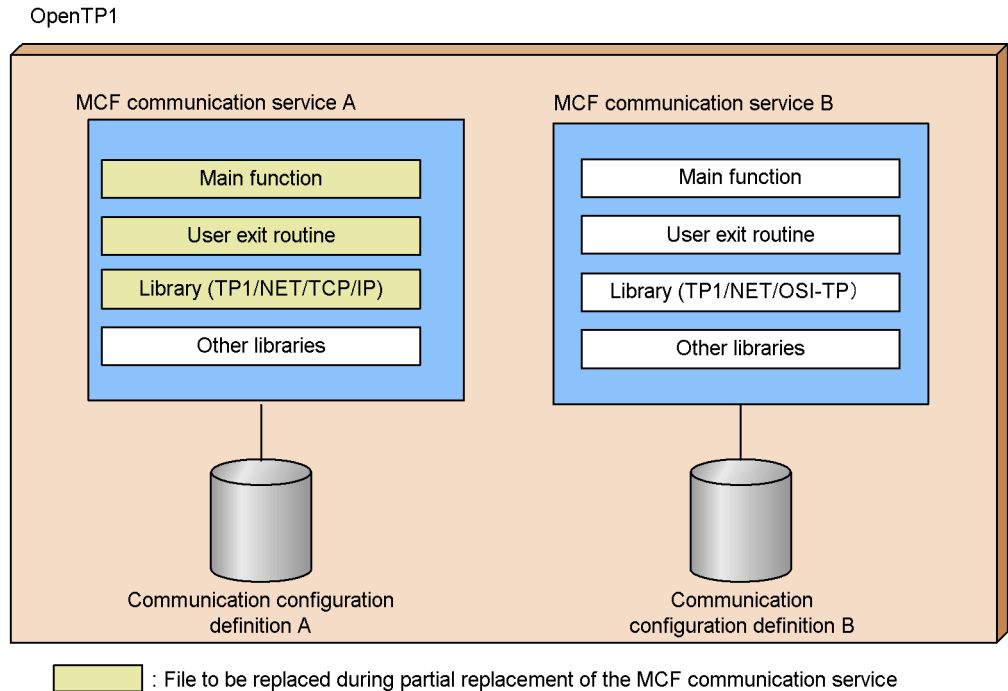
You can replace some files of an MCF communication service during online processing if TP1/NET/TCP/IP, TP1/NET/Secondary Logical Unit - TypeP2, or TP1/NET/OSAS-NIF is used as the communication protocol product. You can stop only the applicable MCF communication service and replace the files without stopping the OpenTP1 system. Use the `mcfstop` command and the `mcfstart` command to replace part of an MCF communication service.

The following files can be replaced partially:

- Load modules of MCF communications services (main functions)
- User-created libraries (user exit routines) incorporated in MCF communication services
- Libraries provided by the protocol product in use (when replacing modules)

The following figure shows how to use this feature to replace part of MCF communications service A without affecting the operation of MCF communication service B.

Figure 3-28: Partially replacing MCF communication service files



To replace the library provided by the protocol product in use, you need to stop all the MCF communication services that are using the protocol product.

You need to shut down the logical terminal and release the connection to prevent the target MCF communication service from processing messages before you execute the `mcftstop` command and stop the target MCF communication service.

To replace part of the MCF communication service:

1. Shut down the logical terminal of the target MCF communication service.

Example: `mcftdctl -s 01 -l "*"`

The sending of messages by the MCF communication service is disabled.

2. Release the connection of the target MCF communication service.

Example: `mcftdctn -s 01 -c "*"`

The receiving of messages by the MCF communication service is disabled.

3. Stop the target part of the MCF communication service.

Example: `mcftstop -s 01`

4. Check that the target part of the MCF communication service has stopped.

Example: `mcftlscom -s 01`

5. Replace the library of the target MCF communication service.

6. Start the target part of the MCF communication service.

Example: `mcftstart -s 01`

7. If necessary, establish a connection and release the logical terminal from shutdown status.

Example of establishing a connection: `mcftactcn -s 01 -c "*"`

Example of releasing the logical terminal from shutdown status: `mcftactle -s 01 -l "*"`

3.13.4 Notes on changing the time

OpenTP1 stores date and time information in status files, journal files, and checkpoint dump files. OpenTP1 uses the information at restart. OpenTP1 also stores date and time information in work files such as trace files, and uses the information during various types of checking. Therefore, if the time information is changed while OpenTP1 is starting, stopping, or is online, problems occur.

In OpenTP1, when changing the time, we recommend that you do not make large time adjustments at one time. Rather, we recommend that you increase or decrease the time in a series of small increments (for example, by using the `adjtime()` system call).

(1) *Advancing the time*

If the OpenTP1 daemons are inactive for a specified length of time, the internal checks performed by the OpenTP1 system will detect a timeout and the OpenTP1 system will go down. Therefore, if you advance the system time, OpenTP1 may incorrectly assume that a timeout has occurred and go down. Do not advance the time in units equal to or greater than seconds while OpenTP1 is starting, stopping, or is online. If you need to advance the time for some tests, stop OpenTP1 and then change the time.

You can advance the time while OpenTP1 is online if the increment is equal to or less than 100 milliseconds. However, in this case, statistics may be displayed incorrectly or the time display of trace information may become invalid.

(2) *Setting back the time*

When you set back the time, the operation is not guaranteed since restart may fail, transactions may be recovered incorrectly, or an invalid system wait may occur. Therefore, do not forcibly set back the time while OpenTP1 is starting, stopping, or online. If you slow the time, files with a more advanced time than the system time remain in the OpenTP1 system, causing the system to fail in switching trace files for wraparound operation. If you need to set the time back to perform a test of some kind,

3. OpenTP1 Online Operation

use the `dcstop` command to stop OpenTP1 temporarily.

After changing the time, wait until the system time is later than when you stopped OpenTP1 before you start it again. If you do not wish to wait, use the `dcsetup` command with the `-d` option specified to delete OpenTP1 from your system, and then rebuild the system including re-creating the OpenTP1 file system.

3.14 Operations using the XA resource service

This section describes how to use the XA resource service with OpenTP1. For an overview of the XA resource service, see the manual *OpenTP1 Description*.

3.14.1 Preparations for using the XA resource service

This section describes the preparation required to use the XA resource service.

(1) Prerequisite facilities

To use the XA resource service, the remote API facility and the following products need to be installed:

When linking with an application server based on J2EE architecture

- TP1/Client/J
- uCosminexus TP1 Connector or Cosminexus TP1 Connector

When linking with a .NET Framework application

- Client .NET
- Connector .NET

(2) Prerequisite definitions

To use the XA resource service, the following definitions must be specified:

- `trn_xar_use=Y` in the transaction service definition
- `jnl_fileless_option=N` in the system common definition (or omit the operand altogether)
- The XA resource service definition (file name: `xar`)

For details about the definitions, see the manual *OpenTP1 System Definition*.

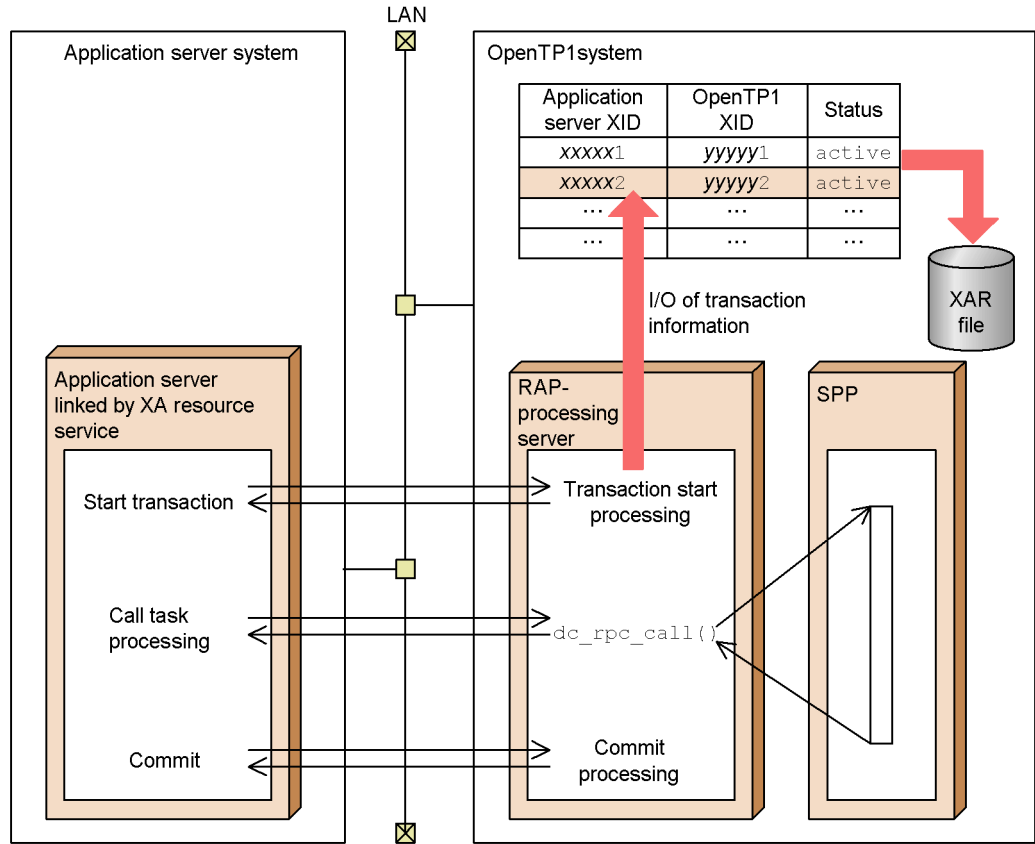
3.14.2 Transaction management using the XA resource service

The XA resource service starts or finishes management of a transaction according to the type of transaction request received from TP1/Client/J or Client .NET. It also manages transaction information in shared memory, by linking the application server XID (transaction identifier) passed from TP1/Client/J or Client .NET with the XID managed by OpenTP1. When MSDTC linkage is used, the XA resource service also manages transaction recovery information (RI). *RI* is created by MSDTC, and is used to recover transactions when an error occurs during transaction determination. The statuses of these transactions are recorded in an XAR file as required.

(1) Operation of the XA resource service

The following figure shows the operation of the XA resource service.

Figure 3-29: Operation of the XA resource service



When using the XA resource service, invalid transactions may remain in the following circumstances:

- A RAP-processing listener or a RAP-processing server was forcibly terminated (using the command `dcsvstop -f`)
- In this case, start a RAP-processing listener or a RAP-processing server (using the command `dcsvstart`).
- An operation was performed which is not guaranteed by the remote API facility
- The definition of an active RAP-processing listener and RAP-processing server was changed by the `rapdfgen` command before the RAP-processing listener and the RAP-processing server ended normally.

In this case, the operation is not guaranteed.

(2) Managing transaction information

The following table shows how transaction information is managed when linked with an application server based on J2EE architecture.

Table 3-31: Transaction information management (when linked with a J2EE application server)

Transaction request type	Transaction information management			Change in transaction status (Normal)
	Start	End	File I/O	
Start ()	Y	N	Y	None -> Active
Call ()	N	N	N	No change
End ()	N	N	N	Active -> Idle
Prepare ()	N	N	N	Idle -> Prepared ^{#1}
		Y	N	Idle -> None ^{#2}
Commit ()	N	Y	N	Prepared -> None ^{#3}
				Idle -> None ^{#4}
Rollback ()	N	Y	N	Prepared -> None ^{#3}
				Idle -> None ^{#4}
Recover ()	N	N	N	No change
Forget ()	N	Y	N	Heuristically Completed -> None

Legend:

Y: Executes processing

N: Does not execute processing

#1

When the transaction branch is *read only*.

#2

For two-phase commit or rollback.

#3

For one-phase commit or rollback.

The following table shows how transaction information is managed when linked with a .NET Framework application.

Table 3-32: Transaction information management (when linked with a .NET Framework application)

Transaction request type	Transaction information management			Change in transaction status (Normal)
	Start	End	File I/O	
Call ()	Y	N	N	None -> Active ^{#1}
	N	N	N	No change
Prepare ()	N	N	Y	Active -> Idle -> Prepared
		Y	Y	Active -> Idle -> None ^{#2}
Commit ()	N	Y	N	Prepared -> None ^{#3}
				Active -> Idle -> None ^{#4}
Rollback ()	N	Y	N	Prepared -> None ^{#3}
				Active -> Idle -> None ^{#4}
Recover ()	N	N	N	No change
Forget ()	N	Y	N	Heuristically Completed -> None

Legend:

Y: Executes processing.

N: Does not execute processing.

#1

For the first Call () request of the transaction.

#2

When the transaction branch is *read only*.

#3

For two-phase commit or rollback.

#4

For one-phase commit or rollback.

(3) XAR file operations

The XA resource service can save transaction information to an XAR file when necessary.

(a) Creating XAR files

Create XAR files using the `xarinit` command, and delete them using the `xarrm` command. Check file information for an XAR file using the `xarfills` command.

Two types of XAR files are required: online XAR files and backup XAR files. Both files must have the same record length and number of records, and be accessible for file I/O operations.

An online XAR file is used in normal operation, but the backup XAR file is automatically swapped in if the normal XAR file is damaged. You should store the online XAR file and the backup XAR file on different disk volumes.

An XAR file uses one record for each transaction branch. Thus, the number of records in an XAR file must be equal to or greater than the maximum number of simultaneous transaction branches (the value specified in the `trn_tran_process_count` operand in the transaction service definition).

- When linked with a J2EE application server

Each unit of record information in an XAR file occupies 512 bytes. When you use character-type special files, I/O operations on the file are performed in units of the sector length you defined in the `-s` option of the `filmkfs` command when you created the OpenTP1 file system. We therefore recommend that you specify a sector length of 512 bytes for the OpenTP1 file system. Note that with some OSs, the minimum sector length is greater than 512 bytes (with HP-UX, it is 1024 bytes). We recommend that you use the minimum sector length allowed by the OS. Specify a sector length that is a multiple of 512 bytes. If you are using the UNIX file system, file I/O operations are performed in 512-byte units by default.

- When linked with a .NET Framework application

Because the size of the transaction recovery information (RI) depends on the environment where MSDTC is operating, in some environments the record length specified for XAR files may be inadequate to store the information. For this reason, when using the MSDTC linkage facility, we recommend that you specify a record length of at least 1024 bytes for XAR files to avoid problems with transaction determination.

When defining the record length for XAR files (by executing the `xarinit` command with the `-s` option specified), specify a multiple of the sector length of the OpenTP1 file system. You can use the `xarfills` or `xarls` commands to find out the maximum size of RI data that XAR files can store.

If transaction determination fails due to the record length for XAR files being too short, recreate the XAR files as follows:

1. Shut down OpenTP1.
2. Recreate the XAR files by executing the `xarinit` command, specifying the `-s` option as the value output as the required length of XAR files in the message KFCA32045-E.
3. Start OpenTP1 in the normal manner.

(b) Specifying XAR files

Use the `xarfile` command in the XA resource service definition to specify the XAR files. You must have already created an XAR file that satisfies the above conditions. If you specify an XAR file that does not exist, an error results. An error also results if you specify an XAR file that contains fewer records than the value specified in the `trn_tran_process_count` operand in the transaction service definition.

Use the `xarfile` definition command with the `-t online` option to specify the online XAR file and the `-t backup` option to specify the backup XAR file. Make sure you specify both.

(c) Notes

OpenTP1 does not support XAR file duplication. Also, the XA resource service cannot manage transaction information in any of the following circumstances:

- OpenTP1 has not been started.
- The remote API facility has not been started.
- The XA resource service has not been started.
- The XA resource service has shut down.

The XA resource service shuts down whenever an error occurs in both the online XAR file and the backup XAR file. To release the XA resource service from shutdown status, execute the `xarres` command. For details, see *3.14.4 Action to take if an error occurs in the XAR file*.

To display the status of the transactions being managed by the XA resource service, use the `xarls` command. Execute the `xarforce` command to complete any undecided transactions.

3.14.3 Starting and stopping the XA resource service

(1) Starting the XA resource service

The XAR files are initialized when you start the XA resource service. If the XA resource service fails to start, OpenTP1 will also fail to start.

For details about the definitions required to use the XA resource service, see *3.14.1(2)*

Prerequisite definitions.

(2) Restarting the XA resource service

To restart the XA resource service, you must restart OpenTP1. Whether or not the XA resource service restarts with OpenTP1 depends on the status of the XA resource service the last time OpenTP1 started normally. If the XA resource service started at the last normal run, the XA resource service also restarts with OpenTP1. If not, the XA resource service does not restart.

The XAR files are initialized when you restart the XA resource service.

If the XA resource service fails to restart, OpenTP1 will also fail to restart.

When you restart OpenTP1, you can change the following definition operands and commands:

- `xar_eventtrace_level`: Output level for XAR event trace information
- `xar_eventtrace_record`: Maximum number of records output to the XAR event trace information file
- `xar_session_time`: Monitoring time for idle transaction branches
- `xar_prf_trace_level`: Acquisition level for XAR performance verification trace information
- `xarfile`: The specified XAR physical files

You can only change the value of the definition command `xarfile` at OpenTP1 restart if an error occurs in an XAR file. If you attempt to change the definition command `xarfile` at OpenTP1 restart in any other circumstance, you will not be able to recover transaction branches. For details about how to recover transaction branches if an error occurs in an XAR file, see *3.14.4 Action to take if an error occurs in the XAR file*.

At restart, the XA resource service restores transaction branches to their previous states, using an XAR file (transaction branch recovery processing). If the backup XAR file was being used, transaction branches are recovered from the backup XAR file; otherwise, transaction branches are recovered from the online XAR file.

In transaction branch recovery processing, the XA resource service examines the prior transaction information for each record in the XAR file. An undecided transaction branch that had the first-phase status `Prepared` is restored to the `Prepared` status. Because the XAR file contains `XID` information for transaction branches, OpenTP1 is again able to accept instructions relating to transaction determination from the application servers linked by the XA resource service.

If a transaction branch had the status `Heuristically completed`, the XA resource service restores the transaction branch to the `Heuristically completed` status, completing the transaction branch heuristically.

All transactions that were not yet prepared (still in `Active` or `Idle` state) are rolled back. Because the `XID` information of these transaction branches is then deleted, subsequent determination requests received for these transaction branches from the application servers linked by the XA resource service cannot be processed.

If transaction recovery processing is performed on a completed transaction branch (a transaction branch that has been committed or rolled back), the determination result of the transaction branch remains unchanged.

The following table shows the recovery processing performed on a transaction branch according to the transaction status.

Table 3-33: Transaction branch recovery processing at OpenTP1 restart

Transaction status		Processing	XID information
Active		Roll back	Deleted
Idle		Roll back	Deleted
Prepared	Before PJ output by OpenTP1	Wait for transaction completion	Left in file
	After PJ output by OpenTP1	Commit	Deleted
Rollback only		Roll back	Deleted
Heuristically completed		Heuristic completion	Left in file

(3) Terminating the XA resource service

The XA resource service terminates along with OpenTP1. However, OpenTP1 cannot terminate unless all transaction branches from all application servers linked by the XA resource service have completed. If uncompleted transaction branches remain, OpenTP1 waits for completion of them. Alternatively, you can use the forced termination command provided by OpenTP1 to forcibly terminate OpenTP1 without waiting for completion of these transaction branches.

3.14.4 Action to take if an error occurs in the XAR file

If an I/O error occurs with the XAR file while the XA resource service is online, the XA resource service automatically swaps the online XAR file for the backup XAR file. This facility is called *swapping*. This facility allows nonstop operation of the XA resource service when there is a problem with the online XAR file.

You can use the swapping facility in the following ways:

If an I/O error occurs in an XAR file, the method to restart the online processing of the XA resource service differs depending on the file and the time the error occurs. The following shows the actions to take each time an error occurs.

1. Use the swapping facility to swap the failed online XAR file with a new XAR file.
2. After step 1, if an error occurs in the backup XAR file and the XA resource service is shut down, replace the failed backup XAR file with a new XAR file.
3. If an error occurs in the online XAR file and OpenTP1 terminates, swap in a new XAR file.
4. If an error occurs in both the online XAR file and the backup XAR file, the XAR resource service is shut down, and OpenTP1 stops, replace both XAR files with new ones.

The first and second methods involve replacing the XAR file that caused the error with a new XAR file without closing down OpenTP1. These methods allow the XA resource service to recover from shared memory the transaction branches that were present before the error occurred.

Since the first method involves measures that are implemented before any damage can occur to the backup XAR file, reliable operation of the XA resource service can be assured.

With the third method, even if OpenTP1 terminates, the XA resource service can access the information in the backup XAR file to recover the transaction branches that were present before the error occurred.

With the fourth method, OpenTP1 has terminated and both the online and backup XAR files have failed. Thus, swapping in a new XAR file does not allow the XA resource service to recover the transaction branches that were present before the error occurred.

Each method is described in detail below.

(1) *Swapping XAR files at failure of the online XAR file*

If an error occurs with the online XAR file, the XA resource service automatically swaps in the backup XAR file and continues uninterrupted operation. At this point, you can replace the failed online XAR file with a new file, allowing the transaction branches that were present before the error occurred to be recovered without having to shut down OpenTP1.

When the XA resource service is running using the backup XAR file, and you wish to replace the failed online XAR file with a new file, use the `xarhold` command to forcibly shut down the XA resource service. You will still be able to execute other OpenTP1 services.

To swap online XAR files at a failure:

1. Execute the `xarhold` command.
The XA resource service shuts down.
2. Create a new online XAR file.

3. OpenTP1 Online Operation

Create this file on a different disk volume from that of the online XAR file that caused the error. Create the new online XAR file with the same record length and number of records as the backup XAR file.

3. Update the definition for the online XAR file in the XA resource service definition, to reflect the XAR file you created.

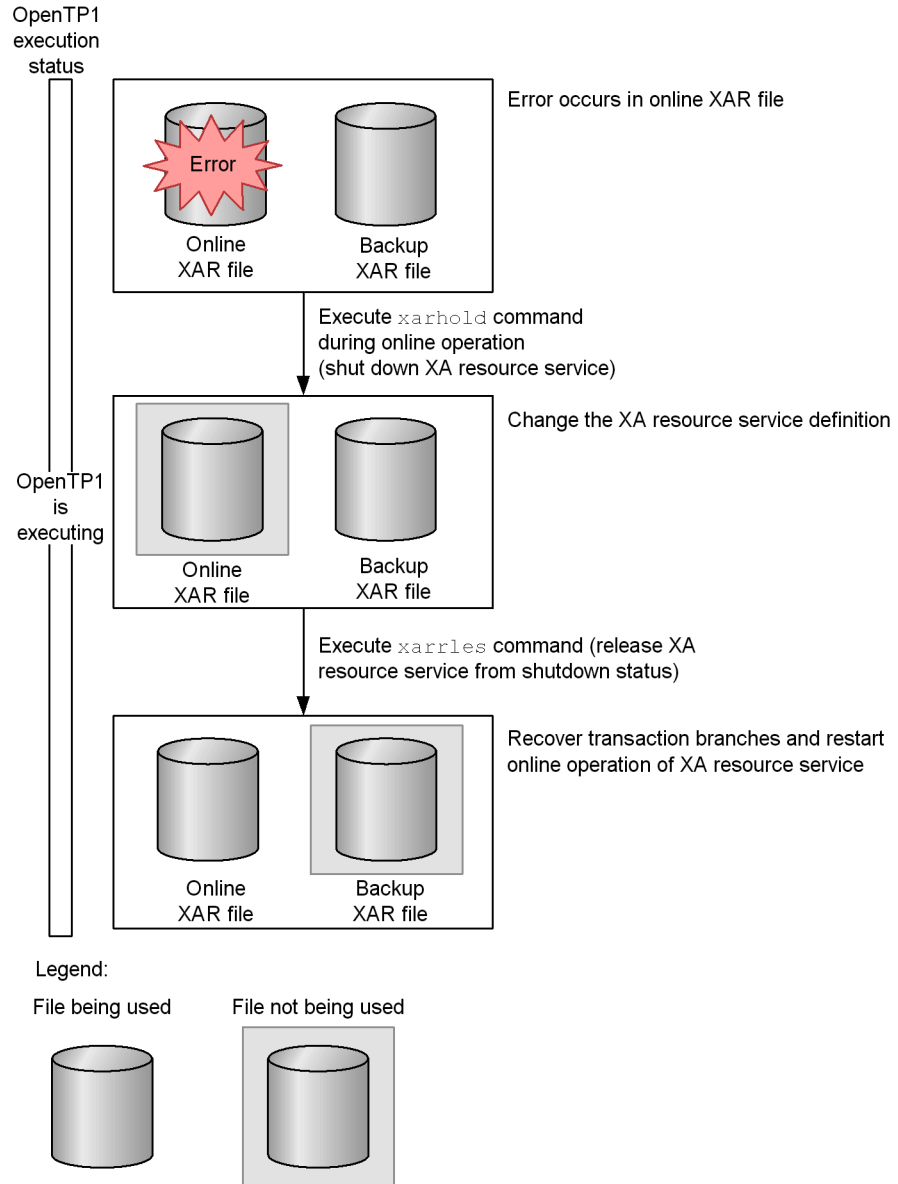
Only change the definition of the online XAR file.

4. Execute the `xarrles` command.

This releases the XA resource service from shutdown status. The transaction branches are recovered from shared memory, and the XA resource service resumes its online status using the newly created online XAR file.

The following figure shows how to swap XAR files when the online XAR file fails.

Figure 3-30: Swapping XAR files at failure of the online XAR file



(2) Swapping XAR files at failure of the backup XAR file

Any failure of the backup XAR file, swapped in to allow continuous operation following an error with the online XAR file, will cause the XA resource service to shut

down. If this occurs, OpenTP1 responds with the error code `XAER_RMERR` to every transaction instruction from an application server linked by the XA resource service. However, since the error is localized to the XA resource service, OpenTP1 continues running, and you can continue to execute OpenTP1 services with the exception of the XA resource service. Once you replace the XAR files with new files, you can recover the transaction branches of the XA resource service.

To replace the XAR file when both the online XAR file and the backup XAR file have failed and the XA resource service has shut down:

1. Create new XAR files.

Create an online XAR file and a backup XAR file on a disk volume that is free from errors. These two files must have the same record length and number of records.

2. Update the definition for the XAR files in the XA resource service definition, to reflect the XAR files you created.

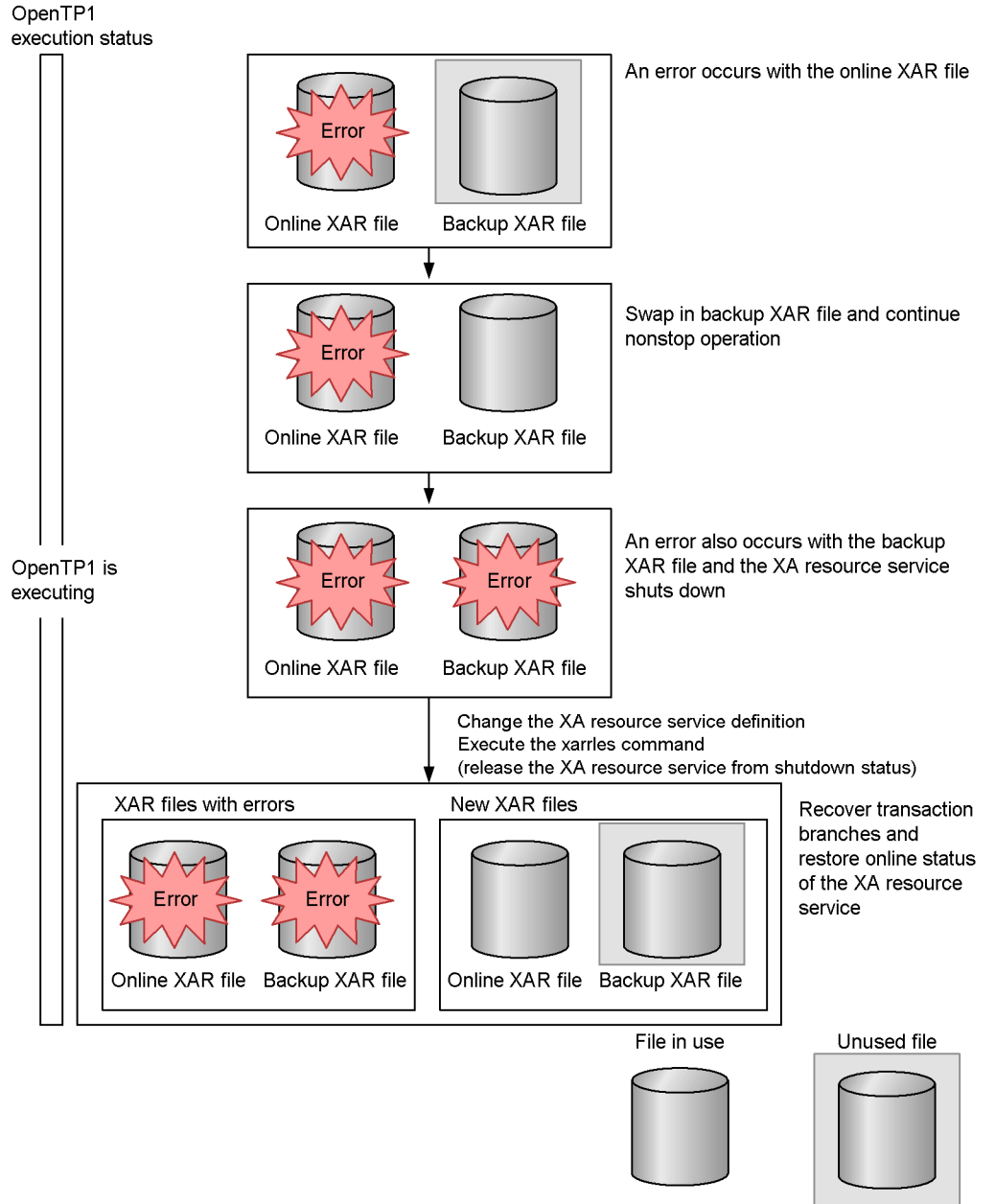
Change the definitions for both the online and backup XAR files.

3. Execute the `xarrles` command.

This releases the XA resource service from shutdown status. The transaction branches are recovered from shared memory, and the XA resource service resumes its online status using the newly created online XAR file.

The following figure shows how to swap XAR files when the backup XAR file fails.

Figure 3-31: Swapping XAR files at failure of the backup XAR file



(3) Swapping XAR files at failure of the online XAR file with OpenTP1 not running

If the online XAR file fails and OpenTP1 terminates (either normally or abnormally), the XA resource service shuts down. To restart the XA resource service, you create a new XAR file and then restart OpenTP1.

To replace the XAR file when the online XAR file fails and OpenTP1 has terminated:

1. Create a new online XAR file.

Create this file on a different disk volume from that of the online XAR file that has failed. The new online XAR file must have the same record length and number of records as the backup XAR file.

2. Update the definition for the online XAR file in the XA resource service definition, to reflect the XAR file you created.

Only change the definition of the online XAR file. Since the XA resource service recovers transaction branches from the backup XAR file, do not change the backup XAR file.

3. Restart OpenTP1.

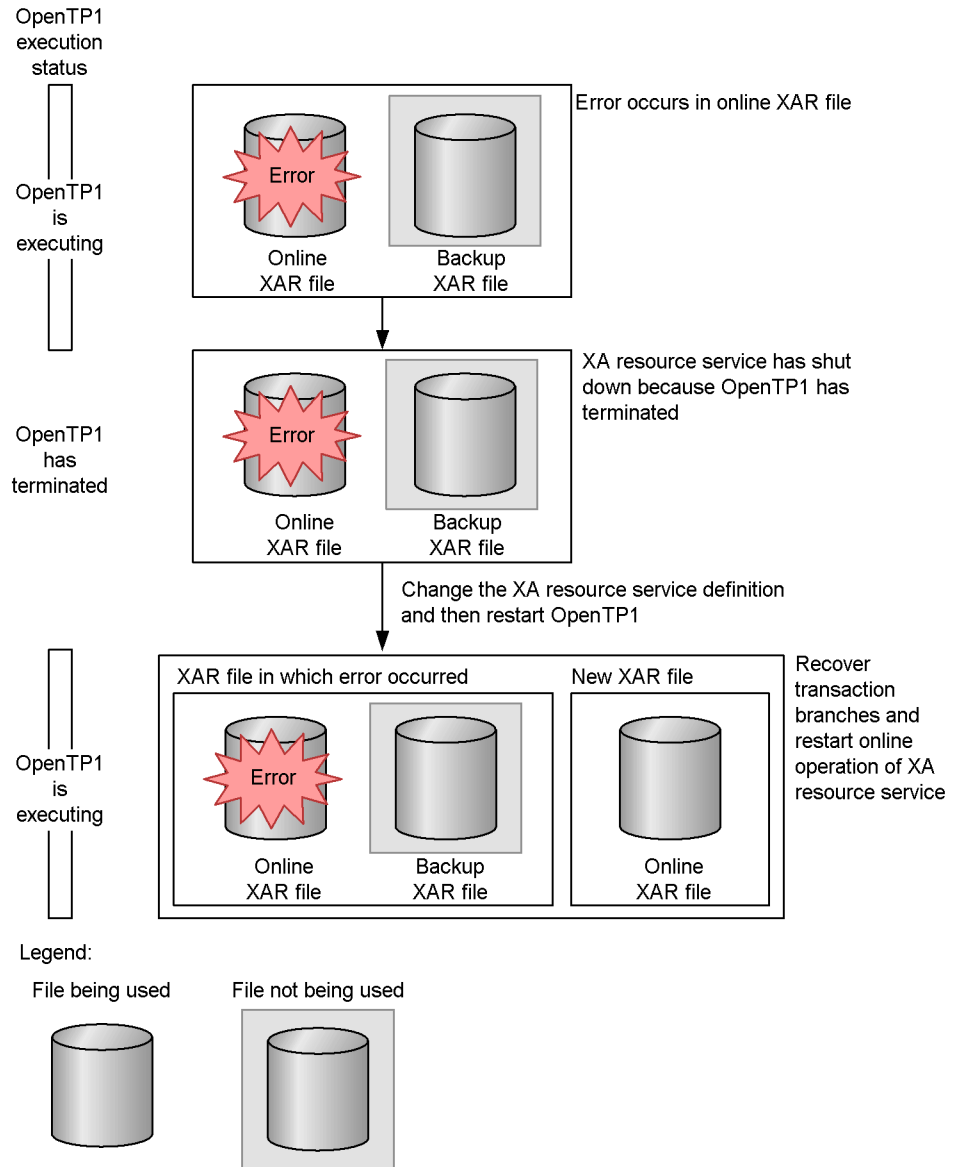
When you restart OpenTP1, the shutdown status of the XA resource service is released. The transaction branches are recovered from the backup XAR file, and the XA resource service restarts, using the newly created online XAR file.

Note

Before you can start OpenTP1 normally, or restart it, I/O access to both the online and backup XAR files must be possible. You cannot start OpenTP1 with a failed XAR file.

The following figure shows how to replace the XAR file when the online XAR file fails and OpenTP1 has been terminated.

Figure 3-32: Swapping XAR files at failure of the online XAR file with OpenTP1 not running



(4) Swapping XAR files at failure of the backup XAR file and with OpenTP1 not running

If both the online and backup XAR files fail and OpenTP1 terminates (either normally

or abnormally), the XA resource service shuts down. This means that the transaction branches that were present before the error occurred cannot be recovered. To restart the XA resource service, create new XAR files, and then forcibly start OpenTP1 (forced normal start).

To replace XAR files when both the online and backup XAR files fail and OpenTP1 has terminated:

1. Create new XAR files.

Create an online XAR file and a backup XAR file on a disk volume that is free from errors. These files should have the same record length and number of records.

2. Update the definition for the XAR files in the XA resource service definition, to reflect the XAR files you created.

Change the definitions for both the online and backup XAR files.

3. Force OpenTP1 to start normally.

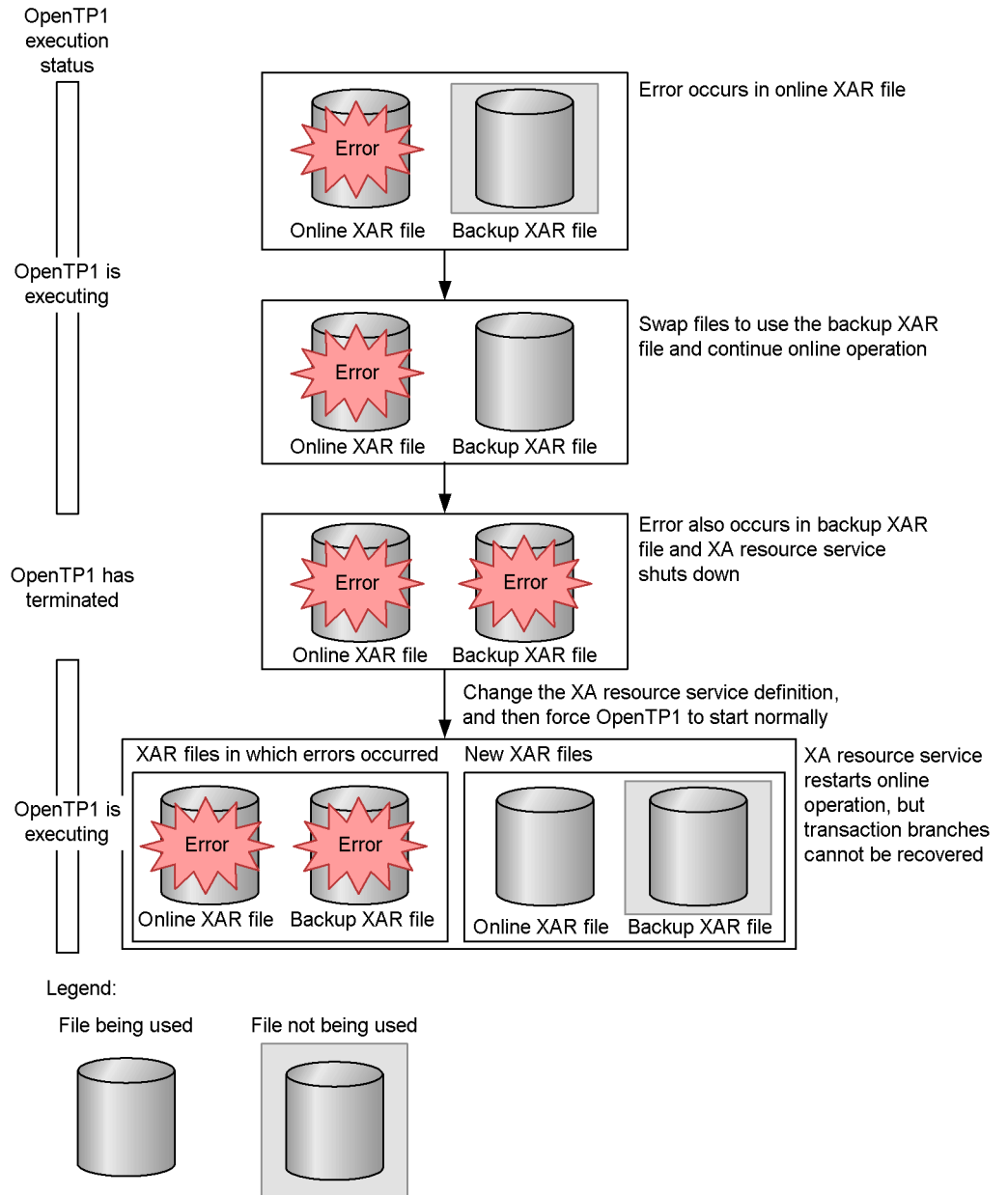
When you restart OpenTP1, the shutdown status of the XA resource service is released. The XA resource service resumes its online status using the newly created online XAR file. However, transaction branches that were present before the error occurred are not recovered.

Note

Before you can forcibly start OpenTP1 normally, I/O access to both the online and backup XAR files must be possible. You cannot start OpenTP1 with a failed XAR file.

The following figure shows how to replace the XAR files when both the online and backup XAR files fail and OpenTP1 has terminated.

Figure 3-33: Swapping XAR files at failure of both the online and backup XAR files with OpenTP1 not running



3.14.5 Trace information acquired by the XA resource service

The XA resource service acquires the following trace information to use for troubleshooting purposes:

- XAR event trace
- XAR performance verification trace

For details about these types of trace information, see the manual *OpenTP1 Description*.

3.15 Operations using Job Management Partner 1

To use Job Management Partner 1 to use scenario templates, use JP1/Base, JP1/AJS2, and JP1/AJS2 - Scenario Operation. For details about JP1/Base, see the manual *Job Management Partner 1/Base Description, User's Guide and Reference*. For details about JP1/AJS2, see the manual applicable to your needs. For details about JP1/AJS2 - Scenario Operation, see the relevant JP1/AJS2 - Scenario Operation manuals.

3.15.1 System operations using scenario templates

A *scenario template* used by OpenTP1 is a component that consists of OpenTP1 commands and an operating procedure for the OpenTP1 commands. By relating a scenario template to a task, you can execute the template as an operating procedure for the task, which is called a *scenario*.

By using a scenario template, you can automate the operation of the OpenTP1 system through using Job Management Partner 1. OpenTP1 provides scenario templates for the following scenarios:

- Scale Out

This scenario creates a new OpenTP1 node, and adds the node to the domain configuration of the OpenTP1 system.

- Scale In

This scenario releases resources of the least heavily loaded nodes on a task or node basis, and allocates the resources to other systems.

- Rolling Update

This scenario applies security-enhancing patches to the OS or UAPs without interrupting system operation.

OpenTP1 provides the following scenario templates:

- OpenTP1_AddNode
- OpenTP1_ChangeNodeID
- OpenTP1_Deploy
- OpenTP1_ScenarioAddNode
- OpenTP1_Start
- OpenTP1_StartUAP
- OpenTP1_Stop
- OpenTP1_StopUAP

3. OpenTP1 Online Operation

- OpenTP1_Undeploy
- OpenTP1_UpdateDomain
- OpenTP1_ScenarioScaleout

For details about these scenario templates, see *M. Details About Scenario Templates*.

To execute an OpenTP1 command from a scenario template, specify the `dcjcmdex` command in the OpenTP1 command for JP1/AJS2 - Scenario Operation to determine the end code of the OpenTP1 command. In addition, set 3 as the priority level of scenario templates so that the priority level of the OpenTP1 commands to be executed from scenario templates is not lowered. For details about setting priority levels, see the relevant JP1/AJS2 - Scenario Operation manuals.

3.15.2 Registering a scenario

To register a scenario template in JP1/AJS2 - Scenario Operation, you need to import the following scenario template definition file in the scenario library folder of JP1/AJS2 - Scenario Operation Manager:

Scenario template definition file

```
/BeTRAN/jp1_template/ScenarioTemplate/  
TP1_ServerBase.sjis.xml
```

Destination scenario library folder

```
/ScenarioLibrary/OpenTP1/TP1_ServerBase
```

For details about how to import the scenario template definition file in the scenario library folder of JP1/AJS2 - Scenario Operation Manager, see the relevant JP1/AJS2 - Scenario Operation manuals.

The following subsection explains the scenario variables that need to be set when a scenario is registered.

(1) Scenario variables

Some information changes depending on the operating environment. Scenario variables represent such information, and they are preset for each scenario. For example, you can set a host name and a file name as scenario variables, and specify values for those scenario variables when the scenario is registered. In this way, you can execute a scenario that matches the operating environment.

The scenario templates provided by OpenTP1 determine the values of environment variables and the user who executes the scenario based on the scenario variables that are entered (input scenario variables).

The following table lists the environment variables whose values are determined from input scenario variables.

Table 3-34: Environment variables whose values are determined from input scenario variables

Environment variable	Description	Value to be set when scenario is executed ^{#1}
DCDIR	OpenTP1 directory ^{#2}	?DCDIR?
DCCONFPATH	Directory containing the OpenTP1 definition file ^{#3}	?DCCONFPATH?

#1

The character string enclosed by question marks (?) indicates the input scenario variable of JP1/AJS2 - Scenario Operation whose value needs to be set.

#2

Specify a full path name not exceeding 50 bytes. You cannot use symbolic links.

#3

Specify a full path name.

When these environment variables are set as the system environment variables of the execution user, you need to set the values of the system environment variables for the input scenario variables as well. For details about operation when both input scenario variables and system environment variables are specified, see the description of the environment variables in the manual *Job Management Partner 1/Automatic Job Management System 2 Setup Guide*.

System environment variables consists of the following variables:

For UNIX:

- Variables defined in the script file that is specified when a scenario job is defined by JP1/AJS2 - Scenario Operation
- Variables defined in the local login script
- Variables defined in the system login script

For Windows:

- System environment variables

Specify the execution user of a scenario by using the input scenario variable `USER_NAME` in each scenario template. Only a user mapped by JP1/Base can be specified as the execution user of a scenario. You need to map (user mapping) the OpenTP1 system administrator before you execute a scenario. For details about user mapping, see the manual *Job Management Partner 1/Base Description, User's Guide and Reference*.

3.15.3 Using Scale Out

When you execute the Scale Out scenario, you can create a new OpenTP1 node and add the node to the domain configuration of the OpenTP1 system. This scenario lets you quickly expand the system.

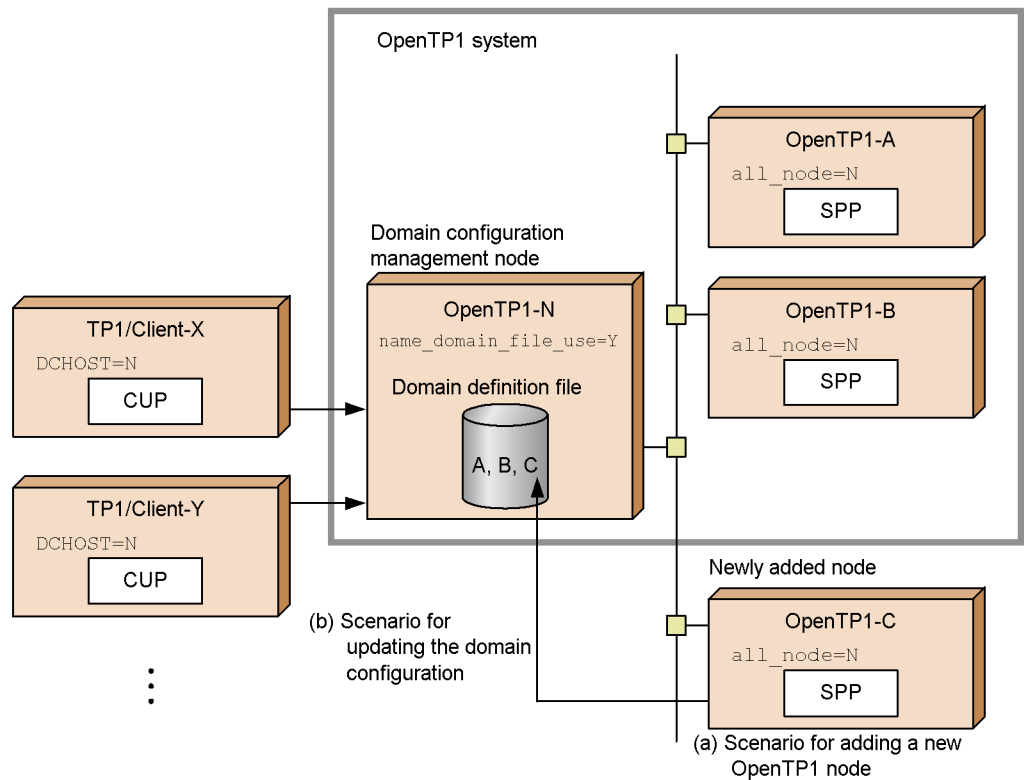
(1) Operating model for Scale Out

When you execute the Scale Out scenario, a scenario for adding a new OpenTP1 node and a scenario for updating the domain configuration are executed.

(a) When the domain configuration management node is used

The following figure shows the operating model for executing Scale Out when the domain configuration management node is used.

Figure 3-34: Operating model for Scale Out (when the domain configuration management node is used)



To execute Scale Out using the domain configuration management node, carry out the following procedure:

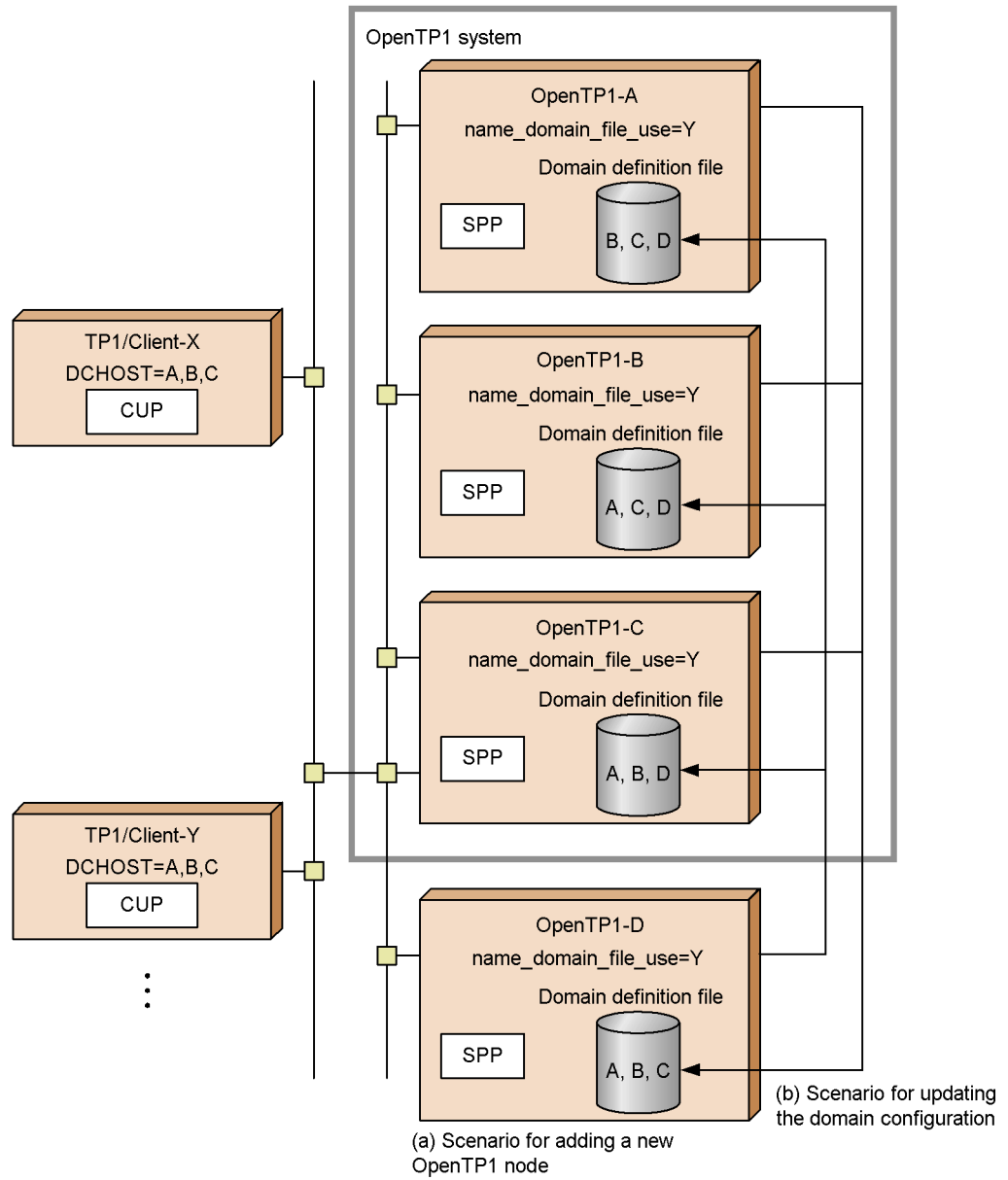
1. Execute scenario (a) for adding a new node and create OpenTP1-C.

2. Execute scenario (b) for updating the domain configuration at OpenTP1-N (domain configuration management node) and add OpenTP1-C to the domain configuration.

(b) When the domain configuration management node is not used

The following figure shows the operating model for executing Scale Out when the domain configuration management node is not used.

Figure 3-35: Operating model for Scale Out (when the domain configuration management node is not used)



In this operating model, nodes manage the domain configuration in the system by configuring one another. When a new OpenTP1 node is created, the domain

configuration set at other nodes is updated so that those nodes can recognize the new node.

To execute Scale Out without using the domain configuration management node, carry out the following procedure:

1. Execute scenario (a) for adding a new node and create OpenTP1-D.
2. Execute scenario (b) for updating the domain configuration in OpenTP1-A, B, and C, and add OpenTP1-D to the domain configuration.
3. Execute scenario (b) for updating the domain configuration in OpenTP1-D and add other nodes in the system (OpenTP1-A, B, C) to the domain configuration.

(2) Configuration of the Scale Out scenario

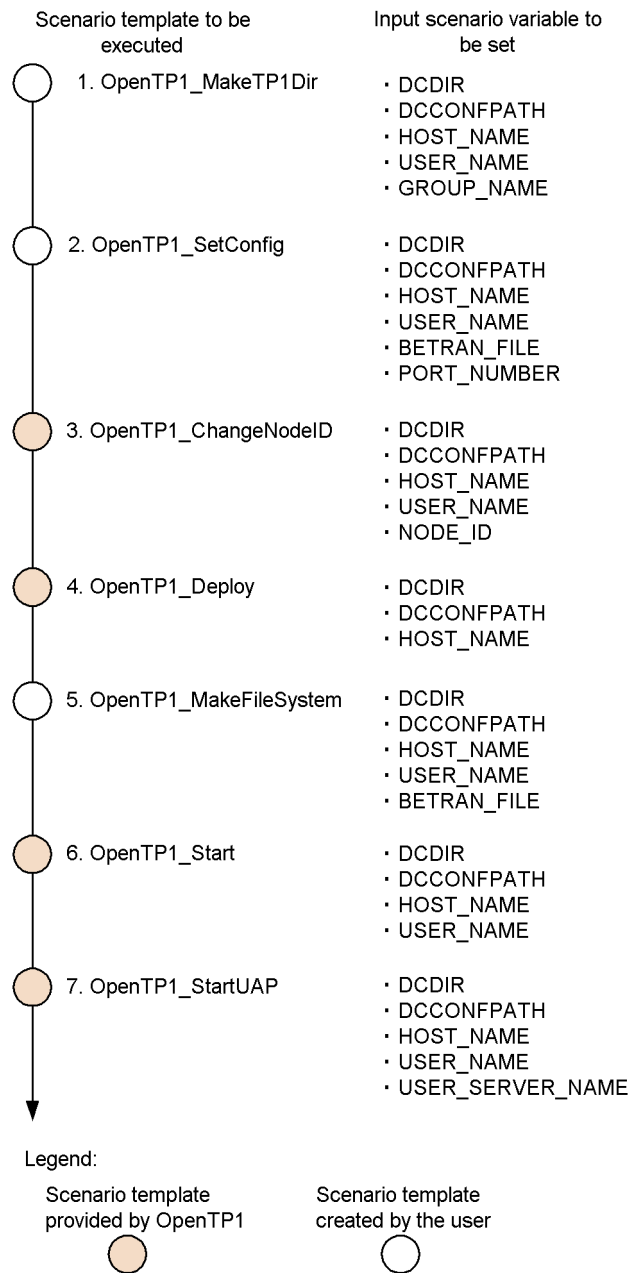
When the Scale Out scenario provided by OpenTP1 is executed, OpenTP1 executes a scenario for adding a new OpenTP1 node and then a scenario for updating the domain configuration.

See the following for details about each scenario.

(a) Scenario for adding a new OpenTP1 node

The following figure shows the sequence of the scenario templates executed by the scenario for adding a new OpenTP1 node.

Figure 3-36: Sequence of scenario templates executed by the scenario for adding a new OpenTP1 node



See the following table for an explanation of each scenario template.

Table 3-35: Scenario templates in the scenario for adding a new OpenTP1 node

No.	Scenario template name	Description
1	OpenTP1_MakeTP1Dir ^{#1}	Creates an OpenTP1 directory.
2	OpenTP1_SetConfig ^{#1}	Sets the OpenTP1 environment.
3	OpenTP1_ChangeNodeID ^{#2}	Sets a node ID.
4	OpenTP1_Deploy ^{#2}	Registers OpenTP1.
5	OpenTP1_MakeFileSystem ^{#1}	Creates OpenTP1 file systems.
6	OpenTP1_Start ^{#2}	Starts OpenTP1.
7	OpenTP1_StartUAP ^{#2}	Starts a UAP.

#1

Created by the user. For details about the samples provided by OpenTP1, see *3.15.6 Using sample scenario templates*.

#2

This scenario template is provided by OpenTP1. For details, see *M. Details About Scenario Templates*.

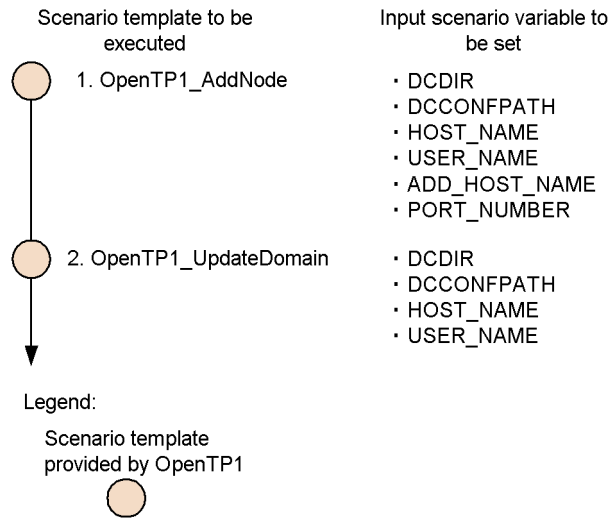
(b) Scenario for updating the domain configuration

The prerequisites for updating the domain configuration are as follows:

- Y is specified in the `name_domain_file_use` operand in the system common definition.
- The node where you update the domain configuration is online.

The following figure shows the sequence of the scenario templates executed by the scenario for updating the domain configuration.

Figure 3-37: Sequence of scenario templates executed by the scenario for updating the domain configuration



See the following table for an explanation of each scenario template.

Table 3-36: Scenario templates in the scenario for updating the domain configuration

No.	Scenario template name	Description
1	OpenTP1_AddNode	Adds a node to the domain definition file.
2	OpenTP1_UpdateDomain	Updates the domain configuration.

Both of these scenario templates are provided by OpenTP1. For details, see *M. Details About Scenario Templates*.

(3) Using Scale Out and DPM for the OpenTP1 environment setup procedure

By executing Scale Out, you can partially automate the procedure for setting the OpenTP1 environment. This subsection describes the procedure for setting the environment in which Scale Out and the disk copy facility of DPM are used. The superuser must perform steps 1 to 8. Perform steps 1 to 5 on the master computer of the OpenTP1 system (disk copy source), step 6 on the computer where DPM is installed, and steps 7 to 9 on the new OpenTP1 node (disk copy destination) to be added.

1. In the `/etc/hosts` file, register the host name used by the OpenTP1 system.
2. Register the OpenTP1 system administrator.

For details, see *1.2.1 Cataloging of OpenTP1 administrator*.

3. Set an OpenTP1 group.

For details, see *1.2.2 Definition of OpenTP1 group*.

4. Install the OpenTP1 program product.

You must create the OpenTP1 program product on the same disk (master disk) that contains the OS.

5. Create a file for setting the OpenTP1 environment, that will be used by Scale Out.

For details, see *3.15.6(3) Customizing a sample scenario template*.

6. Using the disk copy facility of DPM, restore the contents of the disk backed up from the master computer of the OpenTP1 system to the node of the new OpenTP1.

As the network type of the disk copy destination, specify fixed IP, not DHCP. For details about how to use DPM, see the manual *JP1/ServerConductor/Deployment Manager*.

7. Set up JP1/AJS2 - Agent.

For details about how to set up JP1/AJS2 - Agent, see the manual *Job Management Partner 1/Automatic Job Management System 2 Setup Guide*.

8. Allocate a raw device as the OpenTP1 file system.

To run the OpenTP1 file system by using a character type special file, allocate a raw device on the OpenTP1 file system to automate the initialization of the OpenTP1 file system by Scale Out.

9. Execute Scale Out.

When Scale Out is executed, a scenario for adding a new OpenTP1 node is executed. When this scenario is executed, the following processing is performed automatically:

1. Creation of the OpenTP1 directory
2. Creation of the system definition
3. Registration of OpenTP1 in the OS
4. Initialization of the OpenTP1 file system
5. Creation of the OpenTP1 file

3.15.4 Using Scale In

When you execute the Scale In scenario, the resources of nodes with a low load are released and allocated to other systems on a node or task basis. This scenario lets you

use resources effectively.

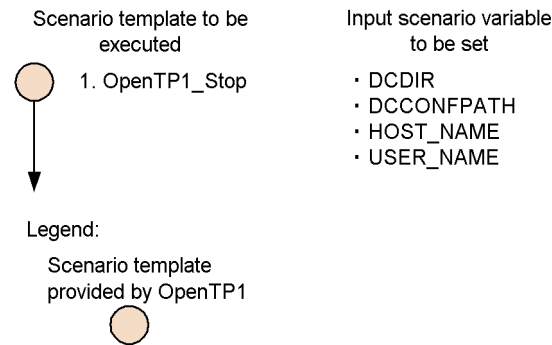
(1) Configuration of the Scale In scenario

The Scale In scenario provided by OpenTP1 includes a scenario for stopping OpenTP1 and releasing resources node by node and a scenario for stopping a UAP and releasing resources task by task. Execute one of the two scenarios. See the following for details about each scenario.

(a) Scenario for stopping OpenTP1 and releasing resources

The following figure shows a scenario for stopping OpenTP1 and releasing the resources of an OpenTP1 node.

Figure 3-38: Scenario for stopping OpenTP1 and releasing resources



See the following table for an explanation of the scenario template.

Table 3-37: Scenario template in the scenario for stopping OpenTP1 and releasing resources

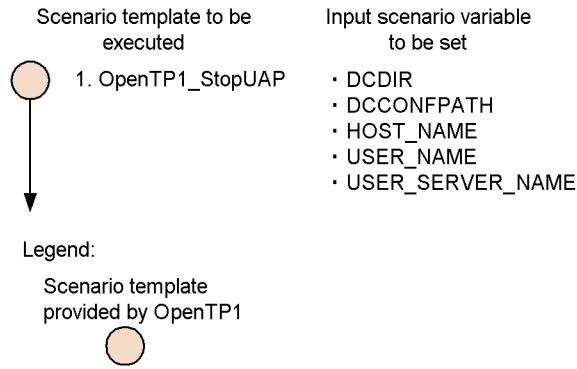
No.	Scenario template name	Description
1	OpenTP1_Stop	Stops OpenTP1.

This scenario template is provided by OpenTP1. For details, see *M. Details About Scenario Templates*.

(b) Scenario for stopping a UAP and releasing resources

The following figure shows a scenario for stopping a UAP and releasing the task resources of OpenTP1.

Figure 3-39: Scenario for stopping a UAP and releasing resources



See the following table for an explanation of the scenario template.

Table 3-38: Scenario template in the scenario for stopping a UAP and releasing resources

No.	Scenario template name	Description
1	OpenTP1_StopUAP	Stops a UAP.

This scenario template is provided by OpenTP1. For details, see *M. Details About Scenario Templates*.

3.15.5 Using Rolling Update

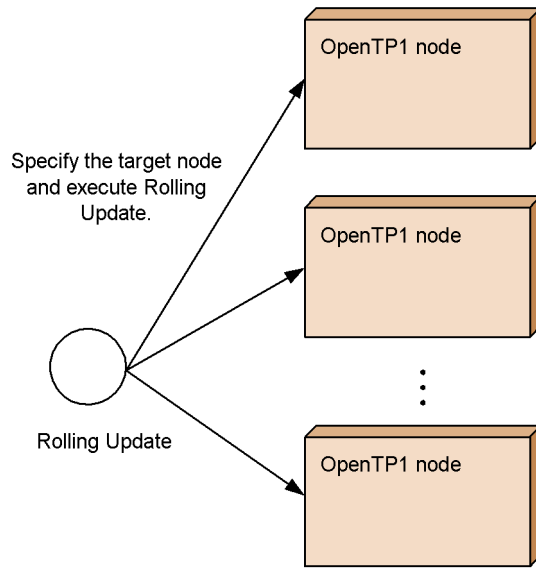
When you execute the Rolling Update scenario, you can apply a security-enhancing patch for the OS or a UAP without stopping system operation. This scenario lets you continue system operation without interruption.

Do not execute Rolling Update on nodes that may stop the OpenTP1 system if OpenTP1 is stopped or deleted.

(1) Operating model for Rolling Update

The following figure shows the operating model for executing Rolling Update.

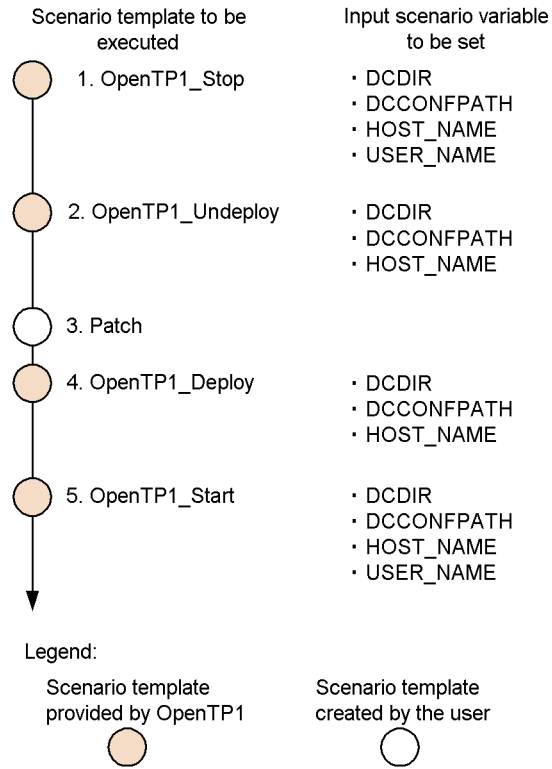
Figure 3-40: Operating model for Rolling Update



(2) Configuration of the Rolling Update scenario

The following figure shows the sequence of the scenario templates executed by Rolling Update.

Figure 3-41: Sequence of scenario templates executed by Rolling Update



See the following table for an explanation of each scenario template.

Table 3-39: Scenario templates in Rolling Update

No.	Scenario template name	Description
1	OpenTP1_Stop ^{#1}	Stops OpenTP1.
2	OpenTP1_Undeploy ^{#1}	Deletes OpenTP1.
3	Patch ^{#2}	Applies a patch.
4	OpenTP1_Deploy ^{#1}	Registers OpenTP1.
5	OpenTP1_Start ^{#1}	Starts OpenTP1.

#1

This scenario template is provided by OpenTP1. For details, see *M. Details About Scenario Templates*.

#2

Created by the user.

3.15.6 Using sample scenario templates

OpenTP1 provides sample scenario templates for the scenario for adding a new OpenTP1 node in the Scale Out scenario.

When you execute the `OpenTP1_ScenarioScaleout` sample scenario template with OpenTP1 installed, a new OpenTP1 environment is created. After creation of the environment, OpenTP1 and a sample user server (`basespp`) start. By customizing sample scenario templates, you can easily execute Scale Out in a way appropriate for your environment.

(1) Overview of sample scenario templates

This subsection provides an overview of sample scenario templates.

The sample scenario templates perform the processing shown in the following table.

Table 3-40: Processing executed by the sample scenario templates

No.	Processing	Scenario template name	Input scenario variables referenced
1	Creating an OpenTP1 directory	<code>OpenTP1_MakeTP1Dir</code>	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME • GROUP_NAME
2	Setting the OpenTP1 environment	<code>OpenTP1_SetConfig</code>	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME • BETRAN_FILE1 • BETRAN_FILE2 • PORT_NUMBER
3	Setting a node ID	<code>OpenTP1_ChangeNodeID</code>	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME • NODE_ID
4	Registering OpenTP1	<code>OpenTP1_Deploy</code>	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME

No.	Processing	Scenario template name	Input scenario variables referenced
5	Creating OpenTP1 file systems	OpenTP1_MakeFileSystem	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME • BETRAN_FILE1 • BETRAN_FILE2
6	Starting OpenTP1	OpenTP1_Start	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME
7	Starting a UAP (sample program)	OpenTP1_StartUAP	<ul style="list-style-type: none"> • DCDIR • DCCONFPATH • HOST_NAME • USER_NAME • USER_SERVER_NAME

See the following for details about each scenario template.

1. Creating an OpenTP1 directory (OpenTP1_MakeTP1Dir)

Use the OpenTP1 directory creation shell file[#] to create the OpenTP1 directory with the following procedure:

1. Acquire a user name and a group name (getUSERNAME).

The specified USER_NAME and GROUP_NAME are acquired and passed to makeTP1Dir. If both USER_NAME and GROUP_NAME are omitted, the user-mapped primary user defined in JP1/Base and the group the user belongs to are acquired.

2. Create the OpenTP1 directory (makeTP1Dir).

Use the OpenTP1 directory creation shell file[#] to create the following OpenTP1 directories:

- Directory: Value set for the input scenario variable DCDIR
- Owner: Value set for the input scenario variable USER_NAME
- Group: Value set for the input scenario variable GROUP_NAME
- Mode: 0755

If the parent directory of the specified OpenTP1 directory does not exist, create the parent directory and the OpenTP1 directory. You need to set the owner, group, and mode only for the OpenTP1 directory.

2. Setting the OpenTP1 environment (OpenTP1_SetConfig)

Use the OpenTP1 directory creation shell file[#] to create the OpenTP1 directory with the following procedure.

1. Create the \$DCDIR/aplib, \$DCDIR/jp1_template/tools, and \$DCCONFPATH directories.
2. Copy aplib, conf, and tools in the /BeTRAN/jp1_template/examples directory to the created directory.
3. Set the node-specific information (such as the path name of the file system) in the system definition.

3. Setting a node ID (OpenTP1_ChangeNodeID)

The scenario template uses the value specified for the input scenario variable NODE_ID as the node ID of OpenTP1.

4. Registering OpenTP1 (OpenTP1_Deploy)

The scenario template executes the dcsetup command to register OpenTP1 in the OS.

5. Creating an OpenTP1 file system (OpenTP1_MakeFileSystem)

Use the OpenTP1 file creation shell file[#] to create an OpenTP1 file system and OpenTP1 files.

The names of the OpenTP1 file systems are the values set for the input scenario variables BETRAN_FILE1 and BETRAN_FILE2.

6. Starting OpenTP1 (OpenTP1_Start)

The scenario template starts OpenTP1.

7. Starting a UAP (sample program) (OpenTP1_StartUAP)

The scenario template starts the UAP on the user server specified in the input scenario variable USER_SERVER_NAME.

#

The shell files depend on the OS being used. The table below shows the shell file paths for each OS.

Table 3-41: Paths for the shell files for each OS

Shell file name	OS	
	UNIX	Windows
OpenTP1 directory creation shell file	/BeTRAN/jp1_template/examples/tools/dcjm_k_dkdir.sh	<i>OpenTP1-installation-directory</i> \jp1_template\examples\tools\dcjm_k_dkdir.bat
OpenTP1 environment setup shell file	/BeTRAN/jp1_template/examples/tools/dcjset_conf.sh	<i>OpenTP1-installation-directory</i> \jp1_template\examples\tools\dcjset_conf.bat
OpenTP1 file creation shell file	/BeTRAN/jp1_template/examples/tools/dcj_mkfs.sh	<i>OpenTP1-installation-directory</i> \jp1_template\examples\tools\dcj_mkfs.bat

(2) Registering sample scenario templates

To execute a sample scenario template, you register it in JP1/AJS2 - Manager from JP1/AJS2 - Scenario Operation View.

The sample scenario templates are stored in the following location:

```
/ScenarioLibrary/OpenTP1/TP1_ServerBase/  
OpenTP1_ScenarioScaleout
```

To register a sample scenario template, set the input scenario variables listed in the following table.

Table 3-42: Information to be set in the sample scenario templates

Input scenario variable	Description	Example
DCDIR	OpenTP1 home directory	/home/user/OpenTP1
DCCONFPATH	Directory containing the OpenTP1 definition file	/home/user/OpenTP1/conf
HOST_NAME ^{#1}	Host name of the OpenTP1 node where the scenario template is to be executed	hostX
USER_NAME ^{#1}	OpenTP1 user name	user
GROUP_NAME ^{#1}	OpenTP1 group name	tp1
BETRAN_FILE1 ^{#2}	Path for an OpenTP1 file system	/home/user/OpenTP1/betranfile1

Input scenario variable	Description	Example
BETRAN_FILE2 ^{#2}	Path for an OpenTP1 file system	/home/user/OpenTP1/betranfile2
USER_SERVER_NAME	User server name	basespp (fixed in the sample)
NODE_ID	OpenTP1 node identifier (unique character string of 4 characters for each node)	smp1
PORT_NUMBER	OpenTP1 name port number	10000

#1

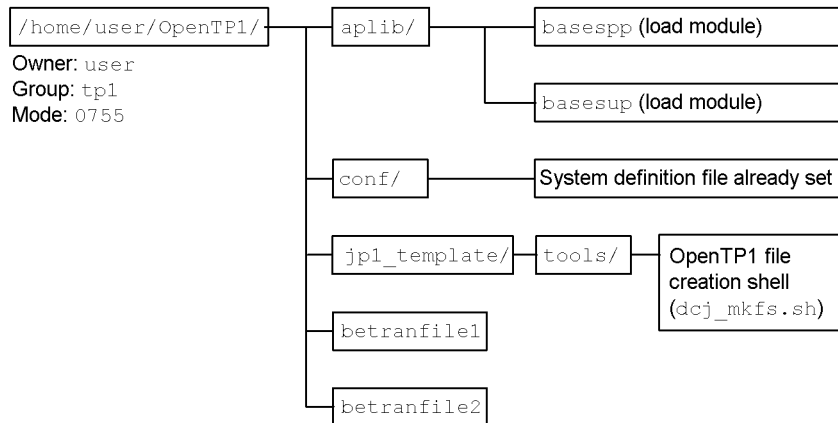
You can omit this specification. You need to either specify both the input scenario variables USER_NAME and GROUP_NAME, or omit both of them.

#2

When you set a character type special file, allocate a partition before you execute a scenario.

When you execute a scenario created by entering into a sample scenario template the values shown in the table above, an OpenTP1 execution environment is created with the directory configuration shown in the following figure.

Figure 3-42: OpenTP1 directory configuration after the sample scenario templates are executed



(3) Customizing a sample scenario template

By customizing a sample OpenTP1 definition file or shell file, you can easily change

the OpenTP1 configuration. You can also store the files customized for your environment and the installation environment settings of OpenTP1 on a master disk, which is created by using DPM, so that you can easily execute Scale Out.

You can customize the following sample files:

- Load modules for the sample programs
- System definition file
- OpenTP1 environment setup shell file
- OpenTP1 file creation shell file

See the following for details about how to customize each file.

(a) Customizing a load module for a sample program

You can create an SUP or an SPP appropriate for your tasks and store the load module (executable file) in the desired directory.

(b) Customizing the system definition file and the OpenTP1 environment setup shell file

To customize the system definition of OpenTP1, you need to change the system definition file and the OpenTP1 environment setup shell file.

- Customizing the system definition file

Preset the items you require in each system definition file of OpenTP1. To change the settings for each node, set the values of character string @DCxxxxxx@ for the sample scenario template. You can change the values of @DCxxxxxx@ by using the `dcjchconf` command.

- Customizing the OpenTP1 environment setup shell file

The files in the following table are copied from the installation directory (/BeTRAN) to the OpenTP1 directory (\$DCDIR) based on the settings in the OpenTP1 environment setup shell file. Therefore, if you change the location of these files, you need to change the settings in the OpenTP1 environment setup shell file as well.

Table 3-43: Settings in the sample OpenTP1 environment setup shell file

File type	Storage location	Copy destination
Load module for a sample program	/BeTRAN/jp1_template/examples/aplib	\$DCDIR/aplib
System definition file	/BeTRAN/jp1_template/examples/conf	\$DCCONFPATH
OpenTP1 file creation shell file	/BeTRAN/jp1_template/examples/tools (file name: dcj_mkfs.sh)	\$DCDIR/jp1_template/tools

The OpenTP1 environment setup shell file sets the values of character string @DCxxxxx@ that are set in the system definition file by executing the dcjchconf command. The values of @DCxxxxx@ are set as shown in the following table in the sample scenario template.

Table 3-44: Values set for character string @DCxxxxx@ in the sample scenario template

File name	Operand name/ definition command name	Value set in the system definition file	Value set in the OpenTP1 environment setup shell file ^{#1}
cdtrn	jnladdpf	@DCbetran@	?BETRAN_FILE1?
env	DCCONFPATH	@DCconfpath@	\$DCCONFPATH
prc	prcsvpath	@DChomepath@	\$DCDIR
sts	sts_file_name	@DCbetran@	?BETRAN_FILE1?
		@DCbetran2@	?BETRAN_FILE2?
sysjnl	jnladdpf	@DCbetran@	?BETRAN_FILE1?
betranrc	node_id ^{#2}	@DCNODE_ID@	?NODE_ID?
	name_port	@DCport_number@	?PORT_NUMBER?

#1

A character string enclosed by question marks (?) indicates a value set for an input scenario variable.

#2

When you execute the scenario template for setting a node ID that was configured from part of the same scenario template, this value is automatically set.

(c) Customizing the OpenTP1 file creation shell file

The sample scenario template creates the files listed in the following table in the OpenTP1 file systems (/home/user/OpenTP1/betranfile1 and /home/user/OpenTP1/betranfile2) that are set for the input scenario variables BETRAN_FILE1 and BETRAN_FILE2. To change the files to be created, customize the OpenTP1 file creation shell file.

Table 3-45: Files created in the OpenTP1 file systems set for the input scenario variables BETRAN_FILE1 and BETRAN_FILE2

File name	File type
cpdf01 ^{#1}	Checkpoint dump file
cpdf02 ^{#1}	
cpdf03 ^{#1}	
stsfil01 ^{#1}	Status file (system A)
stsfil03 ^{#1}	
jnlf01 ^{#1}	System journal file
jnlf02 ^{#1}	
jnlf03 ^{#1}	
stsfil02 ^{#2}	Status file (system B)
stsfil04 ^{#2}	

#1

File created in /home/user/OpenTP1/betranfile1

#2

File created in /home/user/OpenTP1/betranfile2

3.16 Operations for using the real-time statistics service

This section describes how to use the real-time statistics service to acquire real-time statistics.

3.16.1 Preparation for using the real-time statistics service

Perform the following preparation to use the real-time statistics service.

(1) *Setting the execution environment*

Use the `rtsetup` command to set the execution environment for the real-time statistics service. When you execute the `rtsetup` command, execution modules and user service definitions are created. The following table lists the directories and files that are created.

Table 3-46: Directories and files created when the `rtsetup` command is executed

Directory	File name	Description
\$DCDIR/aplib/#	rtssup	Execution module
	rtsspp	
\$DCCONFPATH/	RTSSUP	User service definition
	RTSSPP	

#

This directory is used if you do not specify a path for storing execution modules on the server in the `rtsetup` command. When you specify a path for storing execution modules on the server, files are output to the specified directory.

If the files that have the same names already exist in the storage location for the execution modules and in the directory specified in `$DCCONFPATH`, the `rtsetup` command does not create the files.

(2) *Creating definitions*

Create the real-time statistics service definition and the definition of the items to be acquired in real time. For details about the definitions, see the manual *OpenTP1 System Definition*.

3.16.2 Starting and stopping the real-time statistics service

You must start and stop the real-time statistics service in the following order:

1. Start the real-time statistics service (start `RTSSUP`).

2. Start the extended feature of the real-time statistics service (start RTSSPP).#
 3. Stop the extended feature of the real-time statistics service (stop RTSSPP).
 4. Stop the real-time statistics service (stop RTSSUP).
- #

You need to start the extended feature of the real-time statistics service (RTSSPP) only when you want to update the settings of targets and items to be acquired in real time. For details about changing the settings of targets and items to be acquired, see *3.16.5 Changing the settings of targets and the items to be acquired in real time*.

See the following for details on how to start and stop the real-time statistics service.

(1) Starting the real-time statistics service

You have two ways to start the real-time statistics service:

- Use the `dcsvstart` command to start RTSSUP after starting OpenTP1.
- Use the `dcsvstart` definition command in the user service configuration definition to start RTSSUP.

The real-time statistics service performs the following processing when it starts:

Securing shared memory

The real-time statistics service secures shared memory that the service requires to acquire real-time statistics. If the required shared memory for the real-time statistics service is not secured, the service cancels startup processing. For details about shared memory used by the real-time statistics service, see the manual *OpenTP1 Description*.

Creating RTS log files

For details about RTS log files, see *3.16.4(2) Outputting real-time statistics to an RTS log file*.

Acquiring management information about the real-time statistics service

Use the `rtsls` command to output the management information about the real-time statistics service that has been acquired to the standard output.

(2) Starting the extended feature of the real-time statistics service

You have two ways to start the extended feature of the real-time statistics service:

- Use the `dcsvstart` command to start RTSSPP.
- Use the `dcsvstart` definition command in the user service configuration definition to start RTSSPP.

When you use the second method to start RTSSPP, write the definition as follows to start RTSSUP and then RTSSPP:

```
dcsvstart -u RTSSUP
dcsvstart -u RTSSPP
```

If you start the extended feature of the real-time statistics service before you start the real-time statistics service, RTSSPP outputs a KFCA32723-E message to the message log and a KFCA01812-E message with `SERVER DOWN` as the cause of the error, and goes down. If these messages are displayed, use the `dcsvstart` command to start RTSSUP (this starts the real-time statistics service) and then restart RTSSPP. The extended feature of the real-time statistics service will start.

Do not apply any of the following commands to the extended feature of the real-time statistics service (RTSSPP). If you do so, an error will occur.

- `scdchprc` command^{#1}
- `scdhold` command^{#2}
- `scdrls` command^{#1}
- `scdrsprc` command^{#1}

#1

RTSSPP goes down.

#2

The `rtstats` command will not be able to be executed.

(3) Stopping the extended feature of the real-time statistics service

To stop the extended feature of the real-time statistics service, use the `dcsvstop` command to stop RTSSPP.

(4) Stopping the real-time statistics service

To stop the real-time statistics service, use the `dcsvstop` command to stop RTSSUP. If RTSSPP is running, first stop it with the `dcsvstop` command, and then stop RTSSUP.

(5) Restarting the real-time statistics service

To restart the real-time statistics service, start it after it has stopped. When you restart the real-time statistics service, the service recreates RTS log files.

When the real-time statistics service restarts, it starts to acquire real-time statistics. Whether the real-time statistics service inherits the statistics that have been acquired during the previous operation depends on the status of RTSSUP at startup.

(6) Operation of the real-time statistics service

Depending on the status of `RTSSUP` at startup, there are differences in some parts of the operation of the real-time statistics service (whether to inherit previous real-time statistics and the definition of the real-time statistics service to be applied). The following table shows the relationship between the status of `RTSSUP` at startup and the operation of the real-time statistics service.

Table 3-47: Status of `RTSSUP` at startup and the operation of the real-time statistics service

Status of <code>RTSSUP</code> at startup		Operation of the real-time statistics service	
Startup type	Previous status	Whether previous real-time statistics inherited	Applied definition
Start	Inactive	Not applicable because no real-time statistics have been acquired.	All
Restart	Stopped by the <code>dcsvstop</code> command	Not inherited.	Some
	Forcibly stopped by the <code>dcsvstop</code> command with the <code>-f</code> option	Inherited.	None
	Went down	Inherited.	None
Restart of OpenTP1 [#]	Online	Not inherited.	All
	Forcibly stopped by the <code>dcsvstop</code> command with the <code>-f</code> option	Not inherited.	All

Legend:

All: All the operands and definition commands in the real-time statistics service definition are applied.

Some: All of the real-time statistics service definition except for the following operands is applied:

- `rts_service_max` operand
- `rts_item_max` operand

None: None of the operands and definition commands in the real-time statistics service definition are applied.

#

The real-time statistics service does not start when you use the `dcstart`

command with the `-U` option specified to restart OpenTP1.

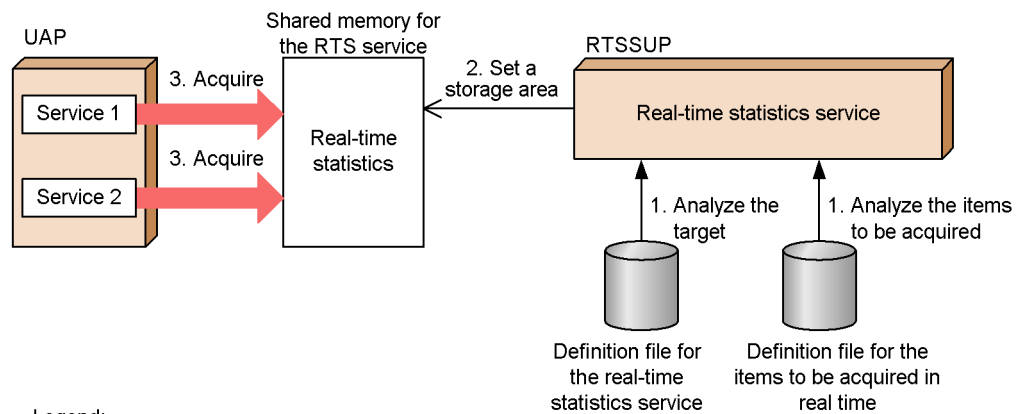
3.16.3 Acquiring real-time statistics

Real-time statistics are acquired in the shared memory for the real-time statistics service.

To acquire real-time statistics, you can specify the execution environment in the definition or use the API to specify a segment. This subsection describes how to specify the execution environment in the definition. For details about using the API to specify a segment, see the manual *OpenTP1 Programming Guide*.

The following figure illustrates acquisition of real-time statistics by specifying the execution environment in the definition.

Figure 3-43: Acquiring real-time statistics



Legend:

- : Flow of real-time statistics
- : Control of real-time statistics

1. The real-time statistics service analyzes the definition file for the real-time statistics service and the definition file for the items to be acquired in real time.
2. The real-time statistics service sets an area for storing real-time statistics in the shared memory for the service.
3. Real-time statistics regarding the target UAP are acquired in the set storage area based on the definition.

(1) Specifying a target

You can specify a target from which you want the real-time statistics service to acquire statistics. To specify a target, use the `rtspat` definition command in the real-time statistics service definition. The maximum number of targets that can be specified in

the `rtspout` definition command is the value specified in the `rts_service_max` operand in the real-time statistics service definition. For details about the targets for the acquisition of statistics, see *E.5 Real-time statistics*.

The following table shows the combinations of the values that can be specified for the options in the `rtspout` definition command and the targets for the acquisition of statistics.

Table 3-48: Values specified for the options in the `rtspout` definition command and acquisition targets

Value specified for the options in the <code>rtspout</code> definition command	Acquisition target
<code>-u sys</code>	Statistics of the entire system ^{#1}
<code>-u srv -s server-name</code>	Statistics of the specified server ^{#2}
<code>-u svc -s server-name -v service-name</code>	Statistics of the specified service
<code>-u obj -o port-number -b IP-address</code>	Statistics for the specified port number and IP address

#1

Statistics can be acquired regardless of the number of targets specified in the `rts_service_max` operand.

#2

The following types of statistics are acquired for the specified server:

- Statistics for the entire server
- Statistics for each service provided by the server
- Statistics for processing other than services (such as the main function)

Therefore, you will specify two more targets than the number of services provided by the server. If you specify a system server or SUP which does not run any services, the overall statistics for the server are acquired.

You do not need to start the server specified as the target for acquiring real-time statistics before you start the real-time statistics service. Real-time statistics can be acquired in the following cases as well:

- When the target server is started after the real-time statistics service has started
- When the target server is stopped and then restarted while the real-time statistics service is running

(2) Specifying the items to be acquired

You can specify the types of real-time statistics to be acquired for each target. The

maximum number of items that can be acquired for one target is the value specified in the `rts_item_max` operand in the real-time statistics service definition.

Use either of the following methods to specify the items to be acquired:

- Specify the item ID of the information item to be acquired in the `-e` option of the `rtspout` definition command.

For details about item IDs, see *E.5 Real-time statistics*.

- Specify the definition file for the items to be acquired in real time in the `-f` option of the `rtspout` definition command.

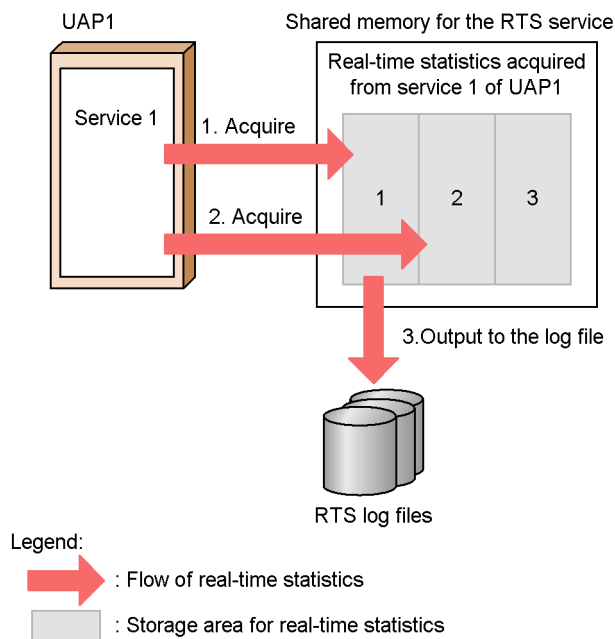
For details about how to specify the definition of the items to be acquired in real time, see the manual *OpenTP1 System Definition*.

(3) Specifying an acquisition interval

Specify the interval for acquiring real-time statistics in the `rts_trcput_interval` operand in the real-time statistics service definition.

The following figure illustrates an acquisition interval.

Figure 3-44: Interval for acquiring real-time statistics



The shared memory for the real-time statistics service contains areas for storing real-time statistics.

1. The real-time statistics service acquires statistics and edits them in the same storage area until the time specified in the `rts_trcput_interval` operand expires.
2. When the specified length of time expires, the storage area is switched.
3. When `Y` is specified in the `rts_log_file` operand in the real-time statistics service definition, the real-time statistics that have already been edited in the storage area are output to an RTS log file.

For details about how to output statistics to an RTS log file, see 3.16.4(2) *Outputting real-time statistics to an RTS log file*.

You can use the `rts_trcput_interval` operand in the real-time statistics service definition to change the interval for acquiring statistics when the real-time statistics service is started normally. For details about when the specification in the definition is applied, see 3.16.2(6) *Operation of the real-time statistics service*.

(4) Notes

- You can acquire real-time statistics only from servers that provide OpenTP1 system services and from UAPs that are started by using the `dcsvstart` command.
- You can acquire real-time statistics service by service only for UAPs. One exception is SUPs and those UAPs that use the XATMI interface. You cannot acquire real-time statistics from those UAPs service by service.
- When you specify `srv` in the `-u` option of the `rtspout` definition command, the names of the services provided by the specified server are acquired from the `service` operand in the user service definition of that server.

Therefore, the directory specified in the `DCCONFPATH` or `DCUAPCONFPATH` environment variable must contain the user service definition file.

If the user service definition file is not stored in the correct directory, the real-time statistics service will not start. If the `service` operand is not specified in the user service definition file, the real-time statistics service acquires real-time statistics on a server-by-server basis.

- If you add or remove services from the user service definition after the real-time statistics service is started, their status as targets of the real-time statistics service does not change. This applies even if the `srv` argument is specified for the `-u` option of the `rtspout` definition command. To change targets after you start the real-time statistics service, use the `rtsstats` command or restart the real-time statistics service (`RTSSUP`).

Services added or deleted using the facility for the dynamic loading of service functions after the real-time statistics service is started are also not automatically added or removed as targets of the real-time statistics service. To change targets

after you start the real-time statistics service, use the `rtstats` command or restart the real-time statistics service (`RTSSUP`).

For details about changing targets by using the `rtstats` command, see 3.16.5 *Changing the settings of targets and the items to be acquired in real time*.

- The real-time statistics service may not be able to acquire real-time statistics for the entire system or for UAPs that use a multi-server if multiple processes are concurrently performing write processing. The reason is that the real-time statistics service does not lock servers and UAPs to acquire statistics.

3.16.4 Outputting real-time statistics

This subsection describes how to output real-time statistics acquired in the shared memory for the real-time statistics service.

(1) Outputting real-time statistics to the standard output

When you execute the `rtsls` command, the real-time statistics acquired in the shared memory for the real-time statistics service are output to the standard output. You can output management information and statistics of the real-time statistics service to the standard output. By outputting this information to the standard output, you can recognize the operating status of the OpenTP1 system in real time.

(2) Outputting real-time statistics to an RTS log file

You can output the real-time statistics acquired in the shared memory for the real-time statistics service to an RTS log file that is created by the real-time statistics service. To output real-time statistics to an RTS log file, specify `Y` for the `rts_log_file` operand in the real-time statistics service definition and start the real-time statistics service.

(a) Creating an RTS log file

The real-time statistics service creates an RTS log file when it starts.

Specify the number of RTS log files to be created in the `rts_log_file_count` operand in the real-time statistics service definition.

Specify the location and name of the RTS log file in the `rts_log_file_name` operand in the real-time statistics service definition. Generation numbers are added to the names of the RTS log files that are created. When you specify the operands as follows, the real-time statistics service creates RTS log files named `rtslog1`, `rtslog2`, and `rtslog3` in a directory directly under `/OpenTP1/spool/dcrtsinf`.

```
set rts_log_file_count=3
set rts_log_file_name=/OpenTP1/spool/dcrtsinf/rtslog
```

When `Y` is specified for the `rts_log_file_backup` operand in the real-time statistics service definition, or no value is specified for the operand, a backup of the RTS log files is created when the real-time statistics service starts. If you specify the

operands in the manner above, which creates RTS log files named `rtslog1`, `rtslog2`, and `rtslog3` in a directory directly under `/OpenTP1/spool/dcrtsinf`, the real-time statistics service creates backup files named `rtslog1.bk`, `rtslog2.bk`, and `rtslog3.bk`, and then re-creates the three RTS log files. The real-time statistics service does not create these backup files if `N` is specified for the `rts_log_file_backup` operand.

Do not create a file or directory with the same name as an RTS log file or backup file at the output destination for RTS log files. If a file with such a name exists, it will be overwritten by the RTS log file or backup file. If you create a directory with the same name as that used for log file output by the real-time statistics service, the real-time statistics service will be unable to create RTS log files or backup files.

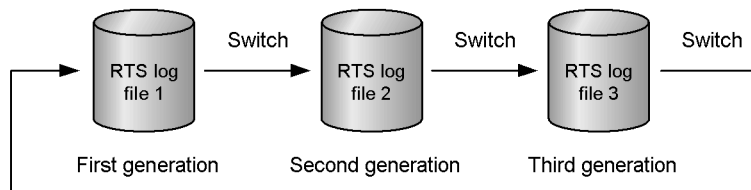
(b) Outputting real-time statistics to an RTS log file

The real-time statistics service outputs the real-time statistics acquired in the shared memory for the real-time statistics service to RTS log files at the interval specified in the `rts_trcput_interval` operand in the real-time statistics service definition.

Specify the size of an RTS log file in the `rts_log_file_size` operand in the real-time statistics service definition. If the size of the real-time statistics to be output to an RTS log file exceeds the value specified in the `rts_log_file_size` operand, the output destination will switch to the next RTS log file. When no RTS log files remain, output of real-time statistics reverts to the first RTS log file.

The following figure shows how RTS log files are switched.

Figure 3-45: Switching RTS log files



(c) Errors in an RTS log file

If an I/O error occurs in an RTS file while the real-time statistics service is online, the real-time statistics service switches to the next RTS log file and continues output processing.

If I/O errors occur in all the RTS log files, the real-time statistics service outputs a `KFCA32708-W` message and stops the output of real-time statistics to the RTS log files. However, the real-time statistics service continues acquiring real-time statistics in the shared memory for the RTS service.

(3) Editing and outputting real-time statistics in an RTS log file

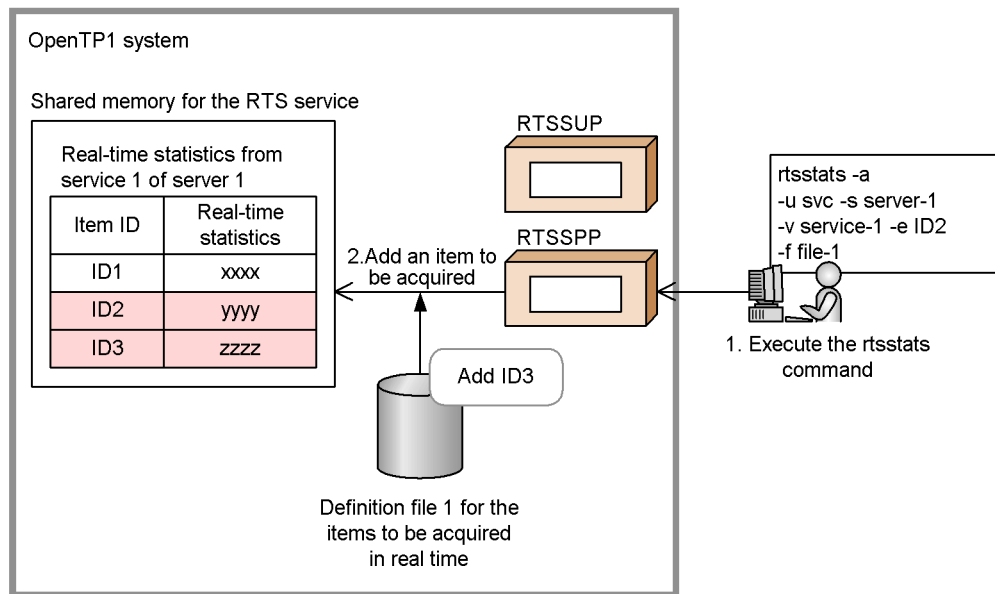
By executing the `rtscedit` command, you can change the format of the statistics output in an RTS log file to CSV format for output to another location.

3.16.5 Changing the settings of targets and the items to be acquired in real time


To change the targets for the acquisition of statistics and the items to be acquired while the real-time statistics service is running, use the `rtssstats` command. To use the command, the real-time statistics service (RTSSUP) and the extended feature of the real-time statistics service (RTSSPP) must have been started.


The following figure shows an example of adding items to be acquired for a target for which real-time statistics are already being acquired.

Figure 3-46: Example of adding items to be acquired



Legend:

 : Storage area for real-time statistics

 : Real-time statistics of the target added by executing the `rtssstats` command

1. Specify the `rtssstats` command as follows and execute it:

```
rtssstats -a -u svc -s server-1 -v service-1 -e ID2 -f file-1
```

2. RTSSPP adds item ID2 specified in the `-e` option and adds item ID3 by using

file-1 specified in the `-f` option.

See the following for details on how to change the settings of targets and the items to be acquired while the real-time statistics service is running.

(1) Adding a target

To add a target, execute the `rtstats` command with the `-a` option specified.

If the number of targets exceeds the value specified in the `rts_service_max` operand in the real-time statistics service definition, the `rtstats` command terminates with a KFCA32742-E message. Because statistics for the entire system are not included in the targets, they can be added even though the number of targets has reached the maximum value.

When you specify `srv` for the `-u` option, acquisition of the following statistics is added for the server specified in the `-s` option:

- Statistics for the entire server
- Statistics for each service provided by the server
- Statistics for processing other than services (such as the main function)

Therefore, you will specify two or more targets than the number of services provided by the server. If you specify a system server or SUP which does not run any services, the overall statistics for the server are acquired.

Use either of the following methods to specify the items to be acquired for the target to be added:

- Specify an item ID in the `-e` option of the `rtstats` command.
The item with the specified item ID is added.
- Specify the name of the definition file for the items to be acquired in real time in the `-f` option of the `rtstats` command.

All the items for which `Y` is specified in the specified definition for the items to be acquired in real time are added.

If the number of items to be acquired exceeds the value specified in the `rts_item_max` operand in the real-time statistics service definition, the `rtstats` command terminates with a KFCA32743-E message.

(2) Adding an item to be acquired

To add an item to be acquired for the target for which real-time statistics are already being acquired, specify the item to be added in the `-a` option in the `rtstats` command and execute the command the same way as if you were adding a target.

(3) Deleting a target

To delete a target, execute the `rtstats` command with the `-d` option specified. Do

not specify the `-e` and `-f` options at this time.

(4) Deleting an item to be acquired

To delete an item to be acquired for the target for which real-time statistics are already being acquired, specify the item in the `-d` option and execute the `rtstats` command.

Specify the item to be deleted in the `-e` or `-f` option. When you specify the `-e` option, the item with the specified item ID will be deleted. When you specify the `-f` option, all the items will be deleted except for the items for which `Y` is specified in the specified definition for the items to be acquired in real time.

(5) Notes

- You can use the `rtstats` command to change targets and the items to be acquired only while the real-time statistics service is running. When you stop the real-time statistics service normally, the changes you made using the `rtstats` command will be discarded the next time the real-time statistics service is started and the targets and the items to be acquired that are specified in the `rtspout` definition command for the real-time statistics service are set again.
- When you specify `srv` in the `-u` option of the `rtstats` command and execute the command, the name of each service provided by the specified server is acquired from the `service` operand in the user service definition of that server. Therefore, the user service definition file must be stored in the directory specified in the `DCCONFPATH` or `DCUAPCONFPATH` environment variable.

If the user service definition file is not stored in the correct directory, the `rtstats` command will not work. If the `service` operand is not specified in the user service definition file, the real-time statistics service acquires real-time statistics on a server-by-server basis.

- If you add or remove services from the `service` operand in the user service definition after the `rtstats` command is executed with `srv` specified for the `-u` option, their status as targets of the real-time statistics service does not change. To add or remove acquisition targets after you start the real-time statistics service, execute the `rtstats` command again.

Services added or deleted by the facility for dynamic loading of service functions after the `rtstats` command is executed are also not added or removed as targets of the real-time statistics service until you execute the `rtstats` command again.

Chapter

4. OpenTP1 File Operations

This chapter explains the operations of files used for OpenTP1.

This chapter contains the following sections:

- 4.1 OpenTP1 file system operations
- 4.2 Status file operations
- 4.3 System journal file operations
- 4.4 Recovery journal file operations
- 4.5 Checkpoint dump file operations
- 4.6 DAM file operations
- 4.7 TAM file operations

4.1 OpenTP1 file system operations

This section describes how to operate an OpenTP1 file system. For details about estimating the size of an OpenTP1 file system, see *K. Estimating the Size of an OpenTP1 File System*.

4.1.1 Creation of OpenTP1 file system

The OpenTP1 administrator must initialize an OpenTP1 file system with the `filmkfs` command. Either a character special file or an ordinary file must be allocated as an OpenTP1 file system area. Initialization is required only once when an OpenTP1 file system area is allocated.

When the OpenTP1 file system is created in a character special file, the OpenTP1 administrator must note that specifying a value larger than the partition size may damage the partitions physically following that partition.

When the OpenTP1 file system is created in an ordinary file, the size of the OpenTP1 file system must be estimated correctly in order to avoid a space shortage.

4.1.2 Display of OpenTP1 file system status

The `filstatfs` command displays the status of an OpenTP1 file system for the following purposes:

- To check the sector length specified during initialization
- To determine the total size of the area allocated to users
- To determine the size of the unused space in the area allocated to users
- To determine the number of existing files and the number of files that can be created
- To check the details of the used and unused (free) areas in the user area

The displayed information includes the total size of the user area, the number of existing files, the file system initialization time, etc.

4.1.3 Display of OpenTP1 file system contents

The `fills` command displays the contents of an OpenTP1 file system for the following purposes:

- To check the existing files in the OpenTP1 file system
- To check the OpenTP1 file owners, access rights, record lengths, numbers of records, and most recent update dates and times
- To check the lock status of the OpenTP1 files

- To output an OpenTP1 file information listing before executing the `filchmod` command (to change the access authorization mode) or the `filchown` command (to change the owner) for a specified file.

The displayed information includes the access rights, owner names, most recent update dates, etc.

Options can be specified in the `fills` command in order to display the files in reverse order of when they were updated or accessed.

4.1.4 Backup of OpenTP1 file system

An OpenTP1 file system can be backed up by the `filbkup` command. OpenTP1 file system backup should be performed periodically for protection against possible OpenTP1 file system errors.

When the `filbkup` command is executed, the attributes of the individual OpenTP1 files are also backed up, including the record length, number of records, and most recent update date and time.

Only a single OpenTP1 file system can be backed up into a backup file. To back up multiple OpenTP1 file systems, a separate file must be specified for each OpenTP1 file system.

It is also possible to specify the names of selected OpenTP1 files for backup processing (this option is usually used in collecting maintenance information).

4.1.5 Restoration of OpenTP1 file system

The information backed up by the `filbkup` command can be restored into an OpenTP1 file system area with the `filrstr` command.

It is also possible to restore particular OpenTP1 files by specifying their names (this option is usually used to collect maintenance information).

4.1.6 Change of OpenTP1 file owner

The OpenTP1 administrator can change the owner of an OpenTP1 file with the `filchown` command.

4.1.7 Change of OpenTP1 file group

The OpenTP1 administrator can change the group for OpenTP1 files to a different group with the `filchgrp` command.

4.1.8 Change of OpenTP1 file access authorization mode

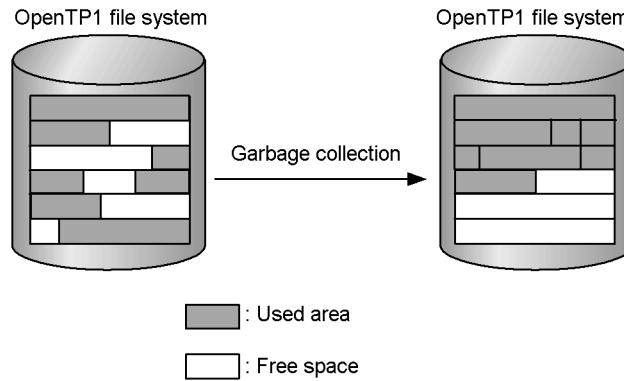
The OpenTP1 administrator can change the access authorization mode for an OpenTP1 file with the `filchmod` command.

4.1.9 OpenTP1 file system garbage collection

As many instances of OpenTP1 file creation and deletion are performed, it may become impossible to create a new OpenTP1 file because a sufficient amount of contiguous area cannot be allocated even though the OpenTP1 file system has adequate free space. Garbage collection rearranges the used areas in the OpenTP1 file system in order to defragment the free space.

The following figure shows garbage collection in an OpenTP1 file system.

Figure 4-1: Garbage collection in OpenTP1 file system



Following is the garbage collection procedure:

1. Back up the OpenTP1 file system with the `filbkup` command
2. Initialize the backed up OpenTP1 file system with the `filmkfs` command
3. Restore the backup file into the initialized OpenTP1 file system with the `filstr` command

Executing the `filstatfs` command with the `-s` option specified enables you to check the status of contiguous areas in the OpenTP1 file system. Then, based on these checking results, when you delete an unneeded OpenTP1 file, the preceding and succeeding free areas are linked, securing a larger contiguous free area.

4.1.10 Change of OpenTP1 file system attributes

The attributes of an OpenTP1 file system, such as the size and the maximum number of files, can be changed. Following is the procedure for changing an OpenTP1 file system's attributes:

1. Back up the OpenTP1 file system with the `filbkup` command
2. Change attributes (such as the sector length), with the `filmkfs` command
3. Restore the backup file into the initialized OpenTP1 file system with the `filstr` command

command

4.1.11 Re-creation of OpenTP1 files

When you create an OpenTP1 file, you can re-create an existing OpenTP1 file by using the appropriate command options.

The following table shows the combinations of commands and options required for re-creating an OpenTP1 file.

Table 4-1: Command options for re-creating an OpenTP1 file

Command for creating an OpenTP1 file	Command for deleting an OpenTP1 file	Option for re-creating an OpenTP1 file
filrstr	Execute one of the following commands depending on the type of the target OpenTP1 file: <ul style="list-style-type: none"> • dandcl • jnlrm • mqadel • querm • stsrn • tandcl 	Specify the -t, -o, or -r option.
mqainit	mqadel	Specify the -r option.
queinit	querm	Specify the -r option.
stsinit	stsrn	No option is required.

Note

For details about creating an ISAM file, see the manual *Indexed Sequential Access Method ISAM*.

When you re-create an OpenTP1 file, a new OpenTP1 file is created and the old OpenTP1 file is deleted. Therefore, an unused area for creating an OpenTP1 file is temporarily required. If unused areas are insufficient, an insufficiency error occurs and the re-creation of an OpenTP1 file fails. In this case, use the appropriate command to delete the OpenTP1 file and then execute the appropriate command to create the OpenTP1 file.

4.2 Status file operations

This section describes how to use status files. For details about estimating the size of a status file, see *H.1 Estimating the size of a status file*.

4.2.1 Creation and definition of status files

- Creation of physical file

Before creating OpenTP1, the user must create status files in an OpenTP1 file system with the `stsninit` command. Status files for both systems A and B must be created. The physical files for system A or B must be created over multiple disks. If all the physical files of either system are created on a single disk, error in the entire disk disables the system. Full recovery of OpenTP1 requires at least one normal file for each of systems A and B, and is impossible with single system's file(s). The name of the physical file must be same as the status file name in the status service definition.

- Correspondence between a physical file and a logical file

Next, the names of the created status files and a logical file name must be specified in the status service definition. A logical file name is a user-assigned name to be used to associate the status files for both systems A and B as a set.

Whether one-system operation is to be selected when only the status file for either system A or B can be used must be specified in `sts_single_operation_switch` of the status service definition.

For details of the status service definition, see the manual *OpenTP1 System Definition*.

4.2.2 Use of status files

Status files are used to record the operating status of each system service comprising OpenTP1, the status of files, system control information for various devices, etc.

OpenTP1 manages status files for both systems A and B in the following statuses:

- Current

Status file is open and is subject to output of system control information. A file entity is required.

- Standby

Status file is not currently subject to output of system control information, but it is open so that the current status file can be switched to this status file if the current status file becomes unavailable due to an I/O error, etc. A file entity is required.

- Invalid

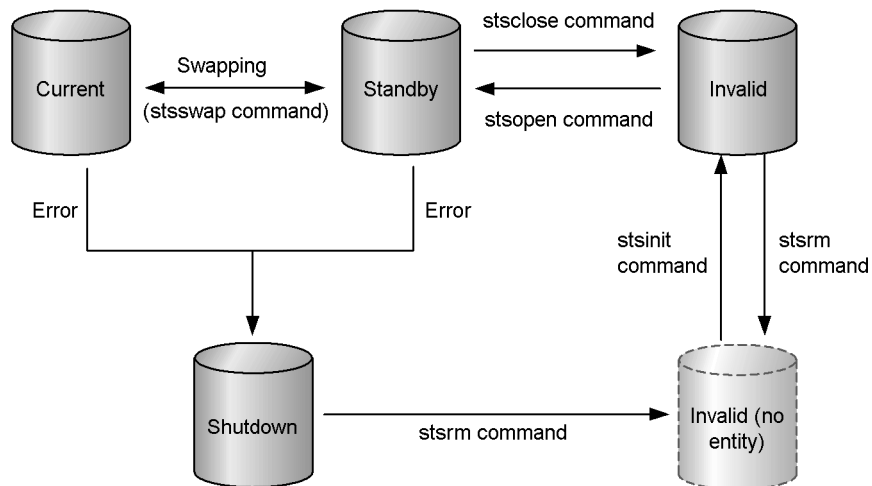
Status file has been specified in the status service definition, but it is currently closed and cannot be used in the online mode until it is opened. When all data has been deleted from a file so that it is empty, the file is called an entity-less invalid file. OpenTP1 cannot be started if an entity-less invalid file exists.

- Shutdown

Status file is currently shut down because an error occurred in the status file in the online mode. A status file on shutdown status can be restored to standby status by deleting its file entity with the `stsrn` command, initializing it with the `stsin` command, and then opening it with the `stso` command.

The following figure shows the status changes of a status file in the online mode.

Figure 4-2: Status changes for status file in online mode



A maximum of seven status files can be specified in the status service definition for each of the systems (A and B). When OpenTP1 is started normally, the first status file specified in the status service definition is placed on current status, any remaining status files that can be opened are placed on standby status, and any status files that cannot be opened are placed on reserve status. A status file that cannot be opened due to an error is placed on shutdown status. When OpenTP1 is restarted, the previous current file is inherited.

The same information is written in the current status files for both systems A and B.

If an I/O error occurs in one of the current status files for either system A or B, OpenTP1 copies the contents of the other system's current file into a standby file of the systems A and B and then places the standby file on current status. Such switching of current files is called swapping. Files can also be swapped by the user with the `stsswap` command.

A status file swapped due to an error becomes a spare file if it is deleted by the `stsrn` command, initialized by the `stsnit` command, and then opened by the `stsope` command.

Where no standby file is available, the processing continues under the normal system alone if one-system operation has been specified in the status service definition (`sts_single_operation_switch=continue`); OpenTP1 terminates abnormally if one-system operation has not been specified (`sts_single_operation_switch=stop`).

If, during one-system operation, the normal system fails or OpenTP1 terminates abnormally when a status file is updated, it becomes impossible to restart OpenTP1. Therefore, it is required that double-system operation (both A and B being active) be restored as early as possible by taking either of the following two procedures:

- Open a reserved file with the `stsope` command for use as a standby file. Execute the `stsswp` command to swap the current file.
- Execute the `stsrn` command to delete the file that has been shut down because of an error. Then initialize the file with the `stsnit` command and open it with the `stsope` command. For one-system operation, if the `stsope` command is executed, the contents of the file for the normal system are copied onto the file for the defective system so that it can be reused as a current file.

4.2.3 Display of status file status

Status file utilization status is displayed by the `stsls` command.

The displayed information includes the logical file status, file record utilization factor, etc.

4.2.4 Display of status file contents

The contents of a status file can be displayed in the offline mode by executing the `stsfills` command.

The displayed information includes the physical file name, initialization time, record length, etc.

If the `-x` option is specified in the `stsfills` command, the status file is first opened under exclusive control and then the contents of the file are displayed.

4.2.5 Status file open and close

An opened status file means that its file entity is on current or standby status. A closed status file means that its file entity is on invalid or shutdown status.

When an invalid status file is opened by the `stsope` command, it is placed on standby status.

If the current status file is under one-system operation, execute the `stsnit` command to

initialize the file that has been shut down because of an error and then execute the `stsoopen` command for use as a current file.

When a standby status file is closed by the `stsclose` command, it is placed on invalid status.

4.2.6 Deletion of status file

An invalid status file and a status file on shutdown status due to an error can be deleted with the `stsrn` command. Current and standby status files cannot be deleted.

4.2.7 Status file space shortage

If a space shortage occurs in a status file in the online mode, OpenTP1 selects a swappable file from the standby files and swaps the files automatically. If there is no swappable standby file of an adequate size, OpenTP1 terminates abnormally. Therefore, if a status file's record utilization factor, which can be displayed with the `stsls` command, becomes high, the size of the status file should be increased by implementing the following procedure:

1. Close one of the standby files with the `stsclose` command
2. Delete the closed file with the `stsrn` command
3. Execute the `stsrn` command specifying for its `-c` option a value greater than the number of records in the current file
4. Open the re-created file with the `stsoopen` command in order to use it as the spare file

4.2.8 Status changes for status files

The following table shows the status changes of status files in the online mode.

Table 4-2: Status changes for status files in online mode

Status		Crnt	Stb	Invalid		Shut-down
		ACT	STB	CLS	ND	BLK
Open/close		Open		Close		
Status number		1	2	3	4	5
Evt	Execution of <code>stsswap</code>	→ 2	→ 1	--	--	--
	Execution of <code>stsclose</code>	--	→ 3	--	--	--
	Execution of <code>stsoopen</code>	--	-	→ 2	--	--

Status		Crnt	Stb	Invalid		Shut-down		
		ACT	STB	CLS	ND	BLK		
Open/close		Open		Close				
Status number		1	2	3	4	5		
Output error	Both systems A and B defective		→ 5	--	--	--	--	
	Either system A or B defective	Standby file available		→ 5	--	--	--	--
		Standby file unavailable	One-system operation available	Both systems in operation	--	--	--	--
				Only one system in operation	→ 5	--	--	--
			One-system operation unavailable		→ 5	--	--	--

Legend:

Crnt: Current

ACT: ACTIVE

STB (or Stb): STANDBY (or Standby)

CLS: CLOSE

ND: NODE

BLK: BLOCKADE

Evt: Event

--: Impossible or inapplicable status change

→ *n*: Target status number

Note

This table assumes that all events process normally. The status of a status file is managed only while the status service is operating. Therefore, if the `stsinit` or `stsrn` command is executed on the closed files listed below during status service operation, the file status to be displayed by the `stsls` command will not change:

- CLOSE
- NONE

- BLOCKADE

4.3 System journal file operations

This section describes how to use system journal files. For the details about estimating the size of a system journal file, see *H.2 Estimating the size of a system journal file*.

4.3.1 OpenTP1 journals

This subsection describes the structure of an OpenTP1 journal.

(1) Records

An OpenTP1 journal contains OLTP records and journal records.

OLTP record

An OLTP record is an I/O unit by which records are read from and written to a journal. Each OLTP record is 4,096 bytes long. The number of records specified in the `-n` option of the `jnlinit` command, and the number of records displayed when the `-d` option is specified for the `jnlis` command, both refer to the number of OLTP records.

Journal record

A journal record is a unit of data that makes up a journal block. A logical unit of information generated by the system, such as a unit of historical or statistical information, constitutes a single journal record. The maximum length of a journal record is the value specified in the `jnl_max_datasize` operand in the system journal service definition. The records displayed when you execute the `jnledit` command are journal records.

(2) Journal blocks

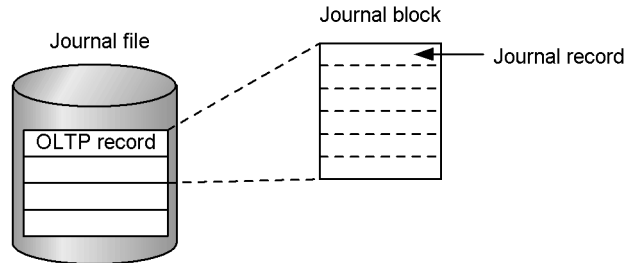
Journal blocks are the units by which the data in the journal buffer is output to a journal file. The space allocated for a single journal buffer is equivalent to the `jnl_max_datasize` operand in the system journal service definition. A single journal block comprises between 1 and n journal records. If the system cannot write any more journal records to the journal buffer when you attempt to acquire journal records, or if a request for acquisition of a synchronization point journal record occurs, the journal records acquired up to that point are output to the journal file as one journal block. When a journal block is written to the journal file, the block number shown when you execute the `jnlis` command is incremented by one.

An OpenTP1 journal outputs a single journal block using between 1 and n OLTP records. If a value of 32,000 is specified in the `jnl_max_datasize` operand in the system journal service definition, the OpenTP1 journal uses between 1 and 8 OLTP records. The equation used to arrive at the maximum of 8 is $32,000/4,096$ (which is then rounded up to a whole number). When the journal buffer cannot acquire any more journal records, the OpenTP1 journal uses eight OLTP records to output the journal

block. However, when a synchronization point journal record is encountered, the OpenTP1 journal acquires the journal records already in the journal buffer, as a journal block. Here the journal block may be less than 4,096 bytes long, in which case only one OLTP record will be used.

The following figure shows the relationship between records and journal blocks.

Figure 4-3: Relationship between records and journal blocks



In Figure 4-3, one journal block comprises six journal records, and is stored in the journal file using two OLTP records.

You can determine how much space an OpenTP1 journal file is using. Multiply the number of records output by the `jnl1s -d` command by the OLTP record length, in this case 4,096 bytes. For example, when the `jnl1s` command produces the execution results shown in the figure below, the capacity used by the journal file `jnlgrp02` will be $(13)_{16} \times 4,096 \text{ bytes} = 77,824 \text{ bytes}$.

Figure 4-4: Execution results of `jnl1s -j sys -d`

```
$ jnl1s -j sys -d
Group  Type  Resource  Generation number  Status  Run ID  Block number
jnlgrp01  sys  sysjnl  1  oc-d--u  3e547830  1  0
ElementElement status  Number of records  Status of system A
jnlgrp01 ou-u--  0  2e oc-u--
Group  Type  Resource  Generation number  Status  Run ID  Block number
jnlgrp02  sys  sysjnl  2  os----u  3e52e7ee  6  9
ElementElement status  Number of records  Status of system A
jnlgrp02 ou-u--  13  2e os-u--
```

4.3.2 Creating and defining system journal files

Before starting OpenTP1, the user must issue the `jnlinit` command in order to create physical files in an OpenTP1 file system. The name of the physical file must be same as the physical file name in the journal service definition.

Once a physical file has been created, the correspondence of the physical file to a *file*

group (a logical file used to manage a number of physical files), and the correspondence of the file group to the element files must be specified in the system journal service definition.

File groups are logical files that are used by OpenTP1 to manage journal generations. During online operation, swapping of journal files takes place on a file group basis. Users can specify from 2 to 256 file groups in the system journal service definition, and give them any name they wish. However, the name must be unique within the system journal service definition.

Element files are logical files that are dispersed when the parallel access facility is used for system journal files. During online mode, journal file access is distributed among the element files, thus distributing the load on the disk.

To use the parallel access facility, in the system journal service definition, a value of 2 or more must be specified for the `jnl_max_file_dispersion` operand, and a value must be specified for the `jnl_min_file_dispersion` operand. One file group can accommodate from 2 to 8 element files, each with an arbitrary user-defined name which must be unique within the system journal service definition. For the `jnl_max_file_dispersion` operand, specify the maximum number of element files in each file group. For the `jnl_min_file_dispersion` operand, specify the number of element files needed to ensure a basic level of parallel access even while an element file is inaccessible due to an error.

You do not need to specify element files if you do not intend to use the parallel access facility.

System journal files can be duplicated. This is done (before starting OpenTP1) by using the `jnlinit` command to create physical files for both systems A and B. Once the physical files have been created, `jnl_dual=Y` must be specified in the system journal service definition and the correspondences between file groups and the physical files for systems A and B must be specified. Additionally, the user must specify in the `jnl_singleoperation` operand of the system journal service definition whether or not one-system operation is to be made available in the event one of the systems becomes unavailable.

Journal files that have been used in online mode must be re-created if you subsequently change any of the following aspects of the OpenTP1 environment:

- The `node_id` operand in the system common definition
- The name of a file group specified by the `-g` option of the `jnladdfg` or `jnladdpf` definition command in the system journal service definition
- The definition file name of the system journal service definition

The message KFCA01240-E is output if you start OpenTP1 without first re-creating the journal files. In this case, you will be unable to use the affected journal files.

For details of system journal service definition, see the manual *OpenTP1 System*

Definition.

4.3.3 Using system journal files

System journal files collect a recovery journal containing historical information about OpenTP1 and a user journal containing historical information about UAPs.

An element file is said to be in *open* status when at least one of its constituent physical files is open, and in *close* status when none of its constituent physical files are open.

When at least the required number of physical files are open, the element file is said to be *usable*. When fewer than the required minimum number of physical files are open, the element file is said to be in *unusable* status. The number of physical files required to constitute an element file is determined by specifications in the system journal service definition:

- jnl_dual=N: 1
- jnl_dual=Y and jnl_singleoperation=Y: 1
- jnl_dual=Y and jnl_singleoperation=N: 2

The following table shows the relationship between element file status and physical file status.

Table 4-3: Relationship between element file status and physical file status

Element file status 1		Element file status 2	Physical file open/close status	
Usable/Unusable status		Open/close status		
One-system operation available	One-system operation unavailable		System A	System B
Y	Y	Open	Open	Open
Y	N	Open	Open	Close
Y	N	Open	Close	Open
N	N	Close	Close	Close

Legend:

Y: Usable

N: Not usable

When at least one of its constituent element files is open, the file group is said to be in *opened* status. When none of the element files are open, the file group is said to be in *closed* status.

When at least the minimum number of element files required for load distribution are open, the file group is said to be *available*. When fewer than the required minimum number of element files are open, the file group is said to be *unavailable*.

The number of element files required for load distribution is determined by specifications in the system journal service definition:

- With parallel access facility disabled: 1
- With parallel access facility enabled: Minimum number of dispersed files for parallel access

If the number of available element files falls below the value specified for the `jnl_min_file_dispersion` operand due to an error or some other reason, the file group will enter closed status. The following table shows the status of file groups, element files, and physical files when an error related to element files occurs.

Table 4-4: Relationship between file statuses when an element file error occurs

Status				File group status	Element file status	Physical file status
Parallel access disabled	Error with non-duplicated system, or errors with both systems A and B			Closed	Closed	Closed
	Error with either system A or B	One-system operation available		Opened	Opened	
		One-system operation unavailable		Opened (Reserve)	Closed	
Parallel access enabled	Error with non-duplicated system, or errors with both systems A and B			Closed	Closed	Closed
	Error with either system A or B	One-system operation available	Usable element files equal to or greater than minimum number of dispersions	Opened	Opened	
			Usable element files less than minimum number of dispersions	Opened	Opened	

Status				File group status	Element file status	Physical file status
		One-system operation unavailable	Usable element files equal to or greater than minimum number of dispersions	Opened	Closed	
			Usable element files less than minimum number of dispersions	Opened (Reserve)	Closed	

OpenTP1 manages a file group of system journal files in the online mode in three statuses:

- Current

The file group is available and is currently subject to journal output; there is always only one file group in this status.

- Standby

The file group is not currently subject to journal output, but it is available and can be switched onto current status.

Standby status can also be further classified into swappable and nonswappable status:

- Swappable status

This standby file group is overwriteable (contains no journal required for recovery) and has been unloaded (copied to a file by the `jnlunlfg` command). When swapping occurs, this file group can immediately become the current file group.

- Nonswappable status

This standby file group is not overwriteable and is waiting to be unloaded. When swapping is required, this file group cannot immediately become the current file group.

- Reserve

The file group is on unusable status.

At least two file groups are required in current and in standby status (but not

necessarily in reserve status).

When OpenTP1 is started normally, the file groups specified in the system journal service definition for which ONL has been specified are opened. Among the opened file groups, the first file group specified is placed on current status, and the rest are placed on standby status. Any file group that cannot be opened or for which ONL was not specified is placed on reserve status. When OpenTP1 is restarted, the previous file group on current status is inherited.

The journal is output to the current file group. When the current file group becomes full, the next file group (on standby status) specified in the system journal service definition is swapped in. If dual journals are used, swapping occurs whenever one of them becomes full. When all file groups in the system journal file become full, journal output resumes in the first file group.

A file group containing a journal required for OpenTP1 recovery is determined to be not overwritable and is protected (a non-overwritable file group cannot be placed on current status).

Once a checkpoint dump is collected by OpenTP1, a non-overwritable file group returns to overwritable status, because the previous journal is no longer needed.

4.3.4 Unloading of system journal files

A file group placed on standby status because it is full, because an error occurred, or because it was swapped out by an operation command is placed on unload wait status. A file group on unload wait status retains its journal until the user unloads it in order to restore a DAM file or to perform a user operation. A file group on unload wait status cannot be placed on current status.

Once a file group on unload wait status is unloaded by using the `jnlunlfg` command or the automatic unloading facility, it is placed on *unload completion status*. A file group on unload wait status enters unload completion status when its journal is copied by using the `jnlunlfg` command or the automatic unloading facility, or its journal is discarded by using the `jnlchgfg` command.

A file created as a result of unloading with the `jnlunlfg` command or the automatic unloading facility is called an *unload journal file*.

(1) Unloading using the `jnlunlfg` command

When OpenTP1 outputs a message indicating that a journal has been discarded from the file group (KFCA01222-I), execute the `jnlunlfg` command.

If the `jnlunlfg` command is specified with its `-f` option specified, a file group is unloaded without checking its status; therefore, the current file group and an unloaded file group can be unloaded. Note that, before unloading an unloaded file group by executing the `jnlunlfg` command with its `-f` option specified, the file group must be closed with the `jnlclsfg` command.

Executing the `jnlunlfg` command with its `-f` option specified cannot change the file group status.

(2) Unloading using the automatic unloading facility

By using the automatic unloading facility, you can automatically unload system journal files while OpenTP1 is running without executing the `jnlunlfg` command. To use the automatic unloading facility, specify `Y` in the `jnl_unload_check` operand and the `jnl_auto_unload` operand in the system journal service definition. Specify the directory for storing the unload journal files in the `jnl_auto_unload_path` operand in the system journal service definition.

The automatic unloading facility can unload only system journal files; it cannot unload archive journal files.

(a) When unloading occurs

The automatic unloading facility starts unloading when the current file group containing system journal files is released and enters the unload wait status. The unload wait status begins at the following times:

- The current file group is swapped with another file group because it is full or has failed, or because an operation command was used.
- The OpenTP1 system ended normally.

(b) Using the unload journal file storage directory

In the `jnl_auto_unload_path` operand in the system journal service definition, specify the directory for storing the unload journal files to be unloaded by the automatic unloading facility. You can specify multiple directories for storage.

When a single directory is specified for storing unload journal files

- If OpenTP1 detects a full disk during unloading, OpenTP1 deletes the unload journal file being created and stops the automatic unloading facility.
- If OpenTP1 detects a disk failure during unloading or the unload journal file storage directory becomes unavailable for some reason, OpenTP1 stops the automatic unloading facility.

When multiple directories are specified for storing unload journal files

The following table shows the first unload journal file storage directory to be used when OpenTP1 starts.

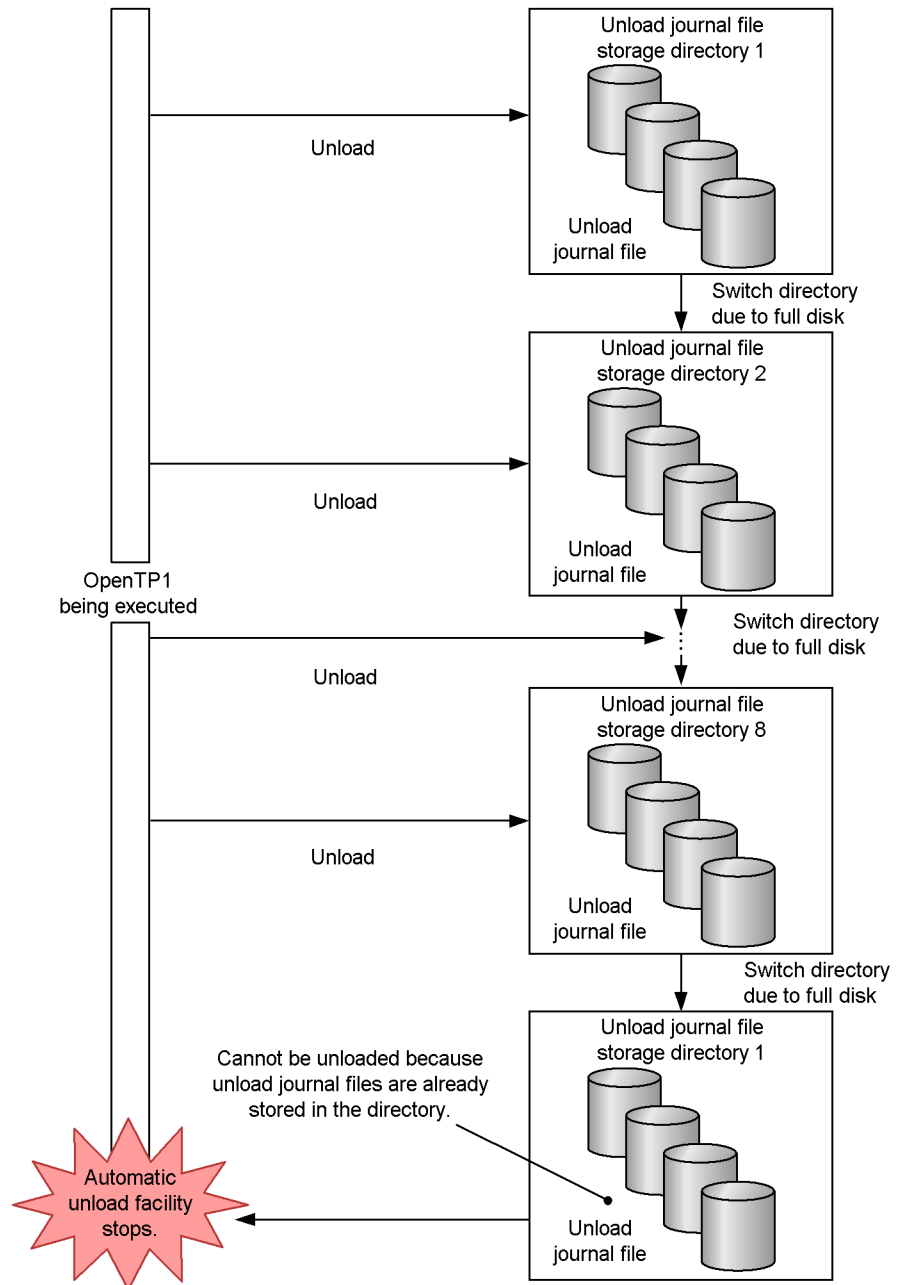
Table 4-5: First unload journal file storage directory to be used

Startup type of OpenTP1	First unload journal file storage directory to be used
Normal start	Of the directories specified in the <code>jnl_auto_unload_path</code> operand in the system journal service definition, the first directory containing no unload journal files that was created by using the automatic unloading facility.
Start after planned stop	Last directory that was used when OpenTP1 was running previously
Restart	

- If OpenTP1 detects a full disk during unloading, OpenTP1 deletes the unload journal file being created, and re-creates the unload journal file in the next directory specified in the `jnl_auto_unload_path` operand in the system journal service definition. If the next directory specified in the `jnl_auto_unload_path` operand in the system journal service definition contains an unload journal file that was created during the previous period of operation, OpenTP1 stores the new unload journal file in the next directory in the sequence. When the directory for the file is determined, OpenTP1 starts unloading the same generation. If OpenTP1 cannot unload the same generation in all the directories specified in the `jnl_auto_unload_path` operand in the system journal service definition, OpenTP1 stops the automatic unloading facility.
- If OpenTP1 detects a disk failure during unloading or if the unload journal file storage directory becomes unavailable for some reason, OpenTP1 unloads unload journal files in the next directory specified in the `jnl_auto_unload_path` operand in the system journal service definition. The directory with a disk failure will be used subsequently for storing files.
- If OpenTP1 detects a full disk error for the directory specified in the last parameter in the `jnl_auto_unload_path` operand in the system journal service definition, OpenTP1 unloads unload journal files to the directory specified in the first parameter in the `jnl_auto_unload_path` operand in the system journal service definition. However, if all the directories, including the directory specified in the first parameter, contain unload journal files that were created during the last period of operation, OpenTP1 stops the automatic unloading facility. You can specify multiple storage directories.

The following figure shows how to use the eight storage directories.

Figure 4-5: Using the unload journal file storage directories



If all the journal files become full during unloading that uses the automatic unloading

facility so that unloading is no longer possible, OpenTP1 outputs message KFCA01220-E and stops the system. If a journal file is closed during unloading that uses the automatic unloading facility, the automatic unloading facility continues processing.

If you store a file with the following name in an unload journal file storage directory, OpenTP1 may delete the file:

(system-journal-service-definition-file-name) _bbbbbbbbbbb_cc....cc

bbbbbbbbbbb: Character string consisting of 12 characters

cc....cc: Character string consisting of 1 to 8 characters

(c) Unload journal files

The unload journal files created by using the automatic unloading facility must follow the following naming convention to prevent duplication of names.

*(system-journal-service-definition-file-name) _ (journal-service-run-ID)
(journal-file-generation-number) _ (journal-file-group-name)*

You can use a maximum 28 characters for a file name. The details about the components of a file name are as follows:

System journal service definition file name: Character string consisting of 8 or fewer characters

Journal service run ID: 8-digit hexadecimal number

Journal file generation number: 4-digit hexadecimal number

If the journal file generation number is fewer than 4 digits, 0s are inserted from the left to make the number 4 digits. If the journal file generation number is 5 digits or more, the last 4 digits are used.

Journal file group name: Character string consisting of 8 or fewer characters

Example

System journal service definition file name: `sysjnl`

Journal service run ID: `40e8ca39`

Journal file generation number: `3`

Journal file group name: `jnlfg02`

Unload journal file name: `sysjnl_40e8ca390003_jnlfg02`

Do not perform any operations on or for the files stored in the directory currently used

as an unload journal file storage directory while the automatic unloading facility is being executed. If you do so, file integrity is not guaranteed. You can check the directory that is being used as the unload journal file storage directory by executing `jnlcatunl -j sys -i`.

You can move and copy unload journal files while the automatic unloading facility is inactive. You can move and copy the unload journal files that are stored in the directory that is not being used as the unload journal file storage directory even if the automatic unloading facility is being used. If you do not use an unload journal file as a file recovery journal, you can delete that file. However, if a directory becomes an unload journal file storage directory while you are manipulating a file in the directory, be careful that you do not manipulate the unload journal files that are created by OpenTP1 in any way.

(d) Restarting the automatic unloading facility

If the automatic unloading facility stops while OpenTP1 is running, you can restart it as follows:

1. Back up the unload journal files in all the directories specified in the `jnl_auto_unload_path` operand in the system journal service definition or delete unnecessary unload journal files to create free space on the disk.

2. Execute `jnlcatunl -j sys -b`.

The automatic unloading facility restarts.

3. Execute `jnlcatunl -j sys -i`.

Check whether the automatic unloading facility is operating.

(3) Inhibiting unload check

A file group on unload waiting status can be allocated as the current one without being unloaded. This is called *inhibiting unload check on journal*. OpenTP1 usually checks the unload status of a file group, and allocates it as the current one only when it is on unload completion status. When unload status check is inhibited, unloading a file group is unnecessary because even a swappable file group on unload waiting status is allocated as the current one.

When unload status check is inhibited, unloading is executed in the offline mode.

Before unloading a file group online for test, close that file group by using the `jnlclsfg` command. If you do not unload an online file group before executing the `jnlclsfg` command, the file group might change to the *current* status, and the data in the file group might become corrupted. In this case, OpenTP1 interrupts unloading because the file group contains both data being unloaded and new data written for the current file group, but does not exit online mode.

The commands that input unloaded-journals file cannot be used unless unloading is executed at least once. The following table lists these commands.

Table 4-6: Commands that input unload journal files

	Function	Command
Management of journal files	Journal integration for file recovery	jnlcolc
	Copying of unloaded-journals file	jnlcopy
	Editing and output of unloaded-journals file	jnledit
	Output of MCF statistics	jnlmcsst
	Recovery of journal files	jnlmkrf
	Output of records of unloaded-journals file	jnlrput
	Merging and chronological sorting of unloaded-journals file	jnlstts
	Output of statistics	jnlstts
DAM file management	Recovery of logical file	damfrc
TAM file management	Recovery of TAM file	tamfrc

Even if the use of the journal unload check inhibit facility is specified, executing the `jnl1s` command displays unload wait status for the file group. However, that file group is allocated as the current one.

4.3.5 Reuse of system journal files

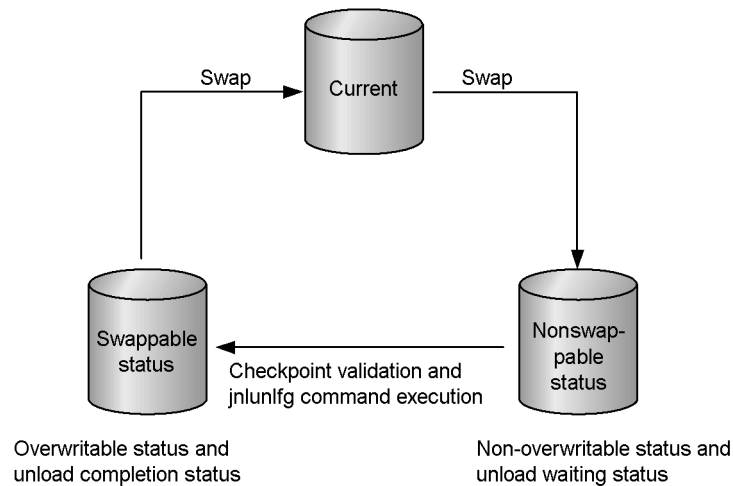
To reuse a file group placed on standby status because it was full or swapped out by an operation command, both of the following conditions must be satisfied:

- The file group is on overwritable status
- The file group is on unload completion status

When `N` is specified in `jnl_unload_check` (facility for inhibiting unload check on journal) in the system journal service definition, the file group is swapped without checking its status.

The following figure shows the status changes for a system journal file in the online mode.

Figure 4-6: Status changes for system journal file in online mode



In the following situation, the system will close a file group and assign the next file group as the current file group. You force a normal start of OpenTP1 that was forcibly terminated or terminated abnormally, and then attempt to assign the journal file that was used as the current file during the previous operation as the current file for the current operation during startup processing.

The closed journal file is not available until you use the `jnlunlfg` command or the `jnlchgfg` command to change the status of the file group and then execute the `jnlpnfg` command.

4.3.6 Display of system journal file information

System journal file information can be displayed by the `jnl1s` command. To display the system journal file information that is read during a restart, use the `jnlrinf` command.

The displayed information includes the file group names, file types, file group statuses, etc.

4.3.7 System journal file open and close

When a reserve file group is opened by the `jnlpnfg` command, it becomes a standby file group.

When an overwritable file group is closed by the `jnlclsfg` command, it becomes a reserve file group. A non-overwritable file group cannot be closed, because it contains a journal required for recovery.

If there is only one overwritable file group, that file group cannot be closed.

The following figure shows the status changes for a system journal file caused by the open and close commands.

Figure 4-7: Status changes caused by open and close commands (system journal file)



#: During a restart, this becomes a standby file group that cannot be overwritten.

4.3.8 Change of system journal file status

The `jnlchgfg` command changes forcibly a standby file group on unload wait status to unload completion status without actually unloading the file group. When a file group is placed forcibly on unload completion status, its journal is not copied.

If a file group placed on unload completion status is overwritable, it can be used as the current file group the next time swapping is performed.

4.3.9 Swapping of system journal files

Whenever the current file group needs to be unloaded or edited, the user can immediately swap system journal files with the `jnlswpfg` command. If there is no swappable file group, an error results.

4.3.10 Nonswappable file group

If the file group becomes full or if a journal output error occurs, OpenTP1 performs swapping. The user must always provide a swappable standby file group so that swapping can always be performed.

If swapping of a journal file group is to be performed, OpenTP1 checks the status in the memory and looks for a swappable standby file group. When OpenTP1 searches the memory for a file group and cannot find any swappable file group, it reads the status from the journal file of the file group being opened and updates the status in the memory to the status in the file. OpenTP1 then checks the status in the memory again and looks for a swappable standby file group. If OpenTP1 still cannot find any file group, it terminates abnormally.

When OpenTP1 terminates abnormally because there was no swappable file group, it can be restarted by unloading the file group with the `jnlunlfg` command or by correcting the erroneous file group. If OpenTP1 still cannot be started, the user must specify a new file group with ONL specified in the system journal service definition and then must restart OpenTP1.

4.3.11 Status changes for system journal files

The following table shows the status changes for system journal files in online mode.

Sts	Current/Standby/Reserve				C	Standby				Reserve			
	Unloaded/Waiting to be unloaded				–	Wtg		Unld		Wtg		Unld	
	Overwritable/Non-overwritable				N	Y	N	Y	N	Y	N	Y	N
	Status number				1	2	3	4	5	6	7	8	9
Evt	Swap due to full disk				→ 3	–	–	→ 1	–	–	–	–	–
	Swap due to error in current system	PAF not used	Error in non-dualized configuration or error in both systems A and B in dualized configuration		→ 7	–	–	→ 1	–	–	–	–	–
			Error in one system in dualized configuration	One-system operation possible	→ 3	–	–	→ 1	–	–	–	–	–
		One-system operation not possible		→ 7	–	–	→ 1	–	–	–	–	–	
		PAF used	Error in non-dualized configuration or error in both systems A and B in dualized configuration		After physical file in which error occurred is shut down	Usable element file count ≥ min. dist. count	→ 3	–	–	→ 1	–	–	–
	Error in one system in dualized configuration		One-system operation not possible	Usable element file count < min. dist. count			→ 7	–	–	→ 1	–	–	–
		One-system operation possible			→ 3	–	–	→ 1	–	–	–	–	–
	Swap using <code>jnlswpfg</code>				→ 3	–	–	→ 1	–	–	–	–	–
	Enabling of checkpoint dump following a swap				–	–	→ 2	–	→ 4	–	→ 6	–	→ 8
	Execution of <code>jnlunlfg</code>				–	→ 4	→ 5	–	–	→ 8	→ 9	–	–

Sts	Current/Standby/Reserve	C	Standby					Reserve			
	Unloaded/Waiting to be unloaded	-	Wtg		Unld			Wtg		Unld	
	Overwritable/Non-overwritable	N	Y	N	Y	N	Y	N	Y	N	
	Status number	1	2	3	4	5	6	7	8	9	
	Execution of <code>jnlchgfg</code>	-	→ 4	→ 5	-	-	→ 8	→ 9	-	-	
	Execution of <code>jnlopnfg[#]</code>	-	-	-	-	-	→ 2	→ 3	→ 4	→ 5	
	Execution of <code>jnlclsfg[#]</code>	-	→ 4	-	→ 8	-	-	-	-	-	

Legend:

Sts: Status

Evt: Event

C: Current

Wtg: Waiting to be unloaded

Unld: Unloaded

Y: Overwritable

N: Non overwritable

PAF: Parallel access facility

min. dist.: minimum distribution

- : Impossible or inapplicable status change

→*n*: Target status number

Note 1

This table assumes that all events are processed normally. The status of a system journal file group is managed only while the journal service is operating. The status of a closed file group (on unload completion/wait status, overwritable/non-overwritable status) is not managed. Therefore, if the `jnlunlfg` or `jnlchgfg` command is executed on a closed file group, the status to be displayed by the `jnlcls` command will not change. The status of the current files can be displayed by executing the `jnlopnfg` and `jnlcls` commands after executing the `jnlunlfg` or `jnlchgfg` command.

Note 2

For the events *Swap due to no space remaining in current file group*, *Swap due to error with current file group*, and *Swap with jnlswpfg*, the system journal files whose status changes from *Current* are the swapped-out files. The system journal files whose status changes from *Standby* are the swapped-in files.

#

The `-a`, `-b`, or `-e` option is omitted.

4.3.12 Output of system statistics to journal

System statistics can be edited and output to a system journal file by entering the `dcstats` command. When the `dcstats` command executes, information is output to the system journal file continuously at the interval specified in the `-m` option. The edited output information includes for each interval the number of event occurrences, the average value, the maximum value, and the minimum value, depending on the type of statistics specified with the `-k` option. It is also possible to specify a server name in order to output this information for the specified server.

Output of system statistics can be terminated by entering the `dcstats` command with the `-r` option specified. If the `-r` option is not specified in the `dcstats` command, output of system statistics continues until OpenTP1 terminates.

To change the type of system statistics or the interval specified in the `dcstats` command, the `dcstats` command must be reexecuted after executing the `dcstats -r` command.

The specification of the `dcstats` command is not inherited when OpenTP1 is restarted. To output system statistics after OpenTP1 has been restarted, the `dcstats` command must be reentered.

Note that system statistics can be edited and output from shared memory to the standard output in real time. To do so, execute the `dcreport` command.

For details about the items to be edited in system statistics, see *Table E-2*.

4.3.13 Chronological sorting and merging of unload journal files

In multinode operation, the contents of unload journal files and global archive unload journal files can be sorted or merged on a block basis by using the `jnl-sort` command. Only a needed range of information can be sorted or merged by specifying the node identifier and journal server run ID.

4.3.14 Copying of unload journal files

The contents of unload journal files can be copied in units of records by the `jnl-copy` command. Only a needed range of information can be copied by specifying a copying range.

4.3.15 Editing and output of unload journal files

The contents of unload journal files can be edited and output in units of blocks or

records by the `jnledit` command. Only a needed range of information can be output by specifying the edit range and journal collection mode. A file listing can also be output by specifying an appropriate option.

4.3.16 Output of records of unload journal files

Information of user journal records in an unload journal file or CPU time information for transaction branches and statistics about response can be output to a standard output without editing its contents by the `jnlrput` command. Only a needed range of information can be output by specifying the output range and journal collection mode. Also, commit-determined user journal records can be output by specifying an appropriate option. The following table lists the data that can be output by the `jnlrput` command.

Table 4-7: Data that can be output by jnlrput command

Data		Contents	
File management information (128 bytes)		This information indicates that the output result is the data obtained by processing output records. It contains the states and contents of the total output results.	
Record information	Record management information (128 bytes)	This information indicates the beginning of record information. It contains the states and contents of the total record information.	
	Record data	Record data header (128 bytes)	This data indicates fixed record data of journal records. It is used as a reserved area if there is no fixed record data.
		Second record data header	This data indicates the second fixed record data of journal records. None is output if there is no fixed record data.
		Record data (variable length)	This data indicates floating record data of journal records. None is output if there is no floating record data.

4.3.17 Output of uptime statistics

Uptime statistics can be collected, edited, and output from an unload journal file by the `jnlstts` command. The uptime statistics that can be edited and output by the `jnlstts` command include system statistics, transaction statistics, statistics about response, and statistics about communication delay. The uptime statistics that can be edited and output by the `jnlmest` command are MCF uptime statistics.

You can use the following commands to output uptime statistics without using an unload journal file:

- `dcreport` command

Allows you to edit the system statistics collected into shared memory, and output them to the standard output in real time.

- `mcfstats` command and `mcfreport` command

Allow you to acquire the MCF uptime statistics about MHPs and TP1/Message Control that have been collected into shared memory, and edit and output the statistics that were collected in a UNIX file.

(1) **Statistics editable with the `jnlstts` command**

(a) **System statistics**

Executing the `jnlstts` command with `sys` specified in the `-e` option (edit item) edits and outputs collected the system statistics chronologically. Only a needed range of information can be edited and output by specifying the editing range.

The contents of system statistics are the same as those which can be output to system journal files. For details, see *Table E-2*.

(b) **Transaction statistics**

Executing the `jnlstts` command with `trn` specified in the `-e` option (edit item) edits and outputs collected transaction statistics chronologically on the basis of the names of, or service names for, the user servers originating transaction branches. Only a needed range of information can be edited and output by specifying the editing range.

Note that, before obtaining transaction statistics, any other than nothing must be specified in `trn_statistics_item` in the user service definition.

The following table shows the edited contents of transaction statistics.

Table 4-8: Edited contents of transaction statistics

Edited contents					Unit
Item	Occurrences	Edited values			
		Average	Maximum	Minimum	
Transaction branch execution time (including time for synchronization point processing)	N	Y	Y	Y	Microseconds
Transaction branch execution time for synchronization point processing	N	Y	Y	Y	Microseconds
Transaction branch determinations	Y	N	N	N	Events
Transaction branch and child-branch determinations	Y	N	N	N	Events
Process type for transaction branch determination	Y	N	N	N	Events

Legend:

Y: Can be edited.

N: Cannot be edited.

(c) Statistics about response

Executing the `jnlstts` command with `rsp` specified in the `-e` option (edit item) edits and outputs the collected statistics about response chronologically on the basis of service names. Only a needed range of information can be edited and output by specifying the editing range.

Note that, before obtaining statistics about response, Y must be specified in `rpc_response_statistics` in the user service definition.

To obtain the RPC type and response time shown in Table 4-9, specify Y for the `rpc_response_statistics` operand in the user service definition for the client UAP, that is, the caller of `dc_rpc_call`. If not specified, the `jnlstts` command outputs zero as the edited results without obtaining these information items.

To obtain the service execution time and service wait time, specify Y in the `rpc_response_statistics` operand in the user service definition for the client or server.

The following table shows the edited contents of statistics about response.

Table 4-9: Edited contents of statistics about response

Edited contents					Unit
Item	Occurrences	Edited values			
		Average	Maximum	Minimum	
RPC type	Y	N	N	N	Events
Response time	Y	Y	Y	Y	Microseconds
Service execution time	Y	Y	Y	Y	Microseconds
Service wait time	Y	Y	Y	Y	Microseconds

Legend:

Y: Can be edited.

N: Cannot be edited.

(d) Statistics about communication delay

Executing the `jnlstts` command with `dly` specified in the `-e` option (edit item) edits and outputs collected statistics about communication delay chronologically on the

basis of destination node identifiers. Only a needed range of information can be edited and output by specifying the editing range.

The following table shows the edited contents of statistics about communication delay.

Table 4-10: Edited contents of statistics about communication delay

Edited contents					Unit
Item	Occurrences	Edited values			
		Average	Maximum	Minimum	
Communication delay	Y	Y	Y	Y	Microseconds

Legend:

Y: Can be edited.

(2) Statistics that the *jnlmcst* command can acquire

The *jnlmcst* command edits and outputs the MCF statistics about the send and receive messages. Statistics can be edited and output on the basis of logical terminal names or application names. In either case, only a needed range of information can be output by specifying the editing range and journal collection mode.

The *jnlmcst* command edits and outputs the MCF uptime statistics created from the statistical journals collected by MCF (IJ, GJ, OJ, and AJ). Therefore, to edit and output the MCF uptime statistics, perform the following definitions in advance:

- System definition

With *mcfaalcap* (application attribute definition) in the MCF application attribute definition, specify that IJ, GJ, and OJ be collected.

- Protocol definition

In the logical terminal definition, specify that AJ be collected.

The following table shows the edited contents of MCF uptime statistics.

Table 4-11: Edited contents of MCF uptime statistics

Information type	Edited contents						Unit
	Event	Output format	Edited values				
			Minimum	Maximum	Average	Total	
Information relating to receiving messages	Number of inquiry message inputs	0 to 4294967295 or *	Y	Y	Y	Y	Inputs

Information type	Edited contents						Unit
	Event	Output format	Edited values				
			Minimum	Maximum	Average	Total	
	Number of receive-only message inputs	0 to 4294967295 or *	Y	Y	Y	Y	Inputs
	Number of receive request inputs	0 to 4294967295 or *	Y	Y	Y	Y	Inputs
	Receive message size	0 to 4294967295, * or -	Y	Y	Y	N	Bytes
Information relating to sending messages	Number of priority branch send request issues	0 to 4294967295 or *	Y	Y	Y	Y	Issues
	Number of general branch send request issues	0 to 4294967295 or *	Y	Y	Y	Y	Issues
	Number of response send request issues	0 to 4294967295 or *	Y	Y	Y	Y	Issues
	Number of outputs (segments)	0 to 4294967295 or *	Y	Y	Y	Y	Outputs

Legend:

Y: Can be edited.

N: Cannot be edited.

(3) Statistics that the `dcreport` command can acquire

By using the `dcreport` command, you can edit the system statistics collected into shared memory, and output them to the standard output in real time.

For details about the system uptime statistics that can be edited and output using this command and the items to be edited, see *Table E-1* and *Table E-2*. When you use this command to edit and output system uptime statistics, you need to specify `set statistics=Y` in the system common definition beforehand.

(4) Statistics that the `mcfstats` and `mcfreport` commands can acquire

The `mcfstats` command acquires the MCF statistics. The `mcfreport` command edits and outputs the acquired MCF statistics.

The following table lists the types of the MCF statistics that these commands can acquire, edit and output. To acquire, edit and output the MCF statistics by these commands, specify `mcfmcomm -w stats=yes` first in the MCF manager definition.

Table 4-12: Types of MCF statistics

Type	Description	Usage
Receive message information (for a service group)	<ul style="list-style-type: none"> • Message-handling count • Total, maximum, and minimum of message-handling wait count and time 	Tuning related to service groups
Send message information (for a logical terminal)		Tuning related to logical terminals
Connection wait information (for a communication server)		Tuning for the maximum processing multipliers (<code>mcftrred -m</code>)

The opportunities to start and terminate outputting the statistics are:

■ Start

The `mcfstats` command starts outputting the MCF statistics to a file.

When you execute the `mcfstats` command, the MCF statistics are output to a file the number of times specified by the `-t` option at the time interval specified by the `-s` option. If you omit the `-t` option, the MCF statistics are output until the termination of the command.

■ Termination

- After outputting the MCF statistics to a file the number of times specified, the command terminates outputting MCF statistics to the file.
- You can specify the `-r` option in the `mcfstats` command to terminate outputting the MCF statistics to a file.
- Output of the MCF statistics to a file terminates at normal termination, planned termination, forced termination and abnormal termination of MCF.

The specification of the `mcfstats` command is not inherited after MCF is restarted. To output the MCF statistics to a file after restarting MCF, reexecute the `mcfstats` command.

- If the MCF statistics that have been output exceed the size of the output file

specified by the `-f` option in the `mcfstats` command, output of the statistics to a file terminates.

The following table lists the contents of the MCF statistics edited by the `mcfreport` command.

Table 4-13: Edited contents of the MCF statistics (`mcfreport` command)

Edit type	Obtained value	Edited values				Unit
		Occurrences	Edited values			
			Average	Maximum	Minimum	
Information about messages to receive	Number of messages in the schedule queue	Y	Y	Y	Y	Requests
	Scheduling wait time	N	Y	Y	Y	Seconds
Information about messages to send	Number of messages in the schedule queue	Y	Y	Y	Y	Requests
	Scheduling wait time	N	Y	Y	Y	Seconds
Information about wait for connection	Number of messages in the schedule queue	Y	Y	Y	Y	Requests
	Scheduling wait time	N	Y	Y	Y	Seconds

Legend:

Y: Can be edited.

N: Cannot be edited.

4.3.18 Journal integration for file recovery

Journal records required for DAM or TAM file recovery can be extracted from unload journal files by the `jnlcolc` command. A file created by executing the `jnlcolc` command is called an integrating journal file. Using an integrating journal file during DAM or TAM file recovery reduces the time required for file recovery processing.

4.3.19 Journal fileless mode

Journal fileless mode is a mode in which system journal files (recovery journals containing historical information about OpenTP1, user journals containing historical information about UAPs, and statistical journals) are not acquired.

OpenTP1 acquires system journal files which can be used to recover the system or investigate aspects of OpenTP1 operation. However, system journal files do not need to be acquired in the following system configurations.

Examples of systems where system journal file acquisition is unnecessary:

- Systems that use TP1/EE to control transactions
- Systems that do not implement system recovery using the journal service
- Systems that use name service nodes instead of transaction control

Using journal fileless mode for systems of this nature can help free resources for other uses.

To enable journal fileless mode, specify `Y` for the `jnl_fileless_option` operand in the system common definition. Environments that use journal fileless mode are said to be running in *journal fileless mode*.

The OpenTP1 facilities listed in the following table cannot be used in journal fileless mode.

Table 4-14: Facilities unavailable in journal fileless mode

Category	Unavailable facility
Facility	Transaction facility ^{#1}
	Transactional linkage using the XA resource service
	Resource managers (such as DAM and TAM) provided by OpenTP1
	Statistical information
	Global archive journal facility
	Automatic unloading facility

4. OpenTP1 File Operations

Category	Unavailable facility
Library function	dc_jnl_ujput
	dc_trn_begin
	dc_trn_unchained_commit
	dc_trn_chained_commit
	dc_trn_unchained_rollback
	dc_trn_chained_rollback
	dc_trn_info
	tx_begin
	tx_close
	tx_commit
	tx_info
	tx_open
	tx_rollback
	tx_set_commit_return
	tx_set_transaction_control
tx_set_transaction_timeout	

Category	Unavailable facility
Operation command	jnl1s
	jnlunlfg
	jnlchgfg
	jnlpnfg
	jnlclsfg
	jnlswpfg
	jnlrinf
	jnlde1pf
	jnladdpf
	jnlardis
	jnlar1s
	jnl1atun1
	jnlmkrf
	dcdefchk (logic checks of the journal service definition, system journal service definition, and checkpoint dump service definition)
Other	System recovery by restarting OpenTP1
	atomic_update operand ^{#2} in the user service definition

#1

Because it is not possible to access resource managers via the XA interface, the `trnlnkrm` command cannot be used to register resource managers with OpenTP1. If a resource manager is registered with OpenTP1 when you start OpenTP1 in journal fileless mode, an error occurs in the startup processing of the monitoring service for the transaction resource manager, and OpenTP1 terminates abnormally.

#2

Because the transaction facility is unavailable, OpenTP1 operates as if the `atomic_update` operand in the user service definition is set to N, regardless of the actual setting.

The OpenTP1 files and definition files listed in the following table cannot be used in journal fileless mode.

Table 4-15: Files unavailable in journal fileless mode

File type	Unavailable file
OpenTP1 file	System journal file
	Checkpoint dump file
Definition file	Journal service definition file
	System journal service definition file
	Checkpoint dump service definition file

4.4 Recovery journal file operations

4.4.1 Transaction recovery journal file operations

(1) *Recovery of transaction recovery journal files*

If an error occurs with a transaction recovery journal file and the transaction cannot be recovered at restart of OpenTP1, you can use the unload journal file to recover the transaction recovery journal file with `jnlmkrf` command.

4.4.2 Server recovery journal file operations

(1) *Recovery of server recovery journal files*

If an error occurs with a server recovery journal file and the service cannot be recovered at restart of OpenTP1, you can use the unload journal file to recover the server recovery journal with the `jnlmkrf` command.

For details about estimating the size of a server recovery journal file, see *H.3 Estimating the size of a server recovery journal file*.

4.5 Checkpoint dump file operations

This section describes how to use checkpoint dump files. For details about estimating the size of a checkpoint dump file, see *H.4 Estimating the size of a checkpoint dump file*.

4.5.1 Creation and definition of checkpoint dump files

A checkpoint dump file consists of multiple (2 to 30) file groups. File groups are logical files. Normally, one physical file is assigned to one file group but two physical files can be assigned to one file group (duplication).

(1) Creation

1. Initialization of a physical file

Before starting OpenTP1, create a physical file for a checkpoint dump file on the OpenTP1 file system, with the `jnlinit` command.

2. Correspondence between a physical file and a logical file (file group)

Associate the created physical file to the file group, with the `jnladdpf` definition command in the checkpoint dump service definition.

(2) To duplicate a physical file

When a checkpoint dump file is duplicated, even if full recovery using the system A file failed due to an error, full recovery can be performed using the file of the other system (system B).

1. Initialization of a physical file

Before starting OpenTP1, create two physical files (one for system A and the other for system B) on the OpenTP1 file system, with the `jnlinit` command.

The physical files for system A or B must be created over several disks. If all the physical files of either system is created on a single disk when one-system operation is unavailable, error in the entire disk disables the system. The OpenTP1 system will go down, assuming that there is no file group available.

2. Definition

Do not forget to specify `jnl_dual=Y` in the checkpoint dump service definition. The correspondence between a file group and the created physical files (for systems A and B) is specified with the `jnladdpf` definition command.

When a physical file is duplicated, one-system operation is available in case of an error. One-system operation can be enabled or disabled with the `jnl_singleoperation` operand in the checkpoint dump service definition.

For details of the checkpoint dump service definition, see the manual *OpenTP1 System Definition*.

4.5.2 Use of checkpoint dump files

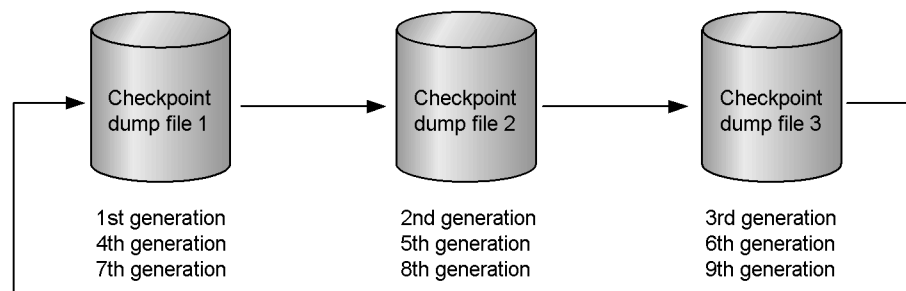
A checkpoint dump file is used to collect table information required for OpenTP1 recovery.

In order to handle possible errors, OpenTP1 collects checkpoint dumps in multiple checkpoint dump files, using the round-robin scheduling method. The most recent checkpoint dump generation is required for full recovery of OpenTP1. During full recovery, all recovery journals collected from the time the most recent checkpoint dump generation was collected until the time OpenTP1 terminated are also required.

If the most recent checkpoint dump generation cannot be read, the immediately preceding checkpoint dump generation is used. If this checkpoint dump cannot be read, the generation collected immediately before it is used. Thus, the checkpoint dump files are searched backward for a usable generation, and all recovery journals collected after collection of the checkpoint dump to be read are required. If needed journals have been lost (because they have been overwritten by other data), OpenTP1 cannot be recovered. Currently, two generations are guaranteed. The journals collected after the most recent checkpoint dump are guaranteed.

The following figure shows the sequence in which checkpoint dump files are used.

Figure 4-8: Sequence of using checkpoint dump files



OpenTP1 collects checkpoint dumps for a server at the following times:

- When server startup or restart processing is completed
- When system journal files are swapped
- When the number of journals specified in the system journal service definition have been collected since the previous checkpoint dump collection
- When server termination preparations are completed

OpenTP1 manages a file group of checkpoint dump files in the online mode in the

statuses listed below.

- **Overwritable status or writing status**
A checkpoint dump file is overwritable when it does not contain a checkpoint dump needed for OpenTP1 full recovery or checkpoint dump collection is currently underway.
- **Non-overwritable status**
A checkpoint dump file is overwrite-protected when it contains a checkpoint dump needed for OpenTP1 full recovery.
- **Reserve**
A checkpoint dump file is on reserve status when it is closed and cannot be used in the online mode without first being opened.

When OpenTP1 is started, all file groups specified in the checkpoint dump service definition for which ONL has been specified are opened. The opened file groups are placed on overwritable status. File groups that cannot be opened or for which ONL is not specified are placed on reserve status.

By using the `jnladdpf` command, a new physical file can be allocated online to a checkpoint dump file for which a file group has been specified in the checkpoint dump service definition but a corresponding physical file has not (that is, with the definition command `jnladdfg` specified but `jnladdpf` not specified). The file thus allocated is in reserved state; it can be used online by opening it with the `jnlopenfg` command.

When checkpoint dump collection occurs, an output destination is selected from the overwritable file groups, and the selected file group is placed on writing status.

When checkpoint dump collection is completed, the file group on writing status is placed on non-overwritable status, and the oldest non-overwritable file group is placed on overwritable status.

When OpenTP1 is started, the guaranteed number of generations plus at least one is required for each file group (except reserve file groups). If use of the fall-back facility is specified in the fall-back operation option of the checkpoint dump service definition (`jnl_reduced_mode=1` or `2`), at least two more unreserved file groups than the guaranteed number of generations are needed. If there is no overwritable file group, OpenTP1 issues an error message to the message log file and terminates abnormally.

4.5.3 Deletion of checkpoint dump files

By using the `jnldehpf` command, physical files can be deleted during online processing from a checkpoint dump file group. The physical files that can be deleted using the `jnldehpf` command are those which are not open or are placed in the shutdown status that were allocated online with the `jnladdpf` command.

4.5.4 Automatic opening of checkpoint dump files

If additional checkpoint dump files are needed in the online mode, reserved file groups are automatically opened and placed on overwritable status. This is implemented by specifying `jnl_reserved_file_auto_open=Y` in the checkpoint dump service definition.

The automatic opening function is executed when the number of remaining checkpoint dump files in use decreases to the guaranteed number of generations. A file once opened (including those which could not be successfully opened) can no longer be opened automatically.

4.5.5 No checkpoint dump collection destination

If checkpoint dump collection is completed in the online mode and no file group is available as the next collection destination, OpenTP1 outputs an error message to the message log file. The user must open a reserve file group with the `jnlpnfg` command in order to place it on overwritable status.

If the next checkpoint dump collection occurs without a collection destination file group having been provided, OpenTP1 terminates abnormally.

4.5.6 Display of checkpoint dump file information

Checkpoint dump file information can be displayed by the `jnlis` command.

The displayed information includes the server names, file group names, corresponding journal file information, etc.

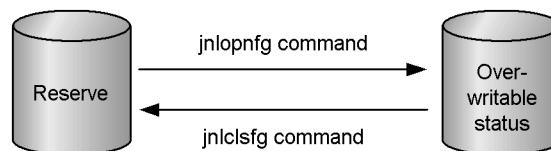
4.5.7 Checkpoint dump file open and close

When a reserve file group is opened by the `jnlpnfg` command, it is placed on overwritable status.

When a file group on overwritable status is closed by the `jnlclsfg` command, it is placed on reserve status.

The following figure shows how the open and close commands change the status of a checkpoint dump file.

Figure 4-9: Status changes by open and close commands (checkpoint dump file)



4.5.8 Duplication of checkpoint dump files

In a duplicated checkpoint dump file configuration, the same checkpoint dump

information is collected in both files for system A and system B. If recovery from the file for system A fails at OpenTP1 recovery, the file for system B can be used to recover OpenTP1; this eliminates the need for the user to return to old generations of checkpoint dump files.

When a checkpoint dump file is duplicated, one-system operation can be enabled or disabled.

When one-system operation is available in a dual system configuration, the file group can be used when at least one of the two physical files is open. Therefore, if an error occurs on either of systems A and B while collecting checkpoint dump, the other system continues the dump collection. When the dump collection is completed, the file group becomes valid.

When one-system operation is unavailable in dual system configuration, the file group can be used only when both of the physical files are open. If either of the physical files cannot be opened, the file group is placed in reserved status and cannot be used. However, if an error occurs while reading from system A, the read is continued using system B. After the read is completed, the file group is reserved.

The following table shows the difference in the file the group status at error between when one-system operation is available and when one-system operation is unavailable.

Table 4-16: Difference in file group status at error between when one-system operation is available and when unavailable

Error type		One-system operation available	One-system operation unavailable
Open error at full recovery	The error occurred on either of the systems.	The other system is opened and the processing is continued.	<ul style="list-style-type: none"> When the other system is overwritable, it is opened (when it becomes non-overwritable, the file group is placed in the reserved status). When the other system is non-overwritable, the file group is placed in the reserved status.
	The error occurred on both of the systems.	The file group is placed in reserved status.	The file group is placed in reserved status.
An error occurred while collecting (writing) checkpoint dump.	The error occurred on either of the systems.	The other system continues the dump collection.	The file group is placed in reserved status and the dump is collected in another file group.

Error type		One-system operation available	One-system operation unavailable
	The error occurred on both of the systems.	The file group is placed in reserved status and the dump is collected in another file group.	This error does not occur when one-system operation is disabled. Take an action when either of the systems failed.
An error occurred while reading checkpoint dump.	The error occurred on system A.	The checkpoint dump is read from system B.	After the read from system B is completed, the file group is placed in reserved status. If system B is non-overwritable, the file group is placed in reserved status when it becomes overwritable.
	The error occurred on system B.	The file group is placed in reserved status and the checkpoint dump is read from the previous generation.	The file group is placed in reserved status and the checkpoint dump is read from the previous generation.

Between when one-system operation is available and when unavailable, there is a difference in online allocation, disconnection, open, and close of only one system.

The following table shows the difference in online operation for only one system.

Table 4-17: Difference in online operation for only one system

Operation	One-system operation available	One-system operation unavailable
Allocating only one system in online mode	Possible	Possible
Disconnecting only one system from online processing	Possible. However, the file group cannot be used when disconnected.	Possible. However, the file group cannot be used when disconnected.
Opening only one system	Possible when both systems are allocated.	Impossible
Closing only one system	<ul style="list-style-type: none"> • Possible when overwritable. • Impossible when non-overwritable. 	Impossible

4.5.9 Status changes for checkpoint dump files

Table 4-18 shows the status changes for checkpoint dump files in the online mode.

Table 4-19 shows the status changes for checkpoint dump files in the online mode when one-system operation is available. Table 4-20 shows the status changes for checkpoint dump files in online mode when one-system operation is unavailable.

Table 4-18: Status changes for checkpoint dump files in online mode

Status		Overwritable	Writing	Non-overwritable	Reserve
Open/close		Open			Close
Status number		1	2	3	4
Event	Checkpoint dump collection timing	→ 2	--	--	--
	Validation of checkpoint dump	--	→ 3	→ 1 [#]	--
	Output error	--	→ 4	--	--
	Execution of jnlclsfg	→ 4	--	--	--
	Execution of jnlopnfg	--	--	--	→ 1

Legend:

--: Impossible or inapplicable status change

→ *n*: Target status number

Note

This table assumes that all events are processed normally.

#

This is the status change for the oldest non-overwritable file.

Table 4-19: Status changes for checkpoint dump files in online mode when one-system operation is available

Status		Overwritable			Writing			Non-overwritable			Rs
Open/close	System A	O	O	CI	O	O	CI	O	O	CI	CI
	System B		CI	O		CI	O		CI	O	
Status number		1	2	3	4	5	5	7	8	9	10
Event	Checkpoint dump collection timing	→ 4	→ 5	→ 6	--	--	--	--	--	--	--
	Validation of checkpoint dump	--	--	--	→ 7	→ 8	→ 9	→ 1 [#]	→ 2 [#]	→ 3 [#]	--

Status		Overwritable			Writing			Non-overwritable			Rs
Open/close	System A	O	O	Cl	O	O	Cl	O	O	Cl	Cl
	System B		Cl	O		Cl	O		Cl	O	
Status number		1	2	3	4	5	5	7	8	9	10
	Output error on system A	--	--	--	→ 6	→ 10	--	--	--	--	--
	Output error on system B	--	--	--	→ 5	-	→ 10	--	--	--	--
Execution of jnlclsfg	with -ab option	→ 10	→ 10	→ 10	--	--	--	--	--	--	--
	with -a option	→ 3	→ 10	--	--	--	--	--	--	--	--
	with -b option	→ 2	--	→ 10	--	--	--	--	--	--	--
Execution of jnlopnfg	with -ab option	--	→ 1	→ 1	--	--	--	--	→ 7	→ 7	→ 1
	with -a option	--	--	→ 1	--	--	--	--	--	→ 7	→ 2
	with -b option	--	→ 1	--	--	--	--	--	→ 7	--	→ 3

Legend:

O: Open

Cl: Close

Rs: Reserve

--: Impossible or inapplicable status change

→ *n*: Target status number*Note*

This table assumes that all events are processed normally.

#

This is the status change for the oldest non-overwritable file.

Table 4-20: Status changes for checkpoint dump files in online mode when one-system operation is unavailable

Status		Overwritable	Writing	Non-overwritable	Reserve	
Open/close	System A	Open			Close	
	System B	Open			Close	
Status number		1	2	3	4	
Event	Checkpoint dump collection timing		→ 2	--	--	--
	Validation of checkpoint dump		--	→ 3	→ 1 [#]	--
	Output error on system A		--	→ 4	--	--
	Output error on system B		--	→ 4	--	--
	Execution of jnlclsfg	with -ab option	→ 4	--	--	--
		with -a option	--	--	--	--
		with -b option	--	--	--	--
	Execution of jnlpnfg	with -ab option	--	--	--	→ 1
		with -a option	--	--	--	--
		with -b option	--	--	--	--

Legend:

--: Impossible or inapplicable status change

→ *n*: Target status number

Note

This table assumes that all events are processed normally.

#

This is the status change for the oldest non-overwritable file.

4.6 DAM file operations

This section describes how to use DAM files. For details about estimating the size of a DAM file, see *H.6 Estimating the size of a DAM file*.

4.6.1 Creation of DAM files

Once an OpenTP1 file system is created, the user can create DAM files with the `damload` command; the access right to the OpenTP1 file system must be set to user. The name of a physical file must be the same as that in the DAM service definition. When the `damload` command is executed, a physical file is allocated, and initial data is output to the physical file. In the online mode, access to the DAM file is performed on the corresponding logical file, not the physical file. The relationship between logical and physical files is specified in the DAM service definition. The physical file access right is not passed on to the logical file.

For details of DAM service definition, see the manual *OpenTP1 System Definition*.

A DAM file can also be created from a UAP; for details, see the manual *OpenTP1 Programming Guide*.

4.6.2 DAM file status management

A DAM file in the online mode is managed in the following three statuses:

- In use
The DAM file is being accessed.
- Logical shutdown
The logical file is placed on logical shutdown status by the `damhold` command.
- Error shutdown
The DAM file is placed on shutdown status because of a disk error, etc.

4.6.3 Display of DAM file status

The status of DAM files can be displayed by the `fills` and `daml`s commands.

(1) Status display by `fills` command

When the `fills` command with a physical file name specified is executed, the status of that physical file is displayed.

The displayed information includes owner name, record length, last update date and time, etc. Although access rights are also displayed, the access rights for physical files are invalid. You can use a UAP to access and write a file for which you are authorized to read but not to write.

(2) Status display by damls command

When the damls command is executed, the status of logical files is displayed.

The displayed information includes the logical file names, file shutdown statuses, block lengths, etc.

4.6.4 DAM file addition and deletion

The damadd command adds a logical file to an online system. This addition processing defines an existing physical file as a logical file. The user must create a physical file before executing the damadd command.

The deferred update facility can be used by specifying the -d option in the damadd command. Specifying this option makes it possible to add specified logical files as files subject to deferred updating.

The -n option of the damadd command enables a DAM file to be additionally registered as a file not to be recovered. If you wish to specify cacheless access for the file, also specify the -f option.

The damrm command removes a logical file from the online system, and the damdel command deletes a physical file. To delete a physical file being used in the online mode, the user must remove the corresponding logical file from the online system with the damrm command before executing the damdel command.

A DAM file used in the online mode (opened) cannot be removed from the online system by the damrm command. To remove the DAM file from the online system, the appropriate logical file must be closed in all services under which it is opened.

4.6.5 Logical file logical shutdown and release of shutdown status

To prohibit access to a DAM file cataloged into an online system, its logical file must be placed on logical shutdown status by the damhold command.

To perform maintenance on a DAM file during online operation, the DAM file must be removed from the online system. The damhold and damrm commands must be entered in this order to remove a DAM file from the online system.

The damrles command makes a DAM file that has been placed on logical shutdown status by the damhold command accessible again in the online mode. The damrles command also makes accessible DAM file that has been restored to the online system after error recovery.

The DAM service disables access to a specified file when it accepts the damhold command. An attempt to logically shut down a file subject to deferred updating will fail as it is placed on the logical shutdown waiting status until the blocks in the buffer area corresponding to that file are output onto the disk. Therefore, no logical files can be deleted until the blocks in the buffer area are output onto the disk. In this case, executing the damrm command results in a command error.

4.6.6 DAM file backup and restore

(1) *DAM file backup*

The `dambkup` command backs up a DAM file. A DAM file can be backed up in the online or offline mode. This is referred to as online backup.

The offline-mode backup procedure is as follows:

1. Execute the `damhold` command to logically shut down a logical file.
2. Execute the `damrm` command to remove the logically shut-down logical file from the online system.
3. Execute the `dambkup` command with its `-o` option not specified to back up the DAM file.

Execute the `dambkup` command with the `-o` option to perform online backup. Online backup does not require a DAM file to be shut down logically (by the `damhold` command) or removed from the online system (by the `damrm` command).

Recovering the DAM file by using an online backed-up file reduces the number of unload journal files that are used at recovery. This means a shorter time required for recovery of the DAM file than recovery by using a file backed up with the `-o` option not specified.

DAM files must always be backed up before an application is started.

(2) *Output destination of backup*

A file or standard output can be specified for the output destination of backup. To output to the file, specify the file name in the `dambkup` command. To output to the standard output, specify the `-s` option in the `dambkup` command.

(3) *DAM file restore*

To restore a backup file, the `damrstr` command is used.

A file or a standard input can be specified for the restoration input source. To input backup data from the file, specify the file name in the `damrstr` command. To input backup data from the standard input, specify the `-s` option in the `damrstr` command.

4.6.7 DAM file recovery

DAM file recovery is referred to as DAM FRC, and is executed by the `damfrc` command. When the `damfrc` command executes, logical files specified in the DAM service definition are recovered to their status immediately before an error occurred, using backup DAM files and specified unload journal files or using integrating journal files. Note that the `damfrc` command can only be executed for files to be recovered. Executing the command for a file not to be recovered has no effect.

When the `damfrc` command is executed, OpenTP1 creates inheritance files to be used

with DAM FRC. An inheritance file is created under the file name `jnlcolc***` (***: key) or `jnlcolc***.bak` (temporary backup file) in the current directory each time unload journal processing is completed. Inheritance files are used when DAM FRC is executed in multiple steps. The journal information that needs to be inherited is collected in the inheritance files. If the `-e` option is specified for DAM FRC, OpenTP1 deletes the inheritance files upon completion of DAM FRC.

For the DAM FRC execution procedure, see *10.2.6 DAM file*.

(1) DAM FRC execution in one step

For DAM FRC execution to be completed in a single step, the `-s` and `-e` options must both be specified in the `damfrc` command. Specifying the `-s` option does not inherit the previous DAM FRC.

(2) DAM FRC execution in multiple steps

If there are multiple unload journal files, DAM FRC can be executed in multiple steps. The `-s` option must be specified in the first `damfrc` command that is executed, and the `-e` option must be specified in the last `damfrc` command that is executed.

(3) Simultaneous execution of multiple DAM FRC steps

If multiple DAM FRC steps are to be executed simultaneously, the inheritance files must be created individually. In this case, the `-k` option must be specified in the `damfrc` command. If a different key is specified with the `-k` option, the corresponding inheritance file is created. The user must specify the `-e` option in the last `damfrc` command that is executed.

(4) DAM FRC execution using an intermediate generation of system journal files

If a DAM file is backed up in the online mode, DAM FRC can be executed using the subsequent generation of the system journal files. In this case, the `-g` option must be specified in the `damfrc` command.

(5) DAM FRC execution using an integrating journal file

Executing DAM FRC using an integrating journal file already created by the `jnlcolc` command reduces the DAM FRC processing time. In this case, the `-c` option must be specified in the `damfrc` command.

4.6.8 DAM file lock

Block lock is performed using the following file name as the resource name:

file-name + hyphen (-) + (relative-block-number + 1 (hex.))

File lock is performed using the following file name as the resource name:

logical-file-name (8 chars.) + space (Δ)

If the logical file name has 1 to 7 characters, the DAM service inserts spaces to make

it an eight-character name. Therefore, for file lock, the resource name always has nine characters.

The following shows resource name examples for block lock and file lock.

1. When block lock is performed on relative block number 30 of logical file `dam1`, the name of the locked resource is `dam1-1f`.
2. When file lock is performed on logical file `dam1`, the name of the locked resource is `dam1 ΔΔΔΔΔ`.

4.6.9 DAM file addition in online mode

Following is the procedure for adding a DAM file in the online mode:

1. Create a physical file with the `damload` command
2. Use the `damadd` command to establish the correspondence between a logical file and the created physical file (a logical file that has not been defined in the DAM service definition must be specified)

4.6.10 Setting the number of cache blocks for a DAM file

The DAM service stores the block data of the DAM file it reads in the DAM service-exclusive shared memory. When there are multiple reference requests for the same block, the DAM service uses the data in the memory to reduce the number of file I/O operations. In the shared memory, several pieces of block data are chain managed for each DAM file. An area in the DAM service-exclusive shared memory storing block data is called a cache block.

When you use the `damchdef` command, you can set the maximum number of cache blocks a DAM file can manage (threshold). By setting a threshold, you can effectively use memory resources since cache blocks are reused.

Determine a threshold for the number of cache blocks as follows.

To determine a threshold:

1. Execute the `damchinf` command to acquire the cache block information.

An example output of executing the `damchinf` command is as follows:

FileNo	FileName	BlkLen	BlkNum	CchBlkNum	PreservNum	LimitNum	ReUse
1	damfile1	504	10000	7900	0	-1	Exist
0	damfile0	504	10000	100	0	-1	Exist
2	damfile2	504	10000	0	0	-1	None

2. Choose a threshold value based on the type of UAP access.

- If the UAP will access `damfile1` very seldom

Only 20% of the cache block area can be allocated to `damfile0` and

`damfile2`. Since the UAP will hardly access `damfile1`, the cache block area for the 7,900 cache blocks that are connected to the cache block chain for `damfile1` will remain secured. Specify a small value as the threshold for `damfile1` to decrease the secured cache block area and increase the cache block area to be allocated to `damfile0` and `damfile2`.

Since many cache blocks are connected to the cache block chain for `damfile1`, clean-up processing will be performed if the cache block area becomes insufficient. Since the output result indicates that the clean-up target is 1, that means that the cache block chain for `damfile1` is the target of clean-up. In this case, release processing will start for these 7,900 cache blocks and system performance will degrade significantly. To prevent this degradation, you should specify a threshold for `damfile1`.

- When the UAP accesses `damfile1` frequently

When the UAP accesses `damfile1` frequently, the cache block chain for `damfile1` is frequently searched. By specifying a threshold, you can shorten the cache block chain, reducing the chain search time. However, if the threshold is too small, the cache efficiency decreases, degrading performance. Determine an optimum value by testing several thresholds.

- When the UAP accesses all DAM files uniformly

When the UAP accesses all DAM files uniformly, the cache block areas allocated to each DAM file should be the same. By setting the same value as the threshold for each DAM file, the cache block area used by each DAM file will be the same.

4.6.11 Extending the block length for a DAM file

To extend the block length for a DAM file, after first backing up the file using the `dambkup` command, execute the `damrstr` command, specifying the block length of the physical file in which to restore the backup data. There are two ways of extending the block length: one method keeps the block structure and the other does not. With the method that keeps the block structure, the restore-destination DAM file keeps the same block structure as the backup-source DAM file. With the method that does not keep the block structure, the data in the backup-source DAM file is written to the restore-destination DAM file from the first block in the file. When you wish to keep the block structure, specify the `damrstr` command with the `-e` option, specifying the extended block length. When you do not wish to keep the block structure, specify the `damrstr` command with the `-p` option, specifying the extended block length. You cannot extend the block length of a file that is backed up in online mode. Also, you cannot specify a DAM file with an extended block length as the recovery definition file in the `damfrc` command arguments.

4.6.12 Extracting user data from a DAM file

To extract only user data from a DAM file while ignoring management information in the file, execute the `dambkup` command with the `-d` option. Note that the extracted user data cannot be restored by the `damrstr` command.

4.7 TAM file operations

This section describes how to use TAM files. For details about estimating the size of a TAM file, see *H.7 Estimating the size of a TAM file*.

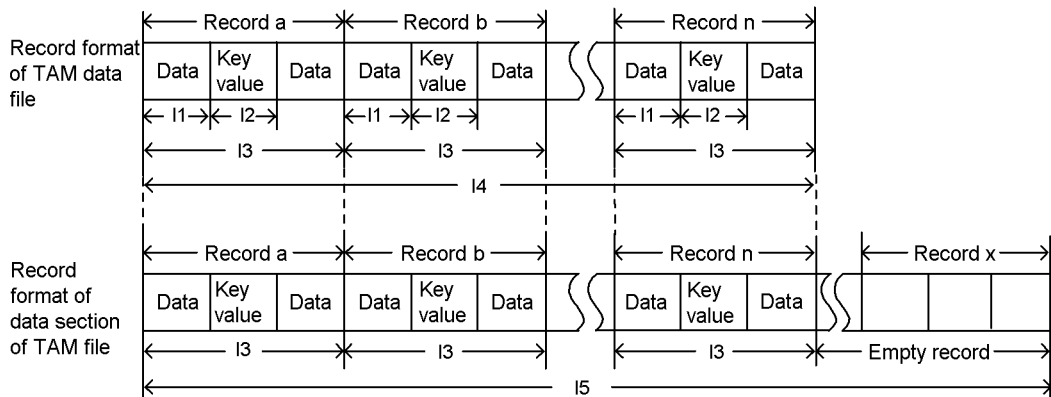
4.7.1 Creation of TAM files

Once an OpenTP1 file system is created, the user can create TAM files with the `tamcre` command; the access right to the OpenTP1 file system must be set to user. The name of a physical file must be the same as that in the TAM service definition. The TAM service definition is used to specify the correspondence between the created TAM files and a TAM table.

For details of the TAM service definition, see the manual *OpenTP1 System Definition*.

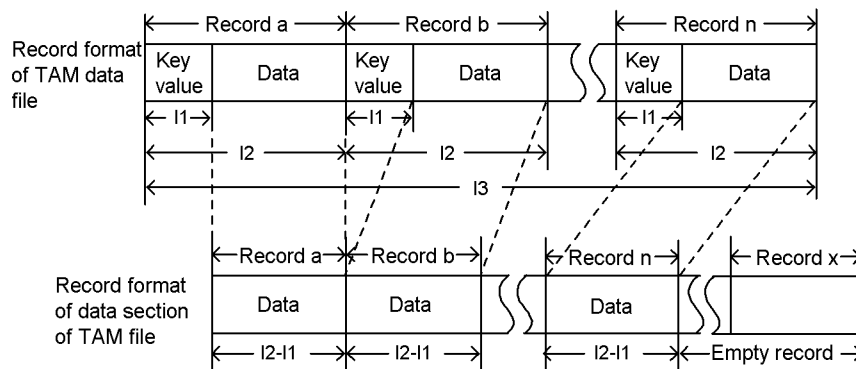
In executing the `tamcre` command, initial data can be stored in the TAM file simultaneously with TAM file creation. To do this, the user must create a **TAM data file** (a file used to store the initial data for the TAM file) and must store the initial data in this file before executing the `tamcre` command. The record format for the data section of the TAM file depends on whether or not the `-s` option is specified in the `tamcre` command. Figure 4-10 shows the relationship between the TAM data file format and the record format for the data section of a TAM file when the `-s` option is not specified, and Figure 4-11 shows the relationship when the `-s` option is specified.

Figure 4-10: Relationship between TAM data file format and record format of data section of TAM file (-s option not specified)



- I1: Key begin position specified in `tamcre` command (length from beginning of record to key begin position)
- I2: Area length specified in `tamcre` command
- I3: Record length specified in `tamcre` command
- I4: Record length specified in `tamcre` command x number of records specified for initial data
- I5: Record length specified in `tamcre` command x maximum number of records

Figure 4-11: Relationship between TAM data file format and record format of data section of TAM file (-s option specified)



Note

When the `-s` option is specified, the key begin position (length from beginning of record to key begin position) is always zero.

I1: Key area length specified in `tamcre` command

I2: Record length specified in `tamcre` command

I3: Record length specified in `tamcre` command x number of records specified for initial data

4.7.2 TAM table status management

A TAM table in the online mode is managed in the following three statuses:

- Logical shutdown

The TAM table has been placed on logical shutdown status by the `tamhold` command.

- Error shutdown

The TAM table has been placed on shutdown status because of a disk error, etc.

- Non-shutdown

The TAM table can be accessed in the online mode.

4.7.3 Display of TAM table status

The status of TAM tables can be displayed with the `tamls` command.

The displayed information includes the TAM table names, numbers of records in use, etc.

4.7.4 TAM table addition and removal

The `tamadd` command adds a TAM table to an online system. The loading time and

access mode of the added TAM table depend on the options specified in the `tamadd` command. Once a TAM table has been added, it can be accessed in the online mode by releasing the TAM table from shutdown status with the `tamrles` command.

When the `-i` option of the `tamadd` command is specified, a TAM table serves as an I/O failure continuation table. If an I/O error occurs during updating of a TAM file, this file will not be placed in the error shutdown status. This means that the file can be accessed by any UAP operating in the same online mode. In order to prevent inconsistent TAM file status at restart of online operation, the TAM table with an error having occurred in the previous online operation (an I/O failure continuation table) will be removed from the online system. Once online operation has restarted, the TAM file must be recovered and added to the online system again.

When updating only a part of a long record, the journal amount can be reduced by collecting the journal of only the updated part. This is specified by the `-j` option in the `tamadd` command when adding a TAM table.

To remove a TAM table from the online system, the `tamrm` command is used.

A TAM table once removed from the online system can be added under the same TAM table name if the file attributes (record length, key length, etc.) are the same.

4.7.5 TAM table logical shutdown and release of shutdown status

To prohibit access to a TAM table cataloged into the online system, the TAM table must be placed on logical shutdown status with the `tamhold` command.

The `tamrles` command makes a TAM table placed on logical shutdown status accessible in the online mode. The `damrles` command also makes accessible a TAM table that has been added to the online system by the `tamadd` command after error recovery.

4.7.6 TAM table loading and unloading

The time at which a TAM table is loaded into memory is determined by the loading time specification in the TAM service definition:

start specified:

Loaded when TAM service is started.

cmd specified:

Loaded when `tamload` command is executed.

lib specified:

Loaded when TAM table is opened.

A TAM table whose loading time is `cmd` can be unloaded by the `tamunload` command.

4.7.7 TAM file backup and restoration

(1) TAM file backup

The `tambkup` command backs up a TAM file.

A TAM file can be backed up in the online or offline mode. This is referred to as online backup.

The offline mode backup procedure is as follows:

1. Execute the `tamhold` command to logically shut down a TAM table.
2. Execute the `tamrm` command to remove the logically shutdown TAM table from the online system.
3. Execute the `tambkup` command with its `-o` option not specified to back up the TAM file.

Execute the `tambkup` command with the `-o` option to perform online backup. Online backup does not require a TAM file to be shut down logically (by the `tamhold` command) or removed from the online system (by the `tamrm` command).

Recovering the TAM file by using an online backed-up file reduces the number of unload journal files that are used at recovery. This means a shorter time required for recovery of the TAM file than recovery by using a file backed up with the `-o` option not specified.

A TAM file must always be backed up before an application is started.

(2) Output destination of backup

A file or standard output can be specified for the output destination of backup. To output to the file, specify the file name in the `tambkup` command. To output to the standard output, specify the `-s` option in the `tambkup` command.

(3) TAM file restore

To restore a backup file, the `tamrstr` command is used.

A file or standard input can be specified for the restoration input source. To input backup data from the file, specify the file name in the `tamrstr` command. To input backup data from the standard input, specify the `-s` option in the `tamrstr` command.

4.7.8 Creation of a TAM data file from a TAM file

After an online operation or removal of a TAM table from the online system, a TAM data file can be created with the `tambkup` command from a TAM file that has been accessed in online mode. Executing the `tambkup` command with its `-d` option specified creates a TAM data file from valid records in a TAM file. The TAM data file thus created contains data sorted in ascending order on the basis of a key value. The user can update the contents of the TAM data file in offline mode. Executing the

`tamcre` command after updating the contents of the TAM data file creates a TAM file that includes the updates made in offline mode.

4.7.9 TAM file deletion

The `tamdel` command deletes a TAM file. To delete a TAM file being used in the online mode, the corresponding TAM table must be removed from the online system with the `tamrm` command and then the `tamdel` command must be executed.

4.7.10 TAM file recovery

TAM file recovery is referred to as TAM FRC, and is executed by the `tamfrc` command. When the `tamfrc` command executes, the TAM file is recovered to its status immediately before the error occurred, either using the backup TAM file, and specified unload journal files or using integrating journal files.

When the `tamfrc` command is executed, OpenTP1 creates inheritance files to be used with TAM FRC. An inheritance file is used when TAM FRC is executed in multiple steps. The journal information that needs to be inherited is collected in the inheritance files. If the `-e` option is specified for TAM FRC, OpenTP1 deletes inheritance files upon completion of TAM FRC.

When the `-f` option is specified in the `tamfrc` command, more than one TAM file can be recovered. Create the recovery target definition file using a text editor and specify in it the TAM files to be recovered. Execute the `tamfrc` command, specifying the `-f` option and the recovery target definition file.

For the TAM FRC execution procedure, see *10.2.7 TAM file*.

(1) TAM FRC execution in one step

For TAM FRC execution to be completed in a single step, the `-s` and `-e` options must both be specified in the `tamfrc` command. Specifying the `-s` option does not inherit the previous TAM FRC.

(2) TAM FRC execution in multiple steps

If there are multiple unload journal files, TAM FRC can be executed in multiple steps. The `-s` option must be specified in the first `tamfrc` command that is executed, and the `-e` option must be specified in the last `tamfrc` command that is executed.

(3) TAM FRC execution using an integrating journal file

Executing TAM FRC using the integrating journal file already created by the `jnlcolc` command reduces the TAM FRC processing time. In this case, the `-j` option must be specified in the `tamfrc` command.

4.7.11 TAM file lock

Record lock is performed by using a record identifier ('R') + TAM table number (5 decimal digits) + record number (10 decimal digits) as the resource name.

Table lock is performed by using a table identifier ('T') + TAM table number (5 decimal digits) as the resource name.

The following shows the format of the record resource name and table resource name.

Record resource name: *Raaaaabbbbbbbbbb* (16 characters)
 Table resource name: *Taaaaa* (six characters)

R

Record identifier

T

Table identifier

aaaaa

Table number (5 digits)

bb...bb

Record number (10 digits)

Specify the resource name for the `tamlockls` command to check the contents of the deadlock information file, timeout information file, or TAM lock resources displayed by the `lockls` command.

4.7.12 TAM file addition in online mode

Following is the procedure for adding a TAM file in online mode:

1. Execute the `tamcre` command to create a TAM file.
2. Execute the `tamadd` command to associate the TAM file created in step 1. with a TAM table. In this case, the user can select a TAM table which has not been specified in the TAM service definition, or a TAM table which has been specified in the TAM service definition but which has been removed from the online system with the `tamrm` command.

4.7.13 Displaying the synonym information after creating a TAM file

To see how many synonyms are generated for a hash-format TAM file that is created using the `tamcre` command, use the `tamhsls` command.

4.7.14 Increasing the number of records in a TAM file

If you want to increase the number of records in a TAM file, first use the `-d` option in the `tambkup` command to back up only user data and then use the `tamcre` command to create a new TAM file with the increased number of records. Next, allocate the data you have backed up using the `-d` option in the `tamcre` command to the TAM file.

Note that if you use the `tamrstr` command to restore the new TAM file with more

4. OpenTP1 File Operations

records you created using the `tamcre` command, the number of records will not be increased, since the control information for the backup file will be overwritten.

Chapter

5. Sending and Receiving Messages

This chapter explains sending and receiving messages. For details about correspondences between the connections explained in this chapter and the protocols used by the user, see the documentation for each protocol.

The following program products are required in order to use the OpenTP1 message control facility:

- TP1/Message Control (for managing the message control facility)
- TP1/NET/Library (for controlling the network)
- TP1/NET/*protocol-name* (for controlling the interface with the communication protocol)

5.1 MCF communication service operations

5.2 Connection operations

5.3 Application operations

5.4 Logical terminal operations

5.5 Service group operations

5.6 Service operations

5.7 Protocol-specific operations

5.8 Monitoring message queues

5.9 Queue operations

5.1 MCF communication service operations

This section explains operations related to the MCF communication service.

(1) Display of MCF communication service status

The `mcfctlscm` command displays the status of the MCF communication service. The displayed items include the MCF communication server name, the MCF communication server process ID, and the status of the MCF communication service.

You can also acquire the status of the MCF communication service by issuing a function from a UAP. For details, see the manual *OpenTPI Programming Guide*.

5.2 Connection operations

This section explains operations related to connections.

(1) *Display of connection status*

A connection status can be displayed by the `mcfctlscn` command. If the `-d` option is specified in the `mcfctlscn` command, information about the logical terminal corresponding to a connection is also displayed. The displayed information includes the connection ID, type of protocol, connection status, etc.

You can also acquire the connection status by issuing a function from a UAP. For details, see the manual *OpenTPI Programming Guide*.

(2) *Connection establishment and release*

If a connection error occurs or the connection is shut down by entry of the `mcftdctcn` command, the `mcfactcn` command can be used to reestablish the connection.

The `mcftdctcn` releases a connection established by the `mcfactcn` command. If connection cannot be released normally, specifying the `-f` option in the `mcftdctcn` command releases the connection forcibly. When a connection is released forcibly, a message being received is discarded and is not cataloged into the input queue. During message send, send processing is cancelled, and the message being sent is retained in the output queue.

You can also establish and release a connection by issuing a function from a UAP. For details, see the manual *OpenTPI Programming Guide*.

(3) *Exchange of connections*

If the remote system terminates abnormally or a line error occurs, the connection can be exchanged by the `mcfatchcn` command to continue message exchange. However, if an error occurs with the connection of the destination, messages cannot be sent or received by executing the `mcfatchcn` command.

The appropriate logical terminal must be shut down by the `mcftdctle` command. Then the connections of both the source and destination must be released by the `mcftdctcn` command. This must be followed by execution of the `mcfatchcn` command, which exchanges the correspondence between the connection and the logical terminal and enables messages to be sent or received at the connection of the destination.

The `mcfatchcn` command must be executed without remaining messages to be sent. If the command is executed with messages remaining in the output queue, they are all sent to the connection of the destination. If the command is executed after an inquiry-response type UAP have received an inquiry message and before it sends a response message, the response message is sent to the connection of the destination.

To update the status of the new connection, the `mcfactcn` command must be executed to establish the connection and then the `mftlscn` command executed to display the connection status for updating.

Before executing the `mftchcn` command, TP1/NET/High Availability must be installed.

(4) Start and stop of acceptance of server-type connection establishment requests

The `mftonln` command starts acceptance of server-type connection establishment requests, and the `mftofln` command stops acceptance of server-type connection establishment requests. You can also use the `mftlsln` command to display the status of establishment request acceptance.

For details, see the applicable *OpenTP1 Protocol* manual.

5.3 Application operations

This section explains operations related to applications.

(1) Display of application status

The status of applications can be displayed by the `mcfalsap` command.

The displayed information includes the application types, application names, application statuses, etc.

(2) Application shutdown and release of shutdown status

Applications can be shut down by the `mcfadctap` command. When the `mcfadctap` command executes, input to the input queue and input queue scheduling are shut down on the basis of specified options. If application attributes definition is specified in such a manner that input to an application input queue and its scheduling are to be shut down (`aplihold=a`) or if shutdown is specified in the schedule of the application (`aplihold=s`) in the event of abnormal termination of the application, the application is shut down whenever it terminates abnormally and the specified application abnormal terminations count has been reached.

When input to an input queue is shut down, all messages already in the input queue are processed normally. If a message is received during shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated.

If a message is received while input to an input queue and input queue scheduling are involved in shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated.

Applications can be released from shutdown status with the `mcfactap` command. When the `mcfactap` command is executed, each application's abnormal terminations count is initialized to 0.

The status changed by the `mcfadctap` or `mcfactap` command (input to input queue and input queue scheduling status) can be inherited during full recovery. If a maximum number of applications is specified in the status inheritance definition of the MCF communication configuration definition, the status of resources is inherited only up to the specified value.

(3) Initialization of application abnormal terminations count

An application's abnormal terminations count is incremented each time the application terminates abnormally. When this count reaches the maximum abnormal terminations count value specified in the application definition, the application or service is shut down automatically. An application's abnormal terminations count can be initialized (to 0) by executing the `mcfac1cap` command. An application's abnormal terminations count is also initialized when the application is released from shutdown status by the

`mcfactap` command.

(4) Display of application timer startup requests

The `mcfalstap` command displays application timer startup requests. The displayed items include the application name and the time of acceptance of the application startup request.

(5) Deletion of application timer startup requests

The `mcfadltap` command terminates startup of applications for which a timer startup request has been issued. When the `mcfadltap` command executes, the timer startup requests for the specified applications are deleted, and startup of the applications is terminated.

You can also delete an application timer startup request by issuing a function from a UAP. For details, see the manual *OpenTPI Programming Guide*.

(6) Startup of application programs

An application program can be directly started up with an operation command. Executing the `mcfuevt` command starts up the service that corresponds to the application name (UCMDEV) specified in the MCF application attribute definition (`mfaalcap`).

5.4 Logical terminal operations

This section explains operations related to logical terminals.

(1) Display of logical terminal status

The status of logical terminals can be displayed with the `mcftlslsle` command.

The displayed information includes the MCF identifiers, logical terminal names, statuses of the logical terminals, etc.

If you are using TP1/NET/OSI-TP, logical terminals have no status.

You can also acquire the status of logical terminals by issuing a function from a UAP. For details, see the manual *OpenTP1 Programming Guide*.

(2) Logical terminal shutdown and release of shutdown status

You can use the `mcftdctlle` command to shut down a logical terminal. The branch message send requests present during shutdown are retained in the output queue.

You can use the `mcftactlle` command to release a logical terminal from shutdown status. When the shutdown status is released, the messages remaining in the output queue are sent.

If the `mcftdctlle` command for shutting down a logical terminal is executed while a message is being received or sent, the operation of OpenTP1 depends on the protocol product you are using. The following table describes the operation of OpenTP1 for each protocol product:

Table 5-1: Operation when the `mcftdctlle` command is entered while a message is being received or sent

Status	Operation of OpenTP1	Protocol product being used
<code>mcftdctlle</code> is entered while a message is being received	OpenTP1 discards the message being received. Messages that are received later are not registered in the input queue.	<ul style="list-style-type: none"> • TP1/NET/C/S560 • TP1/NET/HNA-NIF • TP1/NET/HSC (for HSC2 procedure) • TP1/NET/OSAS-NIF • TP1/NET/User Agent • TP1/NET/User Datagram Protocol • TP1/NET/XMAP3

Status	Operation of OpenTP1	Protocol product being used
	OpenTP1 receives the message even though the logical terminal is shut down. The message reception processing is not affected by the shut-down logical terminal.	<ul style="list-style-type: none"> • TP1/NET/HDLC • TP1/NET/HSC (for HSC1 procedure) • TP1/NET/NCSB • TP1/NET/Secondary Logical Unit - TypeP2 • TP1/NET/TCP/IP • TP1/NET/X25
mcftdctle is entered while a message is being sent	OpenTP1 returns an error for the mcftdctle command. The logical terminal is shut down unless a branch message is being sent.	<ul style="list-style-type: none"> • TP1/NET/NCSB • TP1/NET/Secondary Logical Unit - TypeP2 • TP1/NET/X25
	OpenTP1 cancels the transmission of a branch message. Messages sent from UAPs are stored in the output queue.	<ul style="list-style-type: none"> • TP1/NET/C/S560 • TP1/NET/HDLC • TP1/NET/HSC • TP1/NET/User Agent • TP1/NET/XMAP3
	OpenTP1 cancels the transmission of a branch message. Messages sent from UAPs are discarded.	<ul style="list-style-type: none"> • TP1/NET/HNA-NIF
	OpenTP1 cancels the transmission of a branch message. Messages sent from UAPs are stored in the output queue if the disk queue is used as the output queue. If the memory queue is used as the output queue, the messages from UAPs are discarded.	<ul style="list-style-type: none"> • TP1/NET/OSAS-NIF
	OpenTP1 does not cancel the transmission of a branch message. The logical terminal is shut down after sending of the message being sent has been completed.	<ul style="list-style-type: none"> • TP1/NET/TCP/IP • TP1/NET/User Datagram Protocol

You can also shut down a logical terminal or release a terminal from shutdown status by issuing a function from a UAP. For details, see the manual *OpenTP1 Programming Guide*.

(3) Discarding of logical terminal message queue's initial message

The mcfmspql command discards a message that was erroneously requested for send or that failed to be sent. The first message queued can be discarded by executing the mcfmspql command. In this case, however, the logical terminal must be placed on shutdown status with the mcfmdctle command before the mcfmspql command is executed.

If the TP1/NET/HNA-560/20 protocol is used, the mcfmdctss command must be executed to terminate the session. In addition, the logical terminal must not be a logical

terminal of alternate origin.

The logical terminal status, number of unsent messages, etc., can be checked by displaying the logical terminal status with the `mcfctlslsle` command prior to execution of the `mcfctlspqle` command.

(4) Copying of contents of logical terminal output queues

Where a terminal error occurs, you can use the `mcftdmpqu` command to check the contents of the disk messages remaining in the output queue. Executing this command lets the contents of the output queue copied onto the specified file. If then the `-a` option is specified, the contents of the output queue are again written on the destination file for which the `mcftdmpqu` command was previously executed.

If the `mcftdmpqu` command is to be independently used, the `mcftdctle` command must be executed to shut down the logical terminal before executing the `mcftdmpqu` command. If the `mcftdmpqu` command is to be used in combination with the `mcftdlqle` command (to delete output queues), the output queue process must be held with the `mcfthldoq` command before executing the `mcftdmpqu` command. In this case, disk must be specified in the `-d` option (deletion type) of the `mcftdlqle` command.

(5) Holding process of, and releasing held process of, logical terminal output queues

When you want to copy the contents of the output queue (`mcftdmpqu` command) and delete the messages remaining in the output queue (`mcftdlqle` command) successively, the number of messages to be copied must match the number of messages to be deleted. For this reason, the input and scheduling of the output queue must be held during this period. You can use the `mcfthldoq` command to temporarily hold the input or scheduling of the output queue. Executing the `mcfthldoq` command holds the input or scheduling of the output queue, or both, depending on the option specification.

The held process of an output queue can be released with the `mcftrlsoq` command.

Be careful when you use the `mcfthldoq` command because it affects the OpenTP1 system as follows. Once you execute the `mcfthldoq` command and the target processing terminates, be sure to execute the `mcftrlsoq` command.

- If a UAP sends a message to a logical terminal for which the input of the output queue is held, it stops until the held process is released.
- For a UAP sending messages to multiple logical terminals, if the input of the output queue for one of the logical terminals is held, the messages to the other logical terminals may also be stopped until the held process is released.
- When an attempt is made to terminate OpenTP1 in normal termination mode, or in planned termination A or B mode, OpenTP1 terminates abnormally if any one of the service groups is held.

The status (held status and hold-released status of input and scheduling of an output queue) changed by the `mcftthldoq` or `mcfttrlsoq` command can be inherited during full recovery. If the maximum number of logical terminals is specified in the status inherit definition in the MCF communication configuration definition, the status of resources up to the specified value is inherited. Even after full recovery, the messages remaining in the OTQ because scheduling is held are inherited in the status waiting for the held process to be released. The messages waiting to be input into the OTQ because the input is held are input to the OTQ during full recovery.

(6) Deletion of logical terminal output queues

After establishment of a connection, messages remaining in the output queue can be deleted by the `mcftdlqle` command. Disk queues alone or both disk queues and memory queues can be deleted depending on the specification of the appropriate option of the `mcftdlqle` command.

If the `mcftdlqle` command is to be independently used, the `mcftdctle` command must be executed to shut down the logical terminal before executing the `mcftdlqle` command.

If the TP1/NET/HNA-560/20 protocol is used, the `mcftdctss` command must be executed to terminate the session. In addition, the logical terminal must not be a logical terminal of alternate origin. If the `mcftdlqle` command is to be used in combination with the `mcftdmpqu` command (to copy the contents of input/output queues), the output queue process must be held with the `mcftthldoq` command before executing the `mcftdmpqu` command. In this case, disk must be specified in the `-d` option (deletion type) of the `mcftdlqle` command.

You can also delete logical terminal output queues by issuing a function from a UAP. For details, see the manual *OpenTP1 Programming Guide*.

(7) Output of contents of logical terminal output queues

Where a terminal error occurs, the contents of logical terminal output queues can be output to a file by taking the following procedure:

1. Execute the `mcftthldoq` command to hold the process of output queues.
2. Execute the `mcftdmpqu` command to copy the contents of the output queues onto the specified file.
3. Execute the `mcftdlqle` command to delete the output queues. Specify `disk` with the `-d` option (deletion type).
4. Execute the `mcfttrlsoq` command to release the held process of the output queues. For the format of input/output queue dump files, see Appendix A. *Dump File Format of Input/Output Queues*.

(8) Startup and termination of logical terminal message journal collection

The `mcftactmj` command collects message journals (MJ). When the `mcftactmj`

command executes, collection of message journals for specified logical terminals begins. A message journal is collected from the time an input message editing user exit routine is called to the time an output message editing user exit routine is called.

Collection of message journals can be terminated by the `mcftdctmj` command.

(9) Forced termination of continuous inquiry-response processing for logical terminal

The `mcftendct` command terminates forcibly continuous inquiry-response processing for specified logical terminals.

The `mcftendct` command can only be used under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

For details about continuous inquiry-response processing, see the manual *OpenTP1 Protocol TP1/NET/XMAP3* or *OpenTP1 Protocol TP1/NET/HNA-560/20*.

(10) Startup and termination of alternate sending

The `mcftstalt` command is used where the logical terminal to which a message is to be sent is in shutdown status or an error prevents messages from being sent. Executing the `mcftstalt` command lets another logical terminal alternately output messages. This alternate sending can be terminated by the `mcftedalt` command.

The `mcftstalt` and `mcftedalt` commands can only be used under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

For alternate sending, see the manual *OpenTP1 Protocol TP1/NET/XMAP3* or *OpenTP1 Protocol TP1/NET/HNA-560/20*.

For details about alternate sending, see the manual *OpenTP1 Protocol TP1/NET/XMAP3* or *OpenTP1 Protocol TP1/NET/HNA-560/20*.

5.5 Service group operations

This section explains operations related to service groups.

(1) *Display of service group status*

The status of service groups can be displayed with the `mcfctlssg` command.

The displayed information includes the service group names, statuses of the service groups, numbers of received messages, etc.

(2) *Service group shutdown and release of shutdown status*

Service groups can be shut down with the `mcftdctsg` command. When the `mcftdctsg` command executes, input to input queues and input queue scheduling are shut down on the basis of specified options. When the shutdown of service group scheduling in the event of abnormal termination of an application (`srvghold=s`) is specified in the application attribute definition, the application terminates abnormally and the service group is shut down when the maximum application abnormal termination count is reached.

When input to an input queue is shut down, all messages already in the input queue are processed normally. If a message is received during shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated.

If a message is received while input queue scheduling for the memory queue is involved in shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated. If the scheduling is for the disk queue, the message is rescheduled after the service group is released from shutdown status.

If rescheduling of an input message that was being processed during the previous online operation fails at the time of a restart, input queue scheduling for the corresponding service group is shut down.

If a UAP terminates abnormally with a timeout because any value specified in the operands is exceeded, the service and service group are also shut down.

- `trn_cpu_time`
- `watch_next_chain_time`
- `xat_trn_expiration_time`

Once the cause of a rescheduling failure has been corrected, entry of the `mcfactsg` command releases the service group from shutdown status and reschedules the input message that was being processed during the previous online operation.

The status changed by the `mcftdctsg` or `mcfactsg` command (input to input queue and input queue scheduling status) is inherited during full recovery. If a maximum value for the number of service groups has been specified in the status inheritance

definition of the MCF manager definition, the status of the resources is inherited up to the specified value. In order not to inherit the shutdown status after full recovery, specify the `-r` option in the `mcfstdctsg` command.

(3) Copying of contents of service group input queues

Where the contents of service group input queues cannot be output, you can use the `mcfstdmpqu` command to check the contents of the disk messages remaining in the input queue. Executing this command lets the contents of the input queue copied onto a file. If then the `-a` option is specified, the contents of the output queue are again written on the destination file for which the `mcfstdmpqu` command was previously executed.

If the `mcfstdmpqu` command is to be used in combination with the `mcfstdlqsg` command (to delete input queues), the input queue process must be held with the `mcfsthldiq` command before executing the `mcfstdmpqu` command. In this case, disk must be specified in the `-d` option (deletion type) of the `mcfstdlqsg` command.

(4) Holding process of, and releasing held process of, service group input queues

When you want to copy the contents of the input queue (`mcfstdmpqu` command) and delete the messages remaining in the input queue (`mcfstdlqsg` command) successively, the number of messages to be copied must match the number of messages to be deleted. For this reason, the input and scheduling of the input queue must be held during this period. You can use the `mcfsthldiq` command to temporarily hold the input or scheduling of the input queue. Executing the `mcfsthldiq` command holds the input or scheduling of the input queue, or both, depending on the option specification.

The held process of an input queue can be released with the `mcfstrlsiq` command.

Be careful when you use the `mcfsthldiq` command because it affects the OpenTP1 system as follows. Once you execute the `mcfsthldiq` command and the target processing terminates, be sure to execute `mcfstrlsiq` command.

- If you hold the input of the input queue for a service group which is started up with the MCF application start service, the MCF application start service holds the startup of all the service groups.
- If you hold the input of the input queue for a service group which the MCF communication service starts, the messages to be sent to the logical terminal of the input source also remain in the OTQ until the held process is released.
- If you hold the input of the input queue for a service group which the MCF communication service starts, this may degrade the processing capacity of the MCF communication service.
- If an attempt is made to input a message to the service group for which the input of the input queue is held, the operation command to the logical terminal of the input source and the connection to which the logical terminal belongs may result

in a timeout. The operation command is accepted even if a timeout occurs. The operation command is executed after releasing the service group from hold.

- When an attempt is made to terminate OpenTP1 in normal termination mode, or in planned termination A or B mode, OpenTP1 terminates abnormally if any one of the service groups is held.

The status (held status and hold-released status of input and scheduling of an input queue) changed by the `mcftthldiq` or `mcfttrlsiq` command can be inherited during full recovery. If the maximum number of service groups is specified in the status inherit definition in the MCF manager definition, the status of resources up to the specified value is inherited. Even after full recovery, any messages remaining in the ITQ because scheduling is held are inherited in the status waiting for the held process to be released. Any messages waiting to be input into the ITQ because the input is held are input to the ITQ during full recovery.

When the `-r` option is specified in the `mcftthldiq` command, the hold status is not inherited after restart even when the maximum number of service groups has been specified.

(5) Deletion of service group input queues

Messages remaining in the input queue can be deleted by the `mcftdlqsg` command. Disk queues alone or both disk queues and memory queues can be deleted depending on the specification of the appropriate option of the `mcftdlqsg` command.

If the `mcftdlqsg` command is to be independently used, the `mcftdctsg` command must be executed to shut down the service group before executing the `mcftdlqsg` command.

If the `mcftdlqsg` command is to be used in combination with the `mcftdmpqu` command (to copy the contents of input/output queues), the input queue process must be held with the `mcftthldiq` command before executing the `mcftdmpqu` command. In this case, disk must be specified in the `-d` option (deletion type) of the `mcftdlqsg` command.

(6) Output of contents of service group input queues

Where the contents of service group input queues cannot be output, they can be output to a file by taking the following procedure:

1. Execute the `mcftthldiq` command to hold the process of input queues.
2. Execute the `mcftdmpqu` command to copy the contents of the input queues onto the specified file.
3. Execute the `mcftdlqsg` command to delete the input queues. Specify `disk` with the `-d` option (deletion type).
4. Execute the `mcfttrlsiq` command to release the held process of the input queues. For the format of input/output queue dump files, see Appendix A. *Dump File*

Format of Input/Output Queues.

5.6 Service operations

This section explains operations related to services.

(1) *Display of service status*

The status of services can be displayed with the `mcfctlssv` command.

The displayed information includes the service names, statuses of the service, etc.

(2) *Service shutdown and release of shutdown status*

Services can be shut down with the `mcftdctsv` command. When the `mcftdctsv` command executes, input to input queue and input queue scheduling are shut down on the basis of specified options. If the application attributes definition is specified in such a manner that service input and its scheduling are to be shut down (`servhold=a`) in the event of abnormal termination of an application, the service is shut down when an application terminates abnormally and the specified application abnormal terminations count has been reached.

When input to an input queue is shut down, all messages already in the input queue are processed normally. If a message is received during shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated.

If a message is received while input to an input queue and input queue scheduling are involved in shutdown processing, the MCF event that reports the discarding of a message (`ERREVT2`) is activated.

If the UAP terminates abnormally with a timeout because any value specified in the operands is exceeded, the service and service group are also shut down.

- `trn_cpu_time`
- `watch_next_chain_time`
- `xat_trn_expiration_time`

Services can be released from shutdown status by the `mcfactsv` command.

The status changed by the `mcftdctsv` or `mcfactsv` command (input to input queue and input queue scheduling status) is inherited during full recovery. If a maximum number of services has been specified in the status inheritance definition of the MCF manager definition, the status of resources is inherited up to the specified value.

5.7 Protocol-specific operations

This section explains operations specific to individual protocol products.

5.7.1 Startup and termination of a session

A session can be started up by the `mcfactss` command, and terminated by the `mctdctss` command. The `mcfactss` and `mctdctss` commands can only be used under the TP1/NET/HNA-560/20 protocol. For sessions, see the manual *OpenTP1 Protocol TP1/NET/HNA-560/20*.

5.7.2 Display of buffer group utilization status

The utilization status of buffer groups can be displayed with the `mcfslsbuf` command.

The displayed information includes the MCF identifiers, buffer group numbers, numbers of buffers, etc.

5.7.3 Map file

The mapping service facility is available under the TP1/NET/XMAP3 protocol or TP1/NET/HNA-560/20 protocol.

The `dcmaphg` command and `dcmpls` command can only be used under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

For details about the mapping service facility, see the manual *OpenTP1 Protocol TP1/NET/XMAP3* or *OpenTP1 Protocol TP1/NET/HNA-560/20*.

(1) Change of path name

Use the `dcmaphg` command to change the current map under the mapping service facility. The `dcmaphg` command can change the path name of the following map files.

- Standard physical map input path
- Alternate physical map input path
- Standard PAGEC module input path
- Alternate PAGEC module input path

(2) Display of resources

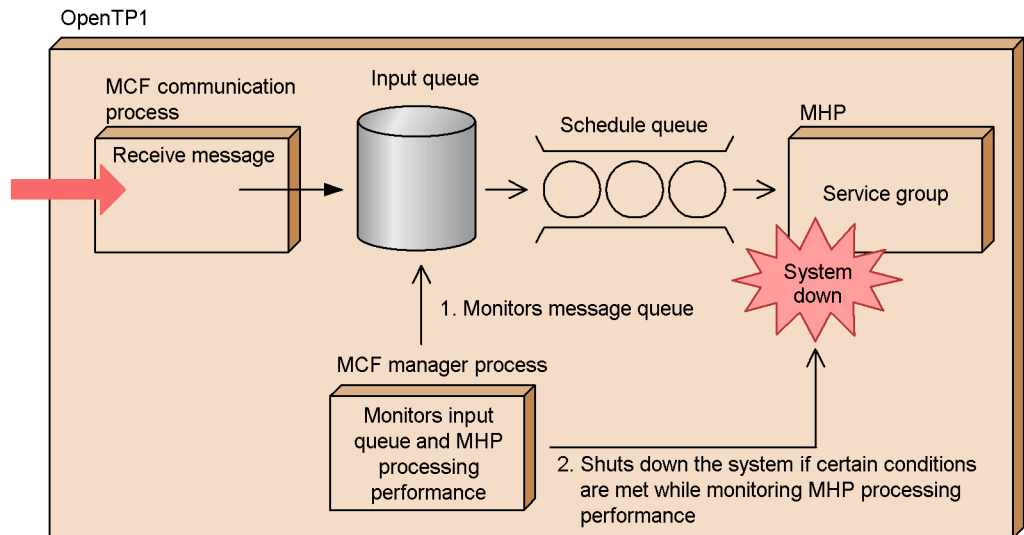
Loaded physical maps or PAGEC modules can be displayed by using the `dcmpls` command with the `-s` option.

The `dcmpls` command with the `-p` option displays the path name of the map file.

5.8 Monitoring message queues

The *message queue monitoring facility* monitors the backlog of messages in the input queue at regular intervals. This facility can only be applied to user servers (MHPs). The following figure shows an overview of the message queue monitoring facility.

Figure 5-1: Overview of message queue monitoring facility



1. After the MCF service starts, the message queue monitoring facility begins monitoring the messages remaining in the input queue.

The phase during which the facility monitors only the number of messages in the message queue is called the *queue-depth monitoring phase*. Monitoring continues until the MCF service terminates.

2. Once the number of messages held in the input queue exceeds a threshold, the facility regularly gauges the ability of the MHP to process those messages.

This is called the *processing-capacity judgment phase*. During this phase, if the message queue monitoring facility determines that the MHP is unable to process the messages at the expected rate, it either outputs the message KFCA11820-W and continues processing, or outputs the message KFCA11821-E and shuts down the OpenTP1 system.

If a service group defined as a monitoring target is not defined in the attribute definitions for the MCF application, the message queue monitoring facility outputs the message KFCA11822-W and continues processing.

(1) Required operands

To use the message queue monitoring facility, specify the following iterations of the `mcfmsvg` definition command in the MCF manager definition.

- `mcfmsvg -g "servgrp=service-group-name"`

Specify the name of the service group whose input queue you want to monitor.

- `mcfmsvg -w "watchcnt=input-queue-message-count-threshold"`

Specify the threshold for the number of messages that can reside in the queue before the message queue monitoring facility transitions from monitoring the queue depth to judging processing capacity.

- `mcfmsvg -w "watchint=input-queue-monitoring-interval"`

Specify the interval at which the message queue monitoring facility checks message numbers and judges the MHP processing capacity.

- `mcfmsvg -w "expectcnt=expected-number-of-service-requests-processed-by-MHP"`

Specify the number of service requests that the MHP is expected to have processed by the next time that processing capacity is judged (the number of requests that must be processed to clear the messages in the input queue).

- `mcfmsvg -w "abort=yes|no"`

Specify whether to shut down the OpenTPI system when MHP is judged to have insufficient processing capacity.

For details about the operands, see the manual *OpenTPI System Definition*.

(2) Flow of message queue monitoring

The processing to monitor the backlog of queued messages is as follows:

1. When the MCF service is started, the message queue monitoring facility starts in the queue-depth monitoring phase, and begins monitoring the number of messages in the input queue at the specified interval (as specified by `mcfmsvg -w watchint`).
2. When the number of messages in the input queue exceeds the number specified by `mcfmsvg -w watchcnt`, the message queue monitoring facility enters the processing-capacity judgment phase.

While in the processing-capacity judgment phase, the message queue monitoring facility uses the following formula to determine the processing capacity of the MHP.

Formula for determining MHP processing capacity:

Number of service requests processed by MHP < Number of service requests expected to be processed by MHP or number of service requests queued at time of previous judgment, whichever is smaller

Depending on the outcome, the following processing takes place once this judgment is made.

- When the formula for determining MHP processing capacity does not hold true:
OpenTP1 continues processing.
- When the formula holds true, and the OpenTP1 system is not set to shut down when MHP processing capacity is deemed inadequate (no is specified for `mcfmsvg -w abort`):
The message KFCA11820-W is output and OpenTP1 continues processing.
- When the formula holds true, and the OpenTP1 system is set to shut down when MHP processing capacity is deemed inadequate (yes is specified for `mcfmsvg -w abort`):
The message KFCA11821-E is output and the OpenTP1 system is shut down.

When the number of messages held in the input queue falls below the number specified by `mcfmsvg -w watchcnt`, the message queue monitoring facility leaves the processing-capacity judgment phase and re-enters the queue-depth monitoring phase.

The following table shows the conditions judged during message queue monitoring, and how MCF behaves in each instance.

Table 5-2: Relationship between judgment conditions during message queue monitoring and MCF behavior

Judgment condition					MCF behavior
Previous judgment phase	Relation between number in queue and threshold	Relation between A and B	Relation between A and C	mcfmsvg -w abort specification	
Queue-depth monitoring phase	Number in queue < Threshold	N/A	N/A	N/A	Stays in queue-depth monitoring phase and continues processing.

Judgment condition					MCF behavior	
Previous judgment phase	Relation between number in queue and threshold	Relation between A and B	Relation between A and C	mcfmsvg -w abort specification		
	Number in queue \geq Threshold	N/A	N/A	N/A	Enters processing-capacity judgment phase and continues processing.	
Processing-capacity judgment phase	Number in queue $<$ Threshold	N/A	N/A	N/A	Enters queue-depth monitoring phase and continues processing.	
					Number in queue \geq Threshold	$A \geq B$
	$A < C$	N/A				
	$A < B$	$A \geq C$	N/A	Yes		Outputs message KFCA11821-E, forcibly terminates MHP, and shuts down OpenTPI system.
		$A < C$	Yes			
					No	Outputs message KFCA11820-W, remains in processing-capacity judgment phase and continues processing.

Legend:

A: Number of service requests processed by MHP

B: Number of service requests expected to be processed by MHP

C: Number of service requests queued at time of previous judgment

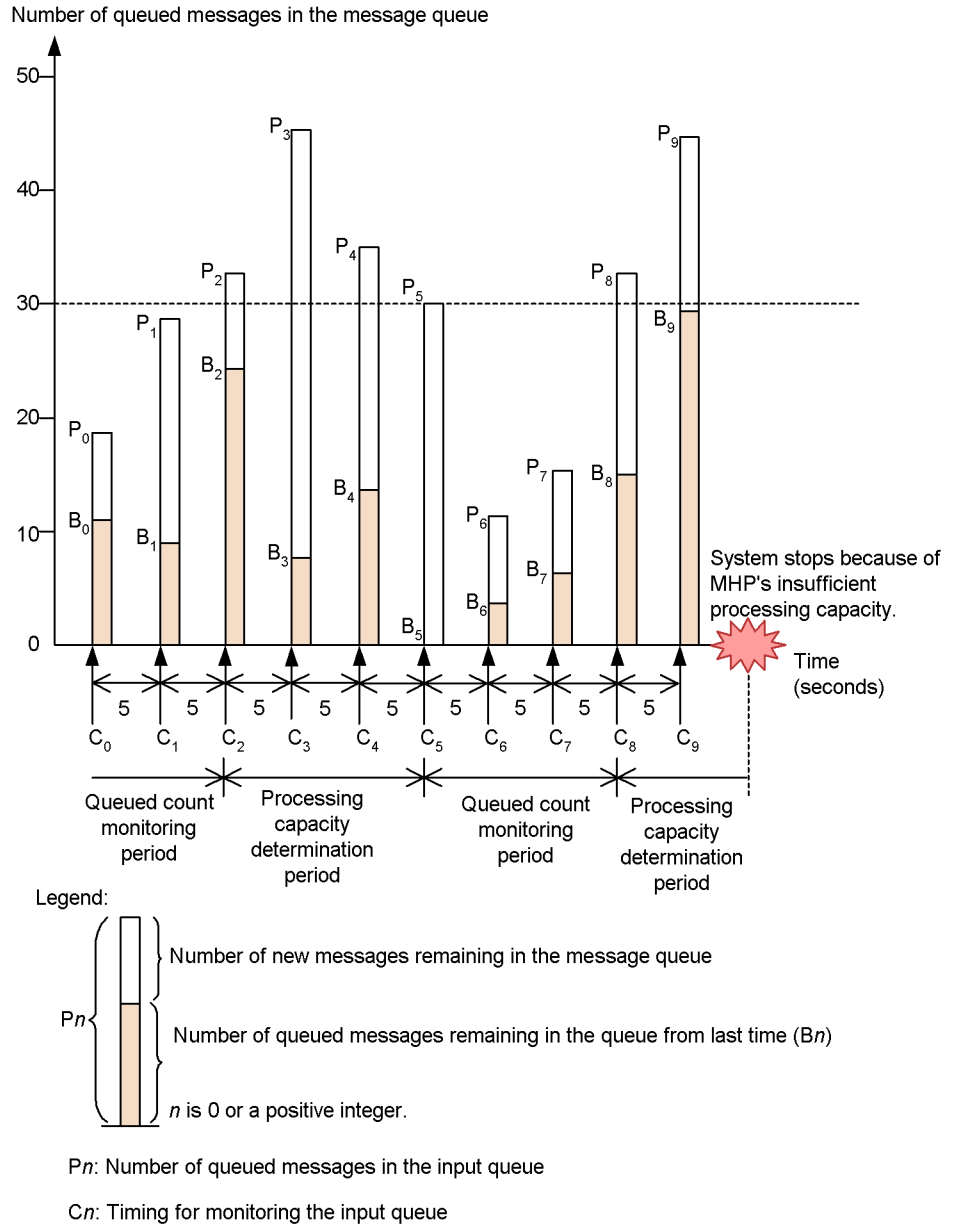
(3) Example of message queue monitoring

The following is an example of the processing of the message queue monitoring facility, assuming a situation in which the `mcfmsvg` definition command is specified as follows in the MCF manager definition:

Specification of `mcfmsvg` definition command in MCF manager definition:

- Input queue message threshold (`mcfmsvg -w "watchcnt=30"`)
- Input queue monitoring interval (`mcfmsvg -w "watchint=5"`)
- Number of service requests expected to be processed by MHP (`mcfmsvg -w "expectcnt=24"`)
- Whether to shut down OpenTP1 system when inadequate MHP processing capacity is detected (`mcfmsvg -w "abort=yes"`)

Figure 5-2: Example of the message queue monitoring facility



Because 30 is specified as the threshold for messages in the input queue, the message queue monitoring facility will enter the queue-depth monitoring phase for segments

C2 to C5, and from C8 onward, in the above figure. For all other segments, the message queue monitoring facility will be in the processing-capacity judgment phase.

The following table shows the relationship between the number of messages in the input queue when the queue status is judged, and the judgment result.

Table 5-3: Queued message counts and judgment results

No.	Number of queued messages carried over from previous judgment (B_n)	Number of queued messages at previous judgment (P_{n-1})	Number of service requests processed since last judgment ($P_{n-1} - B_n$)	Number of service requests expected to be processed by MHP	Judgment result
1	11	-	-	24	Because the number of queued messages has not reached the threshold, the queue-depth monitoring phase remains in effect and OpenTP1 stays online.
2	9	18	9	24	
3	25	28	3	24	Because the number of queued messages has reached the threshold, the processing-capacity judgment phase begins.
4	8	32	24	24	Although the number of queued messages has remained above the threshold, the MHP was able to process more than the expected number of service requests. Thus, the processing-capacity judgment phase remains in effect and OpenTP1 stays online.
5	13	45	32	24	
6	0	35	35	24	
7	3	30	27	24	Because the number of queued messages has fallen below the threshold, the queue-depth monitoring phase begins.
8	5	11	6	24	Because the number of queued messages has not reached the threshold, the queue-depth monitoring phase remains in effect and OpenTP1 stays online.

No.	Number of queued messages carried over from previous judgment (B_n)	Number of queued messages at previous judgment (P_{n-1})	Number of service requests processed since last judgment ($P_{n-1} - B_n$)	Number of service requests expected to be processed by MHP	Judgment result
9	15	17	2	24	Because the number of queued messages has reached the threshold, the processing-capacity judgment phase begins.
10	29	32	3	24	Because the number of queued messages has remained above the threshold, and the MHP was unable to process the expected number of service requests, OpenTP1 is taken offline.

(4) Notes

- The message queue monitoring facility does not monitor service groups that are shut down or being held. Service groups that enter shutdown or held status during the processing-capacity judgment phase are assumed to have transitioned to the queue-depth monitoring phase. Accordingly, for the first monitoring cycle after the service group is released from shutdown or held status, the message queue monitoring facility monitors queue depth regardless of the phase that was in effect when the group was shut down or held. If the number of messages in the queue is found to exceed the threshold, the message queue monitoring facility will then begin the processing-capacity judgment phase.
- The messages in input queues that are deleted by the `mcf tdlqsg` command are not counted in the number of service requests processed by the MHP.
- Do not advance or set back the time by whole seconds while OpenTP1 is online. Because this facility uses data and time information to operate, changing the time while OpenTP1 is online may cause the following problems:
 - If you advance the time, the MCF may wrongly determine that the time to judge the queue status or processing ability has arrived and, depending on the definitions in effect, may shut down the OpenTP1 system.
 - If you set the time back, depending on the definitions in effect, the MCF may not monitor queue depth.

For details about changing the time, see *3.13.4 Notes on changing the time*.

5.9 Queue operations

(1) Display of queue group status

The status of physical files and queue files can be displayed for queue groups with the `quels` command.

The displayed information includes the record length of a physical file, number of records in use, path name of the physical file, etc.

(2) Allocation of a physical file for a message queue

The `queinit` command allocates a physical file to be used as a message queue by the message queue service.

(3) Deletion of the physical file for a message queue

A physical file allocated for a message queue by the `queinit` command can be deleted with the `querm` command. The specified physical file cannot be deleted if it is being used in the online mode.

Chapter

6. OpenTP1 Optional Facility Operation

This chapter explains the OpenTP1 optional facilities.

This chapter contains the following sections:

6.1 Resource manager operations

6.1 Resource manager operations

6.1.1 Display of information about resource manager

The information about a resource manager can be checked by the `trnlstrm` command. Executing this command outputs to a standard output information about the resource manager linked to an OpenTP1, UAP or transaction control object file (such as the resource manager name, attributes, switch name, and object name).

6.1.2 Cataloging and deletion of resource manager

XA interface object files provided by resource managers have not been linked with a transaction service control executable-form program and a client service execution-form program that are used to install an OpenTP1 program product. When the `dcsetup` command is executed, the system determines the installed OpenTP1 program product and automatically links XA interface object files of a resource manager (DAM, TAM, MCF, ISAM, or MQA) provided by OpenTP1. To execute a transaction using a non-OpenTP1 resource manager under OpenTP1, execute the `dcsetup` command and then execute the `trnlnkrm` command to catalog the non-OpenTP1 resource manager before starting OpenTP1. Executing the `trnlnkrm` command re-creates a transaction service control executable-form program, a client service executable-form program and a standard transaction control object file. For information needed when non-Hitachi databases are used, see the manual *OpenTP1 Programming Guide*.

The XA interface object files provided by all resource managers and to be accessed by UAPs working under OpenTP1 must be linked with the transaction service control execution-form program and client service executable-form program. It is impossible to start any UAP that links XA interface object files for the resource manager that is not linked with the transaction service control executable-form program and client service execution-form program.

Up to 32 resource managers can be cataloged under OpenTP1.

The `trnlnkrm` command with the `-D` or `-d` option specified deletes a resource manager that is no longer accessed from a transaction executed under OpenTP1. Executing this command with the `-D` or `-d` option specified deletes XA interface object files for the resource manager specified from a transaction service control executable-form program and a client service executable-form program.

The `trnlnkrm` command can be used in offline mode only.

6.1.3 Creation of transaction control object files

In order for a UAP working under OpenTP1 to access a resource manager within a transaction, a transaction control object file must be linked with that UAP.

For a UAP that accesses all resource managers registered in OpenTP1, the standard transaction control object file named `dc_trn_allrm.o` (a file under `$DCDIR/spool/trnrmcmd/userobj` which is created when the `dcsetup` command is executed and recreated when the `trnlnkrm` command is executed) must be linked with that UAP.

For a UAP that accesses some of the resource managers registered in OpenTP1, a transaction control object file must be created with the `trnmkobj` command and linked with that UAP.

No transaction control object file needs to be linked with a UAP that does not access resource managers within a transaction. If the registered condition of resource managers has been changed with the `trnlnkrm` command, a transaction control object file must be linked again with a UAP with which a standard transaction control object file has been linked.

Before starting OpenTP1, the user must link to the UAP the transaction control object file created by the `trnmkobj` command and the XA interface object file provided by the resource manager.

A UAP to which the transaction control object file or XA interface object file has not been linked produces an error when a function provided by the resource manager is issued.

6.1.4 Resource manager monitor operations

Before using the resource manager monitor (RMM), program product TP1/Resource Manager Monitor must be installed.

(1) Notes on creating commands for resource managers to be monitored

The user must create commands for resource managers to be monitored and define them in the RM-to-be-monitored definition. Following are the notes on creating these commands:

(a) Startup command

The startup command must be created so that the following conditions are satisfied:

- When the startup command is completed, the termination command can be executed.
- When the startup command is completed, all processes to be monitored have been activated.
- When the startup command is completed, the resource manager can monitor processes.
- The startup command terminates normally if termination code `exit()` is 0, and abnormally if the code is not 0.

While awaiting for the startup command to terminate, the RMM service assumes normal termination when the termination code is 0" or abnormal termination when the code is not 0.

If the startup command terminates normally but if system fails to obtain the process ID of the process to be monitored, the startup process will be unsuccessful. OpenTP1 checks the time until the startup command terminates; if the maximum check-time is exceeded, the startup process will also be unsuccessful.

(b) Termination command

The termination command must be created so that the following conditions are satisfied:

- When the termination command is completed, the startup command can be executed.
- When the termination command is completed, all processes to be monitored have been terminated.
- When the termination command is completed, the system resources for all processes to be monitored have been released.
- The termination command terminates normally if termination code `exit()` is 0, and abnormally if the code is not 0.

While awaiting for the termination command to complete, the RMM service assumes normal termination when the termination code is 0 or abnormal termination when the code is not 0.

If the termination command terminates normally but if any process to be monitored remains, the termination process will be unsuccessful. OpenTP1 checks the time until the termination command terminates; if the maximum check-time is exceeded, the termination process will also be unsuccessful.

(c) Forced termination command

The forced termination command must be created so that the following conditions are satisfied:

- When the forced termination command is completed, the startup command can be executed.
- When the forced termination command is completed, the system resources for all processes to be monitored have been released.
- The forced termination command terminates normally if termination code `exit()` is 0, and abnormally if the code is not 0.

The forced termination command can be executed even when there are no processes to be monitored.

While awaiting for the forced termination command to complete, the RMM service assumes normal termination when the termination code is 0 or abnormal termination when the code is not 0.

If the forced termination command terminates normally but if any process to be monitored remains, the forced termination process will be unsuccessful. OpenTP1 checks the time until the forced termination command terminates; if the maximum check-time is exceeded, the forced termination process will also be unsuccessful.

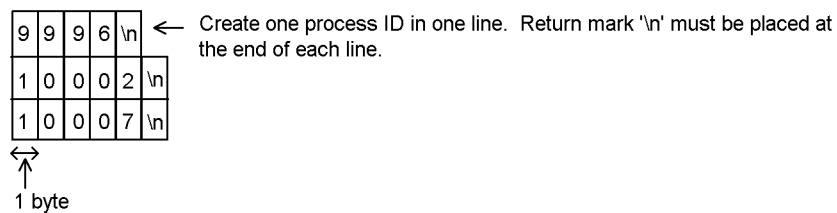
(d) Process ID acquisition command

1. Creation of process ID acquisition command

The process ID acquisition command must be created so that the following conditions are satisfied:

- The process ID of every process to be monitored is output to the standard output.
- The process ID is output in the format shown in the following figure.

Figure 6-1: Process ID output format



- One of the following conditions is reported to the RMM service by using termination code `exit()`:

0

- The resource manager to be monitored is now working.

1

- The resource manager to be monitored is not normal.

2

- The resource manager to be monitored is not working.

RMM provides `rmmoraid` as a model process ID acquisition command. This command has been created so that the following conditions are satisfied:

- The process ID of every process to be monitored is output to the standard output.

- The process ID is output in the format shown in Figure 6-1.
- If the process name of a process is described in the command, its process ID is acquired by using /bin/ps.
- The `exit ()` termination code has the following meaning:

0

- All process IDs have been acquired.

1

- Not all process IDs have been acquired.

2

- No process IDs have been acquired.

2. Restart of the resource manager to be monitored

When the resource manager to be monitored has a facility that performs automatic full recovery at an error, the RMM service does not need to restart the resource manager. If an error occurs, the facility of the resource manager automatically performs full recovery. First output 0 as the standard output of the process ID acquisition command. This prohibits the RMM service from performing full recovery of the resource manager to be monitored.

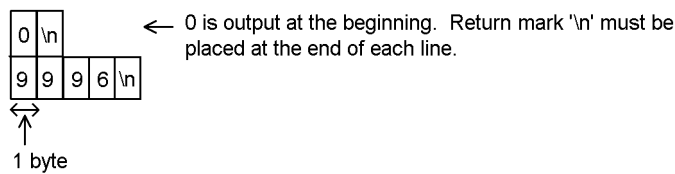
To perform full recovery using the facility of the resource manager to be monitored, satisfy the following conditions:

- There is only one process to be monitored.
- The termination code of `exit ()` is 0 or 2.

If the `exit ()` process ID acquisition command terminates with a termination code of 0, the RMM service assumes it to be completion of the automatic full recovery of the resource manager to be monitored.

The following figure shows the format of the standard output of the process ID acquisition command.

Figure 6-2: Format of standard output of process ID acquisition command



(e) Other commands

- The RMM service provides these four commands that can be used for ORACLE definitions:

Startup command

```
$DCDIR/etc/RMmonitor/rmmorast
```

Termination command

```
$DCDIR/etc/RMmonitor/rmmorasp
```

Forced termination command

```
$DCDIR/etc/RMmonitor/rmmoraab
```

Process ID acquisition command

```
$DCDIR/etc/RMmonitor/rmmoraid
```

All of these commands are implemented in a shell script using Bourne shells. #CHANGEABLE is written in each changeable area; the user can change the description in such an area according to an appropriate environment.

- The RMM service provides these four commands that can be used for the HiRDB (single mode) definitions:

Startup command

```
$DCDIR/etc/RMmonitor/rmmhirst
```

Termination command

```
$DCDIR/etc/RMmonitor/rmmhirsp
```

Forced termination command

```
$DCDIR/etc/RMmonitor/rmmhirab
```

Process ID acquisition command

```
$DCDIR/etc/RMmonitor/rmmhirid
```

All of these commands are implemented in a shell script using Bourne shells. "#CHANGEABLE" is written in each changeable area; the user can change the description in such an area according to an appropriate environment.

- HiRDB has a facility that performs automatic full recovery at an error. First output 0 as the standard output of the process ID acquisition command. HiRDB performs full recovery at an error.
- The shell script created is forked & executed by the RMM service and is different in seven points from command execution from a normal shell:
 1. The following environment variables are available:

DCSVNAME

- `_rmm` (unchangeable)

DCDIR

- Set uniquely for OpenTP1 (unchangeable)

DCCONFPATH

- Set uniquely for OpenTP1 (unchangeable)

LANG

- Set uniquely for OpenTP1 (changeable)

TZ

- Default value for OS (changeable)

PATH

- Follows the specification of `prcsvpath` in the process service definition (changeable)

2. The current directory is `$DCDIR/tmp/home/_rmm.xxx` (`xxx`: process ID of RMM service).
3. The standard input, standard output or standard error output of a process is a console.
4. The user ID and group ID follow the RM-to-be-monitored definition.
5. There is no control terminal.
6. `umask` is `000`.
7. Any operations at signal reception is an OS default.

All of the above conventions for the variables affect the commands to be used in the shell script and the processes to be activated. The user can change the setting of a variable only in the shell script or a user-provided command.

6.1.5 Suspending OpenTP1 startup until the startup of a resource manager

Resource managers provided outside of OpenTP1 (asynchronous RMs) can start and stop without synchronizing with OpenTP1. When you start OpenTP1 while the asynchronous RM to be accessed through the XA interface is inactive and you execute a user server that accesses the asynchronous RM, an `xa_open` function error occurs and the user server may terminate abnormally or shut down. Possibility of an error is especially high when this type of user server starts automatically (using the facility for starting a specified user server when OpenTP1 starts).

To prevent this situation, you should start OpenTP1 after the asynchronous RM is completely started up. This facility checks the startup of the asynchronous RM registered in OpenTP1 when OpenTP1 starts. If the applicable RM is not started, this facility suspends OpenTP1 until all the asynchronous RMs are started. By using this facility, you do not need to be aware of when to start OpenTP1 and asynchronous RMs when you start the system. This reduces the load on the system.

Whether an asynchronous RM is started is determined by whether the `xa_open` function provided by the asynchronous RM terminates normally. Therefore, OpenTP1 does not wait for the start of an asynchronous RM whose `xa_open` function terminates normally even if the RM is not started.

(1) How to use this facility

To use this facility, specify the `trn_wait_rm_open` operand in the transaction service definition. You can specify the number of retries and the interval for checking the startup of an inactive asynchronous RM.

(2) Notes

- OpenTP1 only uses this facility to wait for the start of resource managers that are not provided by OpenTP1 and that have been registered in the OpenTP1 system using the `trnlncrm` command.
- In the system specified with `stop` or `retry_stop` in the `trn_wait_rm_open` operand in the transaction service definition, if a resource manager on which to be waited cannot start because of an error, the OpenTP1 system also cannot start (rerun). To stop the startup of a resource manager and start OpenTP1, change the value specified in the `trn_wait_rm_open` operand to `continue` or `retry_continue` and restart the OpenTP1 system. To cancel the startup processing during a retry, use the `dcstop -f` command to perform a forced termination of the OpenTP1 system.
- Whether an asynchronous RM is started is determined by whether the `xa_open` function provided by the asynchronous RM terminates normally. Therefore, OpenTP1 does not wait for the start of an asynchronous RM whose `xa_open` function terminates normally even if the RM is not started.
- If an `xa_open` function error occurs for the asynchronous RM when you specify `retry_continue` or `retry_stop` in the `trn_wait_rm_open` operand in the transaction service definition, the `xa_open` function is issued for the inactive asynchronous RM for the number of times specified in the `trn_retry_count_rm_open` operand at the interval specified in the `trn_retry_interval_rm_open` operand. Therefore, the actual retry interval for issuing the `xa_open` function is as follows:

`xa_open` function processing time for all the inactive asynchronous RMs + time specified in the `trn_retry_interval_rm_open` operand

- To wait for the start of HiRDB, use HiRDB 02-05-/G or a later version.

6.1.6 Suspending transaction recovery (when Oracle9i RAC is used)

When you use Oracle9i RAC[#], if the system ends abnormally, in-doubt transactions (undetermined transactions) are transferred from the failed running server to the standby server. If the transfer of in-doubt transactions takes too much time, the transaction determination type may become inconsistent between OpenTP1 and Oracle9i because of the specifications of Oracle9i.

Therefore, the transaction recovery by OpenTP1 is suspended during the transfer of in-doubt transactions to avoid the inconsistency of transaction determination types.

#

RAC is an abbreviation for Real Application Clusters. RAC is a cluster system that allows multiple nodes to share a single database on a shared disk.

(1) How to use this facility

To use Oracle9i RAC, specify the `-r` option in the `trnstring` definition command in the transaction service definition and suspend the recovery of OpenTP1 transactions. To enable the `-r` option in the `trnstring` definition command, you need to specify "OPS_FAILOVER=T" as an open character string.

When the `-r` option is specified in the `trnstring` definition command, the system suspends the responses from the resource manager, locking the transaction recovery process. Therefore, increase the number of transaction parallel recovery processes (value specified in the `trn_recovery_process_count` operand) for the number of resource managers specified in the `-r` option in the `trnstring` definition command. If you do not increase the number of transaction parallel recovery processes, transaction recovery processing may be affected.

(2) Notes

If an error occurs in a resource manager during startup processing or online processing while transaction recovery is suspended, the transaction recovery is delayed until the error in the resource manager is corrected.

You can change or delete the `-r` option in the `trnstring` definition command in the transaction service definition at restart. For details, see the description about the system service definition and the definitions that can be changed at restart in the manual *OpenTP1 System Definition*.

6.1.7 Pre-online transaction recovery

When OpenTP1 restarts to recover all transactions, it can determine the managed transactions based on journals.

The pre-online transaction recovery facility acquires the information about undetermined transactions from the resource manager specified in the `-m` option in the

`trnstring` definition command in the transaction service definition. Then the facility recovers the transactions before OpenTP1 goes online when OpenTP1 starts (regardless of normal start or restart).

When you use this facility, recovery processing is repeated at the interval specified in the `trn_start_recovery_interval` operand in the transaction service definition. Recovery processing is repeated until there is no information left about undetermined transactions from the resource manager. Therefore, OpenTP1 goes online only when there is no information about undetermined transactions from the resource manager.

(1) How to use this facility

To use this facility, use the `trnlkrm` command to re-register resource managers in the OpenTP1 system and specify `-m` option in the `trnstring` definition command in the transaction service definition. For details, see the description about the transaction service definition in the manual *OpenTP1 System Definition*.

(2) Notes

- Only the resource managers that are registered in the OpenTP1 system by using the `trnlkrm` command can be specified in the `trnstring` definition command.
- The determination of transactions may be delayed if there is a resource manager that does not immediately respond to the issued `xa_recover` function.

Chapter

7. Operations Under the Multi-node Facility

This chapter explains operations required when the multinode facility is used: environment setup and its procedure, OpenTP1 startup and termination, OpenTP1 node status display, and global journal operations.

This chapter contains the following sections:

- 7.1 OpenTP1 environment setup
- 7.2 OpenTP1 startup and termination
- 7.3 Display of OpenTP1 node status
- 7.4 Global journal operations

7.1 OpenTP1 environment setup

This section explains OpenTP1 environment setup procedures that are required to operate OpenTP1 under the multinode facility. The environment should be set up in the order of section numbers.

7.1.1 Cataloging of OpenTP1 administrator

The superuser must catalog the user ID of an OpenTP1 administrator into the OS. The information to be cataloged includes the login name, user ID, group ID, home directory, and login shell, all of which may be arbitrary.

After cataloging the user ID, set up a password.

The OpenTP1 administrator has the right to:

- Access OpenTP1 system files and directories and prohibit other users from writing data to them, and
- Execute operation commands causing changes in OpenTP1 configuration.

7.1.2 Definition of OpenTP1 group

The superuser must define a group dedicated to OpenTP1. Defining an OpenTP1 group allows only a limited number of non-OpenTP1 group users to access OpenTP1 files, leading to higher security.

7.1.3 Installation of OpenTP1

The superuser must install the OpenTP1 program products. TP1/Server Base and TP1/Multi must be installed at all OpenTP1 nodes.

The OpenTP1 installation directory depends on the OS you are using.

7.1.4 Creation of OpenTP1 directory

The superuser creates the OpenTP1 directory. The length of the name of the OpenTP1 directory depends on your OS.

You cannot use symbolic links for OpenTP1 directories.

Assign one partition for the OpenTP1 directory. Do not use this partition for any other programs. If other programs create files in the partition for the OpenTP1 directory, consuming free space on the disk or the number of i nodes, operation of OpenTP1 may be adversely affected. Specify the owner, the group, and the mode as follows:

Owner

OpenTP1 administrator

Group

OpenTP1 group

Mode

0755

7.1.5 Creation of system definition

The OpenTP1 administrator must create OpenTP1 system definition before the superuser catalogs OpenTP1 into the OS.

The contents of system definition created at a node with the global archive journal service (called an archive-journal target node) differ from those created at a node whose journal is archived by the archive-journal target node (called an archive-journal source node).

For system definition information, see the manual *OpenTP1 System Definition*.

7.1.6 Cataloging of OpenTP1 into OS

The superuser must catalog OpenTP1 into the OS using the OpenTP1 `dcsetup` command after the OpenTP1 administrator has created system definition information.

7.1.7 Creation of OpenTP1 file system areas

The OpenTP1 file systems can be created on character type special files or ordinary files.

To create the OpenTP1 file systems on character type special files, disk partitions dedicated to the system must be allocated. These partitions must not be mounted. If the systems are to be created on ordinary files, disk partitions need not be allocated.

An archive-journal source node must have two kinds of OpenTP1 file systems, one for the system and the other for users. For creation on character type special files, two different disk partitions, one for the system and the other for users, must be allocated. An archive-journal target node need not have the OpenTP1 file system for users.

The character type special files or ordinary files on which the OpenTP1 file systems have been created are called OpenTP1 file system areas.

To prevent unauthorized users from accessing the OpenTP1 file system areas, their owners and access right must be defined as described in the following table. OS commands are used to define the owner and access right.

Table 7-1: OpenTP1 file system area owners and access permissions

Type of OpenTP1 file system area	Owner		Access permissions		
	User ID	Group ID	Owner	Group	Others
For system	OpenTP1 administrator	OpenTP1 group	rw (readable and writable)	r- (readable)	r- (readable)
For users	OpenTP1 administrator	OpenTP1 group	rw (readable and writable)	rw (readable and writable)	r- (readable)

Note

OpenTP1 file systems for the system and for users can be allocated in the same area. In such a case, access permissions must be for users.

7.1.8 Reservation of resources for OpenTP1 internal control

The `dcmakeup` command can be used to reserve OS resources for OpenTP1 internal control. Executing the command reserves resources and stores them under an OpenTP1 directory. How many resources to reserve is determined through analysis of the system definition information under the OpenTP1 directory.

If the value of the `prc_process_count` operand in the process service definition is changed, the `dcsetup` command must be executed and then the `dcmakeup` command executed before using the `dcstart` command. If the `dcmakeup` command is not executed, it will be executed at OpenTP1 startup but much time may be required to reserve a sufficient number of resources.

7.1.9 OpenTP1 administrator environment setup

To execute the OpenTP1 commands, set up the following environment variables in the login environment:

`DCDIR`

Specify the OpenTP1 directory by a full path name. Use a maximum of 50 bytes for the directory name to be set in `DCDIR`. You cannot use symbolic links for OpenTP1 directories.

`DCCONFPATH`

Use a maximum of 246 bytes to specify the full path name of the directory for storing OpenTP1 definition files.

`DCUAPCONFPATH`

If you want to store the OpenTP1 user service definition files in a directory other than the directory specified in the `DCCONFPATH` environment variable, use a maximum of 246 bytes to specify the full name of the directory.

`PATH`

Add `$DCDIR/bin` to `PATH`.

7.1.10 Initialization of OpenTP1 file systems

The OpenTP1 administrator must initialize the OpenTP1 file systems using the `filmkfs` command.

7.1.11 Creation of OpenTP1 files

The OpenTP1 administrator must create and initialize the OpenTP1 files listed in the following table on the OpenTP1 file system that has been initialized by the `filmkfs` command.

Table 7-2: OpenTP1 files to be created and initialized, and operation commands used for creation and initialization

Node	OpenTP1 file		Operation command	OpenTP1 file system area	Remarks
Archive-journal target node	Status file		<code>stsinit</code>	for system	Mandatory in mode
	Archive journal file		<code>jnlinit</code>		
Archive-journal source node	Status file		<code>stsinit</code>	for system	Mandatory in mode
	Journal-related files	System journal file	<code>jnlinit</code>		
		Checkpoint dump file	<code>jnlinit</code>		
	Message queue file		<code>queinit</code>	for users	Optional in node [#]
	DAM file		<code>damload</code>		Optional in node
	TAM file		<code>tamcre</code>		Optional in node

Note

When using an ISAM file, see the manual *Indexed Sequential Access Method ISAM*.

#

Needed when MCF is used.

7.1.12 Creation of non-OpenTP1 files

Execution of OpenTP1 requires the following OS files in addition to the OpenTP1 files:

(1) Files created by the user

The files that must be created by the user are:

- User program file

This file stores execution-form UAPs. The archive-journal target node need not have this file.

- MCF communication process program file and MCF application startup process program file

These files store the executable programs of the MCF communication process and the MCF application startup process that you create to use the message control facility.

- Various definition files

These files store definition information. They must be created as text files using the OS text editor.

The above files are stored under the directories created by the user. The following table lists the files and directories that are created by the user. The directories `$(DCDIR)/aplib` and `$(DCDIR)/conf` are to be created during installation of OpenTP1.

Table 7-3: Files and directories created by the user

File		Directory	File name	File type
User program file		<code>\$(DCDIR)/aplib/#1</code>	Executable program name ^{#2}	Executable file
MCF communication process program file and MCF application startup process program file		<code>\$(DCDIR)/lib/servers/</code>	Executable program name	Executable file
Definition files	System environment definition	<code>\$(DCDIR)/conf</code>	<code>env</code>	Text file
	Other definitions	<code>\$(DCCONFPATH)/</code>	Definition file name	
	System service information definition	<code>\$(DCDIR)/lib/sysconf/</code>	System service information definition file name	

	File	Directory	File name	File type
	System service common information definition ^{#3}	\$DCDIR/lib/sysconf/	mcf	

#1

Can be changed by the process service definition.

#2

Can be changed by the user service definition.

#3

This file is created when OpenTP1 is installed or registered into the OS. However, you need to change the contents of the definition depending on the operation environment.

(2) Files and directories created at installation or cataloging into OS

The following table lists the files and directories created at installation or cataloging into OS.

Table 7-4: Files and directories created at installation or cataloging into OS

	Name	Directory	File name
OpenTP1 program files	OpenTP1 server	\$DCDIR/lib/servers/	..#
	Commands	\$DCDIR/bin/	--
	Header file	\$DCDIR/include/	--
	Archive file	\$DCDIR/lib/	--
Definition analysis files		\$DCDIR/lib/sysconf/	..#
		\$DCDIR/lib/sysdef/	--
Message object file		\$DCDIR/lib/	msgtxt, jmsgtxt
User program file directory		\$DCDIR/aplib/	None
Directory for definition files		\$DCDIR/conf/	None
System management information directory		\$DCDIR/etc/	--
Evacuated core file		\$DCDIR/spool/save/	None

Name	Directory	File name
Command log directory	\$DCDIR/spool/cmdlog/	--
Deadlock information file directory	\$DCDIR/spool/dclckinf/	None
Transaction information directory	\$DCDIR/spool/dctrninf/	None
Journal information directory	\$DCDIR/spool/dcjnlinf/ errinf/	None
Process service information directory	\$DCDIR/spool/dcprcinf/	None
Transaction journal directory	\$DCDIR/spool/dctjlinf/	--
Server recovery journal directory	\$DCDIR/spool/dcsj1/	--
Transaction control object storage directory	\$DCDIR/spool/trnrmcmd/ userobj/	dc_trn_allrm.0
Online tester directory	\$DCDIR/spool/uto/	None
Directory for system internal lock control	\$DCDIR/spool/olkfifs/	--
Directory for system internal synchronization control	\$DCDIR/spool/olkrfsf/	--

Legend:

--: Need not be specified by the user.

Note

Note that some directories and files are not used for archive-journal target nodes.

#

The file needs to be created when you use the MCF.

(3) Files and directories created during execution of OpenTP1

The following table lists the files and directories created at execution of OpenTP1.

Table 7-5: Files and directories created at execution of OpenTP1 (when using the multi-node facility)

Name	Directory	File name	Node 1 ^{#1}	Node 2 ^{#2}
Message log file	\$DCDIR/spool/	dclog1 or dclog2	Y	Y
MCF trace file ^{#3}	\$DCDIR/spool/	mcf ^{t0} XXZZ ^{#4}	N	Y
Scheduling queue information file	\$DCDIR/spool/	scdqid1 or scdqid2	N	Y

Name	Directory	File name	Node 1 ^{#1}	Node 2 ^{#2}
RPC trace file	\$DCDIR/spool/	rpctrl or rpctr2 ^{#5}	Y	Y
Trace information dump file	\$DCDIR/spool/save/	dcmrtdpl or dcmrtdp2	Y	Y
Performance verification trace information file	\$DCDIR/spool/dcprfinf/	prf_001, prf_002, prf_003	Y	Y
Backup file for performance verification trace information file	\$DCDIR/spool/save/	prf_nnn.bk1 ^{#6} and prf_nnn.bk2 ^{#6}	Y	Y
XAR performance verification trace information file	\$DCDIR/spool/dcxarinf/	_xr_nnn ^{#7}	Y	Y
Backup file for XAR performance verification trace information file	\$DCDIR/spool/save/	_xr_nnn.bk1 ^{#6} and _xr_nnn.bk2 ^{#6}	Y	Y
JNL performance verification trace information file	\$DCDIR/spool/dcjnlinf/prfinf/	_jl_nnn ^{#7}	Y	Y
Backup file for JNL performance verification trace information file	\$DCDIR/spool/save/	_jl_nnn.bk1 ^{#6} and _jl_nnn.bk2 ^{#6}	Y	Y
LCK performance verification trace information file	\$DCDIR/spool/dclckinf/prf/	_lk_nnn ^{#7}	Y	Y
Backup file for LCK performance verification trace information file	\$DCDIR/spool/save/	_lk_nnn.bk1 ^{#6} and _lk_nnn.bk2 ^{#6}	Y	Y
MCF performance verification trace information file	\$DCDIR/spool/dcmcfinf/	_mc_nnn ^{#7}	Y	Y
Backup file for MCF performance verification trace information file	\$DCDIR/spool/save/	_mc_nnn.bk1 ^{#6} and _mc_nnn.bk2 ^{#6}	Y	Y

7. Operations Under the Multi-node Facility

Name	Directory	File name	Node 1 ^{#1}	Node 2 ^{#2}
TRN event trace information file	\$DCDIR/spool/dctrninf/ trace/prf/	_tr_nnn ^{#7}	Y	Y
Backup file for TRN event trace information file	\$DCDIR/spool/save/	_tr_nnn.bk1 ^{#6} and _tr_nnn.bk2 ^{#6}	Y	Y
NAM event trace information file	\$DCDIR/spool/dcnaminf/	_nm_001, _nm_002, and _nm_003	Y	Y
Backup file for NAM event trace information file	\$DCDIR/spool/save/	_nm_nnn.bk1 ^{#6} and _nm_nnn.bk2 ^{#6}	Y	Y
Process service event trace information file	\$DCDIR/spool/dcprcinf/	_pr_001, _pr_002, and _pr_003	Y	Y
Backup file for process service event trace information file	\$DCDIR/spool/save/	_pr_nnn.bk1 ^{#6} and _pr_nnn.bk2 ^{#6}	Y	Y
FIL event trace information file	\$DCDIR/spool/dcfilinf/	_fl_001, _fl_002 and _fl_003	Y	Y
Backup file for FIL event trace information file	\$DCDIR/spool/save/	_fl_nnn.bk1 ^{#6} and _fl_nnn.bk2 ^{#6}	Y	Y
RTS log file	\$DCDIR/spool/dcrtsinf/ #8	rtslog[1-10] ^{#8, #9}	Y	Y
MCF statistics file	\$DCDIR/spool/#10	mcfstc ^{#10}	N	Y
Current working directory ^{#11}	\$DCDIR/tmp/home/ server-name.ID#12/	--	Y	Y
UAP trace data file	\$DCDIR/tmp/home/ server-name.ID/	dcuat.map ^{#13}	Y	Y
Backup file for UAP trace data file	\$DCDIR/spool/save/trc/ #14	server-namen.uatmap ^{#15} and server-name_n.uatmap ^{#15}	Y	Y

Name	Directory	File name	Node 1 ^{#1}	Node 2 ^{#2}
Shared memory dump file	\$DCDIR/spool/	Linux: shmdump [1-3] .gz, and shmdump.XXX ^{#16} [1-3] .gz Other operating systems: shmdump [1-3] .Z, and shmdump.XXX ^{#16} [1-3] .Z	Y	Y
Core file	\$DCDIR/spool/save/	server-namen ^{#17}	Y	Y
Deadlock and timeout information files	\$DCDIR/spool/dclckinf/	file-name ^{#18}	N	Y
MCF dump file ^{#3}	\$DCDIR/spool/	mcfdkAXXNN ^{#19}	N	Y
MCF common memory dump file ^{#3}	\$DCDIR/spool/	mcfSAXXY ^{#18}	N	Y
Undetermined transaction information file	\$DCDIR/spool/dctrninf/	file-name ^{#21}	N	Y
Invalid journal information file for archive journal node	\$DCDIR/spool/dcjnlinf/errinf/	an_x ^{#22}	Y	N
Invalid journal information file for archive-journal source node	\$DCDIR/spool/dcjnlinf/errinf/	r_n ^{#23}	N	Y
Copy file of input/output queue contents	\$DCDIR/spool/	file-name ^{#24}	N	Y
UAP trace edit/output file	\$DCDIR/spool/save/	server-nameN ^{#15} .uat	N	Y
OpenTP1 debug information file	\$DCDIR/spool/save/	server-nameN ^{#15} .deb	Y	Y

Legend:

Y: Created.

N: Not created.

--: Need not be specified by the user.

#1

Archive-journal target node

#2

Archive-journal source node

#3

Required only when MCF is used.

#4

AXZZ:

AXX: MCF identifier

A: MCF manager process identifier specified in the *id* operand of *mcfmenv* -*m* in the MCF manager definition

XX: MCF communication process identifier specified in *mcfteenv* -*s* in the MCF communication configuration definition

ZZ: Trace swap file identifier

#5

The file name can be changed in the system common definition.

#6

nnn: Value corresponding to the name of the backup source file.

#7

nnn: Value in the range of 001 to the value specified in the *prf_file_count* operand in the definitions listed below:

- Performance verification trace: Performance verification trace definition
- XAR performance verification trace: XAR performance verification trace definition
- JNL performance verification trace: JNL performance verification trace definition
- LCK performance verification trace: LCK performance verification trace definition
- MCF performance verification trace: MCF performance verification trace definition
- TRN event trace: TRN event trace definition

#8

If you do not specify the `rts_log_file_name` operand in the real-time statistics service definition, RTS log files are created in this directory.

#9

Files are created for the number of files specified in the `rts_log_file_count` operand in the real-time statistics service definition.

#10

The `mcfstc` file is created in the `$DCDIR/spool/` directory if you do not specify the output file name in the operation command (`mcfstats`).

#11

You can use the `prc_current_work_path` operand in the system common definition to change the current working directory.

#12

For user environment setting commands, `_usrCmd` is displayed as the server name.

"ID" is a serial number not less than 1. ID" is not added to operation commands (`dcstart`, `dcstop`, `dcsvstart`, and `dcsvstop`) and user environment setting commands.

#13

This file is created only when `Y` is specified in the `uap_trace_file_put` operand. This operand can be specified in one of the following definitions:

- System common definition
- User service default definition
- User service definition

#14

When the `prc_coresave_path` operand of the process service definition is specified, the directory is `(prc_coresave_path-value)/trc/`.

#15

n: File serial number (1-3)

#16

XXX: Resource manager name (`dam`, `tam`, `ist`, or `ism`)

#17

n

Core file serial number (1-3)

No serial number is assigned to a core file that is output when OpenTP1 is forced to stop (that is, when the `dcsvstop -df` command or `prckill` command is executed or real time monitoring overtime is reached). A process service core file may be acquired in the directory under the name *core*. If the user environment setup command terminates abnormally, the core file is saved under the name `_usrcmd` followed by a sequential number (1-3).

#18

The file name is determined on the basis of the deadlock detection date and time. The length of the file name depends on whether the date is one digit or two digits.

Example:

6: 29: 56 on October 3: Oct3062956

18: 06: 00 on October 10: Oct10180600

#19

KAXXNN:

K: Process type

AXX: MCF identifier

A: MCF manager process identifier specified in the `id` operand of `mcfmenv -m` in the MCF manager definition

XX: MCF communication process identifier specified in `mcfteenv -s` in the MCF communication configuration definition

NN: Dump serial number (01-99)

#20

AXXY:

AXX: MCF identifier

A: MCF manager process identifier specified in the `id` operand of `mcfmenv -m` in the MCF manager definition.

XX: MCF communication process identifier specified in `mcfteenv -s` in the MCF communication configuration definition.

Y: File serial number (1-3)

#21

rl+transaction service start time (8 hexadecimal characters)

#22

n: Resource group serial number of the journal in which invalid data was detected

This serial number is assigned sequentially to the resource groups specified in the `-a` option of the `jnldfs` command of the global archive journal service definition.

X: Generation number of the journal in which invalid data was detected

#23

n

Generation number of the journal in which incorrect data was detected

#24

File name specified by the `mcfdmpqu` command (to copy the contents of input/output queues)

In addition to the files listed in these tables, there are files for OpenTP1 internal processing under the directories `$DCDIR/spool/` and `$DCDIR/tmp/`. This must not be changed the user's discretion. All files for OpenTP1 internal processing under `$DCDIR/tmp/` are cleared when OpenTP1 starts.

7.2 OpenTP1 startup and termination

7.2.1 Startup

The `dcmstart` command enables one OpenTP1 node to start more than one OpenTP1 node that configures multinode areas or subareas.

Executing this command starts the following OpenTP1 nodes in normal startup mode or restart mode:

- All OpenTP1 nodes that configure multinode areas to which the OpenTP1 node corresponding to the command input environment variable `DCDIR` belongs
- All OpenTP1 nodes that configure the specified multinode subareas
- OpenTP1 nodes with the specified node identifiers

When the global archive journal facility is used, the archive-journal target node is started by the `dcmstart` or `dcmstart` command as with other nodes. The global archive journal service must be started before other nodes; otherwise, the global archive journal service started later sometimes cannot perform archive processing.

7.2.2 Termination

(1) Termination modes

(a) Normal termination

One OpenTP1 node can terminate all OpenTP1 nodes configuring multinode areas or subareas in normal termination mode. All journals at the OpenTP1 node that configures a multinode area or subarea are archived.

OpenTP1 nodes will be started in normal termination mode next time. Journals following the last archive file when OpenTP1 terminated will not be archived.

(b) Planned termination

One OpenTP1 node can terminate all other OpenTP1 nodes configuring multinode areas or subareas in normal termination mode or planned termination mode. All journals at the OpenTP1 node configuring a multinode area or subarea are archived.

OpenTP1 will be started in restart mode next time. Journals will be archived following the last archive journal file when OpenTP1 terminated.

(c) Forced termination

One OpenTP1 node can terminate all other OpenTP1 nodes configuring multinode areas or subareas in forced termination mode.

OpenTP1 will be started in restart mode next time. Journals will be archived following the last archive journal file when OpenTP1 terminated.

(d) Abnormal termination

If an error occurs, one OpenTP1 node will terminate OpenTP1 nodes configuring multinode areas or subareas in abnormal termination mode.

OpenTP1 will be started in restart mode next time. Journals will be archived following the last archive journal file when OpenTP1 terminated.

(2) Termination methods

The `dcmstop` command enables one OpenTP1 command to terminate more than one OpenTP1 node that configures multinode areas or subareas. Executing this command terminates the following OpenTP1 nodes:

- All OpenTP1 nodes that configure multinode areas to which the OpenTP1 node corresponding to the command input environment variable `DCDIR` belongs
- All OpenTP1 nodes that configure the specified multinode subareas
- OpenTP1 nodes with the specified node identifiers

The termination modes and the operation commands used for termination are:

Normal termination

```
dcmstop
```

Planned termination

```
dcmstop -a or dcmstop -b
```

Forced termination

```
dcmstop -f
```

The `dclistop` command is used to terminate individual OpenTP1 nodes. The termination modes and the operation commands used for termination are:

Normal termination

```
dclistop
```

Forced normal termination

```
dclistop -n
```

Planned termination

```
dclistop -a or dclistop -b
```

Forced termination

```
dclistop -f
```

Under the global archive journal facility, the archive-journal target node terminates with the `dclistop` or `dcmstop` command as with other nodes. In this case, however, the

archive-journal target node must be terminated concurrently with or after any other nodes. If the `dcstop` or `dcmstop` command is executed before the other nodes, the archive-journal target node awaits for the other nodes to terminate. The maximum waiting time (timeout value) must be specified in `jnl_arc_terminate_timeout` of the global archive journal service definition. For details, see the manual *OpenTP1 System Definition*.

The archive-journal target node can terminate normally only when all other nodes have terminated normally. The user is recommended to specify a termination mode for the archive-journal target node after verifying that all other nodes have terminated normally.

The archive-journal target node remains connected in the following cases:

- The OpenTP1 `dcstop` command (termination command) is executed in the archive-journal source node before that node is connected with the archive-journal target node (message KFCA04130-I is not output).
- The `jnl_arc_terminate_timeout` operand (timeout with the global archive journal service) is set for the archive-journal source node and a timeout occurs when the archive-journal source node is terminated (message KFCA04148-W is output).

To release the connection, execute the `jnlardis` command from the archive-journal target node.

7.3 Display of OpenTP1 node status

The `dcndls` command displays the status of each OpenTP1 node that configures multinode areas. The following status can be displayed depending on the specification of options:

- Status of all OpenTP1 nodes configuring multinode areas or subareas
- Status of the specified OpenTP1 node

7.4 Global journal operations

7.4.1 Configuration, creation, and definition of archive journal files

The global journal service facility can use up to 16 different resource groups depending on the node to be archived or its service. One resource group is to be specified in one archive journal service definition.

The archive journal service definition specifies the correspondence among physical files, logical file groups that control the physical files, and element files.

A file group consists of logical files that are used by OpenTP1 to perform journal generation control. Swapping of journal files in online mode takes place on a file group basis. The user can specify 2-256 file groups for one resource group, giving a desired name to each file group. The name must not be duplicated, however.

Element files are logical files that are dispersed when the parallel access facility is used for archive journal files. Online-mode journal file access is distributed to element files, thus dispersing the load on disk.

To use the parallel access facility, the user must specify `jnl_max_file_dispersion=2` or more and `jnl_min_file_dispersion` of the archive journal service definition. One file group can accommodate 2-8 element files, each given a user-selected name. (The name must not be duplicated, however.) Note that, since a unique number of dispersions is used for resource groups, the same number of element files must be specified for each file group.

When using no parallel access facility, the user need not specify element files. OpenTP1 perform processing as if one file group were one element file.

An archive journal file can be duplicated. To implement a duplicated file configuration, two physical files, one for system A and the other for system B, must be specified for one element file. Journals of the same contents are written in the two physical files. Duplication requires `jnl_dual=Y` to be specified in the archive journal service definition. In a duplicated file system, there may be the case where only one system can be used; `jnl_singleoperation` must be specified to select whether one-system operation is to be available or unavailable.

For a non-duplicated archive journal file configuration, one physical file is to be specified for one element file. Note that, since whether to duplicate is unique for resource groups, the same number of physical files must be specified for each element file.

Before starting OpenTP1, the user must execute the `jnlinit` command to create a physical file on the OpenTP1 file system. The physical file must have the same name as that specified in the archive journal service definition.

After the physical file is created, the correspondence between the physical file and the element file/file group/resource group must be specified in the archive journal service definition.

For the archive journal service definition, see the manual *OpenTPI System Definition*.

For details about estimating the size of an archive journal file, see *H.5 Estimating the size of an archive journal file*.

7.4.2 Use of archive journal files

An archive journal file contains information identical to the contents of system journal files of many other nodes.

Of the physical files that make up an element file, more than one file may be opened and this status is called *element file opened*. If no physical files are opened, the status is called *element file closed*.

Of the physical files that make up an element file, more files than necessary may be opened and this status is called *element file available*.

If more files than necessary are not opened, the status is called *element file unavailable*. The number of required physical files depends on the specification in the archive journal service definition.

- jnl_dual=N: 1
- jnl_dual=Y and jnl_singleoperation=Y: 1
- jnl_dual=Y and jnl_singleoperation=N: 2

The following table shows the relationship between file group status and physical file status.

Table 7-6: Relationship between file group status and physical file status

File group status			Physical file open/close status	
Usable/unusable status		Open/close status	System A	System B
One-system operation available	One-system operation unavailable			
Y	Y	Open	Open	Open
Y	--	Open	Open	Close
Y	--	Open	Close	Open
--	--	Close	Close	Close

Legend:

Y: Usable

--: Unusable

Of the element files that make up a file group, more than one element file may be opened and this status is called *file group opened*. If no element files are opened, the status is called *file group closed*.

Of the element files that make up a file group, more files than necessary may be opened is called *file group available*. If more files than necessary are not opened, the status is called *file group unavailable*.

The number of required element files depends on the specification in the archive journal service definition.

- Parallel access facility not used: 1
- Parallel access facility used: Minimum number of dispersions for parallel access

OpenTP1 manages online-mode archive journal file groups under three status categories.

- Current status

A file group in available status, to which journals can be output currently. There is only one file group that stays in this status.

- Standby status

File groups in available status, to which journals are not to be output currently but which stand by for future use as current-status file groups. This status is further divided into two categories:

- Swappable status

Standby-status file groups that have been unloaded (that is, copied onto a file with the `jnlunlfg` command). Each of them is ready for immediate use as a current-status file group when the next swapping occurs.

- Nonswappable status

Standby-status file groups that are waiting for unloading. Each of them is not ready for immediate use as a current-status file group.

- Reserved status

File groups in unavailable status.

More than one file group is needed except for reserved status.

When OpenTP1 is normally started, all file groups with ONL specified in the archive journal service definition are opened. Out of the opened file groups, the first group plays a role as a current-status file group and the rest stands by. Any file group that could not be opened or for which ONL has not been specified is placed in reserved

status. Upon restart of OpenTP1, the last current-status file group is inherited.

Journals are output to the current-status file group. If the file group becomes full, it is swapped with the standby-status file group specified second in the archive journal service definition. In a duplicated system, swapping occurs if the current-status file group of either system becomes full. If all file groups of the system journal file become full, journal outputting wraps around to the first file group.

7.4.3 Unloading of archive journal files

(1) Unloading

A file group is placed on unload wait status if it is swapped and stays in standby status by reasons of becoming full, an error, or operation command execution. The unload wait status means that journals are stored until the user unloads them in preparations for recovery of a DAM file or use by the user. The file group in unload wait status cannot be used as a current-status file group.

An unload-wait-status file group can be placed on unload completion status by using the `jnlunlfg` command. The unload completion status means that journals are copied from an unload-wait-status file group with the `jnlunlfg` command, or that journals are discarded with the `jnlchgfg` command.

The file created by unloading with the `jnlunlfg` command is called a global archive unload journal file.

The `jnlunlfg` command with its `-f` option specified enables a file group to be unloaded without checking its status, and can be used to unload file groups staying in current status or in unload completion status. Note that a file group in unload completion status must be closed with the `jnlclsfg` command before it is unloaded with the `jnlunlfg` command with its `-f` option specified.

Executing the `jnlunlfg` command with its `-f` option specified cannot change the status of a file group.

(2) Inhibiting unload check

A file group on unload waiting status can be allocated as the current one without being unloaded. This is called *inhibiting unload check on journal*. OpenTP1 usually checks the unload status of a file group, and allocates it as the current one only when it is on unload completion status. When unload status check is inhibited, unloading a file group is unnecessary because even a swappable file group on unload waiting status is allocated as the current one.

Specify `N` in `jnl_unload_check` in the archive journal service definition to use the facility for inhibiting unload check on journal. This specification can be changed when restarting the system.

Before unloading an online file group for testing, close the file group by using the `jnlclsfg` command. If you do not unload an online file group before executing the

`jnlclsfg` command, the file group might change to the *current* status, and the data in the file group might become corrupted. In this case, OpenTP1 interrupts unloading because the file group contains both data being unloaded and new data written for the current file group, but does not exit online mode.

The commands that input unloaded-journals file cannot be used unless unloading is executed at least once. Table 7-7 lists these commands.

Table 7-7: Commands that input unload journal files

	Function	Command
Management of journal files	Journal integration for file recovery	<code>jnlcolc</code>
	Copying of unloaded-journals file	<code>jnlcopy</code>
	Editing and output of unloaded-journals file	<code>jnledit</code>
	Output of MCF statistics	<code>jnlmcst</code>
	Recovery of journal files	<code>jnlmkrf</code>
	Output of records of unloaded-journals file	<code>jnlrput</code>
	Merging and chronological sorting of unloaded-journals file	<code>jnlstts</code>
	Output of statistics	<code>jnlstts</code>
DAM file management	Recovery of logical file	<code>damfrc</code>
TAM file management	Recovery of TAM file	<code>tamfrc</code>

Even if the use of the journal unload check inhibit facility is specified, executing the `jnlis` command displays unload wait status for the file group. However, that file group is allocated as the current one.

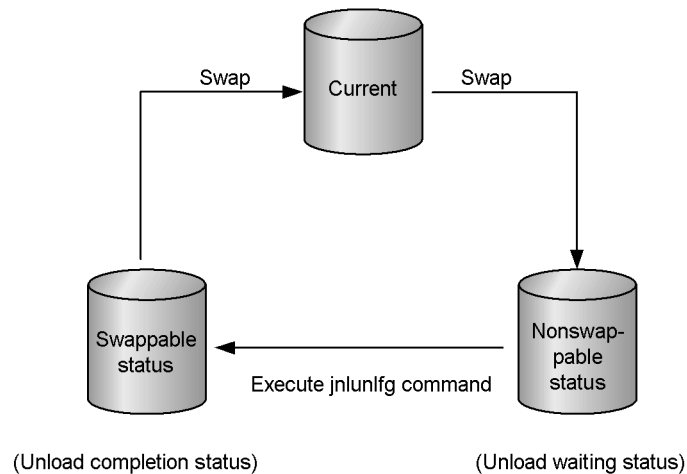
7.4.4 Reuse of archive journal files

To reuse a file group that has been swapped and stays in standby status because of becoming full or operation command execution, the file group must be in unload completion status.

When N is specified in `jnl_unload_check` (facility for inhibiting unload check on journal) in the system journal service definition, the file group is swapped without checking its status.

The following figure shows status changes of an online archive journal file.

Figure 7-1: Status changes of online archive journal file



When you use the global archive journal in the following cases, the file status of the last file group may not become "archived" even though all of the journals are already archived:

- `jnl_arc_terminate_check=N` is specified in the system journal service definition.
- A value other than 0 is specified for `jnl_arc_terminate_timeout` in the journal service definition and the archive service is canceled during processing due to a timeout (message KFCA04148-W is output at the cancellation with reason code 2008).

To change the status of the file to archived, use the `jnlchgfg` command.

7.4.5 Display of archive journal file information

The `jnlis` command displays archive journal file information. To display the archive journal file information that is read during a restart, use the `jnlrinfg` command. The contents that are displayed include the file group name, file type, and file group status.

7.4.6 Display of archive status

The `jnlarls` command displays the relationship between resource groups for the global archive journal service and those for the journal service, and archive status.

7.4.7 Archive journal file open and close

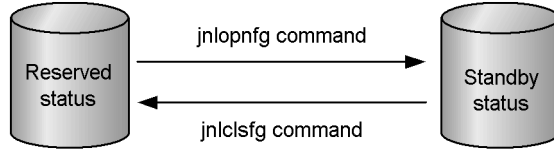
When a reserved-status file group is opened by the `jnlpnfg` command, it is placed on standby status.

When a standby-status file group is closed by the `jnlclsfg` command, it is placed on reserved status.

If there is only one standby-status file group, it cannot be closed.

The following figure shows status changes of an archive journal file caused by the open and close commands.

Figure 7-2: Status changes by open and close commands (archive journal file)



7.4.8 Change of archive journal file status

The `jnlchgfg` command forces only the status of a standby file group now in unload wait status into unload completion without actually unloading it. Note that, if a file group is forced into unload completion status, journals will not be copied.

A file group now in unload completion status can be used as a current-status group at the next swapping if it stays in overwriteable status.

7.4.9 Swapping of archive journal files

The user may want to unload or edit the current-status file group as soon as possible. In such a case, the user can execute the `jnlswpfg` command to swap the file group. If there is no swap target file group, an error occurs and the origin file group cannot be swapped.

7.4.10 Nonswappable target file group

The user is required to make a standby-status file group available so that it can be readily used as a swap target. When swapping of a file group is needed because it becomes full or an error occurs during output of journals, OpenTP1 will terminate abnormally if there is no standby-status file group ready for use as a swap target.

If OpenTP1 terminates abnormally because of the absence of a swap target file group, it can be restarted by executing the `jnlunlfg` command that unloads the file group and removes an error with any file group. If an attempt to restart OpenTP1 fails, specify a new file group with ONL specified in the archive journal service definition and try restarting again.

If the archive source node goes down while the global archive journal service is running and the system journal has no swap target, perform the following:

- To continue the global archive journal service, specify a new file group with ONL in the system journal service definition and restart the archive source node.

- To stop the global archive journal service, use the `jnlunlfg` command to unload all system journal files and then restart the archive source node.

7.4.11 Status changes of archive journal files

The following table shows status changes of an online archive journal file.

Table 7-8: Status changes of online archive journal file

Sts	Current/standby/reserved				Crt	Standby		Reserved					
	Unload completion/wait				--	Wt	Cp	Wt	Cp				
	Status number				1	2	3	4	5				
Evt	Swap due to no space remaining in current file group				→ 2	--	→ 1	--	--				
	Swap due to error with current file group	PAF not used	Error with non-duplicated system, or errors with both systems A and B		→ 4	--	→ 1	--	--				
			Error with either system A or B		One system operation available	→ 2	--	→ 1	--	--			
			One system operation not available			→ 4	--	→ 1	--	--			
	Swap due to error with current file group	PAF used	Error with non-duplicated system, or errors with both systems A and B		After affected physical file is shut down	Usable element files ≥ minimum number of dispersions	→ 2	--	→ 1	--	--		
			Error with either system A or B				One system operation not available	Usable element files < minimum number of dispersions	→ 4	--	→ 1	--	--
			One system operation available						→ 2	--	→ 1	--	--
	Swap with <code>jnlswpfg</code>				→ 2	--	→ 1	--	--				
	Execute <code>jnlunlfg</code>				--	→ 3	--	→ 5	--				
	Execute <code>jnlchgfg</code>				--	→ 3	--	→ 5	--				
	Execute <code>jnl opnfg[#]</code>				--	--	--	→ 2	→ 3				
	Execute <code>jnlclsfg[#]</code>				--	→ 4	→ 5	--	--				

Legend:

- Sts: Status
- Evt: Event
- Crt: Current
- Wt: Waiting
- Cp: Completed
- PAF: Parallel access facility
- : Cannot occur, or no status change
- *n*: Target status number

Note

All events are assumed to be normally terminated.

OpenTP1 manages the status of archive journal file groups only when the global archive journal service is in operation. OpenTP1 does not manage the status (unload completion/wait) of closed file groups. Therefore, executing the `jnlunlfg` or `jnlchgfg` command over closed file groups does not change their status. If the `jnlpnfg` and `jnlis` commands are executed in that order following the `jnlunlfg` or `jnlchgfg` command, the current status of files is displayed.

#

Execute when the `-a` and `-b` options are not specified.

7.4.12 Chronological sorting and merging of global archive unload journal files and unload journal files

The `jnlisort` command enables the contents of a global archive unload journal file and an unload journal file to be sorted and merged chronologically. The result of sorting and merging is output in the format of a global archive unload journal file.

Files can also be sorted and merged in the following ways:

- The journals at all nodes are sorted and merged according to the time they were output to the system journal file of an archive-journal source node.
- Only the journals of specific nodes are sorted and merged.
- Of all nodes, only the journals relating to particular global transactions are extracted, then sorted and merged according to the time they were output to the system journal file.

7.4.13 Editing and output of global archive unload journal files

The `jnledit` command enables the contents of a global archive unload journal file to be edited and output in text format so that the user can see and check them. The journals are edited on a block or record basis.

Executing the `jnledit` command outputs journals from all archive-journal source nodes. Journals can also be edited and output in the following ways:

- The journals at all nodes are sorted, then edited and output according to the time they were output to the system journal file of an archive-journal source node.
- Only the journals of specific nodes are edited and output.
- Of all nodes, only the journals relating to particular global transactions are extracted, then sorted, edited and output according to the time they were output to the system journal file.

7.4.14 Output of records of global archive unload journal files

The `jnlrput` command enables the contents of a global archive unload journal file to be output in binary format so that the user can process them using his own programs.

Executing the `jnlrput` command outputs journals from all archive-journal source nodes. Journals can also be output in the following ways:

- The journals at all nodes are sorted, then output according to the time they were output to the system journal file of an archive-journal source node.
- Only the journals of specific nodes are output.
- Of all nodes, only the journals relating to particular global transactions are extracted, then sorted and output according to the time they were output to the system journal file.

7.4.15 Output of uptime statistics

The `jnlstts` or `jnlmest` command collects uptime statistics at all archive-journal source nodes from a global archive unload journal file, and then edits and outputs them.

The information of all the archive-journal source nodes can be output. In addition, the information can be output in the following way by entering the result of the `jnlsort` command:

- Only the journal information of specific nodes is output.

7.4.16 Journal integration for file recovery

The `jnlcolc` command extracts journals needed for recovery of the files (DAM, TAM and ISAM) at all archive-journal source nodes from a global archive unload journal file. A file created by the `jnlcolc` command is called an integrating journal

file. When a DAM, TAM or ISAM file is recovered, using an integrating journal file can reduce the time required for recovery processing.

7.4.17 Unloading of system journal files at archive-journal source node

Normally, system journal files at archive-journal source nodes need not be unloaded. Journals of an archive journal file are output at timing different from those of a system journal file at an archive-journal source node. If all journals are needed for recovery of a DAM, TAM or ISAM file, it is necessary to check for presence of journals that have not been archived, and then unload the system journal file from the archive-journal source node.

If an error prevents journals from being archived, it is required to unload system journal files from the archive-journal source node.

7.4.18 File recovery at archive-journal source node

If an error occurs with the DAM, TAM or ISAM file at an archive-journal source node, the file can be recovered by a backup file and journal. Use the `jnl1s` command and OS command to compare the size of the backup file with that of the journal needed for updating the defective file. If the size of the backup file is larger, recover the defective file at an archive-journal source node. If the size of the backup file is smaller, recover the defective file at an archive-journal target node. By doing so, a shorter time is required for transfer of the file, making it possible to recover the file more quickly. The procedure for recovering the file at each node is described below.

(1) Backup file larger than journal requirement for recovery

The backup file must be retained at an archive-journal source node before starting recovery. If the backup file was created in the offline mode, the run ID of the online journal service at the next archive-journal source node is required for recovery of the defective file. If the backup file was created in the online mode, the run ID of the online journal service at the archive-journal source node used at that time is required for recovery of the defective file. In the description that follows, this run ID is referred to as *R*.

To recover the defective file, take the following steps:

1. Execute the `jnlunlfg` command to unload the archive journal file needed for recovery. This archive journal file contains journals with run IDs of *R* or later. Check the run IDs by using the `jnlar1s` and `jnl1s` commands.
2. Execute the `jnlunlfg` command to unload all system journal files containing the journals that were not archived at the archive-journal source node.
3. Execute the `jnl1sort` command to integrate the journals from the global archive unload journal file referred to in step 1., using the node identifier and the run ID of *R* at the archive-journal source node as the key.

4. Use the OS command to transfer the integrated journals to the archive-journal source node.
5. Specify the transferred journal file and the unload journal file referred to in step 2. and then execute the `jnlsort` command to integrate more journals.
6. Restore the backup file in a DAM file or a TAM file. For DAM file, use the `damrstr` command to restore; for TAM file, use the `tamrstr` command.
7. Using the integrating journal file referred to in step 5., recover the file at the archive-journal source node. Use the `damfrc` command to recover a DAM file and the `tamfrc` command to recover a TAM file. For recovery of an ISAM file, see the manual *Indexed Sequential Access Method ISAM*.

(2) Backup file smaller than journal requirement for recovery

The backup file must be retained at an archive-journal target node before starting recovery. If the backup file was created in the offline mode, the run ID of the online journal service at the next archive-journal source node is required for recovery of the defective file. If the backup file was created in the online mode, the run ID of the online journal service at the archive-journal source node used at that time is required for recovery of the defective file. In the description that follows, this run ID is referred to as *R*.

To recover the defective file, take the following steps:

1. Execute the `jnlunlfg` command to unload the archive journal file needed for recovery. This archive journal file contains journals with run IDs of *R* or later. Check the run IDs by using the `jnlarls` and `jnlls` commands.
2. Execute the `jnlunlfg` command to unload all system journal files containing the journals that were not archived at the archive-journal source node.
3. Use an OS command to transfer the unload journal file referred to in step 2. to the archive-journal target node.
4. Specify the transferred unload journal file referred to in step 3. and the global archive unload journal file referred to in step 1. and then execute the `jnlsort` command to integrate journals, using the node identifier and run ID of *R* of the archive-journal source node as the key.
5. The archive journal node requires a physical file for the DAM file or TAM file on the OpenTP1 file system, to restore the backup file. If that physical file has not been created, use the `damload` command for DAM file or use the `tamcre` command for TAM file to create it. To restore the backup file in a DAM file or a TAM file, use the `damrstr` command for DAM file or use the `tamrstr` command for TAM file.
6. Using the integrating journal file referred to in step 4., recover the file at the archive-journal target node. Use the `damfrc` command to recover a DAM file and

the `tamfr` command to recover a TAM file. For recovery of an ISAM file, see the manual *Indexed Sequential File Management ISAM*.

7. Back up the restored DAM file or TAM file again.
8. Transfer the backed up file to the archive-journal source node using a command of the OS.
9. Restore the transferred backup file in the original DAM file or TAM file, using the `damrstr` command for DAM file or using the `tamrstr` command for TAM file.

7.4.19 Recovery of recovery journal files

If an error occurs with a recovery journal file, take the following steps to recover it:

1. Confirm the point of journals that is needed to execute the `jnlmkrf` command.
2. Extract the journals downstream of the point referred to in step 1. From the global archive unload journal file.
3. Using an OS command, transfer the file containing the extracted journals to the archive-journal source node.
4. Specify the transferred file and the unload journal file at the archive-journal source node, and execute the `jnlmkrf` command.

Chapter

8. MultiOpenTP1 Operation

This chapter explains the differences between multiOpenTP1 operation and normal OpenTP1 operation in units of nodes.

In the multiOpenTP1 configuration, individual OpenTP1s can be executed independently on the same computer.

This chapter contains the following sections.

- 8.1 MultiOpenTP1 environment setup
- 8.2 Operation command execution environment

8.1 MultiOpenTP1 environment setup

8.1.1 OpenTP1 administrator setup

There are two types of OpenTP1 administrator setup:

- Setting up a different OpenTP1 administrator for each OpenTP1
- Setting up a common OpenTP1 administrator

8.1.2 Creation of OpenTP1 directories

A separate OpenTP1 directory must be created for each OpenTP1.

If the same path name is specified for a user program by the `prcsvpath` command in the process service definitions, the user program file can be shared among multiple OpenTP1s.

When creating an OpenTP1 directory, remember the following:

- Specify a directory other than the root partition so that it is not compressed.
- Do not create an OpenTP1 directory on the remote file system or the directory linked symbolically.

8.1.3 Cataloging of OpenTP1s into OS

The `dcsetup` command with the OpenTP1 directory specified must be executed for each OpenTP1.

8.1.4 Initialization of OpenTP1 file systems

In a multiOpenTP1 configuration, a separate partition or file should be allocated to each OpenTP1 as its OpenTP1 file system.

If the same partition or file is to be shared among multiple OpenTP1s, allocation of the partition or file for the OpenTP1 file system and initialization of the OpenTP1 file system must be executed only once. They must not be executed separately for each OpenTP1 (if they are executed for each OpenTP1, existing files will be deleted).

8.1.5 User environment setup

A specific OpenTP1 can also be run in the multiOpenTP1 configuration by using the environment variables shown below. The environment variables must be defined for each login environment in the user startup files (`.profile`, `.login`).

`DCDIR`

Specifies the full path name of the OpenTP1 directory. Use a maximum of 50 bytes for the directory name to be set in `DCDIR`. You cannot use symbolic links for OpenTP1 directories.

DCCONFPATH

Use a maximum of 246 bytes to specify the full path name of the directory for storing OpenTP1 definitions.

DCUAPCONFPATH

If you want to store the OpenTP1 user service definition files in a directory other than the directory specified in the `DCCONFPATH` environment variable, use a maximum of 246 bytes to specify the full path name of the directory.

8.1.6 Changing a shared library

In multiOpenTP1 configuration, a shared library area resides on the memory of each OpenTP1. When the versions of two OpenTP1 systems are the same, one shared library can be shared by them (only for HP-UX). Since two OpenTP1 systems reference one shared library, memory consumption can be reduced.

To share a shared library by multiple OpenTP1 systems, match all the `SHLIB_PATH` environment variables of the OpenTP1 systems. The following shows the procedure.

(1) Changing `/etc/inittab`

1. Stop OpenTP1 to change `/etc/inittab`. If changing the setting while OpenTP1 is operating, the OpenTP1 system fails.

Before change

```
d1:2:respawn:env SHLIB_PATH=/BeTRAN/lib /BeTRAN/lib
/servers/prcd /BeTRAN
d2:2:respawn:env SHLIB_PATH=/BeTRAN1/lib /BeTRAN1/lib
/servers/prcd /BeTRAN1
```

After change (Changed parts are indicated with an underline.)

```
d1:2:respawn:env SHLIB_PATH=/BeTRAN/lib /BeTRAN/lib
/servers/prcd /BeTRAN
d2:2:respawn:env SHLIB_PATH=/BeTRAN1/lib /BeTRAN1/lib
/servers/prcd /BeTRAN1
```

2. Enter the `dcreset` command to incorporate the changes in `/etc/inittab` into the OS.

Executing `dcsetup -d` deletes the contents of `/etc/inittab`. After `dcsetup -d` is executed, define the contents of `/etc/inittab` using `dcsetup` then change `etc/inittab` again.

(2) Changing the environment variables

Change the environment variables that set the command issuance environment.

Before change

Environment variable	OpenTP1 A	OpenTP1 B
DCDIR	/BeTRAN	/BeTRAN1
DCCONFPATH	/BeTRAN/conf	/BeTRAN1/conf
SHLIB_PATH	/BeTRAN/lib	/BeTRAN1/lib

After change (Changed parts are indicated with an underline.)

Environment variable	OpenTP1 A	OpenTP1 B
DCDIR	/BeTRAN	/BeTRAN1
DCCONFPATH	/BeTRAN/conf	/BeTRAN1/conf
SHLIB_PATH	/BeTRAN/lib	/BeTRAN1/lib

If the versions of the OpenTP1 systems are different, the operation cannot be assured. If the directory specified by SHLIB_PATH does not contain a library, the OS suppresses activation of the process.

(3) Changing the definition

If SHLIB_PATH is defined in the putenv format in the system definition, define it as in paragraph (2).

8.1.7 Creation of OpenTP1 files

The OpenTP1 files listed in Table 8-1 must be created and initialized by the operation commands used to initialize the various files.

Table 8-1: Initialization of OpenTP1 files

OpenTP1 file		Operation command
Status file		stsinit
Journal files	System journal file	jnlinit
	Checkpoint dump file	
Message queue file		queinit
TP1/Message Queue queue file		mqainit
DAM file		damload
TAM file		tamcre

If the same partition is to be shared among multiple OpenTP1s, each OpenTP1 file

must be created and initialized in such a manner that it has a different name in each OpenTP1.

8.1.8 Creation of other files

A separate directory must be created for each OpenTP1 to store corresponding definition files. The OpenTP1 definition files must be created under an appropriate directory. For the definition information, see the manual *OpenTP1 System Definition*.

8.2 Operation command execution environment

Both the OpenTP1 administrator and the users who execute operation commands must define environment variables. An OpenTP1 directory to be subject to operation command processing is defined in `DCDIR`, and a definition storage directory is defined in `DCCONFPATH`.

The correspondence between an OpenTP1 and an operation command is determined by the defined `DCDIR`.

Under the `examples/tools/` directory, there are sample programs that allow the user to enter operation commands into the specified OpenTP1 without consciousness of OpenTP1 environment variables. For more information about the samples, see the manual *OpenTP1 Programming Guide*.

Chapter

9. Operations Under the System Switchover Facility

This chapter explains operations required when the system switchover facility is used, the differences in OpenTP1 operation from when the system switchover facility is not used.

For the system switchover facility, see the relevant HAmonitor manuals.

This chapter contains the following sections:

- 9.1 Preparations for using system switchover facility
- 9.2 Startup and termination
- 9.3 System switchover procedure
- 9.4 Standby user server
- 9.5 Operation commands

9.1 Preparations for using system switchover facility

The following contents must be the same between OpenTP1s using the system switchover facility:

- OpenTP1 system definition information
- User server execution-form program
- Transaction service control execution-form program[#] (if re-created with the `trnlncrm` command)
- OpenTP1 and related program product version
- OpenTP1 administrator environment (user ID, group ID, and environment variables)
- OpenTP1 directory full path name
- OpenTP1 file setup

The OpenTP1 files to be used under the system switchover facility must be allocated to the OpenTP1 file system that has been created on a character type special file on the common disk.

`ha_conf=Y` (exchange systems) must be specified in the system configuration definition. Note, however, that you cannot use the transaction recovery journal file facility when the system switchover facility is being used. Do not specify `trf_put=Y` in the user service definition or the user service default definition.

In order for OpenTP1 to use the system switchover facility, the same host name must be used in the running and standby systems. Also, the same IP address corresponding to the host name must be used in the two systems.

If the `my_host` operand in the system common definition is specified, OpenTP1 uses it. If the operand is not specified, OpenTP1 uses the standard host name of the computer on which OpenTP1 works.

#

You must also execute the `trnlncrm` command to make sure that the order of resource managers registered in OpenTP1 on the primary system is the same as the order on the secondary system.

The following table shows the combinations of host names OpenTP1 uses in the running and standby systems.

Table 9-1: Combinations of host names OpenTP1 uses in running and standby systems

Specification of OpenTP1 my_host operand in running and standby systems	Standard host name	
	Same	Different
Same	Y	Y
Different	N	N
None	Y	N

Legend:

Y: Valid

N: Invalid

9.2 Startup and termination

9.2.1 Startup

OpenTP1s both in the running system and the standby system are to be started as with those using no system switchover facility. The user must start both OpenTP1s.

The startup procedure differs depending on the contents specified in the system environment definition (`mode_conf`).

The user should preferably specify `AUTO` or `MANUAL1` so that OpenTP1s will automatically start after abnormal termination. `AUTO` or `MANUAL1` must be specified depending on the operation procedure because of the difference in starting after activation of the OS.

However, to use OpenTP1 in the standby system for postprocessing, for example, determination of an undecided transaction or management of the integrity of the database, for OpenTP1 in the running system, you need to specify the `-U` option in the `dcstart` command. In this case, do not specify automatic start.

Caution

If you specify `AUTO` or `MANUAL1` for the `mode_conf` operand in the system environment definition, OpenTP1 starts automatically when it ends abnormally. In this case, you cannot use OpenTP1 in the standby system for postprocessing. If you want to use OpenTP1 in the standby system for postprocessing, stop OpenTP1 and then specify the `-U` option in the `dcstart` command to execute OpenTP1. For details about starting OpenTP1 in the standby system by specifying the `-U` option in the `dcstart` command, see the manual *OpenTP1 Description*.

9.2.2 Termination

(1) Termination of OpenTP1 in running system

The `dcstop` command must be executed as with OpenTP1 using no system switchover facility. OpenTP1 in standby system also terminates with termination of OpenTP1 in running system.

(2) Termination of OpenTP1 in standby system

OpenTP1 in standby system terminates with termination of OpenTP1 in running system. One OpenTP1 in standby system can be terminated by executing the `dcstop` command with its `-f` option specified (forced termination). No other than the `-f` option can be specified.

By using an `HAMonitor` command, OpenTP1 in a standby system can be terminated asynchronously with OpenTP1 in the running system. For details about `HAMonitor`

commands, see the HAmonitor manual for your operating system.

9.3 System switchover procedure

If an error occurs with OpenTP1 in the online mode (in the running system), standby OpenTP1 can be selected to continue online operation. After system switchover, OpenTP1 in standby system now becomes active OpenTP1.

Three procedures are available for exchanging between standby and running OpenTP1s.

- Automatic system switchover
- Planned system switchover
- Grouped system switchover

Only the OpenTP1 files on the character type special file on a common disk can be inherited at system switchover.

9.3.1 Automatic system switchover

If an error occurs with OpenTP1 in the running system, OpenTP1 in standby system is automatically selected.

The following shows the errors that triggers the system switchover.

1. System error
 - Hardware error or cut of power of the system
 - Kernel error
 - HAmonitor error
 - RS-232C adapter error
 - System slowdown
2. Hardware error
 - Logical error for OpenTP1
 - Resource error that prevents continuing the system operation
 - Communication management (such as XNF/S) error
 - Delay of OpenTP1

9.3.2 Planned system switchover

When the user executes the HAmonitor planned system switchover command, OpenTP1 in the running system stops and OpenTP1 in standby system selected.

9.3.3 Grouped system switchover

When more than one OpenTP1 is operating in a running system, they can be switched over all at one time if an error occurs. This is called *grouped system switchover*.

The grouped system switchover can be performed by grouping multiple OpenTP1s. When OpenTP1s are grouped, if an error occurs on one of them, the group can be replaced with the standby group of OpenTP1s. Each OpenTP1 in a group can be either an OpenTP1 for which group switchover can be performed or an OpenTP1 for which group switchover cannot be performed.

The environment for the grouped system switchover is set using HAmonitor.

9.4 Standby user server

If `user_server_ha=Y` is specified in the system environment definition, only the user server processes specified in the user service configuration definition can be activated before the standby OpenTP1 (standby system) is placed on standby status. In this case, note the following:

- The user server awaits for the running system to be changed to the standby system in the `dc_rpc_open` function. This means that exchange to the standby OpenTP1 is impossible unless the `dc_rpc_open` function is issued.
- OpenTP1 inherits the previous user server configuration. If the user server specified in the user service configuration definition in the previous online operation has come out of the configuration after exchange to the standby OpenTP1, return value `DCRPCER_STANDBY_END` is returned even when the `dc_rpc_open` function is issued. When that value is returned, the user must terminate the current process. The process of terminating the user server starts after exchange to the standby OpenTP1; much time is required until the process is restarted. To avoid this, the user is recommended to specify a user server in the user service configuration definition that will not end before system termination.
- When the standby OpenTP1 terminates, return value `DCRPCER_STANDBY_END` is also returned if the `dc_rpc_open` function is issued. The user must terminate the current process.

9.5 Operation commands

The operation commands that can be used have the following restrictions:

- Operation commands executable in offline mode must be executed after termination of OpenTP1s both in the running and standby systems. The `dcstart` command can be executed, independently of either system status, if OpenTP1 for the machine is not in operation. For OpenTP1 in standby system, specification of the `-n` option of the `dcstart` command (forced normal termination) is ignored if the command is executed.
- An operation command executable in the online mode can be executed on online-mode OpenTP1 in the running system. Only the `dcstop` command with its `-f` option specified (forced termination) and the `jnlunlfg` command with its `-n` or `-f` option specified (unload journal files) can be executed for OpenTP1 in standby mode.

Chapter

10. Error Handling Procedures

This chapter classifies the OpenTP1 errors and explains the procedures for handling individual errors.

The error information collection procedure is also explained.

This chapter contains the following sections:

- 10.1 Error events and causes
- 10.2 OpenTP1 file errors
- 10.3 File errors
- 10.4 Communication errors
- 10.5 UAP errors
- 10.6 OpenTP1 errors
- 10.7 CPU errors
- 10.8 Information collection when error occurs
- 10.9 Information inherited during full recovery

10.1 Error events and causes

The following table lists the error events and their possible causes.

Table 10-1: Error events and causes

Event	Cause	Reference section for remedy
UAP cannot be started.	System definitions are invalid.	<i>10.5 UAP errors</i>
	Memory is inadequate.	
UAP does not terminate its processing.	Error in UAP.	
UAP terminates abnormally.	Error in UAP or OpenTP1.	
	UAP linkage is invalid.	
UAP deadlock occurs.	Looping of a lock request has occurred between UAPs.	
OpenTP1 cannot be started.	OpenTP1 has not been correctly installed or set up.	<i>10.6 OpenTP1 errors</i>
	OpenTP1 system definitions are invalid.	
	Memory or disk space is inadequate.	
	File required in order to start OpenTP1 is missing.	
	OS configuration is inappropriate to OpenTP1 execution environment.	
	TCP/IP environment has not been set up.	
	Hardware error (disk or LAN).	
	A user initialization command is invalid.	
	Process service processing is interrupted.	
Entering <code>dstop</code> command does not terminate OpenTP1.	UAP processing takes too long, or there is an error in the UAP processing, and the UAP does not terminate its processing.	
	At the time of normal termination, messages accumulated in the output queue are not output due to a communication error, etc.	

Event	Cause	Reference section for remedy
	The OpenTP1 at the archive-journal node does not terminate because the connected journal service does not terminate.	
	The OpenTP1 at the archive-journal source node does not terminate because the connection to the global archive journal cannot be released.	
OpenTP1 terminates abnormally.	OpenTP1 system service or user server was terminated by the kill command.	10.2 <i>OpenTP1 file errors</i>
	Error in OpenTP1.	
	Journal file group to be swapped in is missing.	
	There is no spare status file for A or B system.	
	Current status file for A or B system cannot be read.	
	Error occurred in MCF definition file.	
	Error occurred in MCF queue file.	
	Error in the user exit routine.	
	Error in the UAP.	
OpenTP1 operation command does not terminate normally.	Specification of an option or command argument is invalid.	10.6 <i>OpenTP1 errors</i>
	Environment required in order to execute the operation command has not been set up.	
	User is not authorized to execute the operation command.	
Response-wait overtime occurs for an OpenTP1 operation command.	Processing is not completed within the response wait time, because the OS workload is high.	
File I/O error occurs.	Error has occurred in the file.	10.2 <i>OpenTP1 file errors</i> and 10.3 <i>File errors</i>
File error message is output.		
Communication error message is output.	Error has occurred in the protocol.	10.4 <i>Communication errors</i>

10. Error Handling Procedures

Event	Cause	Reference section for remedy
	Error has occurred on the LAN.	

10.2 OpenTP1 file errors

This section explains the procedures for handling OpenTP1 file errors.

10.2.1 Status files

(1) *During online operation*

Errors that may occur in the online mode can be classified into errors occurring during write operations and errors occurring during read operations.

(a) **During write operations**

(i) Standby file exists

OpenTP1 swaps the current file out and the standby file in.

The erroneous status file must then be initialized by the following procedure:

1. With the `stsrn` command, delete the status file that has been placed on shutdown status due to the error.
2. Create a status file with the `stsinit` command.
3. Open the created status file with the `stsopen` command and place it on standby status.

(ii) No standby file

One-system operation available: If an error occurs with the current file in one system, processing continues with the normal system alone. If an error occurs with the normal system during one-system operation or if the normal system terminates abnormally during updating of a file, OpenTP1 can no longer be restarted. When the current file is put to one-system operation, the user must take either of the following two procedures:

- Swapping with a standby file

To swap the current file with a standby file, take these steps:

1. Open an invalid file with the `stsopen` command to provide a standby file.
2. Execute the `stsswap` command to swap the current file with the standby.

- Recovering the file of the erroneous system

To recover the contents of the file shut down due to an error, take these steps:

1. Delete the file of the shut-down system with the `stsrn` command.
2. Initialize the file with the `stsinit` command.
3. Open the initialized file with the `stsopen` command.

Once the `stsopen` command is executed, the contents of the file in the normal system are copied onto the file in the erroneous system. Upon completion of copying, the current-status determination time is collected and the file is recovered as a current file.

One-system operation unavailable

OpenTP1 terminates abnormally.

After a standby file is provided, OpenTP1 must be restarted, and the erroneous status file must be initialized by the following procedure:

1. With the `stsrn` command, delete the status file that has been placed on shutdown status due to an error.
2. Create a status file with the `stsinit` command.
3. Open the created status file with the `stsopen` command and place it on standby status.

(b) During read operations

A status file is read from the current file in the A system. If an error occurs while this file is being read, the error handling procedure will depend on whether or not the current file in the B system can be read.

(i) Current file in B system can be read

If there are standby files for the A and B systems, OpenTP1 copies into the standby files in both the A and the B system the contents of the current file in the B system.

If there are no standby files for the A and B systems, OpenTP1 terminates. After standby files are provided for the A and B systems, OpenTP1 must be restarted, and a new status file must be provided by the following procedure:

1. With the `stsrn` command, delete the physical file that has been placed on shutdown status due to the error.
2. Initialize the physical file in the A system with the `stsinit` command.
3. Open the initialized file with the `stsopen` command.

(ii) Current files in A and B systems cannot be read

OpenTP1 terminates.

OpenTP1 must be started by the procedure described below. Any processing that was underway cannot be inherited. DAM FRC or TAM FRC must be executed.

1. Initialize the erroneous status file with the `stsinit` command.
2. Forcibly execute normal startup of OpenTP1 with the `dcstart -n` command.

(2) During restart

The error handling procedures depend on specifications in the status service definitions.

(a) sts_initial_error_switch omitted or stop specified

If an error occurs in a status file, OpenTP1 terminates abnormally. If an error occurs in even one of the status files specified in the definition, OpenTP1 will not start. The error handling procedure will depend on the file resulting in the error.

(i) Error occurred in previous current file

If both current files in the A and B systems result in an error, OpenTP1 must be started by the procedure described below. Any processing that was underway cannot be inherited. DAM FRC or TAM FRC must be executed.

1. Initialize the erroneous status file with the `stsinit` command.
2. Forcibly execute normal startup of OpenTP1 with the `dcstart -n` command.

If one of the current files in the A or B system results in an error, OpenTP1 must be started by the following procedure:

1. Check that there is a standby file to swap in for the current file. If there is no standby file, create one with the `stsinit` command.
2. Change the specification of `sts_initial_error_switch` to `continue` in the status service definition.
3. Restart OpenTP1.

(ii) Error occurred in file other than previous current file

OpenTP1 must be restarted by the following procedure:

1. If a file entity resulted in the error, delete it with the `stsrn` command.
2. Initialize the status file with the `stsinit` command.
3. Restart OpenTP1.

(b) sts_initial_error_switch=continue or sts_initial_error_switch=excontinue specified

Even if an error occurs in a status file, processing continues. If an error occurs in the most recent current file in one of the systems, the contents of the error-free file in the other system will be copied into the standby files in both the A and the B system. The current file and the standby file are then swapped, and OpenTP1 is started. However, if there is no standby file for copying information, OpenTP1 terminates abnormally. When `excontinue` is specified an initialized file cannot be the active file.

When `sts_initial_error_switch=continue` is specified, OpenTP1 selects as

the current file the most recent file among those files that were opened during restart processing. If both systems have even one pair of files that cannot be opened, whether or not the selected current file is actually the previous current file cannot be determined by the status service. In such a case, the user can check whether or not the current file selected by OpenTP1 is the correct current file by specifying the name of the logical file used as the most recent current file during the previous online operation with `sts_last_active_file` in the status service definition. If the current file selected by OpenTP1 matches the logical file name specified with `sts_last_active_file`, startup processing can continue. If not, startup processing terminates. The user must take the same step as when both current files in the A and B systems result in an error. If `sts_last_active_file` is not specified and the current file cannot be determined, startup processing terminates.

If an error occurred with the current file of one system during the previous online operation, the user must specify a normal system in `sts_last_active_side` of the status service definition so that an incorrect file may not be selected at restart after one-system operation.

If only the system specified in `sts_last_active_side` can be opened at restart, it is swapped. If swapping fails, OpenTP1 terminates (one-system operation of the current file cannot be performed at start of the status service). If the system specified in `sts_last_active_side` cannot be opened, OpenTP1 stops.

If the files in the A and B systems placed on current status are different in contents, the system having the latest information is assumed to be the current system independently of the system specified in `sts_last_active_side`. The latest information is copied to the system that has old information and then OpenTP1 is restarted.

At restart of OpenTP1, the status service inherits only the status file error that occurred during its activation and the shutdown status of the current file with which an error occurred in the online mode. The service does not inherit the status opened with the `stsoopen` command or closed with the `stsclose` command in the online mode.

Furthermore, the status service does not identify the status of a status file when the `stsin` or `stsr` command is executed in the offline mode.

(3) During normal startup

Errors are handled in the same manner as during restart.

10.2.2 System journal files

(1) During online operation

(a) During write operations

(i) Swap-in file group exists

If an error occurs while a journal is being written into the current file group, the file groups are swapped, and journal data is output to the new current file group.

The current file group used before swapping is closed and placed on reserve status.

If application history needs to be retained, the file group that has been placed on reserve status due to the error must be unloaded with the `jnlunlfg` command.

If the file group cannot be unloaded, the steps described below must be taken. Because the journal cannot be retained in such a case, a backup copy must be made in advance, because DAM FRC or TAM FRC cannot be executed without the journal.

1. With the `jnlrm` command, delete the file group that cannot be unloaded.
2. Re-create the file group with the `jnlinit` command.
3. Open the file group with the `jnlopnfg` command.

(ii) No swap-in file group

If there is no overwritable standby file group that can be used as the swap-in file, OpenTP1 terminates abnormally. The following steps must then be taken:

1. Unload all file groups on unload wait status with the `jnlunlfg` command, and provide an overwritable standby file group.
2. Restart OpenTP1.

If an overwritable standby file group cannot be provided, the definition of a file group with ONL specified must be added to the system journal service definition, and then OpenTP1 must be restarted.

(b) During read operations

If dual system journal files are used, a journal is read from the current file in the A system. If an error occurs while the current file in the A system is being read, the current file is switched to the one in the B system in order to read the journal.

(2) During restart

(a) Recovery of journal output position

If an error occurs when the journal output position is recovered after the file group status has been recovered during restart, OpenTP1 terminates. The cause of the error must be eliminated, based on the reason code displayed with the error message, and then OpenTP1 must be restarted.

(b) Addition of a file group

A new file group can be added to the system journal service definition during full recovery.

(c) Swapping

If `jnl_rerun_swap=Y` is specified in the system journal service definition, the file

groups are swapped after the journal output position has been recovered during restart. Even when Y is not specified, if an error occurs in the file group that is assumed to be the previous current file group during journal output position recovery or if this file group has been unloaded, the file groups are also swapped after the journal output position has been recovered.

(d) Opening a reserve file

If `jnl_rerun_reserved_file_open=Y` is specified in the system journal service definition and the following two conditions are satisfied, a reserve file is opened automatically:

- None of the available file groups can be swapped, except for the current file group
- There is no available file group for swap-in because the size of the available system journal file was exceeded during restart.

10.2.3 Archive journal files

(1) During online operation

(a) During write operation

(i) Swap-in file group exists

If an error occurs while a journal is being written into the current file groups, the file groups are swapped, and journal data is output to the new current file group. The current file group used before swapping is closed and placed on reserve status. This file group must be unloaded with the `jnlunlfg` command.

If the file group cannot be unloaded, the steps described below must be taken. Because the journal file cannot be retained in such a case, a backup copy must be made in advance, as DAM FRC or TAM FRC cannot be executed without the journal.

1. With the `jnlrm` command, delete the file group that cannot be unloaded.
2. Re-create the file group with the `jnlinit` command.
3. Open the file group with the `jnlopnfg` command.

(ii) No swap-in file group

If there is no swap-in file, OpenTP1 terminates abnormally. In this case, however, the OpenTP1 at an archive-journal source node continues. The following steps must then be taken:

1. Unload all file groups on unload wait status with the `jnlunlfg` command.
2. Restart OpenTP1.

(b) During read operation

If dual system archive journal files are used, a journal is read from the current file in the A system. If an error occurs while the current file in the A system is being read, the current file is switched to the one in the B system in order to read the journal.

(2) During restart

If an error occurs when the journal output position is recovered after the file group status has been recovered during restart, reading journals is suspended and the process of restarting OpenTP1 continues. The journal output position must be determined on the basis of the journal that had been read until the error occurred.

Archiving is restarted with the lost journal or the journal that could not be read after completion of restart of OpenTP1.

10.2.4 Recovery journal files**(1) Transaction recovery journal files****(a) During online operation**

If an error occurs with a transaction recovery journal file, an error message is output. The `jnlmkrf` command must be executed according to that message. Executing this command enables OpenTP1 to recover the erroneous transaction recovery journal file. If OpenTP1 fails to recover it, the user must stop OpenTP1, execute the `damfrc` or `tamfrc` command to recover the resource manager, and then force OpenTP1 to start normally.

(b) During restart

If an error occurs with a transaction recovery journal file during restart of OpenTP1, the user must execute the `damfrc` or `tamfrc` command to recover the resource manager and force OpenTP1 to start normally.

(2) Server recovery journal files**(a) During restart**

If an error occurs with a server recovery journal file and a message (KFCA03015-E) is displayed, the user must execute the `jnlmkrf` command to recover the erroneous server recovery journal file. Recovering the file requires an unload journal of the generation indicated in the KFCA03015-E message and a journal of the current generation as well. The user must unload the system journal file of the current generation.

If the server recovery journal file cannot be recovered by executing the `jnlmkrf` command, the user must execute the `damfrc` or `tamfrc` command to recover the resource manager and force OpenTP1 to start normally.

10.2.5 Checkpoint dump files

(1) *During online operation*

(a) **Physical file re-creation**

If an error occurs while a checkpoint dump is being written, OpenTP1 outputs a message to the message log file, closes the erroneous file group, changes it to a reserve file group, reselects the output destination from the overwritable file groups, and outputs the checkpoint dump again.

If the corresponding file group is not placed on reserve status, the erroneous file group must be closed and changed with the `jnlclsfg` command to a reserve file group.

The file group placed on reserve status by the `jnlclsfg` command can be changed to an overwritable file group by the following procedure:

1. Delete the file group with the `jnlrm` command.
2. Re-create the file group with the `jnlinit` command.
3. Open the created file group with the `jnlopnfg` command.

(b) **Physical file reallocation**

If an error occurs on the physical file allocated with the `jnladdpf` command in the online mode, another physical file is allocated.

For dual-system operation, another file can be allocated by manipulating only either system A or system B. Note that the file group is temporarily placed in reserved status. Therefore, the contents of both of the physical files are invalidated. The file group cannot be used if either of the systems is disconnected.

The following shows the re-allocation procedure:

1. Close the file group where the error occurred.
Use the `jnlclsfg` command to close the file group and place it in reserved status.
2. Delete the physical file.
Use the `jnlde1pf` command to delete the physical file where the error occurred. The system (A or B) can be specified.
3. Allocate another physical file.
Use the `jnladdpf` command to allocate a normal physical file. The system (A or B) can be specified.
4. Open the file group.
Use the `jnlopnfg` command to open the file group and make it overwritable.

(c) Re-create the system where the error occurred

When one-system operation is available in a dual system configuration, if an error occurs on either of the systems for the checkpoint dump file, the system where the error occurred can be re-created to make both systems usable. This system re-creation can be executed only when the file group is overwriteable.

The following shows the re-creation procedure:

1. Close the system where the error occurred.

Use the `jnlclsfg` command to close the system where the error occurred. If this command is executed when the system has already been closed, a command error occurs.

Do not execute `jnlclsfg` to a normal system. If executed, the contents of the physical file become invalid.

2. Delete the system where the error occurred.

Use the `jnlrm` command to delete the physical file of the system where the error occurred.

3. Re-create the physical file.

Use the `jnlinit` command to re-initialize the physical file to be allocated to the system where the error occurred.

4. Open the physical file.

Use the `jnlpnfg` command to open the re-created physical file.

(2) During restart

When OpenTP1 is restarted, it opens all the file groups that were open during the previous OpenTP1.

(a) Non-duplicated checkpoint dump file configuration

Any file group that could not be opened or that was on reserve status during the previous online operation becomes a reserve file group.

(b) Duplicated checkpoint dump file configuration

- (i) Either system A or B file failed to open

If one-system operation is available, continue processing by opening the normal system. If one-system operation is unavailable, the file group is placed in reserved status after the read is completed or when the file group becomes overwriteable.

- (ii) Both systems A and B files opened successfully

Recover the system by reading from system A.

If the read from system A fails when one-system operation is available, system B

is used. If the read from system A fails when one-system operation is unavailable, the data is read from system B and after that (when the file group is non-overwritable, after it becomes overwritable), the file group is placed in reserved status.

If the checkpoint dump file required for recovery cannot be read due to an open error, etc., OpenTP1 outputs a message to the message log file and continues reading as much of the checkpoint dump file as possible.

No addition, change, or deletion can be made to the checkpoint dump service definition during restart.

10.2.6 DAM file

(1) DAM FRC procedure

If an error occurs in a DAM file, an error message is displayed, and the DAM file is shut down. DAM FRC must then be executed by the following procedure:

DAM FRC can only be executed for files to be recovered. DAM FRC is invalid for files not to be recovered because they collect no update information in journals.

1. Execute the damrm command to remove the logical file of the erroneous file from OpenTP1.
2. Execute the jnlswpfg and jnlunlfg commands to obtain the journal required for DAM file recovery.
3. With the damrstr command, restore the backup file acquired before startup of online operation or during online operation to place it on the status valid before startup of online operation or during online backup operation. Either the erroneous physical file must be deleted by the damdel command, or the physical file must be allocated in a different partition from the one resulting in the error. The name of the physical file need not be the same as was used during online operation.
4. Define the DAM file to be recovered in the recovery object definition file. The recovery object definition file is created with the OS text editor in the following format:

```
[ Δ ] DAM-file-logical-file-name Δ DAM-file-physical-file-name [ Δ ] (linefeed)
[DAM-file-logical-file-name Δ DAM-file-physical-file-name] [ Δ ] (linefeed)
:
[DAM-file-logical-file-name Δ DAM-file-physical-file-name] [ Δ ] (linefeed)
```

The physical file to be specified is a backup file that has been restored. The user can specify either an online or offline backed-up file that has been restored. Online and offline backed-up files cannot coexist.

5. Execute DAM FRC with the damfrc command.

6. Execute the `damadd` command to add and catalog the recovered DAM file into OpenTP1 again.
7. Execute the `damrles` command to release the shutdown status of the logical file. Once the logical file is released from shutdown status, the DAM file recovered from the error becomes accessible by transactions.

(2) DAM FRC suspended by an error, etc.

If DAM FRC using multiple unload journal files is suspended because of an error, etc., the `damfrc` command must be reexecuted. OpenTP1 will reference the inheritance file and execute DAM FRC using the unload journal files that have already been processed. If the `-s` option is specified, the inheritance file is not referenced; it must therefore not be specified. If there is no processed unload journal file, processing will be suspended due to the invalid journal file. In such a case, the `damfrc` command with the `-s` option specified must be executed.

10.2.7 TAM file

(1) TAM FRC procedure

If an error occurs in a TAM file, an error message is displayed, and the TAM file is shut down. TAM FRC must then be executed by the following procedure:

1. Execute the `tamrm` command to remove the TAM table corresponding to the erroneous TAM file from OpenTP1.
2. Execute the `jnlswpfg` and `jnlunlfg` commands to obtain the journal required for TAM file recovery.
3. Use the `tamrstr` command to restore the backup file that was collected before or during online operation, and recover the TAM file to its status before online operation was started or during execution of online backup. Either the erroneous TAM file must be deleted by the `tamdel` command, or the TAM file must be allocated in a different partition from the one resulting in the error. The name of the TAM file need not be the same as was used during online operation.
4. Execute TAM FRC with the `tamfrc` command.
5. Execute the `tamadd` command to add and catalog the recovered TAM file to OpenTP1 again.
6. Execute the `tamrles` command to release the shutdown status of the TAM table. Once the TAM table is released from shutdown status, the TAM file recovered from the error becomes accessible by transactions.

(2) TAM FRC suspended by an error, etc.

If TAM FRC using multiple unload journal files is suspended because of an error, etc., the `tamfrc` command must be reexecuted. OpenTP1 will reference the inheritance file and execute TAM FRC using the unload journal files that have already been processed.

If the `-s` option is specified, the inheritance file is not referenced; it must therefore not be specified. If there is no processed unload journal file, processing will be suspended due to the invalid journal file. In such a case, the `tamfrc` command with the `-s` option specified must be executed.

(3) I/O failure continuation table in use

(a) Recovery after abnormal online termination

If online operation terminates abnormally during updating using an I/O failure continuation table, the appropriate TAM file can be recovered by taking the following procedure:

1. Execute the `tamrstr` command to restore a backup file of the applicable TAM file.
2. Execute the `jnlunlfg` command to unload the journals from backup acquisition to abnormal online termination.
3. Execute the `tamfrc` command for the specified unloaded-journals file and restored TAM file.
4. Execute the `tamadd` command with the recovered TAM file specified to add and catalog the TAM table to the online system.
5. Execute the `tamrles` command to release the TAM table from its shutdown status.

(b) Recovery during online operation

During updating using an I/O failure continuation table, the appropriate TAM file can be recovered by taking the following procedure:

1. Execute the `tamhold` command to logically shut down the appropriate TAM table at termination of access to the I/O failure continuation table.
2. Execute the `tamrm` command to remove the TAM table from the online system. Or else, execute the `tambkup` command with its `-o` option specified to back up the TAM file in online mode and then the `tamrm` command to remove it from the online system.
3. Execute the `tamrstr` command to restore the backup file of the TAM file.
4. Execute the `jnlunlfg` command to unload the journals from backup acquisition to abnormal online termination. This is not necessary if the `tamunlfg` command with its `-o` option specified was executed in step 2. above.
5. Execute the `tamfrc` command for the specified unloaded-journals file and restored TAM file. This is not necessary if the `tambkup` command with its `-o` option specified was executed in step 2. above.
6. Execute the `tamadd` command with the recovered TAM file specified to add and

catalog the TAM table to the online system. Then, execute the `tamrles` command to release the TAM table from its shutdown status.

(c) Recovery after normal online termination

If online operation terminates normally during updating using the I/O failure continuation table, the user must recover the appropriate TAM file in offline mode. If it is not recovered, it cannot be used in the next online operation because the status of the TAM file differs from that when accessed online.

1. Execute the `tamrstr` command to restore the backup file of the appropriate TAM file.
2. Execute the `jnlunlfg` command to unload the journals from backup acquisition to normal online termination.
3. Execute the `tamfrc` command for the specified unloaded-journals file and restored TAM file.

10.2.8 Message queue files

If an error occurs in a message queue file, an error message is output.

Errors are handled as appropriate to the error messages, as described below.

(1) Physical file space shortage

OpenTP1 must be terminated normally once, and the size of the physical file for the message queue must be increased with the `queinit` command (-r option specified). OpenTP1 can then be started.

(2) File error

OpenTP1 must be terminated normally once. Then, the physical file for the message queue must be deleted with the `querm` command, and a new physical file for the message queue must be allocated with the `queinit` command. OpenTP1 can then be started.

(3) Unusable disk queue

If the disk queue cannot be used for some reason during MCF startup and fallback operation with a memory queue was specified in the extended reservation definition of the MCF manager definitions, processing will resume using the memory queue. When the memory queue is used, messages cannot be recovered during full recovery.

If the disk queue becomes unusable during online operation, it cannot be replaced with the memory queue.

The KFCA11065-W or KFCA11066-W message is output when the fallback operation with a memory queue begins. Whenever either of these messages is output, you must perform recovery from the fallback operation.

The most common cause of a fallback operation is a definition information mismatch. The following are examples of definition information mismatches:

- The queue group ID specified in the application attribute definition of the MCF application definition (`mcfaalcap -g quegrpid= queue-group-ID`) is not specified in the message queue service definition.
- The queue group ID specified in the logical terminal definition of the MCF communication configuration definition (`mcfalcle -k quegrpid= queue-group-ID`) is not specified in the message queue service definition.

In such cases, use the following procedure to recover from the fall-back operation:

1. Stop the OpenTP1 system.
2. Either add the missing queue group ID in the message queue service definition, or correct the queue group ID if it is wrong.
3. If you added a queue group ID in Step 2, use the `queinit` command to allocate the physical file of the corresponding message queue file.
4. If you corrected the MCF application definition or MCF communication configuration definition in Step 2, use the utility startup command to generate a definition object file.
5. Start the OpenTP1 system normally.

If an error occurs in a message queue file, the messages in the queue cannot be recovered. For received messages, the messages that were to be stored in the input queue can be checked by collecting the IJ. For sent messages, the messages that were to be stored in the output queue can be checked by collecting the OJ. Whether or not the IJ and OJ are to be collected is specified in the application attributes definition.

10.2.9 XAR file

If an error occurs in an XAR file, an error message appears and the XAR file is shut down. In this case, the XAR file needs to be recovered. For details about how to recover XAR files, see *3.14.4 Action to take if an error occurs in the XAR file*.

10.2.10 MCF definition file

If an error occurs in the MCF definition file, an error message is output, and OpenTP1 terminates. The cause of the error must be eliminated, as appropriate to the reason code displayed with the error message. OpenTP1 can then be restarted.

10.3 File errors

This section explains the procedures for handling file errors.

10.3.1 Message log file

If an error occurs in a message log file, an error message is output to the standard error output.

The message log file resulting in the error must be deleted, and OpenTP1 must then be restarted. To retain the contents of the erroneous message log file, it must be backed up into a file under a different name.

10.4 Communication errors

This section explains the procedures for handling communication errors.

(1) Communication control unit, terminal, and line errors

If a connection error occurs, the cause of the error must be eliminated, as appropriate to the reason code displayed with the error message. Connection can then be reestablished with the `mctactcn` command.

For details of the error handling procedures, see the applicable protocol manual.

(2) LAN errors

If a LAN error occurs, an error message is output, and the client UAP results in an error during process-to-process communication by the RPCs at both computers.

The cause of the error must be eliminated, as appropriate to the OS and hardware error handling procedures. OpenTP1 can then be restarted.

(3) Communication error under multinode facility

If a communication error occurs between the global archive journal service and the journal service at an archive-journal source node, an error message is displayed. The user must eliminate the cause of the error according to the information contained in the message.

While the communication error remains, the journal at the archive-journal source node is not archived. When there is no swap-in file for the system journal file at the archive-journal source node, the OpenTP1 at the node terminates. The user must unload files individually.

When communication is restored, archiving begins with an untransferred journal even if files are unloaded individually. If journals that must be archived by individual unloading and swapping have been lost, archiving for applicable subsequent nodes is suspended. The user must unload files individually after recovery of communication.

10.5 UAP errors

This section explains the procedures for handling UAP errors.

10.5.1 UAP cannot be started

(1) *Invalid system definitions*

A message is output indicating that the user service configuration definitions or the contents of the user service definitions are invalid. The definition items should be reviewed and the specifications corrected. The UAP can then be restarted.

(2) *Inadequate memory*

A message indicating inadequate memory is output. In the case of the shared memory, the definition items in the system definition should be reviewed. In the case of the process-specific memory, unneeded processes should be terminated.

10.5.2 UAP does not terminate

The status of a UAP can be checked with the `prcls` or `scdls` command or with the OS's `ps` command.

If UAP processing will not terminate, it must be terminated forcibly.

10.5.3 UAP terminates abnormally

A message is output indicating whether UAP processing or OpenTP1 has terminated abnormally.

If UAP processing has terminated abnormally, the UAP should be corrected and restarted.

For HP-UX, linkage bind mode for the UAP must be `immediate`. Otherwise, the UAP may terminate abnormally for unknown reason. Use the OS's `chatr` command to check the linkage bind mode.

If OpenTP1 has terminated abnormally, maintenance personnel should be notified.

10.5.4 UAP deadlocks

The UAP must be corrected so that the resources to be used are specified in the correct order. The UAP can then be restarted.

10.6 OpenTP1 errors

This section explains the procedures for handling OpenTP1 errors.

10.6.1 OpenTP1 cannot be started

(1) *OpenTP1 incorrectly installed or set up*

OpenTP1 installation or setup must be reexecuted.

(2) *Invalid system definitions*

An erroneous specification item in the system definitions is output with a message.

The system definition items must be checked and respecified correctly. OpenTP1 can then be restarted.

(3) *Inadequate memory or disk space*

A message indicating a memory or disk space shortage is output.

Either unneeded processes should be terminated, or unneeded files should be deleted. OpenTP1 can then be restarted.

Additionally, the OS's system constants (especially constants related to the shared memory) should be checked, and the OS should be reconstructed, if necessary. OpenTP1 definition information should also be checked. OpenTP1 can then be restarted.

(4) *Missing file required for OpenTP1 startup*

A message is output indicating that a required file is missing.

The required file must be provided, and OpenTP1 should then be restarted.

(5) *OS configuration inappropriate to OpenTP1 execution environment*

The OS should be reconstructed, the OpenTP1 definition information should be checked, and OpenTP1 should then be restarted.

(6) *TCP/IP environment not set up*

A communication error message is output.

The TCP/IP environment setup should be corrected, and OpenTP1 should then be restarted.

(7) *Error in hardware (disk or LAN)*

An I/O error message is output.

The erroneous device should be repaired or replaced.

(8) Incorrect user environment setting command

If an incorrect user environment setting command is specified by the `user_command` operand in the system environment definition, an attempt to start OpenTP1 may fail without outputting a message or dump.

Check if the user environment setting command specified by the `user_command` operand is correct and take an appropriate action if incorrect. Then, start OpenTP1.

(9) KFCA00715-E output

The `dcreset` command must be executed when the KFCA00715-E message is output. This message is output when OpenTP1 start or restart processing is disabled, indicating that processing of the process service is interrupted. Eliminate the error according to the reason code of the message, and then execute the `dcreset` command. If the `dcreset` command is executed while the error still remains, the KFCA00715-E message appears again.

The `dcreset` command must be executed when OpenTP1 is inactive. For changing the system common definition, this command must be executed after normal termination of OpenTP1. When the KFCA00715-E message is output, the `dcreset` command can be executed after abnormal termination of OpenTP1. Executing the `dcreset` command while OpenTP1 is online causes the system go down.

(10) Excessive load on the CPU

If you have used JP1/AJS2 to set the startup of OpenTP1 from a job, you might not be able to use the `dcstart` command to start OpenTP1. If this problem occurs, check the usage rate of the CPU. The rate is probably 100%, and the system reports that almost all of that amount is being used by the system service processes of OpenTP1.

If you have used JP1/AJS2 to set the startup of OpenTP1 from a job, the OS priority for the `dcstart` command is set to the lowest level (the `nice` value is 39). Therefore, when the load on the CPU increases as the result of many system server processes, CPU time cannot be allocated to the `dcstart` command.

This case often occurs in the following machine configurations:

- Only one CPU is installed.
- The number of system server processes of OpenTP1 is extensive.

If this situation occurs, execute the `ps` command to check the `nice` value for the process in the `dcstart` command. The following shows an example of executing the `ps` command:

For AIX

```
ps -elf|grep dcstart
 200001 A  betran 16382 21390  2  60 20# e7e4 3780          15:56:13 pts/1  0:00
dcstart
```

For HP-UX

```
ps -elf|grep dcstart
  1 S  betran  6995  6932  0 154 20#          41516c00  344          c0fa80 15:56:43
pts/tn 0:00 dcstart
```

#

Indicates the `nice` value. When this value is 39, the OS priority for the `dcstart` command is at the lowest level.

Hitachi recommends that you set the same value for the OS priority for executing OpenTP1 operation commands as the value set for the priority for OpenTP1. To change the priority for OpenTP1 operation commands, in JP1/AJS2-View, change the execution priority level of the job definitions of JP1/AJS2 that starts the process of the OpenTP1 `dcstart` command. For details, see the manual *Job Management Partner 1/Automatic Job Management System 2 Operator's Guide*.

10.6.2 OpenTP1 cannot be terminated

(1) UAP processing continues for a long time, or UAP cannot be terminated due to an error

The status of a UAP can be checked with the `prels` or `scdls` command or with the OS's `ps` command.

If UAP processing does not terminate, either the user can wait for UAP termination or the UAP can be terminated forcibly.

(2) Messages accumulated in output queue are not output during normal termination due to a communication error, etc.

An MCF event that reports the discarding of an unprocessed message (ERREVTVA) is activated after a specified period of time (the unprocessed send message accumulation time specified in the MCF communication configuration definition), and OpenTP1 terminates normally. All unsent messages in the output queue are discarded.

To retain the unsent messages, OpenTP1 must be terminated forcibly before the unprocessed send message accumulation time elapses. The unsent messages can be inherited by then restarting OpenTP1.

(3) OpenTP1 at archive-journal target node does not terminate because connected journal service fails to terminate

Normal or planned termination of OpenTP1 at an archive-journal target node requires that the connections to all journal service be released. To this end, the user must execute the `dcndls` command to check the status of the archive-journal source node being connected and take the following action as may be appropriate:

(a) Archive-journal source node terminated abnormally or forcedly

Execute the `jnlardis` command to forcedly release the connection. Or else, restart OpenTP1 at the archive-journal source node and then perform normal or planned termination.

(b) Archive-journal source node terminated normally or as planned

Execute the `jnlardis` command to forcedly release the connection.

(c) Archive-journal source node in the online mode

Terminate OpenTP1 at the archive-journal source node.

(4) OpenTP1 at archive-journal source node does not terminate because of failure to release connection from global archive journal service

If OpenTP1 at the archive-journal target node is not online, the journal service at the archive-journal source node cannot be released from the connection to the global archive journal service. The user must execute the `dcndls` command to check the status of the archive-journal target node and take the following action as may be appropriate:

(a) Archive-journal target node terminated

Restart OpenTP1 at the archive-journal target node.

(b) Archive-journal target node in online mode

Forcedly terminate OpenTP1 at the archive-journal source node.

10.6.3 OpenTP1 terminates abnormally**(1) OpenTP1 system service or user server directly terminated by kill command**

To continue processing, the user must restart OpenTP1.

(2) OpenTP1 internal error detected

A reason code is output with the message.

Maintenance personnel should be notified.

(3) Abnormal termination of the MCF communication process

The user exit routine may be defective. Check the processing of the user exit routine. Note that if the total amount of the local variables (stack size) used in the user exit routine exceeds 1 Kbyte, the MCF communication process might terminate abnormally.

(4) Abnormal termination of OpenTP1 because of the error in the user application program

OpenTP1 may terminate abnormally if a user application program is defective. Check

if a user application program terminated abnormally before OpenTP1 terminated abnormally. If a user application program has terminated abnormally, check the processing of the user application program. If you cannot find the cause of the abnormal termination, contact maintenance personnel.

10.6.4 OpenTP1 operation command does not terminate normally

(1) Specification of an option or command argument is invalid

An error message is output indicating that the specification of an option or command argument is invalid.

The specification of the option or command argument must be corrected.

(2) No environment is set up to execute operation command

An error message is output indicating that an environment has not been set up to execute the operation command.

The correct environment must be set up, with particular attention being paid to the setup of environment variables.

(3) No authority to execute operation command

The operation command must be executed by a user who is authorized to execute it.

10.6.5 Response wait overtime occurs during OpenTP1 operation command execution

The operation command must be entered when the OS workload is low. Alternatively, the maximum response wait time (`watch_time`) can be increased in the system common definitions.

10.6.6 OpenTP1 errors under multinode facility

(1) OpenTP1 errors at archive-journal target node

If OpenTP1 at an archive-journal target node terminates abnormally, OpenTP1 at an archive-journal source node continues. Journals at the archive-journal source node are not archived. As a result, if there is no swap-in file for the system journal file at the archive-journal source node, OpenTP1 at the archive-journal source node stops. The user must unload journals individually.

When OpenTP1 at the archive-journal target node is restored, archiving begins with an untransferred journal even if files are unloaded individually. If journals that must be archived by individual unloading and swapping have been lost, archiving for applicable subsequent nodes is suspended. The user must unload files individually after recovery of communication.

OpenTP1 at the archive-journal source node cannot terminate normally or as planned before OpenTP1 at the archive-journal target node is restored. Terminate OpenTP1

after restoration of the archive-journal target node or forcedly terminate OpenTP1 at the source node.

If the journal service at the archive-journal source node has a maximum archive-journal termination timeout (specified in `jnl_arc_terminate_timeout` of the journal service definition), OpenTP1 at the archive-journal source node can be terminated normally or as planned. For the journal service definition, see the manual *OpenTP1 System Definition*.

(2) OpenTP1 errors at archive-journal source node

If OpenTP1 at the archive-journal source node terminates abnormally, OpenTP1 at the archive-journal target node continues.

When OpenTP1 at the archive-journal source node is restored, archiving begins with an untransferred journal.

OpenTP1 at the archive-journal target node cannot terminate normally or as planned before OpenTP1 at the archive-journal source node is restored. Terminate OpenTP1 after restoration of the archive-journal source node or forcedly terminate OpenTP1 at the target node.

If the connection is forcedly released with the `jnlardis` command, OpenTP1 at the archive-journal target node can be terminated normally or as planned.

10.7 CPU errors

This section explains the procedure for handling CPU errors.

The cause of an error must be eliminated, as appropriate to the OS and the hardware error handling procedures. Then, the OS on the CPU can be recovered, and OpenTP1 can be restarted.

10.8 Information collection when error occurs

The following table lists the information that is collected when an error occurs.

Table 10-2: Information collection when error occurs

Collected information	Output information	Referencing method	Notes
Message log file (<code>\$DCDIR/spool/dcllog1, dcllog2</code>)	Message output from OpenTP1 or a UAP	Execute the <code>logcat</code> command.	To retain an old message log file, copy its contents before they are lost by using message log file swapping.
Console messages	System information output from OpenTP1	Look at the console.	Output to a printer and output to the centralized console are OS functions.
Standard output and standard error output during command execution	OpenTP1 operation commands' output information and error messages	Check the screen used to enter the operation commands.	To retain this information, redirect it as required.
Core file (<code>./core</code>)	OpenTP1-related process data and stack	--	The core file of an OpenTP1-activated process is saved under the server name <code>n^{#1}</code> in the <code>\$DCDIR/spool/save</code> directory. A maximum of three core files can be saved per server. To retain them, they must be copied.
	UAP trace	Execute the <code>uatdump</code> command.	
Shared memory dump file (In Linux: <code>\$DCDIR/spool/shmdump[1-3].gz</code> , <code>\$DCDIR/spool/shmdump.XXX^{#2}[1-3].gz</code> ; Other operating systems: <code>\$DCDIR/spool/shmdump[1-3].z</code> <code>\$DCDIR/spool/shmdump.XXX^{#2}[1-3].z</code>)	Data retained in the shared memory by OpenTP1	--	To retain an old dump file, it must be copied.
MCF dump file (<code>\$DCDIR/spool/mcfdPXXYY^{#3}</code>)	Data related to MCF	--	Delete any unnecessary files.

10. Error Handling Procedures

Collected information	Output information	Referencing method	Notes
Deadlock and timeout information file (\$DCDIR/spool/dclckinf/ <i>file-name</i>) ^{#4}	Information about resource wait status	Execute the OS's <code>cat</code> or <code>vi</code> command.	Any unneeded file must be deleted by executing the <code>lckrmnf</code> command or by specifying the <code>lck_deadlock_info_remove_level</code> operand of the lock service definition.
MCF trace file (\$DCDIR/spool/mcft <i>AXZZ</i>) ^{#5}	MCF trace	--	To retain an old MCF trace file, it must be copied.
RPC trace file (\$DCDIR/spool/rpctr1,rpctr2)	Text transferred by a UAP using RPC	Execute the <code>rpcdump</code> command.	A receive text may be output to an RPC trace file before a send text. If the size specified for the RPC trace file is too small, data is quickly overwritten in the file, and an adequate amount of information cannot be accumulated.
UAP trace edit/output file (\$DCDIR/spool/save/ <i>server-name</i> ^{#6} .uat)	UAP trace	Execute the OS's <code>cat</code> or <code>vi</code> command.	A maximum of three files can be created per server. To retain them, they must be copied.
OpenTP1 debug information file (\$DCDIR/spool/save/ <i>server-name</i> ^{#6} .deb)	OpenTP1 internal information	--	A maximum of three files can be created per server. To retain them, they must be copied.
XAR event trace information file (\$DCDIR/spool/dcxarinf/trace/xarevtr1,xarevtr2)	Event trace of the XA resource service	Execute the <code>xarevtr</code> command.	To retain old XAR event trace files, they must be copied.
XAR performance verification trace information file (\$DCDIR/spool/dcxarinf/_xr_ <i>nnn</i>) ^{#7}	Performance verification trace of the XA resource service	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old XAR performance verification trace information files, copy these files.
JNL performance verification trace information file (\$DCDIR/spool/dcjnlinf/prfinf/_j1_ <i>nnn</i>) ^{#7}	Event trace of the journal service	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old JNL performance verification trace information files, copy these files.

Collected information	Output information	Referencing method	Notes
LCK performance verification trace information file (<code>\$DCDIR/spool/dclckinf/prf/_lk_nnn#7</code>)	Event trace of the lock service	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old LCK performance verification trace information files, copy these files.
MCF performance verification trace information file (<code>\$DCDIR/spool/dcmcfinf/_mc_nnn#7</code>)	MCF performance verification trace	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old MCF performance verification trace information files, copy these files.
Process service event trace information file (<code>\$DCDIR/spool/dcprcinf/_pr_001, _pr_002, _pr_003</code>)	Event trace of the process service	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old process service event trace information files, copy these files.
FIL event trace information file (<code>\$DCDIR/spool/dcfilinf/_fl_001, _fl_002, _fl_003</code>)	Event trace for the OpenTP1 file	Execute the <code>prfget</code> and <code>prfed</code> commands.	To retain old FIL event trace information files, copy these files.

Legend:

#1

n

Evacuate core file serial number (1-3)

No serial numbers are appended to core files to be output when OpenTP1 is terminated abnormally (that is, when the `dcsvstop -df` command or `prckill` command is executed or a real-time monitoring timeout error occurs).

#2

XXX: Resource manager name (dam, tam, ist, ism)

#3

PXXYY:

P: One of the following as the type of process from which the dump file was output:

m: MCF manager process

c: MCF communication service or MCF application start service

u: User service, etc.

XX: MCF communication process identifier specified in the *-s* option of the *mcfteenv* definition command of the MCF communication configuration definition

YY: File serial number (01-99)

#4

A file name is determined on the basis of the deadlock detection time and date. The length of the file name depends on whether the date consists of one or two digits.

Example:

06: 29: 56 on October 3: Oct3062956

18: 06: 00 on October 10: Oct10180600

#5

AXXZZ:

AXX: MCF identifier

A: MCF manager process identifier specified in the *id* operand of *mcfmenv -m* in the MCF manager definition

XX: MCF communication process identifier specified in *mcfteenv -s* in the MCF communication configuration definition

ZZ: Trace swap file identifier

#6

n

File Serial number (1-3)

#7

nnn: Value in the range from 001 to the value specified in the *prf_file_count* operand in the following definitions:

- XAR performance verification trace information file: XAR performance verification trace definition
- JNL performance verification trace information file: JNL performance verification trace definition

- LCK performance verification trace information file: LCK performance verification trace definition
- MCF performance verification trace information file: MCF performance verification trace definition

10.9 Information inherited during full recovery

The following table lists the information that is inherited during full recovery.

Table 10-3: Information inherited during full recovery

Classification	Inherited information
Status file	Status of a shutdown file
System journal file	Open status of a file
Checkpoint dump file	
Archive journal file	
DAM file	User data
	Shutdown status of a logical file
TAM file	User data
	Status of a TAM table
XAR file	Shutdown status of an XAR file
Message queue	Message serial number
	Queue pointer
	Branch message send
	Terminal configuration
UAP	Shutdown status of a UAP
	User server configuration

Chapter

11. Troubleshooting

This chapter describes what information to acquire when an error occurs in OpenTP1, and how to investigate the problem in an OpenTP1 system.

Section *11.1 Information to acquire and items to check* describes the items to be collected and checked when any kind of error occurs in an OpenTP1 system.

Section *11.2 Troubleshooting procedures* describes procedures for troubleshooting specific problems that may occur in an OpenTP1 system.

11.1 Information to acquire and items to check

11.2 Troubleshooting procedures

11.1 Information to acquire and items to check

This section describes the troubleshooting data you must collect if a problem occurs in an OpenTP1 system.

11.1.1 Information to acquire

The tables in this section list the information you must collect to identify the cause of problem in an OpenTP1 system.

The information described here is the basic information required to investigate various problems. It can be used to identify the cause of problems not specifically mentioned in *11.2 Troubleshooting procedures*.

Tables 11-1 through 11-6 list the information that is needed to identify the cause of a problem. If the remote system is also an OpenTP1 system, collect the information described in the tables both on the affected node and on the remote system node.

Table 11-1: Information required for troubleshooting (for TP1/Server Base, TP1/LiNK, TP1/FS/Direct Access, TP1/FS/Table Access, etc.)

Information to acquire	Remarks
betran.log (OpenTP1 standard output and standard error output destination file) on the affected node	<p>In the UNIX version, information is output to the file specified in <code>\$DCDIR/bin/prcout</code>. If no file is specified, information is output to the following files:</p> <ul style="list-style-type: none"> • <code>\$DCDIR/spool/prclog1</code> • <code>\$DCDIR/spool/prclog2</code> <p>In the Windows version, acquire the event logs (application log and system log). Also, if the standard output redirecting facility is used, acquire the file specified in the <code>redirect_file_name</code> operand of the system environment definition. If no file is specified, information is output to the following files:</p> <ul style="list-style-type: none"> • <code>%DCDIR%\spool\prclog1</code> • <code>%DCDIR%\spool\prclog2</code>
syslog file on the affected node	Not required when using Windows.
\$DCDIR directory information of the affected node	<p>In the UNIX version, execute the following command to acquire information:</p> <pre>ls -laR \$DCDIR</pre>

Information to acquire	Remarks
Files under \$DCDIR/spool on the affected node	If you are unable to acquire all of the files under spool for some reason, back up the files in the directory and then acquire the following files: <ul style="list-style-type: none"> \$DCDIR/spool/rpclog* \$DCDIR/spool/errlog/errlog* \$DCDIR/spool/cmdlog/cmdlog*
Files under \$DCDIR/tmp on the affected node	If the prc_current_work_path operand of the system common definition is specified, also acquire the files on this path.
Files under \$DCDIR/conf and \$DCCONFPATH on the affected node	If \$DCUAPCONFPATH is set, also acquire the files under \$DCUAPCONFPATH. When acquiring definition files, make sure that you also acquire the user service definition.

You use the `dcrasget` command to acquire the information. Note, however, that Windows event logs are not acquired by this command; to acquire Windows event logs, follow the procedure described in the documentation for your operating system.

Table 11-2: Information required for troubleshooting (for TP1/Client/P)

Information to acquire	Remarks
Client environment definition file (%WINDIR%\betran.ini)	If the <code>defpath</code> argument is specified for the <code>dc_clt_cltin_s</code> function, acquire the files specified by the argument. When acquiring data from PCs that use the Windows Terminal Service or MetaFrame, %WINDIR% corresponds to the respective user directories under \Documents and Settings\.
dcerr*.trc (error traces) dcuap*.trc (UAP traces) dcsoc*.trc (socket traces) dcmdl*.trc (module traces)	These files are output to the directory specified for <code>DCTRCPATH</code> in the client environment definition. If <code>DCTRCPATH</code> is not specified, they are output to the current directory of the CUP.

Table 11-3: Information required for troubleshooting (for TP1/Client/W)

Information to acquire	Remarks
Client environment definition file (Log in as the user who starts the CUP, and acquire the execution results of the OS's <code>env</code> command.)	If the <code>defpath</code> argument is specified for the <code>dc_clt_cltin_s</code> function, acquire the files specified by the argument.

Information to acquire	Remarks
dcerr*.trc (error traces) dcuap*.trc (UAP traces) dcsoc*.trc (socket traces) dcmdl*.trc (module traces)	These files are output to the directory specified for DCTRPCPATH in the client environment definition. If DCTRPCPATH is not specified, they are output to the current directory of the CUP.

Table 11-4: Information required for troubleshooting (for TP1/Client/J)

Information to acquire	Remarks
Client environment definition (file names and locations depend on the configuration of the client)	If the deffilename argument is specified for the rpcOpen method, acquire the files specified by the argument.
dcerr*.trc (error traces) dcuap*.trc (UAP traces) dcmd*.trc (method traces) dcdat*.trc (data traces)	Information is output to the following locations: <ul style="list-style-type: none"> • Error traces: Directory specified by the dcerrtracpath operand (information is output only when dcerrtrace=Y is specified) • UAP traces: Directory specified by the dcuaptracpath operand (information is output only when dcuaptrace=Y is specified) • Method traces: Directory specified by the dcmethotracedpath operand (information is output only when dcmethotraced=Y is specified) • Data traces: Directory specified by the dcdatatracedpath operand (information is output when dcdatatraced=Y is specified)
Debug traces (dcClExxxxxxxxxxxxx.dmp) (where xxxxxxxxxxxx is the timestamp of the trace)	This data is output to the TP1clientJ directory in the home directory of the user who executed the CUP (or the user who executed JavaVM).

Table 11-5: Information required for troubleshooting (for TP1/Client for .NET Framework)

Information to acquire	Remarks
Client configuration definition	Acquire application configuration files or machine configuration files (for example CUP.exe.config).

Information to acquire	Remarks
dcerr*.trc (error traces) dcuap*.trc (UAP traces) dcmttd*.trc (method traces) dcdat*.trc (data traces)	This data is output to the following locations: <ul style="list-style-type: none"> • Error traces: the directory specified for the path attribute of the <errTrace> element (if use="true") • UAP traces: the directory specified for the path attribute of the <uapTrace> element (if use="true") • Method traces: the directory specified for the path attribute of the <methodTrace> element (if use="true") • Data traces: the directory specified for the path attribute of the <dataTrace> element (if use="true")
Debug traces (dcCltxxxxxxxxxxxxxx.dmp) (where xxxxxxxxxxxxxx is the timestamp of the trace)	This data is output to the following locations: <ul style="list-style-type: none"> • When the path attribute of the <debugTrace> element is specified: <i>value-specified-by-path-attribute</i>\Hitachi\OpenTP1\TP1ClientNET • When the path attribute of the <debugTrace> element is omitted: C:\Document and Settings\user-name\Application Data\Hitachi\OpenTP1\TP1ClientNET

Table 11-6: Information required for troubleshooting (for TP1/Message Control and TP1/NET/***)

Information to acquire	Remarks
Information listed in <i>Table 11-1</i>	--
The following definitions: <ul style="list-style-type: none"> • MCF manager definition • MCF communication configuration definition (common definition and individual definition) • MCF application definition 	Any storage destinations can be specified for the definitions and files. Although the definition files used online are object files, acquire the definition files before object generation.
System service information definition (files under %DCDIR/lib/sysconf/ with names beginning with mcfu)	--
System service common information definition (%DCDIR/lib/sysconf/mcf)	--
MCF traces (%DCDIR/spool/mcftXXXnn, where XXX is the MCF identifier and nn is a sequentially assigned number)	--

Legend:

-- None.

For details of the information required to troubleshoot errors relating to TP1/Message Queue, TP1/Message Queue - Access, and Message Queue - Operation, see the relevant manual for each program product.

11.1.2 Items to check

This subsection lists, by product, the items to check when a problem occurs.

Items to check for all OpenTP1-related products

- Version numbers of OpenTP1 products
Check the detailed version number of the product. For modified versions of OpenTP1 products, also check the code appended to the version notation.
Example: 07-00-01, 06-03-/A etc.
- The data and time when the problem occurred
- For multi-node systems, the time difference between the nodes
- The workload on the node at the time when the problem occurred (whether the CPU was under a high load, and other aspects for which data exists)
- Whether the problem occurred in a real-world environment
- If the problem occurred on a user server, the history of operations on the user server
- Whether any changes were made to the OpenTP1 environment immediately before the event occurred

Items to check when using TP1/Client

- The time difference between the server running TP1/Client and the remote server

Items to check when using TP1/Message Queue

- The name of the product to which TP1/Message Queue connects
- If TP1/Message Queue connects to a product from another provider, see what information is available regarding the state of the product at the time the problem occurred

Items to check when using TP1/Message Queue - Access

- The type of MQC client used by the user application
- When using XA linkage, the name of the transaction monitoring product being used

11.1.3 Method of checking traces

(1) UAP traces

(a) Overview of UAP traces

Some of the procedures described in *11.2 Troubleshooting procedures* involve looking

at UAP trace information. A UAP trace is trace information acquired when a function provided by OpenTP1 is invoked. By viewing the UAP trace for an OpenTP1 function, you can find out to what stage the function had progressed before the process of the function stopped. `dc_***` (entry) appears in the UAP trace if the process stopped inside the function, and `dc_***` (exit) appears if the process stopped outside the function. Consequently, by viewing UAP trace information, you can find out whether the processing stopped inside or outside the function.

(b) Checking UAP trace information

UAP trace information is acquired only when a UAP terminates abnormally. If OpenTP1 is running when the UAP terminates, UAP trace information is output to the file `user-server-name.uat#` in the directory `$DCDIR/spool/save`.

A UAP trace file is in text format, and can be viewed by a standard editor such as `vi`. To locate a specific UAP trace, search the trace information file for the PID in the message KFCA01820-E that was output when the UAP terminated abnormally.

UAP trace information may not be output to `$DCDIR/spool/save` if the entire OpenTP1 system goes down when the user server terminates abnormally. If you are unable to find the UAP trace under `$DCDIR/spool/save`, look for the `core` file in the current directory (`$DCDIR/spool/tmp/home/user-server-name`). If you find the `core` file, execute the command `uatdump core > desired-file-name` to edit and output the UAP trace file.

#

n

The generation number of the file.

(2) Checking stack trace information

Some of the procedures described in *11.2 Troubleshooting procedures* involve acquiring a stack trace. You can acquire a stack trace as follows:

In UNIX:

Examples are given for AIX and Linux. Because the procedure differs for each OS, see the manual for your operating system.

AIX: Execute `dbx load-module-name core-file`. Enter `where` at the prompt to display the stack trace.

Linux: Execute `gdb UAP-load-module-name core-file | tee-file-name` (optional). At the prompt, enter `bt`. Press Enter to view more lines. Once you have displayed the entire stack trace, enter `q` to quit. The stack trace is output to the specified file.

In Windows:

11. Troubleshooting

From the Command Prompt, execute the `drwtsn32 -i` command to ensure that Dr Watson logs (which include stack trace information) are output to the specified folder when a process terminates abnormally. For details, see the manual for the applicable operating system.

11.2 Troubleshooting procedures

This section describes how to investigate problems, through the use of specific examples.

11.2.1 When the message KFCA00307-E is output

(1) Error event

The following message is output, and OpenTP1 fails to obtain a file descriptor.

KFCA00307-E insufficient file descriptor for RPC.

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

- The information in *Table 11-1*

(4) Troubleshooting procedure

The message KFCA00327-E is output when OpenTP1 is unable to obtain a file descriptor to use for communication. The `max_socket_descriptors` operand specifies the maximum number of file descriptors. Common causes of this problem are as follows:

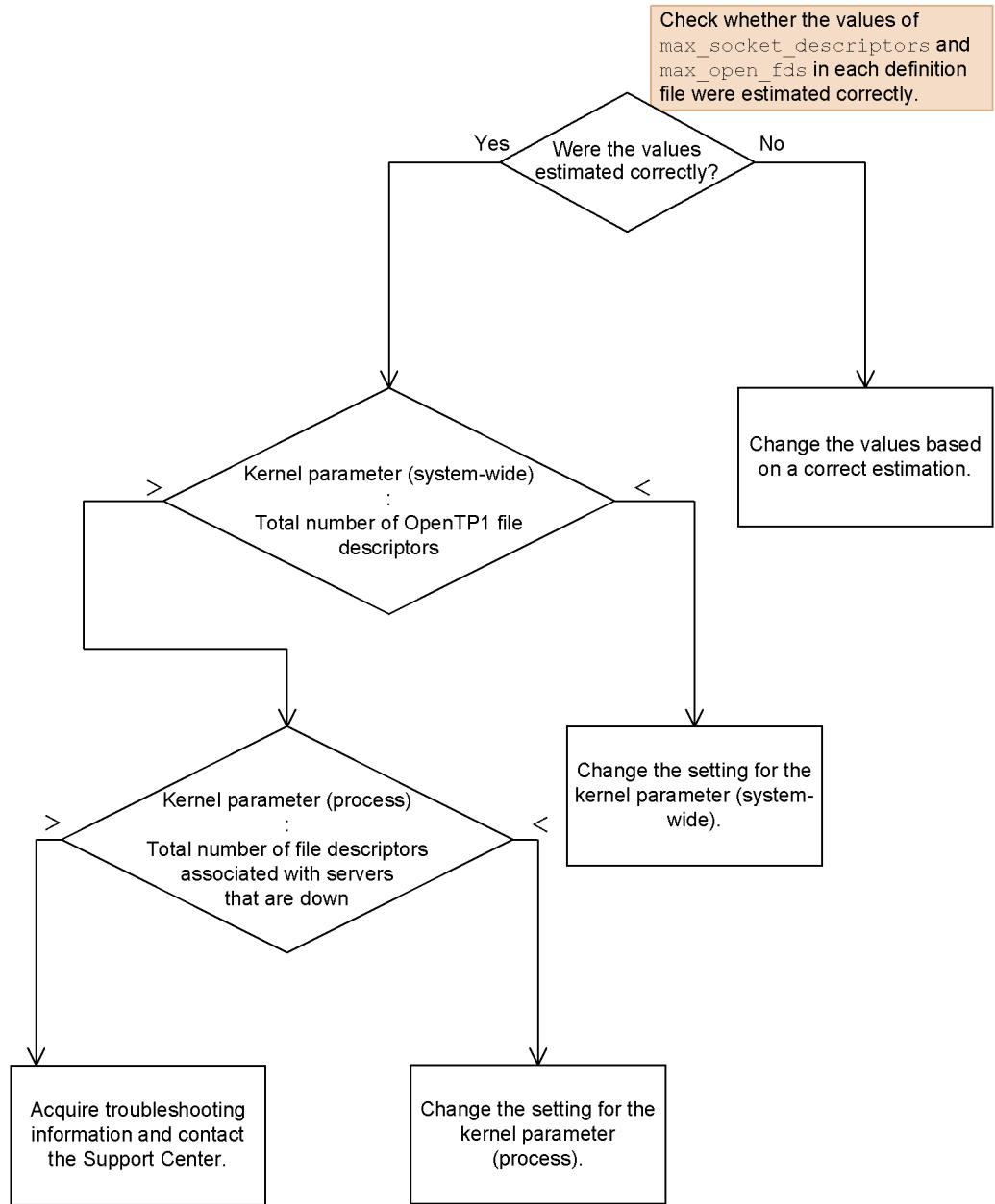
- Insufficient file descriptors are available because too small a value is specified for the `max_socket_descriptors` operand.
- A value specified for an OS kernel parameter[#] is too small.

#

The names of the relevant kernel parameters depend on your OS. Different parameters determine how many file descriptors are available system-wide, and how many are available to individual processes. In the flow chart below, Kernel parameter (system-wide) refers to the maximum number of file descriptors available to the system in its entirety, and Kernel parameter (process) refers to the maximum number of file descriptors available to individual processes. If you change either kernel parameter, specify a value that offers sufficient leeway.

Investigate the cause of the error using the flow chart below.

Figure 11-1: Troubleshooting procedure when message KFCA00307-E is output



The total number of file descriptors used by each server is the sum of the `max_socket_descriptors` and `max_open_fds` operands of the server. The total number of file descriptors used throughout the OpenTP1 system is the sum of the values of the `max_socket_descriptors` and `max_open_fds` operands of all the servers.

Check the values specified for the `max_socket_descriptors` operands of each server by referring to the table below. Calculate the file descriptors process-by-process.

Table 11-7: Values specified for max_socket_descriptors operand

No.	Server name	Valid value of max_socket_descriptors operand
1	prcd, itvd, stsd, cpdd, tjld, qued, damd, tamd, tamioid, ismbd, istd, logd, prfiod, cltcond, cltd, cltrnd, xatd, xatcd, rmmmd, admrsvre, mcfmngrd	The value specified in the system common definition (<code>betranrc</code>). If omitted, a value of 64 is used.
2	namd, namaudtd	The value specified in the name service definition (<code>nam</code>). If there is no name service definition, the value specified in the system common definition (<code>betranrc</code>) is used. If omitted, a value of 64 is used.
3	scdd, scdmld	The value specified in the schedule service definition (<code>scd</code>). If there is no schedule service definition, the value specified in the system common definition (<code>betranrc</code>) is used. If omitted, a value of 64 is used.
4	trnd, trnrvd, trnrmd	The value specified in the transaction service definition (<code>trn</code>). If there is no transaction service definition, the value specified in the system common definition (<code>betranrc</code>) is used. If omitted, a value of 64 is used.
5	jnlld, jnl iod, jnl sdd, jnl swd, jard, jarswd, jarrvd	The value specified in the journal service definition (<code>jnl</code>). If there is no journal service definition, the value specified in the system common definition (<code>betranrc</code>) is used. If omitted, a value of 64 is used.
6	mcfmngrd, mapsmgrd, mcfcmdsv, mqtmgd	Fixed value of 64.
7	MCF application service, MCF communication service	The value specified in the system service common information definition (<code>\$DCDIR/lib/sysconf/mcf</code>). If omitted, a value of 64 is used.

No.	Server name	Valid value of max_socket_descriptors operand
8	mquad, mqaiod, mqacmd, mqamnd	TP1/Message Queue version 05-12 or earlier: The value specified in the system common definition (betranrc). If omitted, a value of 64 is used. TP1/Message Queue version 05-13 or later: The value specified in the MQA service definition (mqa). If omitted, a value of 64 is used.
9	mqtatcp	TP1/Message Queue version 05-13 or earlier: Fixed value of 304. TP1/Message Queue version 05-14 or later The value specified in the MQT service definition (mqt). If omitted, a value of 64 is used.
10	RAP-processing listener, RAP-processing server	The value specified in the RAP-processing listener service definition (name varies). If omitted, a value from one of the following is used (with 1. having priority over 2.). 1. User service default definition (usrcc) 2. System common definition (betranrc) If all three are omitted, a value of 64 is used.
11	User server	The value specified in the user service definition (name varies). If omitted, a value from one of the following is used (with 1. having priority over 2.). 1. User service default definition (usrcc) 2. System common definition (betranrc) If all three are omitted, a value of 64 is used.
12	RTSSUP, RTSSPP	Fixed value of 64.

Check the values specified for the max_open_fds operands of each server by referring to the table below. Calculate the file descriptors process-by-process.

Table 11-8: Values specified for max_open_fds operand

No.	Server name	Valid value of max_open_fds operand
1	prcd, namd, namaudtd, scdd, scdmtd, trnd, trnrvd, trnrmd, jnld, jnlswd, jnliod, jnlsdd, jard, jarswd, jarrvd, itvd, stsd, cpdd, tjld, qued, damd, tamd, tamiod, ismbd, istd, logd, prfiop, cltcond, cltd, cltrnd, xatd, xatcd, rmmnd, admrsvre, mcfmngd, mcfrmgd, mapsmgd, mcfcmdsvmqad, mqaiod, mqacmd, mqamnd, mqtatcp, mqtmgd	Fixed value of 50.

No.	Server name	Valid value of max_open_fds operand
2	MCF application service, MCF communication service	The value specified in the system service common information definition (<code>\$DCDIR/lib/sysconf/mcf</code>). If omitted, a value of 100 is used.
3	RAP-processing listener, RAP-processing server	The value specified in the user service default definition (<code>usrrc</code>). If omitted, a value of 50 is used.
4	User server	The value specified in the user service definition (name varies). If there is no user service definition, the value specified in the user service default definition (<code>usrrc</code>) is used. If omitted, a value of 50 is used.
5	RTSSUP, RTSSPP	Fixed value of 50.

11.2.2 When the message KFCA00327-W is output

(1) Error event

The following message is output, and an RPC returns an error code.

```
KFCA00327-W remote procedure call error occurred.
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the source and destination user servers of the RPC
- Any changes made to the OpenTP1 environment immediately before the event occurred
- When the source of the RPC is on a different node from the destination, the time difference between the nodes

(3) Information to acquire

- The information in *Table 11-1*
- The `hosts` file (except when using a DNS)
- When the source node of the RPC is different from the destination node: The information in *Table 11-1* for the destination node

(4) Troubleshooting procedure

The message KFCA00327-W is output when an RPC returns an error code. Common causes of this problem are as follows:

- An invalid service group name or service name is specified in an argument of the

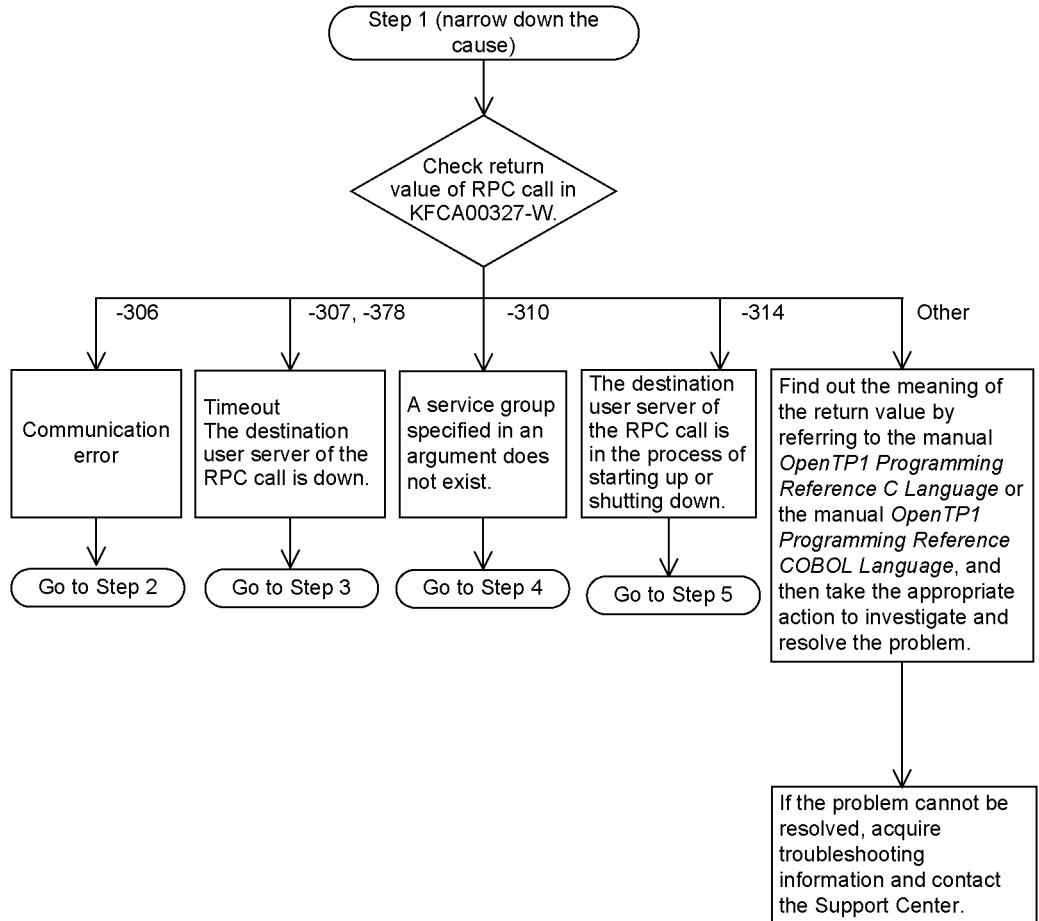
RPC.

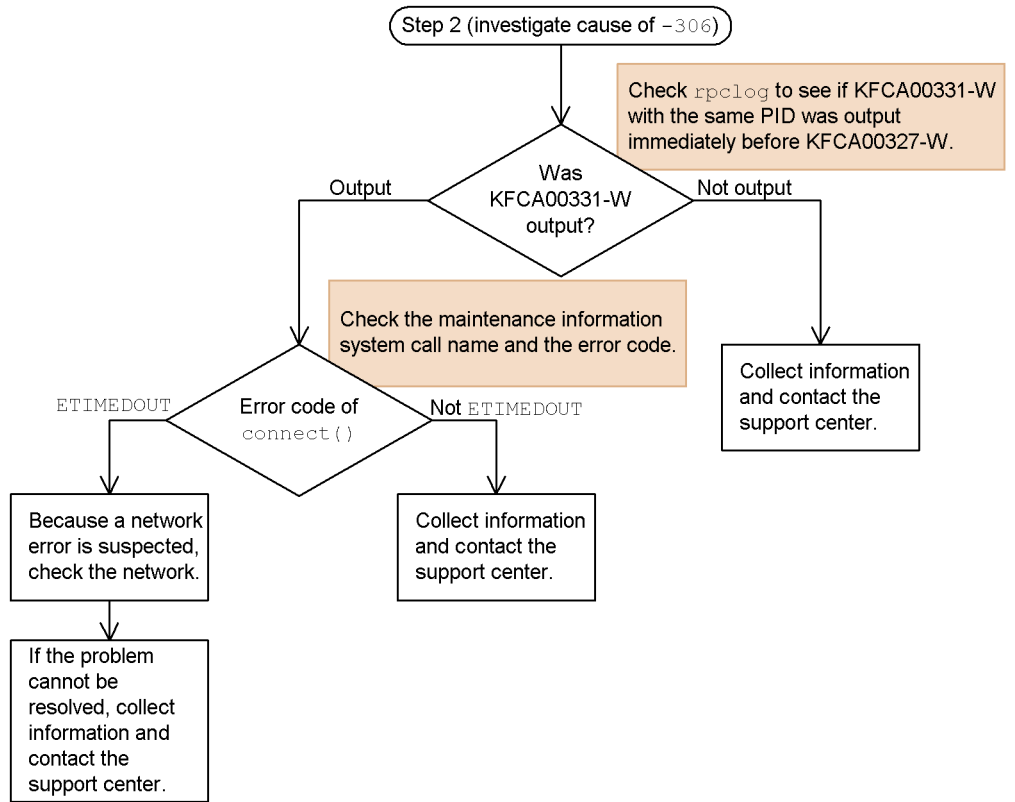
- The user server that is the destination of the RPC is down.

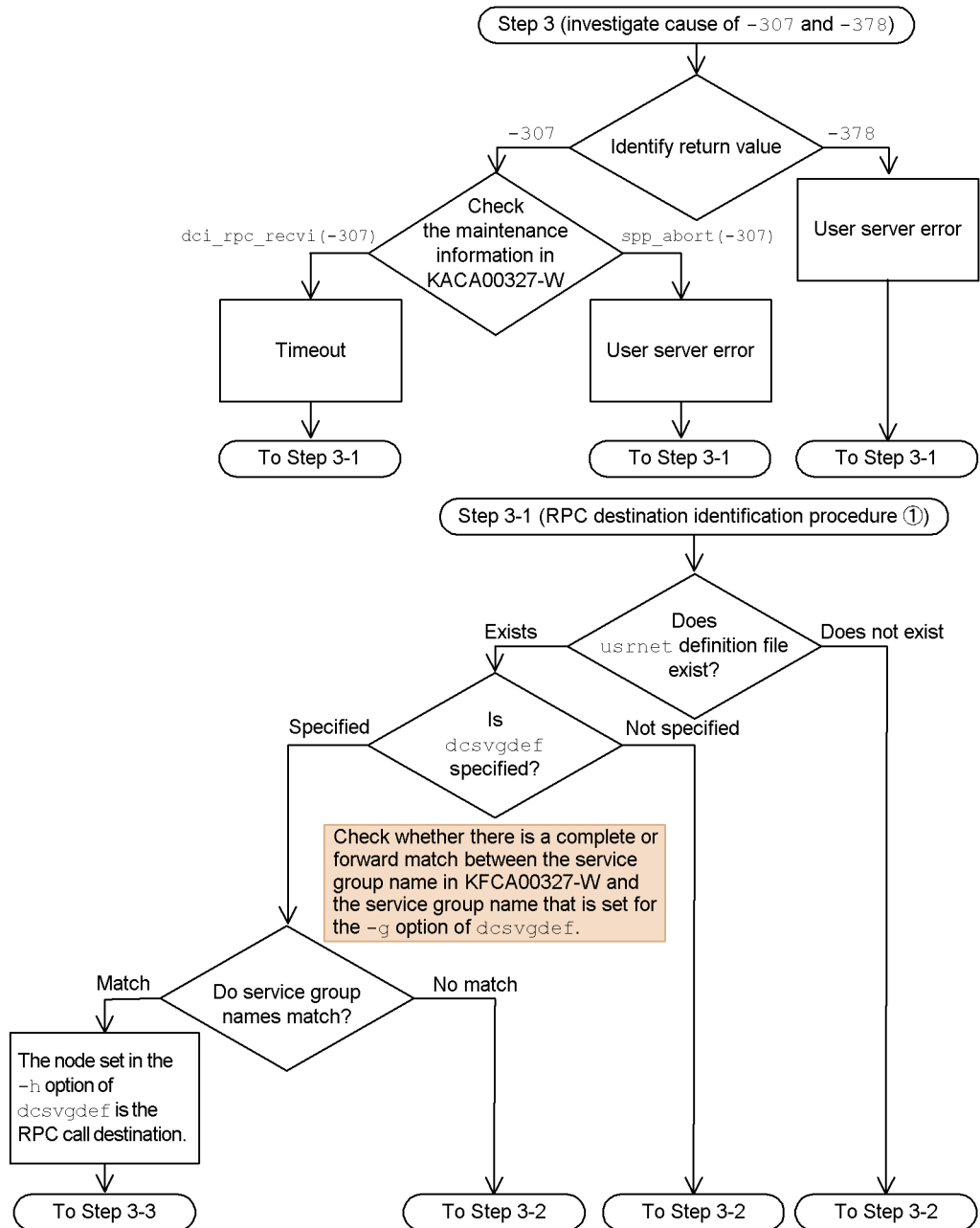
Note that you may be unable to identify the RPC destination if `name_global_lookup=Y` is specified in the name service definition (`nam`).

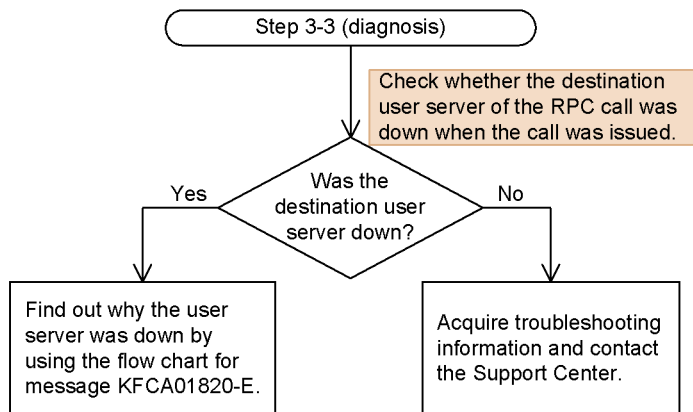
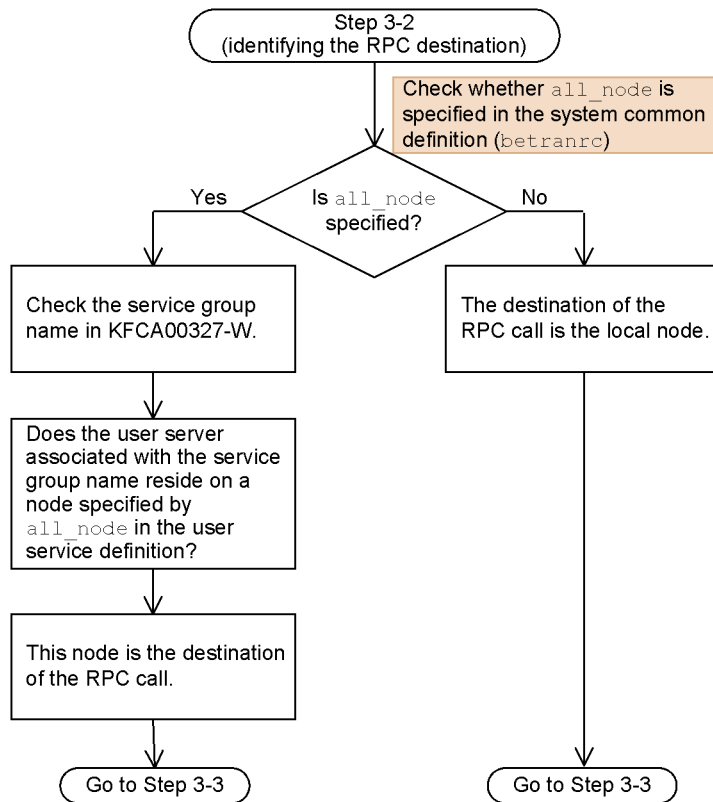
Investigate the cause of the error using the flow charts below.

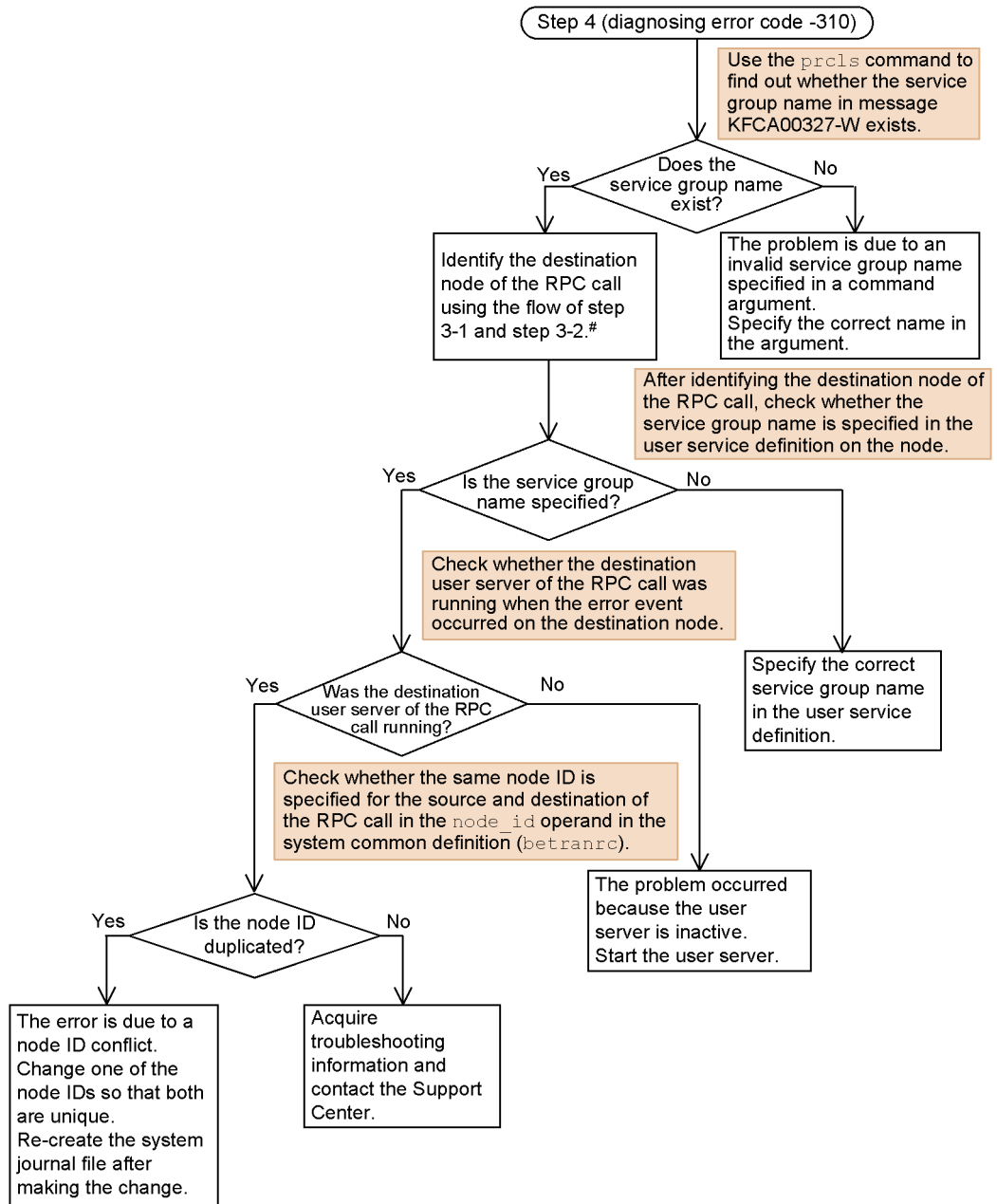
Figure 11-2: Troubleshooting procedure when message KFCA00327-W is output

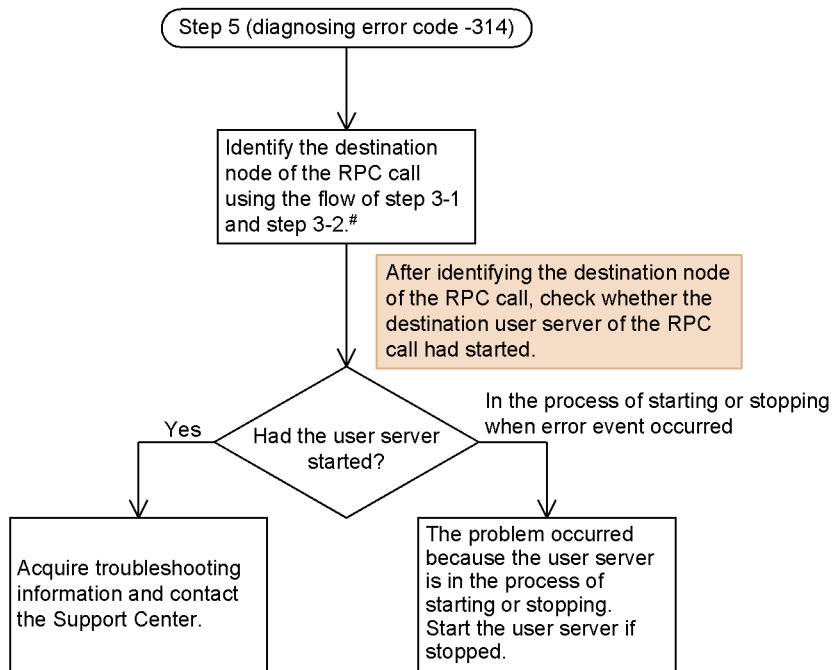












#: When using these procedures only to identify the node, you do not need to proceed to step 3-3.

11.2.3 When the message KFCA00328-W is output

(1) Error event

The following message is output, and an attempt to respond to an RPC fails.

```
KFCA00328-W reply send error occurred at server(aa...aa) .
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

- The information in *Table 11-1*
- The `hosts` file (except when using a DNS)

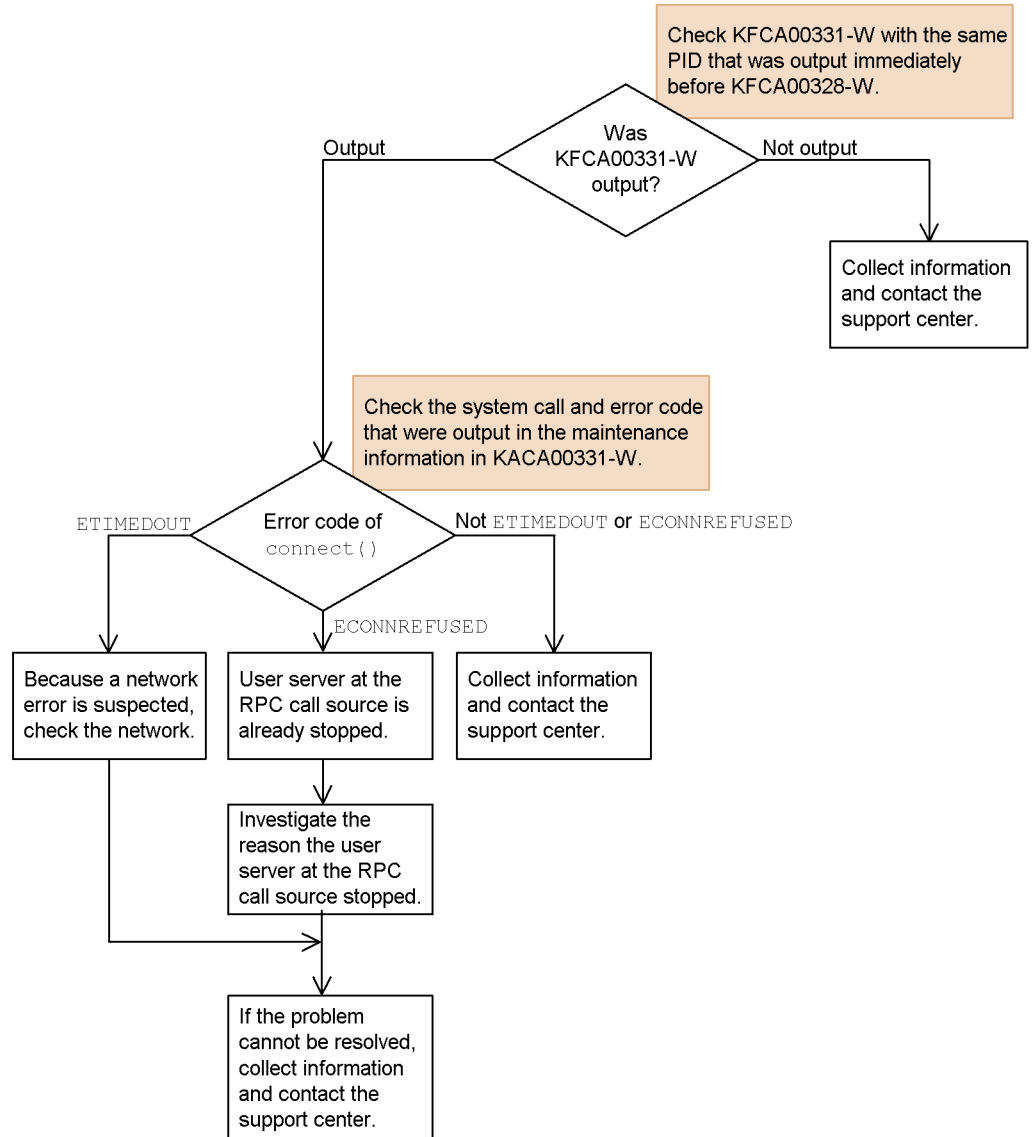
- When the destination of the response is another node: The information in *Table 11-1* for the destination node
- When the destination of the response is TP1/Client: The information in *Table 11-2*, *Table 11-3*, or *Table 11-4* for the destination

(4) Troubleshooting procedure

The message KFCA00328-W is output when an attempt to respond to an RPC fails. The most likely cause is that the user server that issued the RPC cannot be found.

Investigate the cause of the error using the flow chart below.

Figure 11-3: Troubleshooting procedure when message KFCA00328-W is output



Possible causes include a user logic stoppage caused by an RPC call timeout, abnormal termination, or forced termination caused by expiration of timer monitoring.

11.2.4 When the message KFCA00502-I is output

(1) Error event

The following message is output, and a timeout occurred in a process being monitored in real time.

```
KFCA00502-I timeout occurred during real time monitoring at server
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

- The information in *Table 11-1*

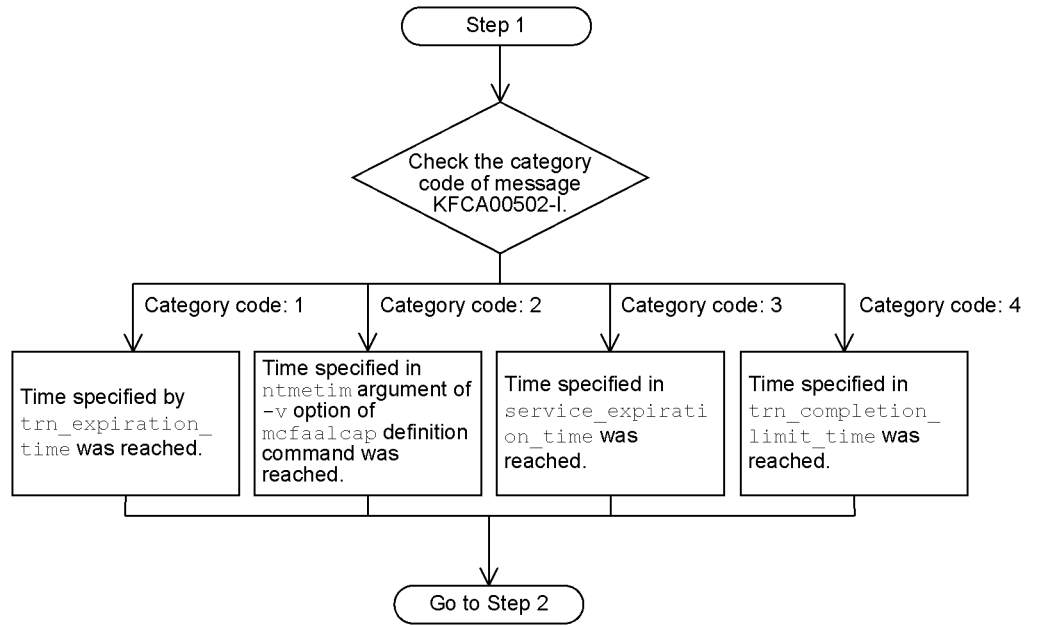
(4) Troubleshooting procedure

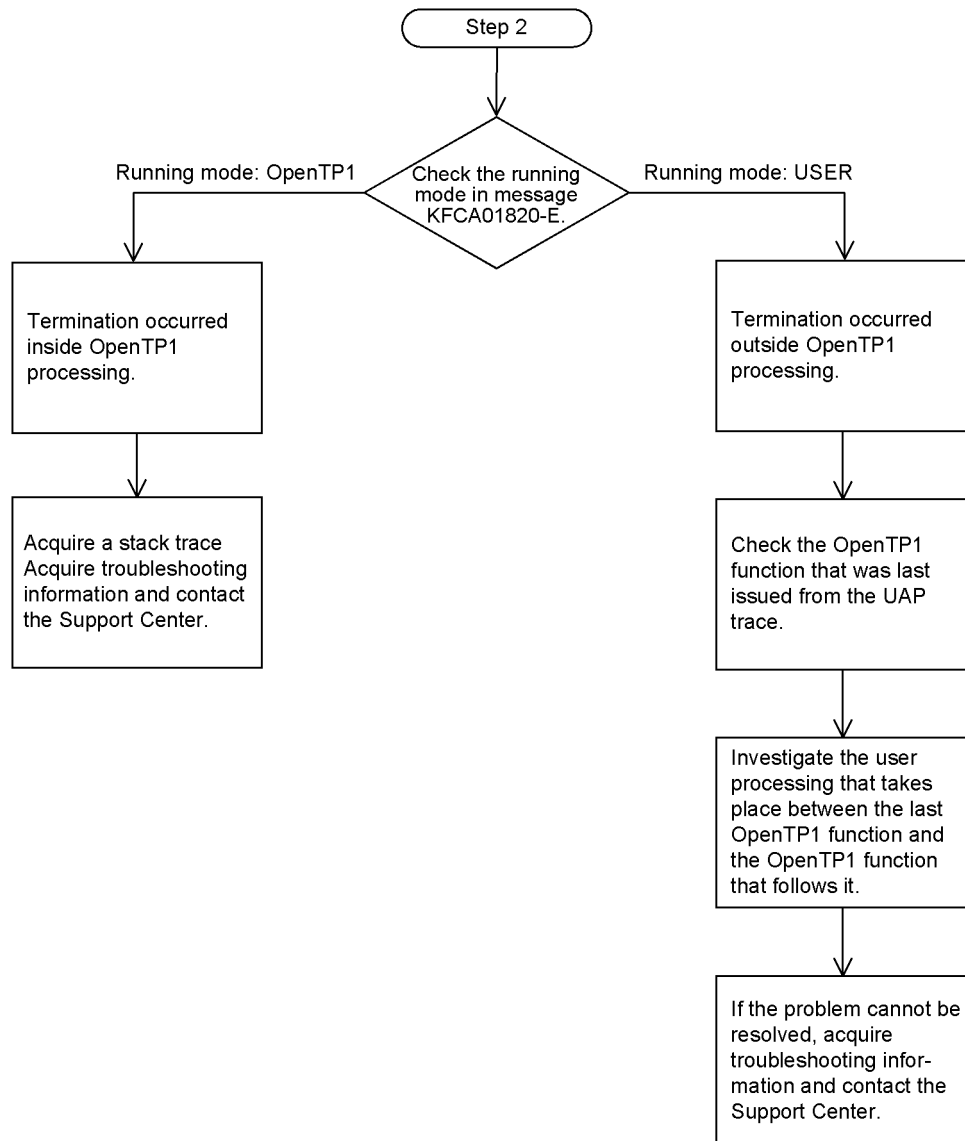
The message KFCA00502-I is output when the value specified in `trn_expiration_time`, `service_expiration_time`, or another such operand is reached. Because the affected process is forcibly terminated, the message KFCA01820-E is also output. Common causes of this problem are as follows:

- The user processing portion of the process takes too long.
- Too much time elapses before a response is issued to a `dc_rpc_call` function.

Investigate the cause of the error using the flow charts below.

Figure 11-4: Troubleshooting procedure when message KFCA00502-I is output





11.2.5 When the message KFCA00837-I is output

(1) Error event

The following message is output when executing the `dcsvstop` command fails to stop the specified user server.

```
KFCA00837-I now holding termination processing for server aa...aa.
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server that failed to stop
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

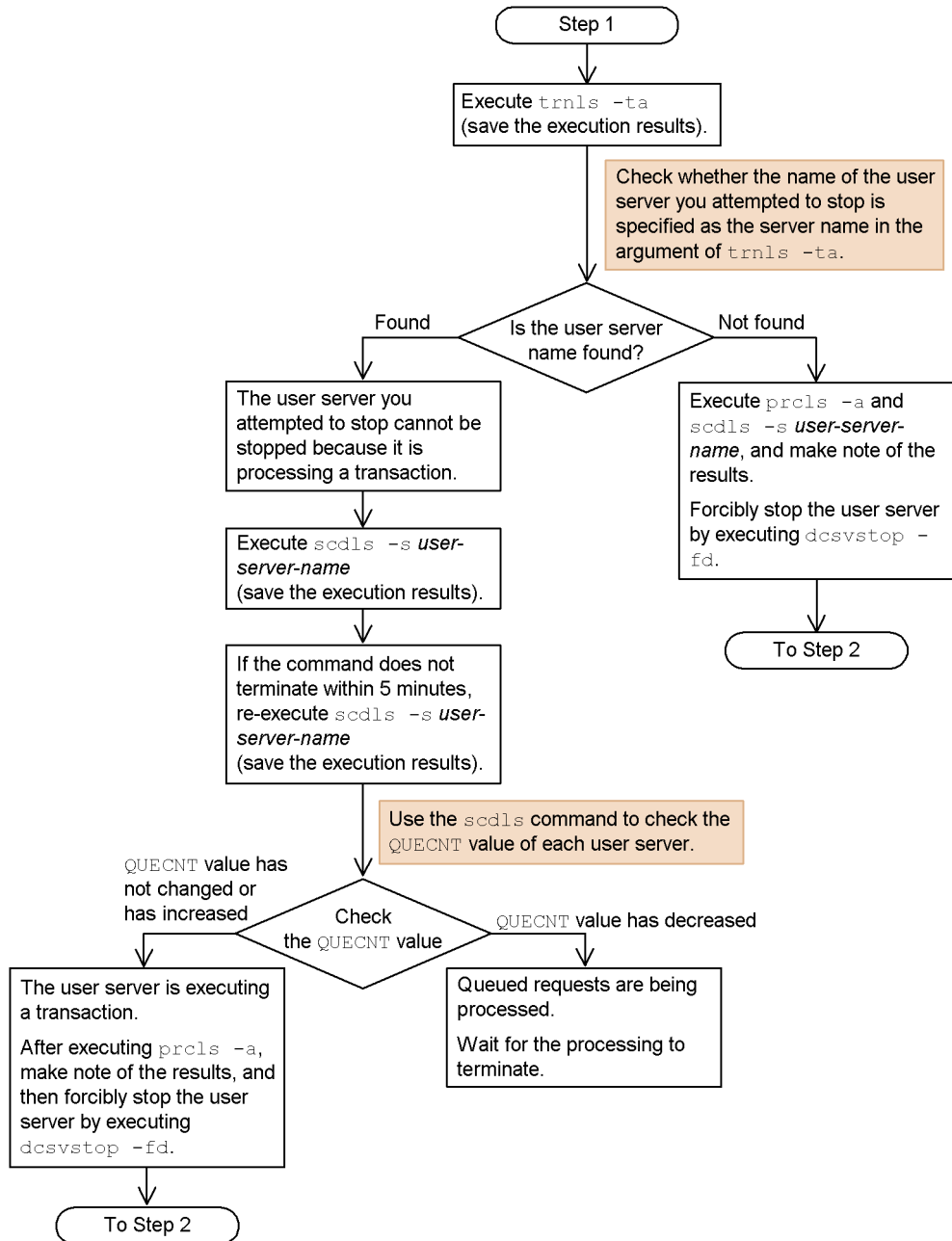
- The information in *Table 11-1*

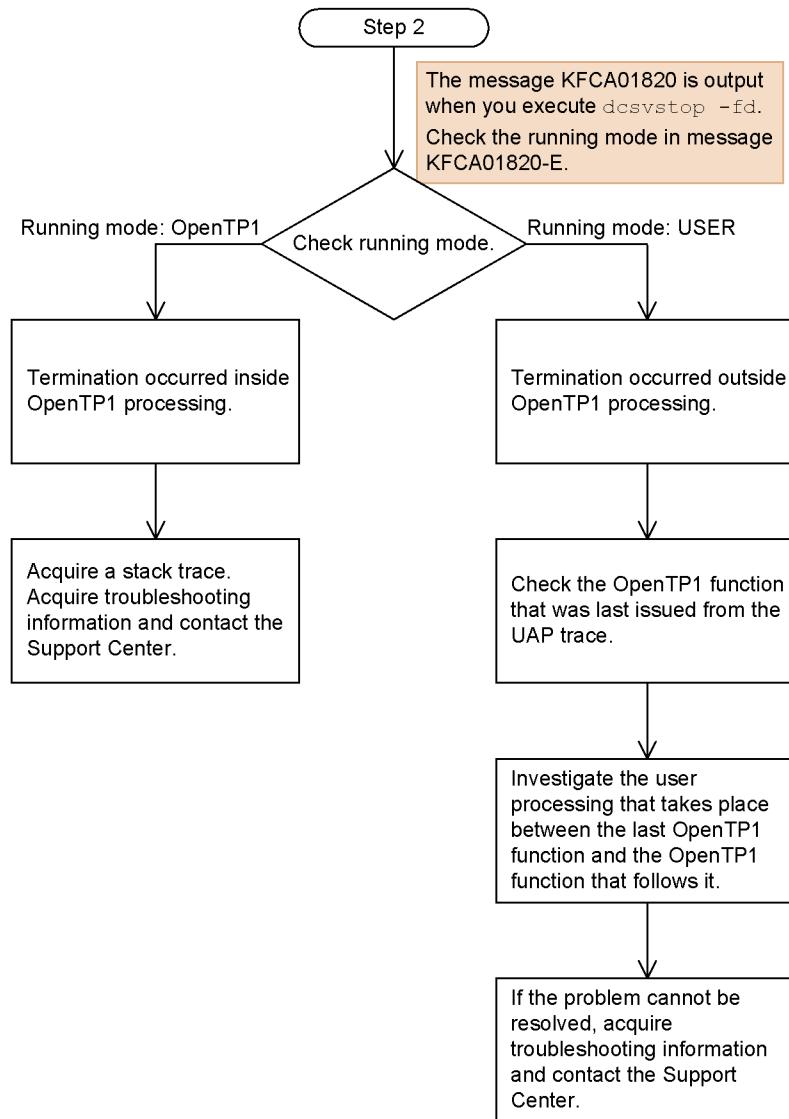
(4) Troubleshooting procedure

The most likely cause of this problem is that the user server you attempted to stop is busy dealing with requests. User servers that are processing a request or have requests stored in the schedule queue when you execute the `dcsvstop` command do not shut down until the requests have all been processed.

Investigate the cause of the error using the flow chart below.

Figure 11-5: Troubleshooting procedure when message KFCA00837-I is output





11.2.6 When the message KFCA00854-E is output

(1) Error event

The following message is output, and the buffer pool for storing messages runs out of space.

```
KFCA00854-E insufficient memory in message buffer pool.
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- Execute the `scdls` command to check the status of a schedule queue (keep a copy of the execution results)
- Execute the `trnls -ta` command to check the status of pending transactions (keep a copy of the execution results)

(3) Information to acquire

- The information in *Table 11-1*
- If time passes and the error event continues to occur, execute the command `dcsvstop -fd user-server-name` to forcibly terminate the user server if possible. If you are able to execute the command, acquire the data in `$DCDIR/spool` after doing so.

(4) Troubleshooting procedure

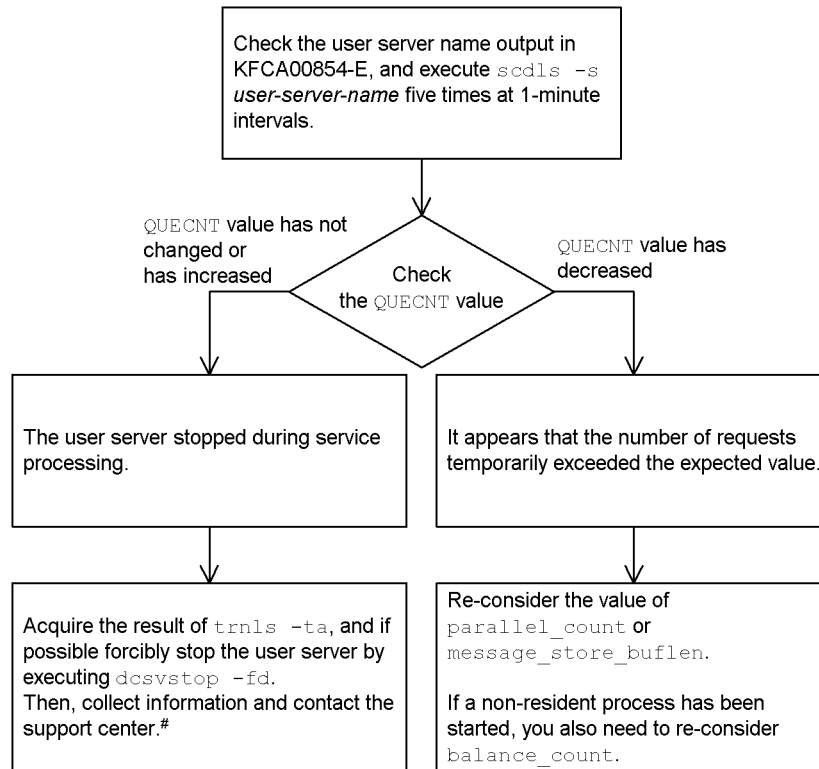
The message KFCA00854-E is output when the size of the buffer pool for storing messages (as specified by the `message_store_buflen` operand) for the particular user server is inadequate. Common causes of this problem are as follows:

- The value specified for the `message_store_buflen` operand is too small.
- Too few user servers are permitted to run concurrently (as specified by the `parallel_count` operand).
- The number of issued RPC requests was greater than expected.
- A service function of the user server is taking a long time to process, causing a backlog of RPC requests in the schedule queue.

Investigate the cause of the error using the flow chart below.

This procedure in this flow chart is designed to be performed immediately after the error event occurs. You may be unable to investigate this type of problem if too much time has passed, though the most likely causes will still be the four listed above. If you are able to identify the cause of the problem from the acquired information, take the appropriate action to resolve it. If you are unable to identify the cause, acquire the data and contact the Support Center.

Figure 11-6: Troubleshooting procedure when message KFCA00854-E is output



#

If a command cannot be executed because of job operation constraints, you must wait for the service processing to terminate. If you execute a command and collect information, you can investigate the cause. However, in some cases information obtained after automatic recovery may not be usable for cause investigation.

Changing the value specified for the `parallel_count` and `message_store_buflen` operands requires that you review the values specified for certain other operands. The following table lists the operands whose values must be modified when you increase the value specified for `parallel_count`. Changes can be made to the `balance_count` operand without affecting any other operands.

Table 11-9: Operands that require modification when the `parallel_count` operand value is increased

No.	Definition file name	Operand	Remarks
1	Process service definition (<code>prc</code>)	<code>prc_process_count</code>	--

11. Troubleshooting

No.	Definition file name	Operand	Remarks
2	Timer service definition (tim)	tim_watch_count	--
3	Transaction service definition (trn)	trn_tran_process_count	--
4	Lock service definition (lck)	lck_limit_for***	For the four operands (ending with user, dam, tam, and mqa), review the values to suit the processing of the applicable user server)
5	<ul style="list-style-type: none"> • User service definition • User service default definition (usrrc) • System common definition (betranrc) • System service common information definition 	max_socket_descriptors	--
6	System environment definition (env)	static_shmpool_size	--
7	System environment definition (env)	dynamic_shmpool_size	Review the value for this operand if you change the value of the dam_tran_process_count operand.
8	DAM service definition (dam)	dam_tran_process_count	Review the value for this operand if a user server whose parallel_count operand you change accesses DAM.
9	TAM service definition (tam)	tam_max_trnnum	Review the value for this operand if a user server whose parallel_count operand you change accesses TAM.
10	RAP-processing listener service definition (name varies)	rap_max_client	Review the value for this operand if a user server whose parallel_count operand you change accesses the RAP-processing listener.

Legend:

--: None

Table 11-10: Operands that require modification when the message_store_buflen operand value is increased

No.	Definition file name	Operand	Remarks
1	System environment definition (<i>env</i>)	<code>static_shmpool_size</code>	For details about estimating shared memory requirements, see the applicable manual or the <i>Release Notes</i> .

11.2.7 When the message KFCA00906-E or KFCA00907-E is output

(1) Error event

Either of the following messages is output, and attempts to access the resource manager fail.

```
KFCA00906-E error occurred while accessing resource manager.
```

```
KFCA00907-E error on XA function of resource manager.
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

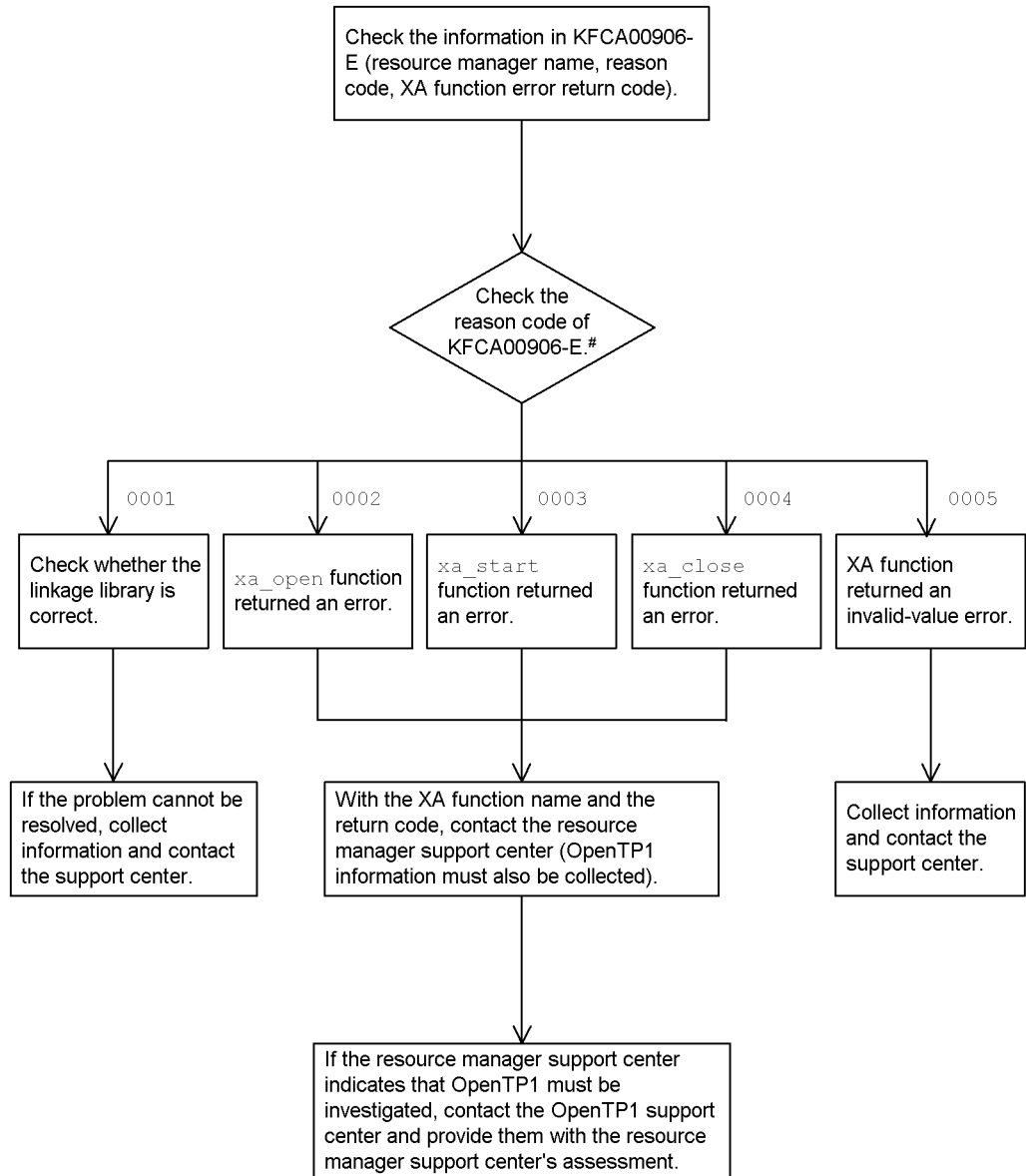
- The information in *Table 11-1*

(4) Troubleshooting procedure

The messages KFCA00906-E and KFCA00907-E are output when an attempt to access a resource manager fails. For linking with outside resource managers (such as Oracle and HiRDB), OpenTP1 uses the industry-standard XA interface to access these resource managers.

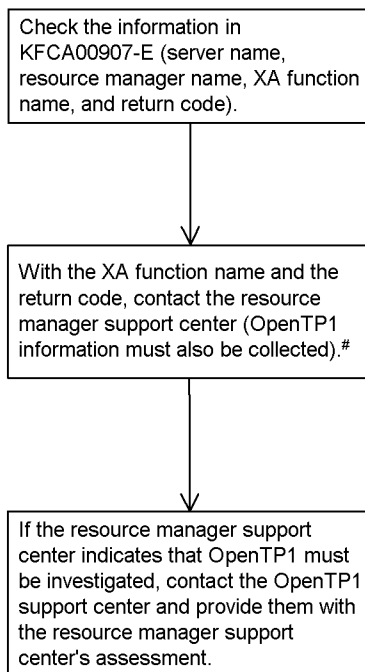
These error messages are output based on information that is returned by the resource manager, not generated by OpenTP1. For this reason, any investigation of this type of error must take place on the resource manager. Use these flow charts to narrow down the cause of the problem, and contact the Support Center for OpenTP1 or the resource manager if necessary.

Figure 11-7: Troubleshooting procedure when message KFCA00906-E is output



#: For the meanings of the reason codes, see the manual *OpenTP1 Messages*.

Figure 11-8: Troubleshooting procedure when message KFCA00907-E is output



#

If KFCA00907-E is output, KFCA00908-E is also output. Because the XID is output in KFCA00908-E, report this XID as well to the support center for the resource manager.

11.2.8 When the message KFCA01803-I is output

(1) Error event

The next time OpenTP1 was started after a successful shutdown, the following message appears and the start mode is set to restart.

```
KFCA01803-I OpenTP1 start mode is determined. start mode: R
```

(2) Items to check

- The date and time when the event occurred
- Whether any operations were performed on the status file while OpenTP1 was shut down

(3) Information to acquire

- The information in *Table 11-1*

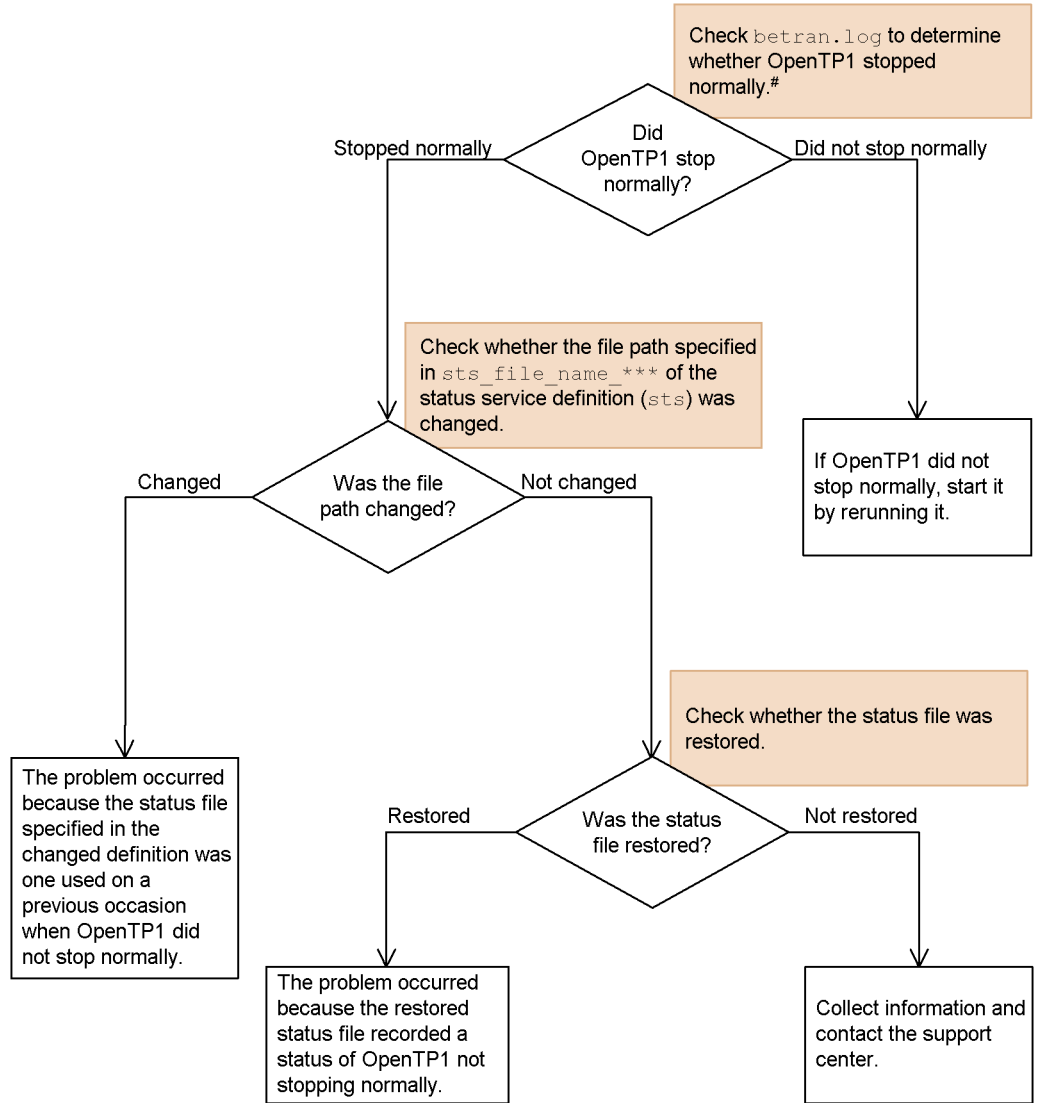
- In a node switching system: The information in *Table 11-1* for both the running and standby systems

(4) Troubleshooting procedure

The most likely cause is that the status file was manipulated in some way (a definition was changed or the file was restored) while OpenTP1 was stopped.

Investigate the cause of the error using the flow chart below.

Figure 11-9: Troubleshooting procedure when message KFCA01803-I is output



 If the messages listed below are output when OpenTP1 stops, you can confirm whether OpenTP1 stopped normally.

```

KFCA01840-I: Now terminating OpenTP1. Termination mode: NORMAL
:
KFCA01893-I: all system servers terminated.
  
```

11.2.9 When the message KFCA01820-E is output

(1) Error event

The following message is output, and a process is forcibly terminated or terminates abnormally.

```
KFCA01820-E server aa....aa has failed.
```

(2) Items to check

- The date and time when the event occurred
- The name of the user server where the event occurred
- The history of operations on the user server
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

- The information in *Table 11-1*

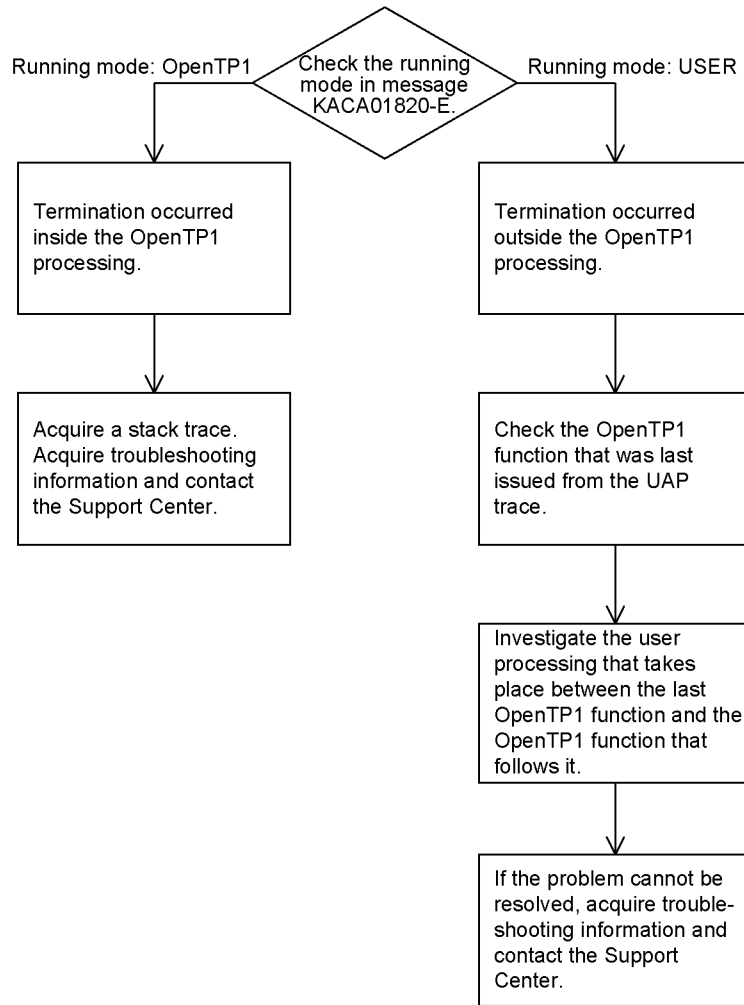
(4) Troubleshooting procedure

The message KFCA01820-E is output when a process is forcibly terminated or terminates abnormally. Common causes of this problem are as follows:

- A timeout occurs in a process which is monitored in real time for a timeout.
- A logic problem related to a user process causes the forced or abnormal termination of a process.

Investigate the cause of the error using the flow chart below.

Figure 11-10: Troubleshooting procedure when message KFCA01820-E is output



11.2.10 When the message KFCA01864-E is output

(1) Error event

The following message is output, and the `dstop` command runs out of time.

```
KFCA01864-E timeout occurred during execution of command dstop.
```

(2) Items to check

- The date and time when the event occurred

- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

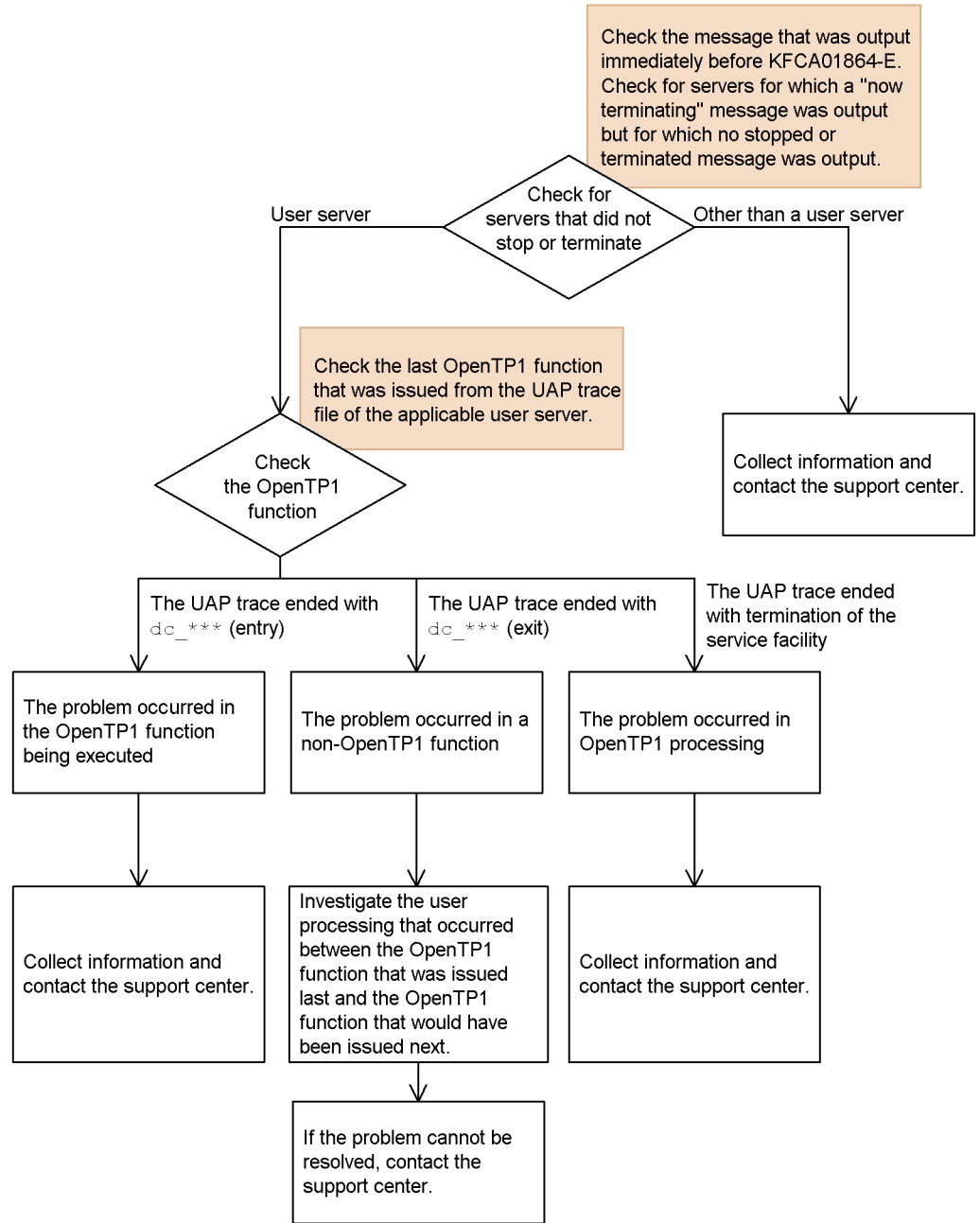
- The information in *Table 11-1*

(4) Troubleshooting procedure

The message KFCA01864-E is output when the length of time specified by the `system_terminate_watch_time` operand in the system environment definition (`env`) has elapsed since the `dcstop` command was executed. The most likely cause is that there was not enough time during OpenTP1 shutdown processing to shut down the user servers.

Investigate the cause of the error using the flow chart below.

Figure 11-11: Troubleshooting procedure when message KFCA01864-E is output



11.2.11 When the error -902: DCTRNER_ROLLBACK is returned

(1) Error event

The `dc_trn_unchained_commit` function or the `dc_trn_chained_commit` function returns an error with error code: -902 and return value: DCTRNER_ROLLBACK.

(2) Items to check

- The date and time when the event occurred
- Whether the `dc_rpc_call` function or `dc_rpc_call_to` function was issued on the user server
 - If either function was issued, the structure of the global transaction
 - If the transaction involved different nodes, the time difference between the nodes
- When using resource managers, the resource manager accessed by the user server

(3) Information to acquire

- The information in *Table 11-1*
- When an RPC is used to generate a child branch on another node: The information in *Table 11-1* for the destination of the RPC

(4) Troubleshooting procedure

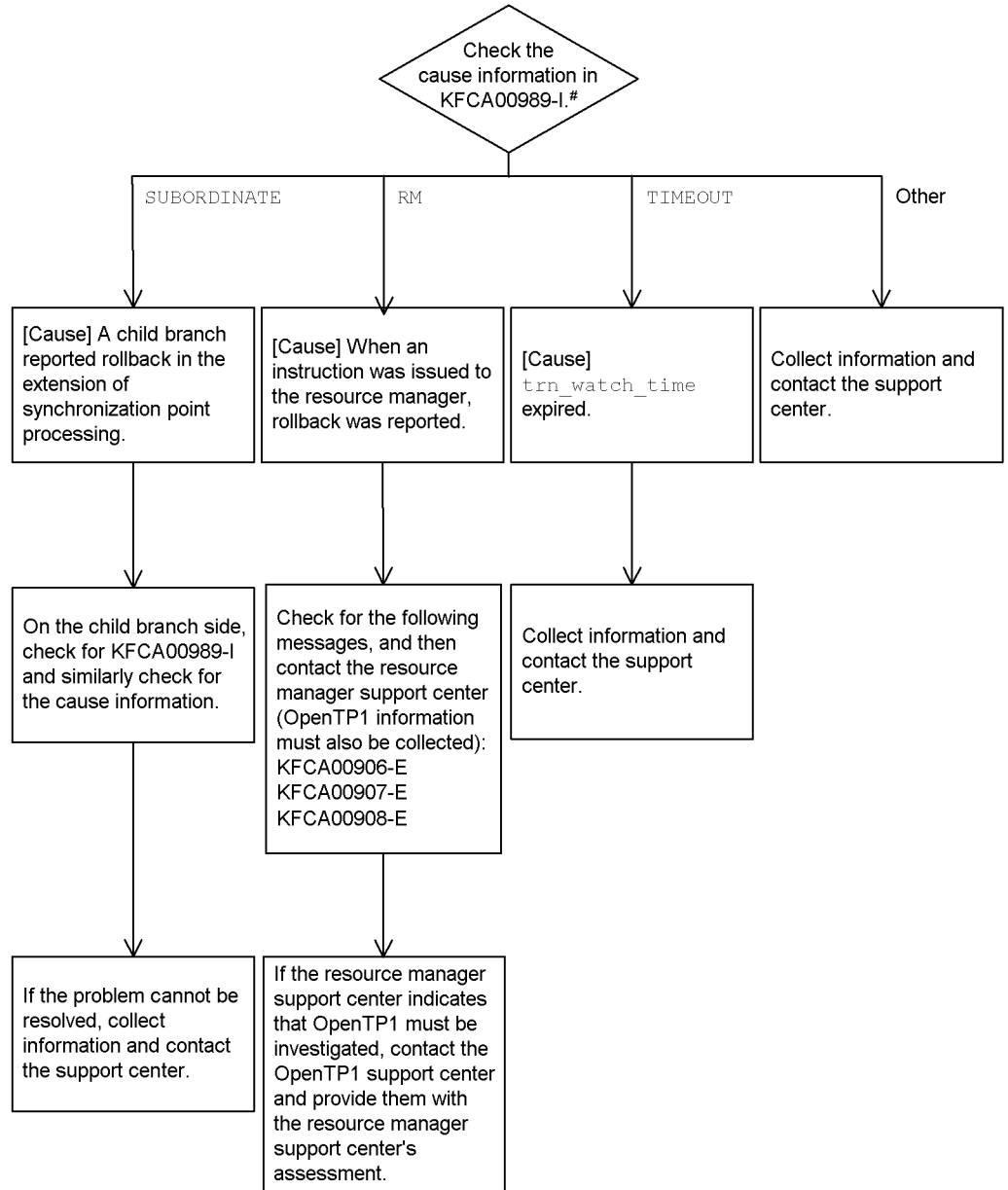
Common causes of this problem are as follows:

- The user server used an RPC to generate a child branch, and the child branch had reason to roll back the transaction after synchronization point processing had begun.
- The user server uses a resource manager other than OpenTP1 (such as Oracle and HiRDB), and the resource manager had reason to roll back the transaction after synchronization point processing had begun. (In this case, the XA function returns an error, and as a result the message KFCA00906-E, KFCA00907-E, or KFCA00908-E is output).
- The value specified by the `trn_watch_time` operand was reached during synchronization point processing.

If `trn_rollback_information_put=all` is specified, the message KFCA00989-I which gives the reason for the rollback will be output. We recommend that you specify this setting as this can help greatly during the troubleshooting process.

The flow chart below describes how to investigate the problem by using the message KFCA00989-I. Use this flow chart to narrow down the cause of the problem, and contact the Support Center for OpenTP1 or the resource manager if necessary.

Figure 11-12: Troubleshooting procedure when the error -902: DCTRNER_ROLLBACK is returned



If KFCA00989-I is not output, collect information and contact the support center.

11.2.12 When a user server fails to start

(1) Error event

A user server did not start during OpenTP1 startup processing, despite a user service structure definition (`usrconf`) being specified.

(2) Items to check

- The date and time when the event occurred
- Any changes made to the OpenTP1 environment immediately before the event occurred

(3) Information to acquire

- `betran.log` on the affected node
When using Windows, acquire the event logs (application log and system log).
- `syslog` file on the affected node
Not required when using Windows.
- Contents of `$DCDIR/conf` on the affected node
If `$DCCONFPATH` specifies a folder other than `$DCDIR/conf`, also acquire the contents of the location specified by `$DCCONFPATH`.
- `$DCDIR/spool/errlog/errlog***`

(4) Troubleshooting procedure

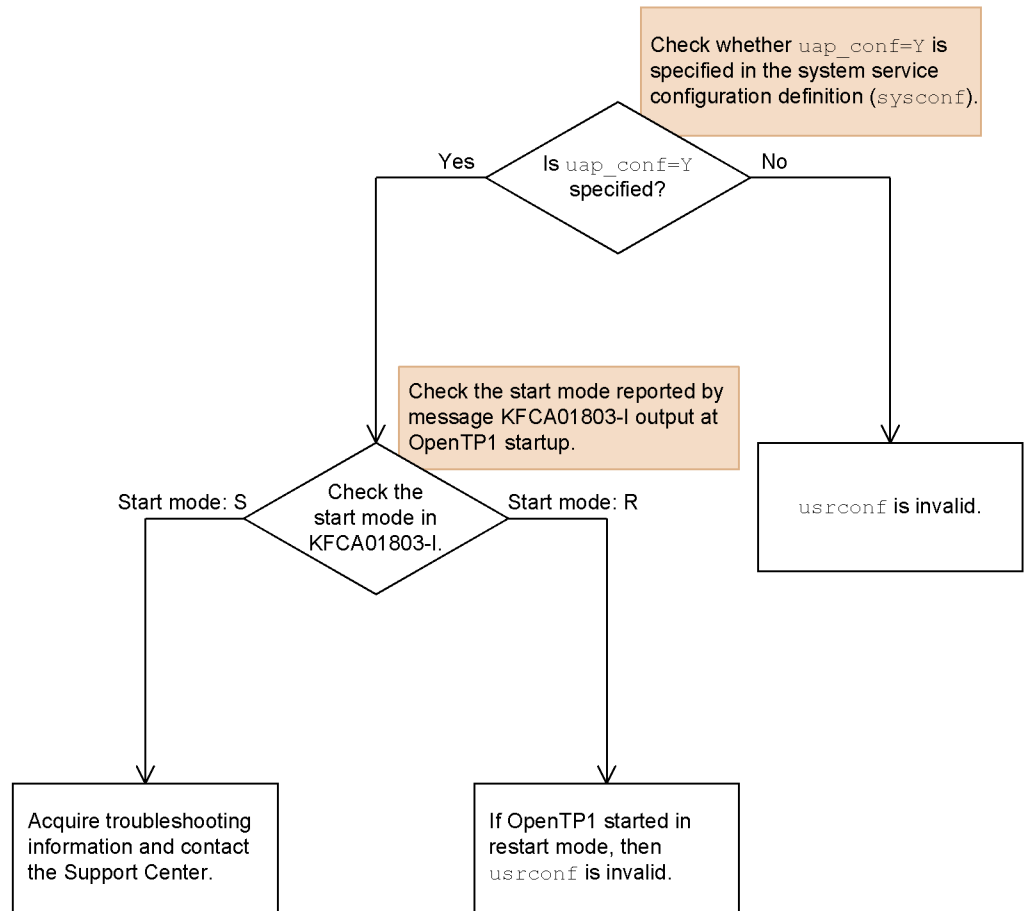
To start a user server specified in the user service structure definition (`usrconf`), both of the following conditions must be satisfied:

- `uap_conf=Y` is specified in the system service structure definition (`sysconf`)
- OpenTP1's startup mode is set to normal startup

The most likely cause of this problem is that the user server did not start because OpenTP1 was started in restart mode.

Investigate the cause of the error using the flow chart below.

Figure 11-13: Troubleshooting procedure when a user server does not start



A similar problem may occur where certain user servers do not start when OpenTP1 is started in restart mode. In restart mode, OpenTP1 resumes the online status of the previous session. Under normal circumstances, user servers that were active in the previous session are also started when OpenTP1 starts in restart mode; however, this does not include user servers for which `node_down_restart=N` is specified. If a user server does not start when OpenTP1 starts in restart mode, check the value of the `node_down_restart` operand for the user server.

Chapter

12. Operation Commands

This chapter explains the operation commands used with OpenTP1 in terms of the input, specification, and details of these commands.

This chapter contains the following sections:

12.1 Overview of operation commands

12.1 Overview of operation commands

12.1.1 Operation command input methods

OpenTP1 operation commands can be input from the shell or can be executed within UAPs.

For details of the procedures for command execution within UAPs, see the manual *OpenTP1 Programming Guide*.

12.1.2 Operation command specification format

Following is the specification format for an operation command:

<i>command-name options command-arguments</i>

(1) Command name

The command name is the file name of the command that is to be executed.

The OpenTP1 operation commands are located under `$DCDIR/bin/`; therefore, `DCDIR/bin` must be added to `PATH`.

(2) Options

In the explanation below, `$` indicates the shell prompt, and `cmd` indicates a command name.

1. An option is a character string beginning with a minus sign (-) that has either no or one flag argument.

Following is the option specification format:

<i>-option-flag</i>

or

<i>-option-flag flag-argument</i>

Explanation:

option-flag

One alphanumeric character (uppercase characters are distinguished from lowercase characters)

flag-argument

Argument for the option flag

2. A series of option flags that do not have flag arguments can be specified consecutively following a single minus sign.

Example:

The following two specifications have the same meaning:

```
$ cmd -a -b -c
$ cmd -abc
```

3. If an option flag requires a flag argument, the flag argument cannot be omitted.
4. When a command is input from the shell and its flag argument contains a whitespace character, the entire flag argument must be enclosed in double quotation marks (").

Example:

Option f with the argument 1 2 must be specified as follows:

```
$ cmd -f "1 2"
```

5. If the same option flag is specified more than once, the last specification is effective.

Example:

```
$ cmd -a 1 / Ineffective -a 2 / Effective
```

6. Options must be specified before command arguments.

Example:

If option flag a requires no flag argument and the following command is entered, `file` and `-b` are both regarded as command arguments:

```
$ cmd -a file -b
```

7. Two consecutive minus signs (--) indicate the end of the options.

Example:

If the following command is entered, `-b` is regarded as a command argument:

```
$ cmd -a -- -b
```

8. An option consisting of a minus sign only cannot be entered.

Example:

If the following command is entered, the minus sign is regarded as a command argument:

```
$ cmd -
```

(3) **Command arguments**

A command argument is used to specify an object of the command's processing.

Usually, up to 1,024 command arguments can be specified. However, the upper limit of command arguments is sometimes less than 1,024 depending on the command.

The following lists specification examples of command arguments. In the explanation below, `$` indicates the shell prompt, `cmd` indicates a command name, `-a` indicates the option flag which requires flag arguments, `-b` indicates the option flag which requires no flag argument, and `param1`, `param2`, ... `paramN` indicates command arguments.

1. If no command argument is specified, the number of command arguments is 0.

Example:

```
$ cmd
$ cmd -b
$ cmd -a 1
```

2. Only `param1` is a command argument and therefore the number of command arguments is 1.

Example:

```
$ cmd param1
$ cmd -b param1
$ cmd -a 1 -b param1
```

3. The elements following `param1` are command arguments and therefore the number of command arguments is `N` (`N` is up to 1,024).

Example:

```
$ cmd param1 param2 ... paramN
$ cmd -b param1 param2 ... paramN
$ cmd -a 1 -b param1 param2 ... paramN
```

12.1.3 Displaying how to use operation commands

If you want to know how to use an operation command, add `-h` as an option to the desired command name to display a description of how to use the command.

However, you cannot specify the `-h` option for the following commands:

- `dcmstart`
- `dcmstop`
- `dcndls`

- dcrasget
- dcreport
- dcreset
- dcsetup
- dcstart
- dcstop
- dcsvstart
- dcsvstop
- jnlardis
- prckill
- prcls
- prcpath
- prcpathls
- prctee
- tptrnls

12.1.4 List of operation commands

The table below lists the OpenTP1 operation commands.

Commands that are listed in the table as being executable in online mode require that the values of the following environment variables specify the OpenTP1 directories:

- DCDIR
- DCCONFPATH

Table 12-1: List of OpenTP1 operation commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
System management	Catalog OpenTP1 into OS, delete OpenTP1 from OS	dcsetup	N	Y	N	Y	N	N	N	Y	N	Superuser
	dcreset (restart process service and reflect definition)	dcreset	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Reserve or release resources for OpenTP1 internal control	dcmakeup	N	Y	N	Y	N	N	N	Y	Y	OpenTP1 administrator
	Start OpenTP1	dcstart	N	Y	N	Y	N	N	N	Y	Y	OpenTP1 administrator
	Terminate OpenTP1	dcstop	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
	Start and terminate collecting system statistics	dcstats	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Start multinode area/subarea	dcstart	N	N	N	Y	Y	Y	Y	N	N	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Terminate multinode area/subarea	dcmstop	N	N	N	Y	Y	Y	Y	N	N	OpenTP1 administrator
Execute an OpenTP1 command from a scenario template	dcjcmde x	N	Y	Y	Y	Y	Y	N	Y	Y	OpenTP1 administrator
Set the values of operands in the system definition	dcjchconf	N	Y	N	Y	N	N	N	N	N	OpenTP1 administrator
Update the domain definition file	dcjnamch	N	Y	Y	Y	Y	Y	Y	Y	Y	OpenTP1 administrator
Display status of OpenTP1 node	dcndls	N	N	N	Y	Y	Y	Y	N	N	General user
Display shared memory utilization status	dcshmls	N	N	Y	N	Y	Y	Y	N	Y	General user
Display execution status of temporary closing	rpcstat	N	N	Y	N	Y	Y	Y	N	Y	General user

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Redirect standard output and standard error output	prctee	N	Y	N	Y	N	N	N	N	N	OpenTP1 administrator
Start and restart of <i>prctee</i> process	prctctrl	N	Y	Y	Y	Y	Y	N	Y	Y	Superuser
Acquire maintenance data	dcrasget	N	Y	Y	Y	Y	Y	Y	N	N	OpenTP1 administrator
Edit system statistics and output them to standard output in real time	dcreport	N	N	Y	N	N	Y	Y	N	N	OpenTP1 administrator
Delete troubleshooting information	dccspool	N	Y	Y	Y	Y	Y	Y	Y	N	OpenTP1 administrator
Check system definitions	dcdefchk	N	Y	Y# 2	Y	N	Y#2	N	Y	Y	OpenTP1 administrator
Display product information	dcpplist	N	Y	Y	Y	Y	Y	Y	N	N	OpenTP1 administrator
Server management	dcsvstart	Y	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
	Terminate server	dcsvstop	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Display status of server	prcls	N	N	Y	N	Y	Y	Y	N	General user
	Display search path names for user server and for command activated from user server	prcpathls	N	N	Y	N	N	Y	Y	N	General user
	Forcibly terminate OpenTP1 process	prckill	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Change search path names for user server and for command activated from user server	prcpath	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
Schedule management	Display scheduling status	scdls	N	N	Y	N	N	Y	Y	N	General user
	Shut down scheduling	scdhold	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
	Restart scheduling	scdrls	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Change the number of processes	scdchprc	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Terminate and restart processes	scdrsprc	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
Transaction management	Display status of transactions	trnls	N	N	Y	N	N	Y	Y	N	General user
	Commit transactions	trncmt	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Roll back transactions	trnrbk	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Terminate transactions forcibly	trnfgt	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Start and terminate collecting transaction statistics	trnstics	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator
	Delete undecided transaction information file	trndlinf	N	Y	Y	Y	Y	Y	Y	Y	OpenTP1 administrator

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
	Display undecided transaction information for OSI TP communication	tptrnls	N	N	Y	N	N	Y	N	N	General user	
XA resource management	Display XAR event trace information	xarevtr	N	Y	Y	Y	N	Y	N	N	General user	
	Display the status of XAR files	xarfills	N	Y	Y	Y	N	Y	Y	N	General user	
	Change the status of XAR transactions	xarforce	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator	
	Shut down the XA resource service	xarhold	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator	
	Create an XAR file	xarinit	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Display XAR transaction information	xarls	N	N	Y	N	N	Y	Y	N	N	General user

12. Operation Commands

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
	Release an XA resource service from shutdown status	xarrles	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Delete an XAR file	xarrm	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
Lock management	Display lock information	lckls	N	N	Y	N	N	Y	Y	N	N	General user
	Display lock control table pool information	lockpool	N	N	Y	N	N	Y	Y	N	N	General user
	Delete deadlock information file and timeout information file	lckrminf	N	Y	Y	Y	Y	Y	Y	Y	Y	OpenTP1 administrator
Name management	Check OpenTP1 startup, clear cache	namaliv echk	N	N	Y	N	N	Y	Y	Y	N	OpenTP1 administrator

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
Register or delete domain representative schedule service	namdomain_setup	N	N	Y	Y	Y	Y	Y	N	N	Superuser	
Change domain configuration (by using the system common definition)	namndchg	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator	
Change the domain configuration (by using the domain definition file)	namchgfl	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator	
Perform forced invalidation of startup notification information	namunavl	N	N	Y	N	N	Y	N	Y	N	OpenTP1 administrator	
Display OpenTP1 server information	namsvinf	N	N	Y	N	N	Y	Y	N	N	OpenTP1 administrator	

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
	Manipulate RPC suppression list	namblad	N	N	Y	N	N	Y	Y	Y	OpenTP1 administrator	
Message log management	Display message log file	logcat	N	N	Y	N	Y	Y	Y	N	N	General user
	Switch message log realtime output function	logcon	N	N	Y	N	Y	Y	Y	Y	N	OpenTP1 administrator
Audit log management	Set the environment for audit log output	dcaudit setup	N	Y	N	Y	N	N	N	Y	Y	Superuser
OpenTP1 file management	Initialize an OpenTP1 file system	filmkfs	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Display status of an OpenTP1 file system	filstat fs	N	Y	Y	Y	Y	Y	Y	N	N	General user
	Display contents of an OpenTP1 file system	fills	N	Y	Y	Y	Y	Y	Y	N	N	General user
	Back up an OpenTP1 file system	filbkup	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Restore an OpenTP1 file system	filrstr	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	filchgrp	N	Y	Y	Y	Y	Y	Y	Y	N	OpenTP1 administrator
	filchmod	N	Y	Y	Y	Y	Y	Y	Y	N	OpenTP1 administrator
	filchown	N	Y	Y	Y	Y	Y	Y	Y	N	OpenTP1 administrator
Status file management	stsinit	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
	stsls	N	N	Y	N	Y	Y	Y	N	N	General user
	stsfills	N	Y	Y	Y	Y	Y	Y	N	N	General user
	stsopen	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
	stsclose	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
	stsrn	N	Y	Y	Y	Y	Y	Y	Y	N	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
	Swap status files	stsswap	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
Journal file management	Initialize journal files	jnlinit	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
	Display journal information	jnlis	N	Y	Y	Y	Y	Y	Y	N	N	General user
	Display journal information that is read during restart	jnlrinf	N	N	Y	N	Y	Y	N	N	N	OpenTP1 administrator
	Open journal files	jnlpnfg	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
	Close journal files	jnlclsfg	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator
	Allocate journal physical files	jnladdpf	Y	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Delete journal physical files	jnldepf	Y	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Swap journal files	jnlswpfg	N	N	Y	N	Y	Y	Y	Y	Y	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Delete journal files	jnlrm	N	N	Y	N	Y	Y	N	Y	N	OpenTP1 administrator
Change status of journal files	jnlchgfg	N	Y	Y	Y	Y	Y	N	Y	Y	OpenTP1 administrator
Unload journal files	jnlunlfg	N	Y	Y	Y	Y	Y	N	Y	Y	OpenTP1 administrator
Control the automatic unloading facility	jnlatl	N	N	Y	N	N	Y	N	Y	Y	OpenTP1 administrator
Recover journal files	jnlmkrf	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
Integrate file recovery journals	jnlcolc	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
Copy unload journal files	jnlcopy	N	Y	Y	Y	N	Y	N	Y	N	OpenTP1 administrator
Display archive status	jnlarls	N	N	N	N	Y	Y	Y	N	N	General user

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Edit and output unload journal files or global archive unload journal files	jnlredit	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
Output records of unload journal files or global archive unload journal files	jnlrput	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
Chronologically sort and merge unload journal files and global archive unload journal files	jnlstts	N	N	N	Y	Y	Y	N	Y	N	OpenTP1 administrator
Output uptime statistics	jnlstts	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator
Output MCF uptime statistics	jnlmct	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
	Forces to release connection to resource group	jnlardis	N	N	N	N	Y	N	N	Y	Y	OpenTP1 administrator
DAM file management ^{#3}	Initialize a physical file	damload	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Display status of logical files	damls	N	N	Y	N	N	Y	Y	N	N	General user
	Add a logical file	damadd	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Remove a logical file	damrm	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Shut down a logical file logically	damhold	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Release logical file from shutdown	damrles	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Delete a physical file	damdel	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Back up a physical file	dambkup	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
	Restore a physical file	damrstr	N	Y	Y# 4	Y	Y#4	Y#4	N	Y	N	OpenTP1 administrator
	Recover physical files	damfrc	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Set a threshold for the number of cache blocks	damchdef	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Acquire the number of cache blocks	damchinf	N	N	Y	N	N	Y	Y	N	N	General user
TAM file management ^{#5}	Initialize a TAM file	tamcre	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Display status of TAM files	tamlis	N	N	Y	N	N	Y	Y	N	N	General user
	Add a TAM table	tamadd	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Remove a TAM table	tamrm	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator
	Shut down a TAM table logically	tamhold	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
Release TAM table from shutdown	tamrles	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator	
Load a TAM table	tamload	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator	
Unload a TAM table	tamunload	N	N	Y	N	N	Y	Y	Y	Y	OpenTP1 administrator	
Delete a TAM table	tamdcl	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator	
Back up a TAM table	tambkup	N	Y	Y	Y	Y	Y	N	Y	N	OpenTP1 administrator	
Restore a TAM table	tamrstr	N	Y	Y#6	Y	Y#6	Y#6	N	Y	N	OpenTP1 administrator	
Recover a TAM table	tamfrcl	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator	
Convert TAM locked resource names	tamlckls	N	N	Y	N	N	Y	Y	N	N	General user	
Display synonym information about a hash-form at TAM file and TAM table	tamhsls	N	Y	Y	Y	N	Y	N	N	N	OpenTP1 administrator	

12. Operation Commands

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
Message queue file management	Display status of queue groups	que1s	N	N	Y	N	N	Y	N	N	General user	
	Allocate a physical file for message queue	queinit	N	Y	Y	Y	N	Y	N	Y	OpenTP1 administrator	
	Delete a physical file for message queue	querm	N	Y	Y	Y	N	Y	N	Y	OpenTP1 administrator	
Resource manager management	Display resource manager information	trnlstrm	N	Y	Y	Y	N	Y	N	N	Y	General user
	Catalog resource manager	trnlkr m	N	Y	N	Y	N	N	N	Y	Y	OpenTP1 administrator
	Create a transaction control object file	trnmkobj	N	Y	Y	Y	N	Y	N	Y	Y	OpenTP1 administrator
Trace management	Edit and output UAP trace information	uatdump#7	N	Y	N	Y	N	N	N	N	N	General user
	Merge RPC trace information	rpcmrg	N	Y	N	Y	N	N	N	N	N	General user

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
	Output RPC trace information	rpcdump	N	Y	N	Y	Y	N	N	N	N	General user
	Output shared memory dump information	usmdump	N	N	Y	N	Y	Y	Y	N	N	OpenTP1 administrator
Remote API management	Set up environment for execution the remote API facility	rapsetup	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Automatically generate definitions for using the remote API facility	rapdfgen	N	Y	Y#8	Y	Y#8	Y#8	N	Y	N	OpenTP1 administrator
	Display status of a RAP-processing listener or a RAP-processing server	rapls	N	N	Y	N	N	Y	N	N	N	OpenTP1 administrator

12. Operation Commands

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
Management of performance verification traces	Edit and output trace information files	prfed	N	Y	Y	Y	Y	Y	N	N	N	OpenTP1 administrator
	Get trace information files	prfget	N	Y	Y	Y	Y	Y	N	Y	Y	OpenTP1 administrator
Management of real-time statistics	Edit and output RTS log files	rtsedit	N	Y	Y	Y	Y	Y	N	N	N	General user
	Output real-time statistics to standard output	rtsls	N	N	Y	N	N	Y	N	N	N	General user
	Set execution environment for real-time statistics service	rtsetup	N	Y	N	Y	N	N	N	Y	N	OpenTP1 administrator
	Change settings for real-time statistics	rtstats	N	N	Y	N	N	Y	N	Y	Y	OpenTP1 administrator
Connection management ^{#9,10}	Display status of connections	mcftlscn	N	N	Y	N	N	Y	Y	N	N	General user

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Establish connections	mcfactcn	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
Release connections	mcfdtcn	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
Switch connections	mcfthcn	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
Display network status	mcfslsn	N	N	Y	N	N	Y	Y	N	N	General user
Start accepting server-type connection establishment requests	mcfonln	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
End accepting server-type connection establishment requests	mcfofln	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
Display the status of multi-processing of messages	mcfstord	N	N	Y	N	N	Y	Y	N	N	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Application management #9, #10	Display status of applications	mcfalsapp	N	N	Y	N	N	Y	N	N	General user
	Shutdown applications	mcfadctap	N	N	Y	N	N	Y	N	Y#11	OpenTP1 administrator
	Release shutdown of applications	mcfactap	N	N	Y	N	N	Y	N	Y#11	OpenTP1 administrator
	Initialize abnormal termination count for applications	mcfaclocap	N	N	Y	N	N	Y	N	Y#11	OpenTP1 administrator
	Display application-related timer start requests	mcfalstap	N	N	Y	N	N	Y	N	N	General user
	Delete timer activation requests for applications	mcfadltap	N	N	Y	N	N	Y	N	Y#11	OpenTP1 administrator
Application operation support #9, #10	Start application programs	mcfuevt	N	N	Y	N	N	Y	N	Y#11	General user

Function		Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
				Off	On	Off	On					
							A-J avb	A-J not avb				
Logical terminal management ^{#9} . #10	Display status of logical terminals	mcfctlsl e	N	N	Y	N	N	Y	Y	N	N	General user
	Shutdown logical terminals	mcftdct le	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Release shutdown of logical terminals	mcfctact le	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Skip first message in a logical terminal message queue	mcfctspq le	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Hold process of a logical terminal output queue	mcfthld oq	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Release held process of a logical terminal output queue	mcftrls oq	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Delete output queues for logical terminals	mcftdlq le	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Start message journal collection for logical terminals	mcftactmj	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	mcftdctmj	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	mcftendct	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	mcftstalt	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	mcftedalt	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
Service group management ^{#9, #10}	mcftlssg	N	N	Y	N	N	Y	Y	N	N	General user
	mcftdctsg	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
Release service groups from shutdown	mcfactsg	N	N	Y	N	N	Y	N	Y#1 1	OpenTP1 administrator		
Hold process of input queue for service groups	mcfthldiq	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator	
Release held process of input queues for service groups	mcftrlsiq	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator	
Delete input queues for service groups	mcftdlqsg	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator	
Service management ^{#9} , #10	Display status of services	mcftrlssv	N	N	Y	N	N	Y	Y	N	N	General user
	Shut down services	mcftdctsv	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
	Release services from shutdown	mcfactsv	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator

12. Operation Commands

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right
			Off	On	Off	On					
						A-J avb	A-J not avb				
Session management ^{#9, #10}	Start a session	mcftactss	N	N	Y	N	N	Y	N	Y ^{#1} ₁	OpenTP1 administrator
	Terminate a session	mcftdctss	N	N	Y	N	N	Y	N	Y ^{#1} ₁	OpenTP1 administrator
Buffer management ^{#9, #10}	Display status of buffer group usage	mcftlsbuf	N	N	Y	N	N	Y	N	N	General user
Map management ^{#9, #10}	Change map file path name	dcmapchg	N	N	Y	N	N	Y	N	N	OpenTP1 administrator
	Display loaded map file resources	dcmapls	N	N	Y	N	N	Y	N	N	OpenTP1 administrator
Queue management ^{#9, #10}	Output contents of input/output queues	mcftdmpqu	N	N	Y	N	N	Y	N	Y ^{#1} ₁	General user
MCF trace collection management ^{#9, #10}	Swap an MCF trace file forcibly	mcftswptr	N	N	Y	N	N	Y	N	N	OpenTP1 administrator
	Start acquiring MCF trace	mcftstrtr	N	N	Y	N	N	Y	N	N	OpenTP1 administrator

Function	Cmd name	Df	Not installed		Installed			U	A log	C log	Access right	
			Off	On	Off	On						
						A-J avb	A-J not avb					
	End acquiring MCF trace	mcfstptr	N	N	Y	N	N	Y	N	N	OpenTP1 administrator	
Management of MCF statistics	Edit MCF statistics	mcfreport	N	Y	N	Y	N	N	N	N	OpenTP1 administrator	
	Output MCF statistics	mcfstats	N	N	Y	N	N	Y	Y	N	Y#1 1	OpenTP1 administrator
MCF communication service management ^{#9, #12}	Stop part of MCF communication service	mcfstop	N	N	Y	N	N	Y	N	N	Y#1 1	OpenTP1 administrator
	Start part of MCF communication service	mftstart	N	N	Y	N	N	Y	N	N	Y#1 1	OpenTP1 administrator
	Reference MCF communication service status	mftlsc	N	N	Y	N	N	Y	N	N	N	OpenTP1 administrator
User timer management ^{#9, #10}	Display user timer monitoring status	mftlstm	N	N	Y	N	N	Y	Y	N	N	General user

Legend:

Cmd name: Command name

Df: Included in definition

Not installed: TP1/multi-uninstalled node

Installed: TP1/multi-installed node

Off: Executeable in offline mode

On: Executable in online mode

A-J avb: Archive journal file available

A-J not avb: Archive journal file not available

U: Executable from UAP

A log: Output of audit log

C log: Output of command log

Y: Can be installed or executed.

N: Cannot be installed or executed.

Note 1:

Each protocol has special commands. For details about protocol-specific commands, see the manual for that protocol.

Note 2:

For information about command logs and their output destinations, see 3.6.2(1) *Command log output format*.

#1: When executing this command from UAP, execute it on the background.

#2: When the command is executed while OpenTP1 is running, the applicable values are those specified in the system definition before the command is executed, not the values valid for the current operation of OpenTP1.

#3: The commands are available only when DAM files are used.

#4: This command can be executed during an online session even after a logical file has been shut down with the `damhold` command or disconnected with the `damrm` command.

#5: The commands are available only when TAM files are used.

#6: This command can be executed during an online session even after a TAM table has been shut down logically with the `tamhold` command or after a TAM table file has been disconnected with the `tamrm` command.

#7: For details, see the manual *OpenTP1 Tester and UAP Trace User's Guide*.

#8: The command can be executed only when the RAP-processing listener and the RAP-processing server are stopped normally.

#9: The commands are available only when MCF is used.

#10: If processing is not completed within 180 seconds from the time the operation command is entered, a timeout error occurs and the processing terminates.

#11: To acquire command logs, you need to set an environment variable. For details about acquiring command logs, see *3.6.2(2) Acquiring command logs*.

#12: Restrictions may apply depending on the defined environment or the timing of entering the operation command. For details, see *3.13.3 Replacing part of an MCF communication service* and the command's description in *13. Details of Operation Commands*.

12.1.5 Notes on operation command entry

- If an operation command is entered during OpenTP1 termination, forced termination, or abnormal termination, the command's processing may terminate abnormally.
- To specify an MCF operation command in an argument of the rsh (remote shell) command, enclose it in a pair of single quotation marks (' ').
- The operation commands are processed in the order they are entered. When many operation commands are entered, they have to be queued. The more operation commands are entered, the longer the waiting time. Therefore, when including the operation commands into part of an application's operation, design the operation to minimize the number of inputs.
- Hitachi recommends that you set the same value for the OS priority for executing operation commands as the value for the priority for OpenTP1. If you execute the `dcstart` command with a lower set priority level, OpenTP1 might not start. For details, see *10.6.1(10) Excessive load on the CPU*.

Chapter

13. Details of Operation Commands

This section explains the operation commands in alphabetical order.

damadd (Add a logical file)

Format

```
damadd [{-d|-n[-f]}] [-1 boundary-for-reusing-cache-blocks]
      logical-file-name physical-file-name
```

Function

The `damadd` command adds and catalogs a specified logical file.

Another logical file cannot be allocated to an existing physical file.

Options

- `-d`

Adds and catalogs a logical file as a file using the deferred update facility. This option cannot be specified if no file using the deferred update facility is specified in the DAM service definition.

This option cannot be specified with the `-n` option.

- `-n`

Adds and catalogs a logical file as a file not to be recovered.

If the `-n` option is not specified, a logical file is added and cataloged as a file to be recovered.

The `-n` and `-d` options are mutually exclusive.

- `-f`

Adds and catalogs a logical file as a file not to be recovered, for access by cacheless access. Specify this option together with the `-n` option.

- `-1 boundary-for-reusing-cache-blocks~<unsigned integer>((0-4000000))<<0>>`

Specify the boundary for reusing cache blocks for a DAM file.

If you do not specify this option, the boundary for reusing cache blocks for the registered DAM file is assumed to be zero regardless of whether the `dam_default_cache_num` operand is specified in the DAM service definition.

Command arguments

- `logical-file-name~<identifier, 1-8 characters>`

Specifies the name of the logical file to be added and cataloged.

■ *physical-file-name*~<path name>

Specifies the name of the physical file that is to correspond to the logical file, expressed as the full path name.

Output messages

Message ID	Description	Output destination
KFCA01650-E	Specification of a command argument is invalid.	Standard error output
KFCA01651-E	Logical file name exceeds 8 characters.	Standard error output
KFCA01652-E	Physical file name exceeds 63 characters.	Standard error output
KFCA01653-I	Help message	Standard output
KFCA01654-E	User does not have access right to a special file.	Standard error output
KFCA01655-E	User does not have access right to the file.	Standard error output
KFCA01656-E	Block length is too large.	Standard error output
KFCA01657-E	Specified file is not a DAM file.	Standard error output
KFCA01658-E	Logical file name already exists.	Standard error output
KFCA01659-E	Physical file name already exists.	Standard error output
KFCA01660-E	Initialization has not been performed.	Standard error output
KFCA01661-E	I/O error occurred.	Standard error output
KFCA01662-E	Memory is inadequate.	Standard error output
KFCA01663-E	Physical file was not found.	Standard error output
KFCA01664-E	Additional area is inadequate.	Standard error output
KFCA01665-E	Number of opened files exceeds the maximum.	Standard error output
KFCA01666-E	Physical file name is invalid.	Standard error output
KFCA01667-E	Versions do not match.	Standard error output
KFCA01679-E	DAM service has not been started.	Standard error output
KFCA01680-E	Name service has not been started.	Standard error output
KFCA01681-E	Communication error occurred.	Standard error output
KFCA01682-E	Network error occurred.	Standard error output
KFCA01683-E	Overtime occurred.	Standard error output

damadd (Add a logical file)

Message ID	Description	Output destination
KFCA01684-E	DAM service is terminating.	Standard error output
KFCA01695-E	DAM file is incomplete.	Standard error output
KFCA01696-E	Error occurred during definition analysis.	Standard error output
KFCA01698-E	Insufficient record lock segment	Standard error output
KFCA02524-E	No additional file can be cataloged by deferred specification.	Standard error output

Notes

- A logical file corresponding to a physical file cannot be added or cataloged if its block length is greater than the block length of the existing physical file.
- The number of files to be added and cataloged cannot exceed the value defined in the DAM service definition of the maximum number of logical files to be added in the online mode.

dambkup (Back up a physical file)

Format

```
dambkup [-c batch-processing-blocks-count] [-o|-d]
        {physical-file-name backup-file-name|-s physical-file-name}
```

Function

The `dambkup` command backs up the contents of a specified physical file into a specified file or standard output.

Options

- `-c batch-processing-blocks-count`~<unsigned integer> ((1-2147483647)) <<10>>

Specifies the number of blocks to be input or output simultaneously. The specified number of blocks will be input from or output to disk.

- `-o`

Backs up physical files in the online mode. This option is valid only when the target DAM file is in the online mode. If the option is not specified, physical files are backed up in the offline mode. In this case, take the following steps to back up a file:

1. Execute the `damhold` command to shut down the logical files logically.
2. Execute the `damrm` command to remove the logically shut down logical files from the online system.
3. Execute the `dambkup` command with its `-o` option not specified to back up the DAM file.

- `-d`

Backs up only the user data in the backup source physical file, excluding the management block and the check bytes at the start of the file.

- `-s`

This option must be specified to output the backup to the standard output.

Command arguments

- `physical-file-name`~<path name>

Specifies the name of the backup source physical file, expressed as the full path name.

- `backup-file-name`~<path name>

Specifies the name of the backup destination file.

This command argument cannot be specified with the `-s` option.

Notes

- The `dambkup` command cannot be executed for DAM files used in the online mode if its `-o` option is not specified.
- Upon completion of backing up, a message (KFCA02531-I) is displayed. This message is usually output to the standard output, but output to the standard error output when the `-s` option is specified.
- The message contains the generation number and block number of the journal file to be recovered. Unload journal files with higher generation numbers and block numbers are needed when the DAM file is recovered.
- The `dambkup` command that is executed with its `-o` and `-s` options specified may terminate abnormally and the KFCA00107-E message is output when the amount of available area under current directory is smaller than the following:
Capacity of DAM file to be backed up + 128 bytes or greater ((Number of blocks + 1) x block length)
In this case, check the disk capacity, delete the unnecessary file, and then allocate the area.
- Data backed up using the `-d` option cannot be restored to a physical file.
- You cannot extract only user information from a DAM file that is being used in online mode.

damchdef (Set a threshold for the number of cache blocks)

Format

```
damchdef logical-file-name threshold-for-the-number-of-cache-blocks
```

Function

The `damchdef` command dynamically sets a threshold for the number of cache blocks during online operation.

Command arguments

- *logical-file-name*~<identifier of 1 to 8 characters>

Specify the name of a logical file for which you want to set a threshold for the number of cache blocks. The logical file to be specified must be in the logical shutdown status.

- *threshold-for-the-number-of-cache-blocks*~<unsigned integer>((0-4000000))

Specify a threshold for the number of cache blocks managed by a single DAM file.

Output messages

Message ID	Description	Output destination
KFCA01650-E	The specified command argument is incorrect.	Standard error output
KFCA01651-E	The name of the logical file exceeded 8 characters.	Standard error output
KFCA01670-E	The applicable logical file is not found.	Standard error output
KFCA01679-E	The DAM service is not started.	Standard error output
KFCA02568-I	Help message	Standard output
KFCA02571-E	The specified threshold for the number of cache blocks is invalid.	Standard error output

Notes

- The `damchdef` command is valid only for logical files shut-down logically.
- The threshold specified using the `damchdef` command is only valid while the target logical file is online. If you delete the target logical file from online operation using the `damrm` command, the specified threshold becomes invalid.

damchinf (Acquire the number of cache blocks)

Format

```
damchinf [-c] [-i] [logical-file-name]
```

Function

The `damchinf` command acquires the number of cache blocks connected to the current cache block chain.

Options

- `-c`
Adds a comma (,) to separate information items.
- `-i`
Displays only the total information about the cache memory. If you omit this option, the cache memory information about each logical file is also displayed.

Command arguments

- *logical-file-name* ~<identifier of 1 to 8 characters>
Specify the name of the logical file for which cache memory information needs to be displayed. If you omit this argument, the cache memory information about all the logical files managed by the DAM service is displayed.

Output format

```
CleanUP Count:aa...aa Next CleanUP FILE-No:bb...bb Using Rate:cc...cc%
FileNo  FileName  BlkLen  BlkNum  CchBlkNum  PreservNum  LimitNum  ReUse
dd...dd  ee...ee   ff...ff  gg...gg  hh...hh    ii...ii    jj...jj   kk...kk
```

1. This section is not output when you specify the `-i` option.
2. This section is output repeatedly for all the logical files managed by the DAM service if you do not specify the logical file name.

aa...aa

Number of times clean-up processing is performed

bb...bb

Target file number for the next clean-up processing

cc...cc

Percentage used of the shared memory pool for the DAM cache memory

dd...dd

File number (this is not always output sequentially)

ee...ee

Logical file name (up to 8 characters)

ff..ff

Length of a block in the logical file (*ee...ee*)

gg...gg

Number of blocks in the logical file (*ee...ee*)

hh...hh

Number of cache blocks connected to the logical file (*ee...ee*)

ii...ii

Maintenance information

jj..jj

Threshold for the number of cache blocks

kk...kk

Whether the reuse cache block chain exists

Exist: The reuse cache block chain exists.

None: The reuse cache block chain does not exist.

Output messages

Message ID	Description	Output destination
KFCA01650-E	The specified command argument is incorrect.	Standard error output
KFCA01651-E	The name of the logical file exceeded 8 characters.	Standard error output
KFCA01670-E	The applicable logical file is not found.	Standard error output
KFCA01679-E	The DAM service is not started.	Standard error output
KFCA02548-I	Help message	Standard output

Notes

When you execute the `damchinf` command with the `damadd` command or the `damrm` command, information may not be output or invalid information may be output. If that

damchinf (Acquire the number of cache blocks)

occurs, reexecute the damchinf command.

damdel (Delete a physical file)

Format

```
damdel physical-file-name
```

Function

The `damdel` command deletes a specified physical file.

The specified physical file cannot be deleted if it is being used in the online mode or in some other offline processing. To delete a physical file being used in the online mode, the corresponding logical file must be removed from the online system by the `damrm` command, and then the `damdel` command can be executed.

Command arguments

- *physical-file-name*~<path name>

Specifies the name of the physical file to be deleted, expressed as the full path name.

damfrc (Recover logical files)

Format

```
damfrc [-s] [-e] [-g] [-k key] [-n] [-m] [-c] recovery-
definition-file-name journal-file-name [[  $\Delta$ journal-file-name] . . .]
```

Function

The `damfrc` command uses specified journal files to restore logical files specified in the DAM service definition to their status immediately before an error occurred.

If the `damfrc` command is executed for a file not to be recovered, the file cannot be recovered.

Options

■ `-s`

Specifies that the previous DAM FRC is not to be inherited.

When this option is omitted, the previous DAM FRC is inherited.

■ `-e`

Specifies that inheritance files are to be deleted upon termination of DAM FRC.

When DAM FRC is executed with this option specified, the `-s` option must be specified the next time DAM FRC is executed.

When this option is omitted, inheritance files are not deleted.

■ `-g`

Specifies that DAM FRC is to be executed even if the `-s` option is specified and an unload journal file with journal generation number 1 is not specified.

When this option is omitted, processing is cancelled if the `-s` option is specified and journal generation number 1 is not specified as an unload journal file.

When the `-s` option is omitted, this option is ignored, if specified.

■ `-k key~((001-999)) <<001>>`

Specifies the key of the DAM FRC that is to be executed when multiple DAM FRCs are to be executed. A different key must be specified for each DAM FRC.

In addition, when you execute the `damfrc` command together with the `tamfrc` command and the `mqafrc` command, or execute the `tamfrc` command, `mqafrc` command, and `damfrc` command in succession, you must specify a different key for each command.

To inherit a previous DAM FRC, its key must be specified.

■ `-n`

Specifies that the DAM files are to be recovered from the data obtained by changing the FJ in the integrated journal file.

When this option is omitted, the data before and after the FJ change in the integrating journal file is checked for its validity. If the data is valid, the DAM files are recovered from the data obtained after the change.

■ `-c`

Specifies that the DAM files are to be recovered by treating the journal files specified in the command arguments as an integrating journal file created by the `jnlcolc` command.

When this option is omitted, the journal files specified in the command arguments are treated as unload journal files. Therefore, the `damfrc` command must be executed after an integrating journal file is created with the `jnlcolc` command.

■ `-m`

Integrates journal records needed for file recovery on a file. If this option is not specified, a buffer area is allocated on memory and journal records are integrated in the area.

Command arguments

■ *recovery-definition-file-name*~<path name>

Specifies the name of the file that defines the DAM files to be recovered.

The definition file subject to recovery must be created using the text editor in the following format:

```
[Δ] logical-file-name Δphysical-file-name [Δ] (linefeed)
[logical-file-name Δphysical-file-name] [Δ] (linefeed)
:
```

logical-file-name~<identifier, 1-8 characters>

Specifies the logical file name of a DAM file to be recovered.

physical-file-name~<path name>

Specifies the recovery destination physical file name, expressed as the full path name.

The physical file to be specified is a restored file that has been backed up. Specify a restored file backed up in either the online or offline mode. Online and offline backed-up files cannot coexist.

■ *journal-file-name*~<path name>

Specifies the name of a journal file to be used during DAM FRC execution.

If the `-c` option is specified, the name of an integrating journal file must be specified; if it is omitted, the names of unload journal files must be specified.

If multiple generations of a journal are to be processed, the corresponding journal file names must be specified. No more than one integrating journal file can be specified.

When an online backed-up file is recovered, not all unload journal files need be specified. Specify unload journal files with higher generation numbers and block numbers than those of the target journal file specified in the message (KFCA02531-I) that was displayed at completion of online backup. To recover a file backed up in the offline mode, specify all unload journal files.

Notes

- If the `damfrc` command is incorrectly specified, a message with no message ID may be output to suggest the correct use of the `damfrc` command. The `jnlcolc` command is used inside the `damfrc` command; an error message for the `jnlcolc` command may be displayed. Take proper action according to the instructions contained in the manual *OpenTPI Messages*.
- Set the write authority in the current directory to allow the `damfrc` command to create a temporary file in the current directory under the following conditions. The disk capacity of the temporary file is as follows:
 1. If the `damfrc` command is executed without the `-c` option to recover a DAM file restored from the offline backup file^{#1}:
Maximum $4096 + a$ (bytes)
 2. If the `damfrc` command is executed without the `-c` option to recover a DAM file restored from the online backup file^{#2}:
Maximum $90 + 4096 + a$ (bytes)
 3. Other than above (the `damfrc` command has the `-c` option specified):
No temporary file is created.

Legend:

a: Total disk space used by the unload journal file^{#3} specified at execution of the `damfrc` command.

#1

When the `dambkup` command was executed to back up the DAM file in offline mode (without the `-o` option).

#2

When the DAM file was backed up by using the `dambkup` command with the `-o` option.

#3

Use the `ls` command in UNIX to find out the total disk space used by the unload journal file. If two or more unload journal files are specified, the sum of their disk capacity applies.

- Do not specify an unload journal file that is acquired by specifying the `-t` option in the `jnlunlfg` command.

Examples

The examples are based on the following information, where generation numbers are assigned consecutively (1,2,...):

Definition files subject to recovery:

```
damdef, damdef1,...
```

Unload journal files:

```
jnl001, jnl002,...
```

Integrating journal file

```
jnlcolc01
```

- (1) *Execution of DAM FRC in one step using multiple unload journal files*

Definition file name subject to recovery:

```
/work/damdef
```

Unload journal file names:

```
/jnldir/jnl001, /jnldir/jnl002
```

```
damfrc -se /work/damdef /jnldir/jnl001 /jnldir/jnl002
```

- (2) *Execution of DAM FRC in three steps*

Definition file name subject to recovery:

```
/work/damdef
```

Unload journal file names:

```
/jnldir/jnl001, /jnldir/jnl002, /jnldir/jnl003, /jnldir/jnl004
```

```
Step 1:  
damfrc -s /work/damdef /jnldir/jnl001 jnldir/jnl002  
Step 2:  
damfrc /work/damdef /jnldir/jnl003  
Step 3:  
damfrc -e /work/damdef /jnldir/jnl004
```

■ (3) *Execution of two DAM FRCs simultaneously in one step*

Definition file names subject to recovery:

/work/damdef1, /work/damdef2

Unload journal file names:

/jnldir/jnl001 (used for both DAM FRCs)

/jnldir/jnl002 (used for the second DAM FRC only)

```
First DAM FRC:  
damfrc -se -k001 /work/damdef1 /jnldir/jnl001  
Second DAM FRC:  
damfrc -se -k002 /work/damdef2 /jnldir/jnl001  
/jnldir/jnl002
```

■ (4) *Execution of DAM FRC in one step using intermediate unload journal files*

Definition file name subject to recovery:

/work/damdef

Unload journal file names:

/jnldir/jnl002, /jnldir/jnl003

```
damfrc -seg /work/damdef /jnldir/jnl002 /jnldir/jnl003
```

■ (5) *Execution of DAM FRC in one step using an integrating journal file*

Definition file name subject to recovery:

/work/damdef

Integrating journal file name:

/jnldir/jnlcolc01

```
damfrc -sec /work/damdef /jnldir/jnlcolc01
```

damhold (Shut down a logical file logically)

Format

```
damhold logical-file-name
```

Function

The `damhold` command shuts down logically a specified logical file.

Command arguments

- *logical-file-name*~<identifier, 1-8 characters>

Specifies the name of the logical file to be shut down logically.

Output messages

Message ID	Description	Output destination
KFCA01650-E	Specification of a command argument is invalid.	Standard error output
KFCA01651-E	Logical file name exceeds 8 characters.	Standard error output
KFCA01662-E	Memory is inadequate.	Standard error output
KFCA01667-E	Versions do not match.	Standard error output
KFCA01670-E	There is no corresponding logical file.	Standard error output
KFCA01671-I	Help message	Standard output
KFCA01672-E	Logical file is already on error shutdown status.	Standard error output
KFCA01673-E	Logical file is already on logical shutdown status.	Standard error output
KFCA01674-I	Waiting for logical shutdown.	Standard output
KFCA01679-E	DAM service has not been started.	Standard error output
KFCA01680-E	Name service has not been started.	Standard error output
KFCA01681-E	Communication error occurred.	Standard error output
KFCA01682-E	Network error occurred.	Standard error output
KFCA01683-E	Overtime occurred.	Standard error output
KFCA01684-E	DAM service is terminating.	Standard error output
KFCA01696-E	Error occurred during definition analysis.	Standard error output

damhold (Shut down a logical file logically)

Notes

The DAM service disables access to the specified file upon accepting the `damhold` command. A file for which deferred updating has been specified is not shut down logically but waits for logical shutdown until the block of the buffer area corresponding to that file is output to a disk. Thus, the logical file cannot be deleted before the block is output a disk. Executing the `damrm` command results in a command error.

damload (Initialize a physical file)

Format

```
damload [-b batch-processing-blocks-count] block-length blocks-count
        physical-file-name [input-file-name]
```

Function

The `damload` command allocates a physical file in an OpenTP1 file system and then outputs data from a specified input file to the allocated physical file. The size of the physical file will be (block length + 8) x (blocks count + 1). The access right to the physical file is 0644 (right to read and write by the owner, right to read by a group, and right to read by others).

Options

- `-b batch-processing-blocks-count`~<unsigned integer> ((1-2147483647)) <<10>>
Specifies the number of blocks to be output simultaneously. When you specify 0, 10 is assumed.

Command arguments

- `block-length`~<unsigned integer> ((sector length x n-8))
Specifies the block length of the physical file to be created.
This specification is also required for the input file.
- `blocks-count`~<unsigned integer> ((1-2147483647))
Specifies the number of blocks in the physical file to be created. If too many blocks are specified, the chain of cache blocks that are managed within the shared memory by DAM may become too large, with possible adverse effects on performance. When you create a DAM file with many blocks, use the `damch1mt` definition command in the DAM service definition to specify a threshold for the number of cache blocks.
- `physical-file-name`~<path name>
Specifies a name for the physical file to be created in the OpenTP1 file system, expressed as the full path name.
- `input-file-name`~<path name>
Specifies the name of the file to which data is to be input.
When this command argument is omitted, physical file allocation only is performed.

Notes

- If an error occurs during execution of the `damload` command, the physical file will remain allocated. Therefore, the physical file must be deleted by the `damdel` command before reexecution of the `damload` command.
- If the number of blocks specified in the `damload` command differs from the number of blocks in the input file, the following results:

Specified blocks count > number of blocks in input file

NULL data blocks are output through the end of the physical file.

Specified blocks count < number of blocks in input file

Input of blocks from the input file is cancelled, the KFCA02599-W message is output, and the `damload` command terminates normally.

- The maximum block length you can specify in the `damload` command is 32760.
- The DAM file is created on the OpenTP1 file system. Therefore, specify the block length and the block count that satisfy the following expression, in the `damload` command.

OpenTP1 file system capacity > ((Block length + 8) x (Block count + 1))

damls (Display status of logical files)

Format

```
damls [-i] [logical-file-name]
```

Function

The `damls` command outputs to the standard output the status of a specified logical file or of all logical files.

Options

- `-i`

Specifies that the total number of logical files currently in use in the online mode and the number of logical files that can be added are to be displayed.

When this option is omitted, neither the total number of logical files nor the number of addable logical files is displayed.

Command arguments

- `logical-file-name`~<identifier, 1-8 characters>

Specifies the name of a logical file whose status is to be output. When this command argument is omitted, the status of all logical files managed by the DAM service is displayed.

Output format

```
Logical file name = aa...aa
Physical file name = bb...bb
Block length = cc...cc  Number of blocks = dd...dd  Shutdown status = ee...ee
Attribute = ff...ff
Security = g
Addition date and time = hh...hh
} 1

Total number of logical files = ii...ii
Number of addable logical files = jj...jj
} 2
```

1. If no logical file name is specified, this information is output individually for each logical file managed by the DAM service.
2. This information is output when the `-i` option is specified.

`aa...aa`

Logical file name (up to 8 characters)

damlS (Display status of logical files)

bb...bb

Physical file name (up to 63 characters)

cc...cc

Block length

dd...dd

Number of blocks

ee...ee

DAM file's shutdown status:

Not shutdown

Logical shutdown

Error shutdown

Under shutdown request

ff...ff

Attribute of DAM file

Quick write

Deferred write

No recovery

Not to be recovered

Cache less

Not to be recovered; cacheless access

g

Security

Y=Security available

N=Security unavailable

hh...hh

Date and time (17 characters in the format year/month/day hour:minute:second, where the year is expressed as the last two digits of the year)

ii...ii

Total number of logical files

jj...jj

Number of addable logical files

Output messages

Message ID	Description	Output destination
KFCA01650-E	Specification of a command argument is invalid.	Standard error output
KFCA01651-E	Logical file name exceeds 8 characters.	Standard error output
KFCA01662-E	Memory is inadequate.	Standard error output
KFCA01667-E	Versions do not match.	Standard error output
KFCA01670-E	There is no corresponding logical file.	Standard error output
KFCA01678-I	Help message	Standard output
KFCA01679-E	DAM service has not been started.	Standard error output
KFCA01680-E	Name service has not been started.	Standard error output
KFCA01681-E	Communication error occurred.	Standard error output
KFCA01682-E	Network error occurred.	Standard error output
KFCA01683-E	Overtime occurred.	Standard error output
KFCA01684-E	DAM service is terminating.	Standard error output
KFCA01696-E	Error occurred during definition analysis.	Standard error output

damrles (Release logical file from shutdown)

Format

```
damrles [-o] logical-file-name
```

Function

The `damrles` command releases a specified logical file from shutdown status.

Options

- `-o`

Specifies that the logical file is to be released from error shutdown status.

When this option is omitted, a logical file placed on shutdown status by the `damhold` command is released from shutdown status.

Command arguments

- *logical-file-name*~<identifier, 1-8 characters>

Specifies the name of the logical file to be released from shutdown status.

Output messages

Message ID	Description	Output destination
KFCA01650-E	Specification of a command argument is invalid.	Standard error output
KFCA01651-E	Logical file name exceeds 8 characters.	Standard error output
KFCA01662-E	Memory is inadequate.	Standard error output
KFCA01667-E	Versions do not match.	Standard error output
KFCA01670-E	There is no corresponding logical file.	Standard error output
KFCA01675-I	Help message	Standard output
KFCA01676-E	Logical file is not on error shutdown status.	Standard error output
KFCA01677-E	Logical file is not on shutdown status.	Standard error output
KFCA01679-E	DAM service has not been started.	Standard error output
KFCA01680-E	Name service has not been started.	Standard error output
KFCA01681-E	Communication error occurred.	Standard error output
KFCA01682-E	Network error occurred.	Standard error output

Message ID	Description	Output destination
KFCA01683-E	Overtime occurred.	Standard error output
KFCA01684-E	DAM service is terminating.	Standard error output
KFCA01696-E	Error occurred during definition analysis.	Standard error output

damrm (Remove a logical file)

Format

```
damrm logical-file-name
```

Function

The `damrm` command closes a specified logical file and removes it from the online system. Before a logical file can be removed, it must be placed on shutdown status.

A file on error shutdown status can also be removed from the online system.

Command arguments

- *logical-file-name*~<identifier, 1-8 characters>

Specifies the name of the logical file to be removed from the online system.

Output messages

Message ID	Description	Output destination
KFCA01650-E	Specification of a command argument is invalid.	Standard error output
KFCA01651-E	Logical file name exceeds 8 characters.	Standard error output
KFCA01662-E	Memory is inadequate.	Standard error output
KFCA01667-E	Versions do not match.	Standard error output
KFCA01668-I	Help message	Standard output
KFCA01669-E	Logical file is not on shutdown status.	Standard error output
KFCA01670-E	There is no corresponding logical file.	Standard error output
KFCA01679-E	DAM service has not been started.	Standard error output
KFCA01680-E	Name service has not been started.	Standard error output
KFCA01681-E	Communication error occurred.	Standard error output
KFCA01682-E	Network error occurred.	Standard error output
KFCA01683-E	Overtime occurred.	Standard error output
KFCA01684-E	DAM service is terminating.	Standard error output
KFCA01696-E	Error occurred during definition analysis.	Standard error output
KFCA01697-E	File cannot be deleted because it is used by others.	Standard error output

Message ID	Description	Output destination
KFCA01698-E	Record lock segment is inadequate.	Standard error output
KFCA02573-E	File cannot be deleted because there is a transaction accessing the file.	Standard error output

Notes

A DAM file being opened in the online mode cannot be removed from the online system by executing the `damrm` command. To remove the file, it is necessary to close the logical file for all applicable services.

However, if a transaction that accesses the logical file remains uncompleted, you cannot delete the logical file, even if the file is closed. In this case, the message KFCA02573-E is output, and processing is interrupted. For this reason, take care when you execute the `damrm` command on a DAM file that is accessed by a large number of transactions.

damrstr (Restore a physical file)

Format

```
damrstr [-c batch-processing-blocks-count] [-e restore-destination-physical-file-block-length] -p
restore-destination-physical-file-block-length {backup-file-name physical-file-name | -s physical-file-name}
```

Function

The `damrstr` command restores to a specified physical file data or standard input data from a file backed up by the `dambkup` command.

Options

- `-c batch-processing-blocks-count` ~<unsigned integer> ((1-2147483647)) <<10>>
Specifies the number of blocks to be input or output simultaneously.
- `-e restore-destination-physical-file-block-length` ~<unsigned integer> ((*sector-length* x *n*-8))
Specifies the block length of the restore destination physical file in bytes, up to a maximum of 32,760 bytes. Specify a value that is either the same as or larger than the block length of the backup source physical file. Specifying this option maintains the block structure of the backup source physical file. The extended portions of each block in the restore destination DAM file are filled with NULL characters. If you omit this option and the `-p` option, the file is restored with the same block length as the backup source physical file.
- `-p restore-destination-physical-file-block-length` ~<unsigned integer> ((*sector-length* x *n*-8))
Specifies the block length of the restore destination physical file in bytes, up to a maximum of 32,760 bytes. Specify a value that is either the same as or larger than the block length of the backup source physical file. If you specify this option, the restore destination physical file does not keep the block structure of the backup source physical file because the backup data is saved starting from the first block in the destination file. If you omit both this option and the `-e` option, the file is restored with the same block length as the backup source physical file.
- `-s`
This option must be specified when the restore source is a standard input.

Command arguments

- `file-name` ~<path name>
Specifies the name of the restore source file.

This command argument cannot be specified with the `-s` option.

■ *physical-file-name*~<path name>

Specifies the physical file name of the restore destination DAM file, expressed as the full path name.

Notes

- If message KFCA02587-E is output with cause code CREATE (physical file allocation processing) and reason code 1514, take the following countermeasures:
 - When no block length is specified for the restore destination physical file

The block length of the specified backup file is not supported in the environment where the `damrstr` command was executed. Execute the `damrstr` command again, specifying the block length of the restore destination physical file (using the `-e` or `-p` option).
 - When a block length is specified for the restore destination physical file

The block length specified for the restore destination physical file (using the `-e` or `-p` option) is not supported in the environment where the `damrstr` command was executed. As the physical file block length, specify the value obtained by the formula (*sector-length-of-restore-destination-file-system* × *n* - 8). Specify a block length that is the same as or larger than that of the backup source physical file.
- You cannot restore a backed up file to an existing physical file that has a shorter block length than the backup source physical file.
- When you specify the `-e` option, you cannot restore a backed up file to an existing physical file that has a lower block count than the backup source physical file.
- When you specify the `-p` option, you cannot restore a backed up file in which the user data occupies a greater number of blocks than the block count of the existing restore destination physical file.
- When you specify the `-p` option, you can determine the block count required of the restore destination physical file by using the following formula (rounding up to a whole number):

$$\frac{\text{backup-source-physical-file-block-length} \times \text{backup-source-physical-file-block-count}}{\text{restore-destination-physical-file-block-length}}$$
- You cannot specify the `-e` or `-p` option when you restore physical file data that was backed up using the `dambkup` command in online mode.
- You cannot specify a DAM file whose block length has been extended as the recovery definition file in the `damfrc` command arguments.

damrstr (Restore a physical file)

- The maximum block length you can specify with the `damrstr` command is 32,760.
- With the `-e` or `-p` option, if you specify a larger block length than that of the backup source physical file, review the size of the I/O area specified as an argument of the DAM access function that is accessing the physical file.
- If the physical file specified in the physical file name argument is yet to be allocated, the access rights to the physical file will be 0644 (read/write permission by the owner, read permission by the group, and read permission by others).

dcauditsetup (Set up the environment for the audit log facility)

Format

```
dcauditsetup OpenTP1-directory
```

Function

The `dcauditsetup` command sets up the environment for using the audit log facility.

To change an operand related to audit logs (a `log_audit_xxxx` operand) in the log service definition, stop OpenTP1, change the log service definition, and then execute the `dcauditsetup` command. Also, whenever you change the location specified by the `DCCONFPATH` operand in the system environment definition (`env`), the log service definition at the new location will not take effect until you execute the `dcauditsetup` command.

The audit log facility will be disabled if the `log_audit_out` operand is omitted from the log service definition, or if `N` is specified for the operand.

The `dcauditsetup` command references the environment variable `DCCONFPATH` in the file *OpenTP1-directory-in-command-argument/conf/env* and sets up the environment for the audit log facility. If `env` does not exist at this location, or the environment variable `DCCONFPATH` is not defined, the command proceeds with the environment setup process using *OpenTP1-directory/conf* as the value of `DCCONFPATH`.

The following table lists the files and directories that are created when you execute the `dcauditsetup` command. If you create the directories in advance, make sure that you assign the appropriate access permissions as shown in the table.

Table 13-1: Files and directories created by the dcauditsetup command

File or directory	User ID	Group ID	Access permission	Description
<code>\$DCDIR/auditlog[#]</code>	User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0777	Directory for storing audit log files
<code>\$DCDIR/auditlog/audit.log[#]</code>	User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0666	Audit log file

[#]

This directory is used when the `log_audit_path` operand is omitted from the log service definition. If the operand is specified with a directory, that directory is

used.

For details about the log service definition, see the manual *OpenTP1 System Definition*.

The following tables show what happens when you execute the `dcauditsetup` command, if the directory serving as the output destination for audit logs already exists or an audit log file already exists at the specified location.

Table 13-2: Execution result of the `dcauditsetup` command when the audit log output directory already exists

User ID of directory	Group ID of directory	Directory access permission	Command execution result
User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0777	Successful
		Other than 0777	Error
User ID of OpenTP1 system administrator	Other group ID	0777	Error
		Other than 0777	Error
Other user ID	Group ID of OpenTP1 system administrator	0777	Error
		Other than 0777	Error
Other user ID	Other group ID	0777	Error
		Other than 0777	Error

Table 13-3: Execution result of the `dcauditsetup` command when the audit log file already exists

User ID of audit log file	Group ID of audit log file	Audit log file access permission	Command execution result
User ID of OpenTP1 system administrator	Group ID of OpenTP1 system administrator	0666	Successful
		Other than 0666	Error
User ID of OpenTP1 system administrator	Other group ID	0666	Successful
		Other than 0666	Error
Other user ID	Group ID of OpenTP1 system administrator	0666	Successful
		Other than 0666	Error

User ID of audit log file	Group ID of audit log file	Audit log file access permission	Command execution result
Other user ID	Other group ID	0666	Successful
		Other than 0666	Error

Command arguments

- *OpenTP1-directory*~<path name>

Specifies the OpenTP1 directory. Use no more than 50 characters.

Output messages

Message ID	Description	Output destination
KFCA33500-I	Help message	Standard error output
KFCA33501-E	Error occurred during command execution.	Standard error output
KFCA33502-I	Log service definition has been analyzed.	Standard output
KFCA33503-I	Audit log facility was enabled/disabled.	Standard output

Notes

Only the superuser can execute the `dcauditsetup` command.

dcccspool (Delete troubleshooting information)

Format

```
dcccspool [-i] [-d number-of-days] [-k {dump|all}]
```

Function

The `dcccspool` command deletes the troubleshooting information created in the `$DCDIR/spool` directory.

Options

■ `-i`

When this option is specified, OpenTP1 outputs a confirmation dialog box to the standard output asking whether it is all right to delete the files containing troubleshooting information. When you enter `y`, the applicable files are deleted.

If you omit this option, the applicable files are deleted without outputting a confirmation dialog box to the standard output. The names of the deleted files are output to the standard output.

■ `-d number-of-days~<unsigned integer>((0-24855))<<1>>`

Deletes the files that were created before the calculated time. OpenTP1 first calculates the number of hours equivalent to the number of days specified in this option by multiplying the specified value by 24. It then deletes the files that were created before the time obtained by subtracting the obtained number of hours from the time that the `dcccspool` command was executed. For example, if you specify `10` as the number of days, OpenTP1 deletes all files that were created prior to 240 hours before the time that the command was executed and earlier.

If you omit this option, OpenTP1 deletes all files that were created prior to 24 to hours before the command was executed.

If you specify `0`, all the files are deleted regardless of the time that they were created.

■ `-k dump|all~<<dump>>`

Specifies the files to be deleted.

`dump`

The following files are deleted:

- Files in `$DCDIR/spool/save`
- Shared memory dump file in `$DCDIR/spool`

all

The following files are deleted:

- Files in \$DCDIR/spool/save
- Shared memory dump file in \$DCDIR/spool
- Deadlock information file and timeout information file in \$DCDIR/spool/dclckinf
- Incomplete-transaction information file in \$DCDIR/spool/dctrninf
- Invalid message information file in \$DCDIR/spool/dcrapinf

The following table lists the files are deleted by using the dccspool command:

Table 13-4: Files deleted by using the dccspool command

File path	File type	Specification of -k option	
		dump	all
Files in \$DCDIR/spool/save	Core file, UAP trace, and OpenTP1 debug information	Y	Y
\$DCDIR/spool/file-name ^{#1}	Shared memory dump file. The file contains the data stored in shared memory by OpenTP1.	Y	Y
\$DCDIR/spool/dclckinf/file-name ^{#2}	Deadlock information file and timeout information file. The files contain the resource wait status.	N	Y
\$DCDIR/spool/dctrninf/file-name ^{#3}	Incomplete-transaction information file. The file contains the information about incomplete transactions.	N	Y
\$DCDIR/spool/dcrapinf/ RAP-processing-listener-name.msg \$DCDIR/spool/dcrapinf/ RAP-processing-client-name.msg	Invalid message information file. The file contains the information about invalid messages.	N	Y

Legend:

Y: Deleted.

N: Not deleted.

#1: The files shmdump, shmdump[1-3], shmdump[1-3].Z, shmdump.XXX, shmdump.XXX.Z, shmdump.XXX[1-3], and shmdump.XXX[1-3].Z (where XXX is the resource manager name dam, tam, ist, or ism) are all deleted. Note that in Linux,

dccspool (Delete troubleshooting information)

the files will have the extension `.gz` instead of `.z`.

#2: The file name is determined based on the time and date that a deadlock was detected. The length of the file name differs depending on whether the date is one digit or two digits.

#3: The file name format is `r1 + time-when-the-transaction-service-is-started` (unique 8-digit hexadecimal number).

Output message

Message ID	Description	Output destination
KFCA01861-E	An error occurred during command processing.	Standard error output

Notes

- If the `prc_coresave_path` operand is specified in the process service definition and the core file is backed up, the `dccspool` command does not delete the backup core file.
- If a shared memory dump file is created in a directory other than `$DCDIR/spool/save` by using the `usmdump` command, the `dccspool` command does not delete the file.

dcdefchk (Check system definitions)

Format

```
dcdefchk [-r] [-l] [-c] [-w] [-e]
```

Function

The `dcdefchk` command checks the values specified in the system definition files (including the domain definition file) in the directories that store definitions (`$DCCONFPATH` and `$DCUAPCONFPATH`). For details about the items to be checked and the procedure, see *1.3.1 Creation and checking of the system definition*.

The command outputs the results of checking the definitions in the form of messages with one of the following indicating the message type labels. Note that some output messages might not be assigned a category.

Labels indicating the message types:

- **ERROR:** This type of message is output on detection of a problem that prevents OpenTP1 from starting or stopping, or impairs some aspects of OpenTP1 operation.
- **WARNG:** This type of message is output when a specified value is not the recommended value.
- **CHECK:** This type of message is output to prompt the user to check the validity of a specified definition.

Options

■ **-r**

Outputs a message (KFCA00254-R) asking you whether you want to continue command processing if an error is found during the syntax check of system definition files.

■ **-l**

Checks the definitions in the system definition files.

Before using this option, we recommend that you first execute the command `dcdefchk` command without the `-l` option. If the command outputs an error message (of type E) at this time, take the appropriate action to resolve it. If you execute the `dcdefchk` command with the `-l` option without first resolving such errors, the same error message might be output multiple times.

■ **-c**

Disables outputting a CHECK message when a problem is detected in a definition in the

dcdefchk (Check system definitions)

system definition files.

■ -w

Does not output a `WARNNG` message when a problem is detected in a definition in the system definition files.

■ -e

Does not output an `ERROR` message when a problem is detected in a definition in the system definition files.

Output messages

Message ID	Description	Output destination
KFCA00200-E	An unspecifiable option is specified.	Standard error output
KFCA00253-I	The definition check command starts.	Standard output
KFCA00254-R	An error was found in a value specified in the definition.	Standard output
KFCA00255-E	An error occurred during command execution.	Standard error output
KFCA00256-I	Help message	Standard output
KFCA00257-E	A directory that has the same name as a definition file was found.	Standard error output
KFCA00258-I	The applicable definition was treated as the system service definition and the syntax of <code>set-format</code> operands in the system service definition was checked.	Standard output
KFCA00283-W	An error occurred during the definition check.	Standard error output
KFCA00284-W	An insignificant error occurred during the definition check.	Standard error output
KFCA00285-W	A mandatory operand is not specified.	Standard error output
KFCA00286-W	A mandatory definition file does not exist.	Standard error output

Note

In addition to the above messages, the messages KFCA00215-E to KFCA00223-E, KFCA00240-E to KFCA00252-E, and KFCA00259-W to KFCA00282-W might be output, as well as the messages mentioned in the description of definition checks in the manual *OpenTP1 System Definition*.

Notes

- You can execute the `dcdefchk` command while `OpenTP1` is operating. If you do so, the command checks the values specified in the system definition before the command is executed, not the values that are valid for the current operation of `OpenTP1`. For example, when you use the `prcpath` command to change the

value of `prcsvpath` specified in the process service definition, the `dcdefchk` command checks the original value specified in the process service definition.

- The `dcdefchk` command uses the directory specified in the `$DCDIR` environment variable as the OpenTP1 directory. The `$DCDIR` environment variable is set in the environment where the `dcdefchk` command is to be executed. Therefore, if the directory specified in the `$DCDIR` environment variable differs from the path name for the OpenTP1 directory set in the environment where the `dcstart` command is to be executed, the `dcdefchk` command cannot correctly check definitions.
- If the directory paths specified in `$DCCONFPATH` and `$DCUAPCONFPATH` are longer than 50 bytes, only part of a definition file name is output in the message (KFCA00242-E) that is output if an error is found during the syntax check. To avoid this, copy the target definition file to a directory whose path name is 50 bytes or less, specify the copy destination directory in the `DCCONFPATH` environment variable that is specified in `putenv` format in `$DCDIR/conf/env`, and execute the `dcdefchk` command.
- In the directory for storing definitions, store only the files for which you want to perform a definition check. If you store files other than system definition files and user service definition files under the directories for storing system definitions (`$DCCONFPATH` and `$DCUAPCONFPATH`), the `dcdefchk` command cannot correctly check definitions. This precaution does not include the directories for storing domain definition files and domain definition files.
- For files other than system service definition files under the directories for storing system definitions (`$DCCONFPATH` and `$DCUAPCONFPATH`), the `dcdefchk` command does not check files whose names begin with periods (.) and underscores (_) and names that are 9 or more bytes.
- The `dcdefchk` command only checks the definition of the operands that are listed in the manual *OpenTP1 System Definition*.
- Do not execute the `dcdefchk` command at the same time as a command that creates OpenTP1 files (such as `jnlinit`).
- If a syntax check detects a problem with a value specified for an operand, the logic check will proceed as if the default is specified for that operand.
- The `dcdefchk` command checks file and directory access permissions based on the UID/GID of the user who executed the command.
- A message may be output during a logic check if the OpenTP1 file system is inaccessible for some reason, such as being in a standby state.
- Some of the messages output by the `dcdefchk` command during the checking process do not belong to any of the message types associated with the `dcdefchk` command. For this reason, if you run the `dcdefchk` command with the `-e` option

dcdefchk (Check system definitions)

specified, messages with the message type E may still be output.

- The `dcdefchk` command may be unable to check the logic of environment variables (definitions in `putenv` or `dputenv` format) correctly.

dcjchconf (Set the value of an operand in the system definition)

Format

```
dcjchconf [-f file-name] [-n] operand-name operand-value
```

Function

The `dcjchconf` command sets the value of an operand in the system definition file or the operation script file.

When setting the value of an operand with this command, you must enclose the name of the operand with at marks (@) in the target file as shown below:

Setting the value of `node_id`

```
set node_id = @DCNODE_ID@
```

When you execute the `dcjchconf` command, the operand name enclosed in at marks (@) is converted to the value set in the applicable operand in the `dcjchconf` command.

Options

- `-f file-name~<path name>`

Specifies the path for the file containing the operand whose value you want to specify or the path for the directory containing this file.

When you specify a directory, all the files in the specified directory are subject to processing.

If you omit this option, all the files in `$DCCONFPATH` are subject to processing.

- `-n`

Does not insert a linefeed character after the value of the operand has been set.

If you omit this option and the length of one line exceeds 80 bytes after the operand value is set, OpenTP1 inserts a continuation character (\) as the 80th byte and inserts a new line.

If you do not want to insert a new line, specify the `-n` option.

Command arguments

- `operand-name~<character string of 1-4096 characters>`

For *operand-name*, specify a character string prefixed with DC.

dcjchconf (Set the value of an operand in the system definition)

- *operand-value*~<character string of 1-4096 characters>

Specifies the value to be set for the operand.

Output messages

Message ID	Description	Output destination
KFCA01815-E	An error occurred in the OS.	Standard error output
KFCA01860-E	The input format of the <code>dcjchconf</code> command is incorrect.	Standard error output
KFCA01861-E	Processing of the <code>dcjchconf</code> command failed.	Standard error output

Notes

- The `dcjchconf` command creates a directory named `.dcjchconf` in the directory containing the file specified by the `-f` option (or in the directory specified by the `-f` option) and makes a backup copy of the old file. If the `dcjchconf` command fails to specify the value of an operand, use the backup file in the `.dcjchconf` directory to restore the original file.
- The `dcjchconf` command creates a temporary file named `.dcjchconf.tmp` in the directory containing the file specified by the `-f` option (or in the directory specified by the `-f` option). If the `dcjchconf` command fails during execution, the temporary file may remain.
- The `dcjchconf` command ends normally even if the specified operand name is not found in the target file.

Examples

- (1) *Specifying `smpl` for `node_id` in the `$DCCONFPATH/betranrc` file*

```
dcjchconf -f $DCCONFPATH/betranrc DCNODE_ID smpl
```

`$DCCONFPATH/betranrc` before the `dcjchconf` command is executed

```
set node_id = @DCNODE_ID@
```

`$DCCONFPATH/betranrc` after the `dcjchconf` command is executed

```
set node_id = smpl
```

- (2) *Specifying `hostA` and `hostB` for all `_node` in the `$DCCONFPATH/betranrc` file when the value of the operand to be specified contains a double quotation mark (")*

```
dcjchconf -f $DCCONFPATH/betranrc DCALL_NODE '"hostA", "hostB"'
```

`$DCCONFPATH/betranrc` before the `dcjchconf` command is executed

```
set all_node = @DCALL_NODE@
```

\$DCCONFPATH/betranrc after the dcjchconf command is executed

```
set all_node = "hostA","hostB"
```

To include a double quotation mark (") in the value of the operand to be specified, enclose the value in single quotation marks ('). To include a space character in the value of the operand to be specified, enclose the value in double quotation marks (") or single quotation marks (').

If your OS is Windows and you want to include a double quotation mark (") in the value of the operand to be specified, write \" instead of the double quotation mark ("). To include a space character in the value of the operand to be specified, enclose the value between double quotation marks (").

dcjcmdex (Execute an OpenTP1 command from a scenario template)

Format

```
dcjcmdex OpenTP1-command-name [command-argument-name [  $\Delta$ command-argument-name] . . .]
```

Function

The `dcjcmdex` command executes a specified OpenTP1 command from a scenario template defined by using JP1/AJS2 - Scenario Operation. First, however, add `$DCDIR/bin` in the `PATH` environment variable (the environment variable is valid only for the command that is executed).

When the executed OpenTP1 command ends normally, the `dcjcmdex` command returns 0. If the command ends abnormally, the `dcjcmdex` command returns 8.

Command arguments

- *OpenTP1-command-name*

Specifies the name of the OpenTP1 command to be executed from the scenario template defined by using JP1/AJS2 - Scenario Operation.

- *command-argument-name*

Specifies an option or a command argument of the specified OpenTP1 command.

Output messages

Message ID	Description	Output destination
KFCA01815-E	An error occurred in the OS.	Standard error output
KFCA01860-E	The input format of the <code>dcjcmdex</code> command is incorrect.	Standard error output
KFCA01886-E	Processing of the OpenTP1 command failed.	Standard error output

Notes

Do not specify the `dcsetup` command as the OpenTP1 command name. If you do, the `dcjcmdex` command may not operate normally.

Examples

- (1) Using the `dcsvstart` command to start a user server named `basespp`

```
dcjcmdex dcsvstart -u basespp
```


- (2) Using the *dcjchconf* command to specify *hostA* and *hostB* for all *node* in the system definition file when a parameter contains double quotation marks (")

```
dcjcmdex dcjchconf DCALL_NODE '"hostA","hostB"'
```

To include a double quotation mark (") in the value of the operand to be specified, enclose the value between single quotation marks (').

If your OS is Windows and you want to include a double quotation mark (") in the value of the operand to be specified, write \" instead of the double quotation mark (").

dcjnamch (Update the domain definition file)

Format

```
dcjnamch [-e] [-f domain-definition-file-name] node-name [port-number]
```

Function

The `dcjnamch` command adds a new node to the domain definition file.

If you execute the `namchgfl` command after you update the domain definition file by using the `dcjnamch` command, you can change the domain configuration without changing the values specified in the system common definition during online operation. The `dcjnamch` command is available only when `y` is specified in the `name_domain_file_use` operand in the system common definition.

Options

- `-e`

Adds a new node to the domain definition file specified in the `all_node_ex` operand in the system common definition.

If you omit this option, a new node is added to the domain definition file specified in the `all_node` operand in the system common definition.

- `-f domain-definition-file-name`~<file name>

Specifies the name of the domain definition file to which a node is to be added.

Specify a domain definition file stored in the `$DCCONFPATH/dcnamnd` directory. If the specified file does not exist, a new file is created.

If you omit the `-f` option, OpenTP1 creates the `$DCCONFPATH/dcnamnd/dcj_allnd` file and adds the specified node to that file.

When you specify both the `-f` option and the `-e` option, OpenTP1 uses the domain definition file in the `$DCCONFPATH/dcnamndex` directory.

If you omit the `-f` option and specify the `-e` option, OpenTP1 creates the `$DCCONFPATH/dcnamndex/dcj_allndex` file and adds the specified node to that file.

Command arguments

- `node-name`~<identifier of 1-255 characters>

Specifies the name of the node to be added to the OpenTP1 system.

The host name or the IP address defined in `/etc/hosts` becomes the node name.

■ *port-number*~<unsigned integer>((5001-65535))

Specifies the port number to be used as the well-known port number by the name server. If you omit the port number, OpenTP1 assumes the port number of the name service specified in the *name_port* operand in the system common definition. If the *name_port* operand is not specified, OpenTP1 assumes 10000.

Output messages

Message ID	Description	Output destination
KFCA01815-E	An error occurred in the OS.	Standard error output
KFCA01860-E	The input format of the <i>dcjnamch</i> command is incorrect.	Standard error output
KFCA01861-E	Processing of the <i>dcjnamch</i> command failed.	Standard error output
KFCA01870-E	Memory is insufficient.	Standard error output

Note

The domain definition file is created in the `$DCCONFPATH/dcnamnd` directory. Note, however, that if you specify the `-e` option, the domain definition file specified in the *all_node_ex* operand in the system common definition is created in the `$DCCONFPATH/dcnamndex` directory. For details about how to change the definition file directly, see *3.10.1 Changing the domain configuration*.

dcmakeup (Reserve or release resources for OpenTP1 internal control)

Format

```
dcmakeup [-d] OpenTP1-directory
```

Function

The `dcmakeup` command reserves or releases OS resources that are used by OpenTP1 for internal control.

The reserved resources are stored under an OpenTP1 directory. How many numbers to reserve is determined through analysis of the system definition information under the OpenTP1 directory.

If the value of the `prc_process_count` operand in the process service definition is changed, the `dcsetup` command must be executed and then the `dcmakeup` command executed before using the `dcstart` command. If the `dcmakeup` command is not executed, it will be executed at OpenTP1 startup but much time may be required to reserve a sufficient number of resources.

Resources once reserved are reused by OpenTP1 and need not be released.

If the `dcmakeup` command is executed after necessary resources have been reserved, it is returned normally and no new resources need be reserved.

If a system definition is changed after necessary resources have been reserved, the user may execute the `dcmakeup` command again or start OpenTP1. This reserves as many additional resources as the new value minus the old value specified in the system definition if the new value is greater than the old value. For a new system definition smaller than the old one, the decreased resources are not released.

Options

- `-d`

Releases the resources for OpenTP1 internal control.

If the `-d` option is not specified, the OS resources to be used by OpenTP1 for internal control are reserved and stored under the specified OpenTP1 directory.

Command arguments

- `OpenTP1-directory~<path name>`

Specifies an OpenTP1 directory that stores reserved resources. For multiOpenTP1, this option specifies a directory for each OpenTP1.

dcmaphg (Change map file path name)

Format

```
dcmaphg { {-mM|-mP|-aM|-aP|-m|-a} mapping-service-identifier path-name}
```

Function

The `dcmaphg` command changes the reading path name of a physical map file or PAGEC module, or the standard reading path name of a map file.

The `dcmaphg` command can be used under the TP1/NET/XMAP3 protocol or TP1/NET/HNA-560/20 protocol.

Options

- `-mM`
Changes the standard physical map reading path name. (TP1/NET/XMAP3)
- `-mP`
Changes the standard PAGEC module reading path name. (TP1/NET/XMAP3)
- `-aM`
Changes the alternate physical map reading path name. (TP1/NET/XMAP3)
- `-aP`
Changes the alternate PAGEC module reading path name. (TP1/NET/XMAP3)
- `-m`
Changes the standard reading path name of the map file specified in `MAPPATH` of the mapping service definition file. (TP1/NET/HNA-560/20)
- `-a`
Changes the alternate reading path name of the map file specified in `ALTPATH` of the mapping service definition file. (TP1/NET/HNA-560/20)

Command arguments

- `mapping-service-identifier`~<1-8 alphanumeric>
Specifies the mapping service identifier specified with the mapping service definition.
- `path-name`~<1-255 character path name>
Specifies the path name after change.

dcmaphg (Change map file path name)

Examples

- (1) Change the standard physical map input path to /tmp

```
dcmaphg -mM mapserv1 /tmp
```

- (2) Change the alternate PAGEC module input path name to /tmp

```
dcmaphg -aP mapserv1 /tmp
```

Output messages

Message ID	Description	Output destination
KFCA10865-I	Help message	Standard output
KFCA10870-E	Specification of an option mode (%s) is invalid.	Standard output
KFCA10871-E	Specification of a mapping service identifier (%s) is invalid.	Standard output
KFCA10872-E	Invalid dcmaphg command was entered.	Standard output
KFCA10873-E	Error occurred in the socket interface (%s).	Standard output
KFCA10874-E	Error code is (%s).	Standard output
KFCA10875-E	Error occurred in the communication with the mapping service process.	Standard output
KFCA10876-E	Path name failed to be changed.	Standard output
KFCA10877-E	No (%s) exists in the mapping service definition.	Standard output
KFCA10878-I	Path name was changed.	Standard output
KFCA10879-E	Mapping service has not been activated.	Standard output
KFCA10880-E	Specification of path name (%s) is invalid.	Standard output
KFCA10886-E	An error occurred during command processing.	Standard output

dcmpls (Display loaded resources of map file)

Format

```
dcmpls {{-s|-d|-p} mapping-service-identifier}
```

Function

The `dcmpls` command displays the loaded physical maps or PAGEC modules of a map file on the standard output.

The `dcmpls` command can be used under the TP1/NET/XMAP3 protocol or TP1/NET/HNA-560/20 protocol.

Options

- `-s`
Displays in the standard format.
- `-d`
Displays in the detailed format.
- `-p`
Displays input path names.

Command argument

- *mapping-service-identifier*~<1-8 alphanumeric>
Specifies the mapping service identifier specified in the mapping service attribute definition.

Examples

1. *Display loaded resource information for the mapping service identifier mapserv1 in the standard format:*

```
dcmpls -s mapserv1
```

2. *Display loaded resource information for the mapping service identifier mapserv1 in the detailed format:*

```
dcmpls -d mapserv1
```

Output format

- (1) When the *-s* option is specified

KFCA10888-I	Starts MPS status display.
KFCA10890-I	<i>ii...ii</i> MAPNAME <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10890-I	<i>ii...ii</i> MAPPATH <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10890-I	<i>ii...ii</i> ALTPATH <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10890-I	<i>ii...ii</i> PGCMNAME <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10890-I	<i>ii...ii</i> PGCSTDPN <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10890-I	<i>ii...ii</i> PGCALTPN <i>pp...pp</i>
KFCA10891-I	<i>mm...mm</i>
KFCA10889-I	Terminates MPS status display.

When the path name specification is omitted from the mapping service attribute definition, only the KFCA10888-I and KFCA10890-I messages are output.

When eight or more map names is displayed for standard physical map input path (MAPPATH) or alternate physical map input path (ALTPATH), the KFCA10891-I message is output immediately after the KFCA10890-I message.

MPS

Mapping service

MAPNAME

Resident physical map input path

MAPPATH

Standard physical map input path

ALTPATH

Alternate physical map input path

PGCMNAME

Resident PAGEC module input path

PGCSTDPN

Standard PAGEC module input path

PGCALTPN

Alternate PAGEC module input path

ii...ii

Mapping service identifier

pp...pp

Path name

mm...mm

Physical map name or PAGEC module name

■ (2) When the *-d* option is specified

KFCA10888-I	Starts	MPS	status	display.					
KFCA10890-I	<i>ii...ii</i>	MAPNAME	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...1.
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...1.
KFCA10890-I	<i>ii...ii</i>	MAPPATH	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10890-I	<i>ii...ii</i>	ALTPATH	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10890-I	<i>ii...ii</i>	PGCMNAME	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...1.
KFCA10890-I	<i>ii...ii</i>	PGCSTDPN	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10890-I	<i>ii...ii</i>	PGCALTPN	<i>pp...pp</i>						
KFCA10892-I	<i>mm...mm</i>	<i>ss...ss</i>	<i>hh...hh</i>	<i>ll...ll</i>	<i>oo...oo</i>				...2.
KFCA10889-I	Terminates	MPS	status	display.					

1. A physical map specified to be resident or PAGEC module but failed due to an I/O error or other problem during startup of the mapping service is not displayed.
2. The physical maps managed by LRU or PAGEC modules are displayed from the last-used one.

When the path name specification is omitted from the mapping service attribute definition, only the KFCA10888-I and KFCA10889-I messages are output.

MPS

Mapping service

MAPNAME

Resident physical map reading path

MAPPATH

Standard physical map reading path

ALTPATH

Alternate physical map reading path

dcmapl (Display loaded resources of map file)

PGCMNAME

Resident PAGEC module reading path

PGCSTDPN

Standard PAGEC module reading path

PGCALTPN

Alternate PAGEC module reading path

ii...ii

Mapping service identifier

pp...pp

Path name

mm...mm

Physical map name or PAGEC module name.

sssss

Physical map size or PAGEC module size (decimal)

ss...ss:

Physical map size or PAGEC module size (decimal)

hh...hh

Maintenance information 1 (hexadecimal)

ll...ll

Maintenance information 2 (hexadecimal)

oo...oo

Maintenance information 3 (hexadecimal)

■ (3) *When the -p option is specified*

KFCA10888-I	Starts MPS status display.
KFCA10890-I	<i>ii...ii</i> MAPNAME <i>pp...pp</i>
KFCA10890-I	<i>ii...ii</i> MAPPATH <i>pp...pp</i>
KFCA10890-I	<i>ii...ii</i> ALTPATH <i>pp...pp</i>
KFCA10890-I	<i>ii...ii</i> PGCMNAME <i>pp...pp</i>
KFCA10890-I	<i>ii...ii</i> PGCSTDPN <i>pp...pp</i>
KFCA10890-I	<i>ii...ii</i> PGCALTPN <i>pp...pp</i>
KFCA10889-I	Terminates MPS status display.

When the path name specification is omitted from the mapping service attribute definition, only the KFCA10888-I and KFCA10890-I messages are output.

MPS

Mapping service

MAPNAME

Resident physical map input path

MAPPATH

Standard physical map input path

ALTPATH

Alternate physical map input path

PGCMNAME

Resident PAGEC module input path

PGCSTDPN

Standard PAGEC module input path

PGCALTPN

Alternate PAGEC module input path

ii...ii

Mapping service identifier

pp...pp

Path name

Output messages

Message ID	Description	Output destination
KFCA10866-I	Help message	Standard output
KFCA10870-E	Specification of an option mode <%s> is invalid.	
KFCA10871-E	Specification of a mapping service identifier <%s> is invalid.	
KFCA10872-E	Invalid command was entered.	
KFCA10875-E	Error occurred in the communication with the mapping service process.	
KFCA10877-E	No mapping service identifier <%s> is defined in the mapping service definition file.	
KFCA10879-E	Mapping service has not been activated.	
KFCA10881-E	Display of the map name listing failed.	

dcmapls (Display loaded resources of map file)

Message ID	Description	Output destination
KFCA10886-E	Error occurred during command processing.	
KFCA10888-I	Information display starts.	
KFCA10889-I	Information display terminates.	
KFCA10890-I	Information (map identifier, path type, path name) is displayed.	
KFCA10891-I	Information (map name, PAGEC module name) is displayed.	
KFCA10892-I	Information (name, size, and maintenance information) is displayed.	

dcmstart (Start multinode area or subarea)

Format

```
dcmstart [-n] [-p] [-t start-verify-time] {-g multinode-subarea-identifier|-w node-identifier [, node-identifier] ... }
```

Function

The `dcmstart` command starts the following OpenTP1 nodes normally at the same time or restarts them at the same time:

- All OpenTP1 nodes that configure multinode areas to which the OpenTP1 node corresponding to the command input environment variable `DCDIR` belongs
- All OpenTP1 nodes that configure the specified multinode subareas
- OpenTP1 nodes with the specified `w` option

Processing for an OpenTP1 node differs depending on its status taken when the `dcmstart` command is executed.

The `dcmstart` command watches all OpenTP1 nodes in the specified range until it verifies that they have been successfully started up or failed to start up. The command reports (by issuing a message) OpenTP1 nodes under startup process every about 30 seconds.

If the command cannot verify success in, or failure to, start up before a certain time (the value specified in `dcmstart_watch_time` of the multinode configuration definition) from entry of the command, the system outputs a message and discontinues watching the remaining nodes. Startup process continues for these nodes, but some may be delayed in starting. The user must check the pending nodes and take appropriate actions.

You cannot verify the startup of an OpenTP1 node using the `dcmstart` command in the following case:

- Although a start instruction was issued to the OpenTP1 node, the status of the OpenTP1 node did not change from "stopped" (`TERM`) during the start verify time.

Typical cases are as follows:

1. Although the OpenTP1 node received a start instruction, initialization takes too much time and the status of OpenTP1 does not change to "started".
2. For some reason, the OpenTP1 node start command cannot be activated in the OpenTP1 node that you request to start.

For an OpenTP1 node using the system switchover facility, startup process takes place

for its running and standby systems. When issued, a message relates to both systems.

Options

- **-n**
 Forces to start each OpenTP1 node normally independently of the previous termination mode. If this option is not specified, the startup mode is determined on the basis of the previous termination mode.
 Previous termination mode was normal termination => normal startup
 Previous termination mode was other than normal termination => restart
- **-p**
 The command returns the moment it requests an OpenTP1 node to start. The result of startup processing for the node is not checked.
- **-t *start-verify-time*~((0-65535))**
 Specify the length of time to verify the startup of an OpenTP1 node in seconds.
 Since a start verification is performed every 5 seconds, you may need to wait longer than the specified time. If you do not specify this option, 20 is assumed. If you specify a value between 1 and 19, 20 is also assumed. When you specify 0, the system continues monitoring the node until it confirms that the OpenTP1 node is completely started.
 If the value specified for this option is smaller than the value specified in the `dcmstart_watch_time` operand in the multinode configuration definition, the value specified in the `dcmstart_watch_time` operand takes precedence.
- **-g *multinode-subarea-identifier*~<1-8 character identifier>**
 Specifies one identifier for the multinode subarea to start. If an asterisk (*) is specified as the subarea name, it is assumed that a multinode area has been specified. Use \ or " instead of * so that the name is not extended to a shell. The -g option enables normal startup or restart of all OpenTP1 nodes configuring multinode areas or subareas specified in the definition command `dcmarea` of the multinode configuration definition.
- **-w *node-identifier*~<4 character identifier>**
 Starts the specified OpenTP1 node. When more than one node identifier is specified, a comma (,) must be placed between identifiers.

Output messages

Message ID	Description	Output destination
KFCA01860-E	dcmstart command input format is invalid.	Standard error output

Message ID	Description	Output destination
KFCA04603-I	OpenTP1 node began the startup processing.	Standard output
KFCA04604-I	OpenTP1 node became online.	Standard output
KFCA04605-W	OpenTP1 node has not been activated.	Standard output
KFCA04606-W	Startup of OpenTP1 node could not be confirmed.	Standard output
KFCA04613-E	The <code>dcmstart</code> command argument is invalid.	Standard error output
KFCA04615-E	Error exists in the definition file.	Standard error output
KFCA04616-E	Error exists in the multinode configuration definition.	Standard error output
KFCA04617-E	Command specification does not match the specification in the multinode configuration definition.	Standard error output
KFCA04619-W	OpenTP1 node exceeded its startup watch time.	Standard output
KFCA04620-I	OpenTP1 node is being activated.	Standard output
KFCA04621-E	Error occurred during executing the multinode function.	Standard error output
KFCA04625-I	OpenTP1 node is now in standby mode.	Standard output

Notes

- In the process of starting each OpenTP1 node with the `dcmstart` command, inquiry-response messages are suppressed.
- Each OpenTP1 node can be started independently with the `dcmstart` command. The `dcmstart` and `dstart` commands must not be executed concurrently. Executing these two concurrently may cause the `dcmstart` command's message to become invalid.
- If a response from the OpenTP1 node to which a startup instruction is sent does not come within the maximum wait time specified in the multinode configuration definition, a message indicating the occurrence of timeout is output and the processing for the OpenTP1 node discontinues.
- System operation is not guaranteed when the `dcmstart` command is executed at occurrence of system switchover.
- The `dcmstart` command must not be executed concurrently for the same OpenTP1 nodes. Otherwise, the `dcmstart` command's message may become invalid.
- With one OpenTP1 node duplicated in more than one multinode subarea, startup process is valid even if any multinode subarea identifier is specified in the `-g` option.

dcmstart (Start multinode area or subarea)

- For a node for which no host name can be obtained from the multinode physical definition, '*****' is placed in the host name area of a message.

dcmstop (Terminate multinode area or subarea)

Format

```
dcmstop [-abf] [-p] {-g multinode-subarea-identifier|-w
node-identifier [, node-identifier] . . . }
```

Function

The `dcmstop` command terminates the following OpenTP1 nodes at the same time:

- All OpenTP1 nodes that configure multinode areas to which the OpenTP1 node corresponding to the command input environment variable `DCDIR` belongs
- All OpenTP1 nodes that configure the specified multinode subareas
- OpenTP1 nodes with the specified `w` option

Processing for an OpenTP1 node differs depending on its status taken when the `dcmstart` command is executed.

The `dcmstop` command watches all OpenTP1 nodes in the specified range until it verifies that they have been successfully terminated or failed to terminate. The command reports (by issuing a message) OpenTP1 nodes under termination process every about 30 seconds.

If the command cannot verify success in, or failure to, terminate before a certain time (the value specified in `dcmstop_watch_time` of the multinode configuration definition) from entry of the command, system outputs a message and discontinues watching the remaining nodes. Termination process continues for these nodes, but some may be delayed in terminating. The user must investigate the pending nodes and take appropriate actions.

For an OpenTP1 node using the system switchover facility, termination process takes place for its running and standby systems. When issued, a message relates to both systems.

Options

- `-a`

Terminates each OpenTP1 node belonging to the multinode area or subarea in planned termination A mode.

- `-b`

Terminates each OpenTP1 node belonging to the multinode area or subarea in planned termination B mode.

dcmstop (Terminate multinode area or subarea)

■ -f

Terminates each OpenTP1 node belonging to the multinode area or subarea in forced termination mode.

The -a, -b and -f options cannot be specified at the same time.

If none of the options is specified, each OpenTP1 node belonging to the multinode area or subarea is terminated normally.

■ -p

The command returns the moment it requests an OpenTP1 node to start. The result of termination processing for the node is not checked.

■ -g *multinode-subarea-identifier*~<1-8 character identifier>

Specifies one identifier for the multinode subarea to terminate. If an asterisk (*) is specified as the subarea name, it is assumed that a multinode area has been specified. Use \ or " instead of * so that the name is not extended to a shell. The g option enables termination of all OpenTP1 nodes configuring multinode areas or subareas specified in the definition command `dcmarea` of the multinode configuration definition.

■ -w *node-identifier*~<4 character identifier>

Terminates the specified OpenTP1 node. When more than one node identifier is specified, a comma (,) must be placed between identifiers.

Output messages

Message ID	Description	Output destination
KFCA01860-E	dcmstop command input format is invalid.	Standard error output
KFCA04607-I	OpenTP1 node began the termination processing.	Standard output
KFCA04608-I	OpenTP1 node terminated.	Standard output
KFCA04609-W	OpenTP1 node cannot be terminated.	Standard output
KFCA04610-W	Termination of OpenTP1 node could not be confirmed.	Standard output
KFCA04613-E	The dcmstop command argument is invalid.	Standard error output
KFCA04614-W	OpenTP1 node exceeded termination watch time.	Standard output
KFCA04615-E	Error exists in the definition file.	Standard error output
KFCA04616-E	Error exists in the multinode configuration definition.	Standard error output
KFCA04617-E	Command specification does not match the specification in the multinode configuration definition.	Standard error output
KFCA04618-I	OpenTP1 node is terminating.	Standard output

Message ID	Description	Output destination
KFCA04621-E	Error occurred during executing the multinode function.	Standard error output
KFCA04626-I	Standby termination process has begun for OpenTP1 node.	Standard output

Notes

- Each OpenTP1 node can be terminated independently with the `dcmstop` command. The `dcmstop` and `dcstop` commands must not be executed concurrently. Executing these two concurrently may cause the `dcmstop` command's message to become invalid.
- If a response from the OpenTP1 node to which a termination instruction is sent does not come within the maximum wait time specified in the multinode configuration definition, a message indicating the occurrence of timeout is output and the processing for the OpenTP1 node discontinues.
- System operation is not guaranteed when the `dcmstop` command is executed at occurrence of system switchover.
- The `dcmstop` command's message must not be executed concurrently for the same OpenTP1 nodes. Otherwise, the `dcmstop` command's message may become invalid.
- With one OpenTP1 node duplicated in more than one multinode subarea, termination process is valid even if any multinode subarea identifier is specified in the `-g` option.
- For a node for which no host name has been collected from the multinode physical definition, '*****' is displayed in the host name area of a message.

dcndls (Display status of OpenTP1 node)

Format

```
dcndls { -g multinode-subarea-identifier | -w node-identifier
        [, node-identifier] . . . }
```

Function

The `dcndls` command outputs the status of the specified OpenTP1 nodes to the standard output.

For an OpenTP1 node using the system switchover facility, the `dcndls` command outputs the status of its running and standby systems.

Options

- `-g multinode-subarea-identifier`~<1-8 character identifier>

Specifies one identifier for the multinode subarea to be displayed. If an asterisk (*) is specified as the subarea name, it is assumed that a multinode area has been specified. Use \ or " instead of * so that the name is not extended to a shell. The `-g` option enables displaying the status of all OpenTP1 nodes configuring multinode areas or subareas specified in the definition command `dcmarea` of the multinode configuration definition.

- `-w node-identifier`~<4 character identifier>

Displays the status of the specified OpenTP1 node. When more than one node identifier is specified, a comma (,) must be placed between identifiers.

Output format

node	status	subarea	host
<i>aaaa</i>	<i>bb...bb</i>	<i>cc...cc</i>	<i>dd...dd</i>
:	:	:	:
:	:	:	:

aaaa

Node identifier of an OpenTP1 node (4 characters)

bb...bb

Status of an OpenTP1 node (1-13 characters)

NOTUP

Communication impossible

The possible causes are:

- dcsetup to be executed for the OpenTP1 node (dcsetup not yet executed or to be reexecuted)
- multinode physical definition error (OpenTP1 node not yet cataloged or invalid host name or port number)
- communication error (power of host with the OpenTP1 node remaining off or a network error)

TERM

Not in working or being abnormally terminated

START_NORMAL

Being normally started up

START_RECOVER

Being restarted

ONLINE

Online

STOP

Being terminated (normal termination)

STOPA

Being terminated (planned termination A)

STOPB

Being terminated (planned termination B)

STANDBY_START

Standby system being started

STANDBY

Being in standby status

STANDBY_STOP

Standby system being terminated

NETDOWN

Network error

TIMEOUT

Timeout

????

Failure to collect status of OpenTP1 nodes

Displayed when an error has occurred at the appropriate OpenTP1 node. Recheck the applicable definition or check if TP1/Multi has been installed.

cc...cc

Multinode subarea identifier (1-8 characters)

'*****' is displayed if the identifier refers to a multinode subarea that does not belong to any multinode area.

If an OpenTP1 node belonging to the multinode subarea identifier specified in the -g option also is duplicated in another multinode subarea, the multinode subarea identifier specified in the -g option is displayed.

If an OpenTP1 node specified by '*' in the -w or -g option is duplicated in more than one multinode subarea, the first multinode subarea identifier specified in the multinode configuration definition is displayed.

dd...dd

Host name (1-64 characters)

The host name specified in the multinode physical definition is displayed.

***** is displayed for a node for which no host name has been collected from the multinode physical definition.

Output messages

Message ID	Description	Output destination
KFCA01860-E	dcndls command input format is invalid.	Standard error output
KFCA04613-E	The dcndls command argument is invalid.	Standard error output
KFCA04615-E	Error exists in the definition file.	Standard error output
KFCA04616-E	Error exists in the multinode configuration definition.	Standard error output
KFCA04617-E	Command specification does not match the specification in the multinode configuration definition.	Standard error output
KFCA04621-E	Error occurred during executing the multinode function.	Standard error output

Notes

The dcndls command must not be executed for the current or standby system at an OpenTP1 node using the system switchover facility.

dcplist (Display product information)

Format

```
dcplist
```

Function

The `dcplist` command displays the product name, version number, and other information about products operating in an environment set up in the OpenTP1 directory.

Output format

#	Product ID	Version	Product Name
aaa	bb...bb	cc...cc	dd...dd

aaa

Serial number (3 digits)

bb...bb

Product ID (20 characters maximum)

cc...cc

Version (9 characters maximum)

dd...dd

Product name (40 bytes maximum)

Output messages

Message ID	Description	Output destination
KFCA01815-E	An error occurred in the OS.	Standard error output
KFCA01860-E	Invalid usage of the <code>dcplist</code> command	Standard error output
KFCA01861-E	Error occurred during <code>dcplist</code> command processing.	Standard error output

Notes

- The `dcplist` command uses data from the Hitachi Program Product Installer.
- The `dcplist` command displays information about all products installed by the Hitachi Program Product Installer, including products unrelated to OpenTP1.

dcplist (Display product information)

- If you use the OpenTP1 installation directory as the OpenTP1 directory, executing the `dcplist` command will display a list of currently installed products.
- uCosminexus may not appear in some product names displayed by the `dcplist` command.

dcrasget (Acquire maintenance data)

Format

```
dcrasget [-c] [-g] [-l] storage-directory
```

Function

The `dcrasget` command acquires the maintenance data required for troubleshooting OpenTP1 and stores it in the specified directory.

This command must be executed by users with OpenTP1 administrator permissions when they are operating OpenTP1. Confirm that the environment variables required for OpenTP1 operation are correctly set.

When this command is entered, the system displays the information about the target directory before acquiring maintenance data, and asks the user whether to continue the processing.

The storage directory needs to have a sufficient empty area to store maintenance data. This command acquires mainly the information under the `$DCDIR/spool` directory and the `$DCDIR/tmp` directory. The storage directory is also used as a temporary working area for acquiring maintenance data. Before you execute this command, check the size of the information under the `$DCDIR/spool` directory or the `$DCDIR/tmp` directory and specify a directory with sufficient space as the storage directory.

If there is an executable file (`$DCCONFPATH/usrrasget`) such as a shell script, the file is also executed when the command is executed. If you want to execute some processing in synchronization with the execution of the `dcrasget` command, create an executable file.

Options

- `-c`

Compresses the file that contains the acquired maintenance data.

- `-g`

Executes the command without displaying information about the target directory before maintenance data is acquired, and without asking the user whether to continue processing.

- `-l`

Does not acquire large amounts of information such as `$DCDIR/spool`, and acquires only information such as command execution results.

Command arguments

- *storage-directory* ~<path name>

Specifies a full path for the directory that will store the acquired maintenance data.

Since users with the OpenTP1 administrator permissions execute this command, set an appropriate access permission for the directory to be specified so that the user who executes the command can create files and directories.

A data file with the following name is created in the specified directory:

```
dcrasget.HOSTNAME.mmddHHMMSS.AAA.tar[.BB]
```

HOSTNAME: Standard host name

mmddHHMMSS: Month, day, hour, minutes, seconds

.z: Added only when the *-c* option is specified.

AAA: Unit of information to be acquired

SAV: Information under the *\$DCDIR/spool/save* directory

SPL: Information under the *\$DCDIR/spool* directory (except for the *save* directory)

TMP: Information under the *\$DCDIR/tmp* directory

CNF: Definitions and related information under the *\$DCDIR/conf* directory

INF: Information in information files and the like

.BB: Appended to the file name when the *-c* option is specified. The appended characters are *.gz* in Linux, and *.Z* in other operating systems.

Output messages

Message ID	Description	Output destination
KFCA01861-E	An error occurred during command processing.	Standard error output
KFCA01897-I	Description about how to use the command	Standard output

dcreport (Edit and output system statistics to standard output in real time)

Format

```
dcreport [-l [-n]] [-c] [-r] [start-ID [Δ end-ID]]
```

Function

The `dcreport` command edits the system statistics collected into shared memory, and outputs them to the standard output in real time.

This command is available when `set statistics=Y` is specified in the system common definition. If you use the `dcstats` command to output system statistics to a journal file, the `dcreport` command can output the system statistics to the standard output regardless of the specification in the system common definition.

For the IDs of statistics to be output, see the *ID for edition by dcreport* columns in *Table E-1* and *Table E-2*.

Options

- `-l`

Outputs system statistics to the standard output and the message log.

- `-n`

Suppresses the output of system statistics to the standard output.

- `-c`

Outputs system statistics to the standard output in CSV format.

If you specify both `-l` and `-c` simultaneously, system statistics are output to the message log using the normal format.

- `-r`

Outputs system statistics and resets the accumulated values.

Command arguments

- `[start-ID[Δ end-ID]] ~<unsigned integer>((1-213))`

Specify the range of IDs of the statistics to be output. For details about the IDs used for statistics, see the *ID for edition by dcreport* columns in *Table E-1* and *Table E-2*.

When you specify only a start ID, the command outputs all the system statistics after the start ID.

When you specify a start ID and an end ID, the command outputs the system statistics from the start ID up to the end ID.

If you omit this option, the command outputs all the system statistics.

When you specify both a start ID and an end ID, make sure that the end ID is equal to or greater than the start ID.

Output format

■ (1) Standard output

id	count	average	max	min
aaa	bbbbbbbbbb	cccccccccc	dddddddddd	eeeeeeeeee

aaa

ID of the statistics (decimal number of 3 digits)

bbbbbbbbbb

Count (decimal number of 10 digits)

cccccccccc

Average (decimal number of 10 digits)

dddddddddd

Maximum (decimal number of 10 digits)

eeeeeeeeee

Minimum (decimal number of 10 digits)

When all the output options (count, average, maximum, and minimum) are not acquired for a desired item, 0000000000 is output for the count and ----- is output for the average, max, and min.

If only the count is acquired for an item, ----- is output for the average, max, and min.

If an overflow occurs since the count or other acquired values are accumulated excessively, ***** is output for the count and average. In this case, the max and min are output.

■ (2)-c option specified (output format for the standard output (CSV format))

<i>aaa,bb...bb,cc...cc,dd...dd,ee...ee</i>
--

aaa

ID of the statistics (decimal number of up to 3 digits)

bb...bb

Count (decimal number of up to 10 digits)

cc...cc

Average (decimal number of up to 10 digits)

dd...dd

Maximum (decimal number of up to 10 digits)

ee...ee

Minimum (decimal number of up to 10 digits)

When all the output options (count, average, maximum, and minimum) are acquired for a desired item, 0 is output for the count and - is output for the average, max, and min.

When only the count is acquired for an item, - is output for the average, max, and min.

If an overflow occurs since the count or other acquired values are accumulated excessively, * is output for the count and average. In this case, max and min are output.

■ (3)-l option specified (output format for the log file)

KFCA01890-I statistics:ID=aaa, number of events=*bb...bb*, average=*cc...cc*, max=*dd...dd*, min=*ee...ee*

aaa

ID of the statistics (decimal number of up to 3 digits)

bb...bb

Number of events (decimal number of up to 10 digits)

cc...cc

Average (decimal number of up to 10 digits)

dd...dd

Maximum (decimal number of up to 10 digits)

ee...ee

Minimum (decimal number of up to 10 digits)

When all the output options (number of events, average, maximum, minimum) are not acquired for a desired item, no information is output.

When only the number of events is acquired for an item, - is output for the average,

dcreport (Edit and output system statistics to standard output in real time)

max, and min.

If an overflow occurs since the count or other acquired values are accumulated excessively, * is output for the number of events and average. In this case, the max and min are output.

Output messages

Message ID	Description	Output destination
KFCA01860-E	The input format of the <code>dcreport</code> command is incorrect.	Standard error output
KFCA01861-E	An error occurred during command processing.	Standard error output

Note

The accumulated values are not reset even if the `-r` option is specified while the `dcstats` command is outputting system statistics to a journal file.

dcreset (Restart process service and reflect definitions)

Format

dcreset

Function

The `dcreset` command reflects the changes in the system common definition to OpenTP1. When the KFCA00715-E message is output, the `dcreset` command resumes the interrupted start or restart processing (restarting interrupted process service).

When the system common definition has been changed, terminate OpenTP1 normally before executing the command.

When the KFCA00715-E message is output, eliminate the error according to the reason code of the message, and then execute the `dcreset` command while OpenTP1 is inactive.

Notes

- The system goes down if the `dcreset` command is executed while OpenTP1 is online.
- Only the superuser, or user who has the same group ID of the OpenTP1 administrator can execute the command.
- When `AUTO` is specified in the `mode_conf` operand of the system environment definition, OpenTP1 starts automatically when this command executes.

dcsetup (Catalog OpenTP1 into OS, delete OpenTP1 from OS)

Format

```
dcsetup { [-j] | -d[-y|-n] } OpenTP1-directory
```

Function

The `dcsetup` command catalogs OpenTP1 under a specified OpenTP1 directory into the OS, and starts or terminates a part of OpenTP1 with the OS.

The environmental variables `TZ` and `LANG` set in the `dcsetup` command execution environment are set in the OS as OpenTP1 environmental variables.

- `LANG`
- `TZ`

If environmental variable `TZ` or `LANG` is set in the `putenv` format of the system definition, they are overwritten when the definition phrase becomes effective.

The `dcsetup` command also deletes OpenTP1 from the OS. Terminate OpenTP1 before attempting deletion.

Only the superuser can execute the `dcsetup` command.

The input and output results of the `dcsetup` command are shown in message `KFCA01895-I` or `KFCA01896-I`. These messages are sent to `syslog`.

Options

■ `-j`

Prevents any installed OpenTP1 resource managers from being cataloged into the system when the `dcsetup` command is executed.

If `Y` is specified for the `jnl_fileless_option` operand in the system common definition, specify the `-j` option when you use the `dcsetup` command to set up OpenTP1.

■ `-d`

Specifies that the OpenTP1 under a specified OpenTP1 directory is to be deleted from the OS.

When this option is omitted, the OpenTP1 under the specified OpenTP1 directory is cataloged into the OS.

When the command is executed with this option specified and the `-y` and `-n` options both omitted, the system asks the operator whether files that are needed for execution

are to be deleted from the OpenTP1 directory.

If *y* is entered in response to this inquiry, the files needed for execution are deleted from the OpenTP1 directory. If *n* is entered, these files are not deleted.

To suppress output of this inquiry during command execution, execute the command with the *-y* or *-n* option specified.

■ *-y*

Suppresses querying of the operator about whether to delete the files required for execution of OpenTP1 in the specified OpenTP1 directory. When you specify this option, the files required for execution of OpenTP1 in the specified OpenTP1 directory are deleted.

■ *-n*

Suppresses querying of the operator about whether to delete the files required for execution of OpenTP1 in the specified OpenTP1 directory. When you specify this option, the files required for execution of OpenTP1 in the specified OpenTP1 directory are not deleted.

Command arguments

■ *OpenTP1-directory*~<path name>

Specifies the OpenTP1 directory. A directory name can be 1-50 characters long.

If multiple OpenTP1s are to be created, they must all be specified in the directory.

Notes

- Save the file name *inittabX* (X is the serial number from 1 to 3) for backup under *\$DCDIR/conf/Inittab*.
- While entering the *dcsetup* command, do not use an editor to edit */etc/inittab*. If you edit it, */etc/inittab* may be destroyed.
- Do not abort the *dcsetup* command processing by using a command such as the *kill* command. If processing stops, */etc/inittab* may be destroyed.
- You can only execute the *dcsetup* command when the instance of OpenTP1 you are cataloging is not running.
- The *dcsetup* command creates a temporary file in */tmp*. If the *dcsetup* command fails to create the temporary file, */etc/inittab* may be destroyed. Check that there is sufficient free space in */tmp* before executing the *dcsetup* command.
- Do not use a symbolic link to the OpenTP1 directory.
- Compilation processing by the *dcsetup* command searches for the target C compiler in the following sequence:

dcsetup (Catalog OpenTP1 into OS, delete OpenTP1 from OS)

1. /bin/cc and /lib/ccom

Both /bin/cc and /lib/ccom are required.

2. /usr/bin/cc

3. /usr/vac/bin/cc

If the target C compiler is not found in the above search sequence, OpenTP1 assumes the setting specified in the `PATH` environment variable of the process that executes the `dcsetup` command.

dcshmls (Display utilization status of common memory)

Format

```
dcshmls [-d {stt|dyn|all}] [-r]
```

Function

The `dcshmls` command outputs the utilization status of shared memory to the standard output during OpenTP1 operation.

Options

- `-d {stt|dyn|all}`

Specifies the shared memory whose utilization status is to be displayed.

`stt`

Detailed utilization status of static shared memory

`dyn`

Detailed utilization status of dynamic shared memory

`all`

Detailed utilization status of static and dynamic shared memories

If this option is not specified, the following information is displayed:

- Shared memory pool type
- Size of shared memory pool
- Total size of shared memory in current use
- Current utilization percentage of shared memory
- Maximum size of shared memory

- `-r`

Displays approximate values of the allocated size and the size in use, shared memory identifiers, and user type.

Output format

TYPE	SHMPOOL	SIZE(byte)	USING	SIZE	RATE(%)	USED	MAXIMUM	}	1
<i>aa...aa</i>		<i>0xbbbbbbbb</i>		<i>0xcccccccc</i>	<i>dd</i>	<i>0xeeeeeeee</i>			
:	:	:	:	:	:	:	:	}	2
Static_shm pool information									
USER		LOCATION		SIZE		OTHER		}	3
<i>fff(class=0xgg)</i>		<i>0xhhhhhhhh</i>		<i>0xiiiiiii</i>		<i>0xjjjjjjj</i>			
:	:	:	:	:	:	:	:	}	4
Dynamic_shm pool information									
USER		LOCATION		SIZE		OTHER		}	3
<i>kk...kk</i>		<i>0xlllllll(0xmmmmmmmm)</i>		<i>0xnnnnnnnn</i>		<i>oo...oo</i>			
:	:	:	:	:	:	:	:	}	4
SHMID SHMGET SIZE(byte) ACTIVE SIZE OWNER									
<i>pp...pp</i>		<i>0xqqqqqqq</i>		<i>0xrrrrrrr</i>		<i>ss...ss</i>		}	4
:	:	:	:	:	:	:	:		

1. Displayed when the `-d` option is specified or has not been specified.
2. Displayed when `-d stt` or `-d all` is specified.
3. Displayed when `-d dyn` or `-d all` is specified.
4. Displayed when the `-r` option is specified.

aa...aa

Shared memory pool type

static: Static common memory

dynamic: Dynamic common memory

bbbbbbbb

Size of shared memory pool (hexadecimal)

ccccccc

Total size of shared memory in current use (hexadecimal)

dd

Current utilization percentage (%) of shared memory (fractions rounded down)

eeeeeee

Maximum size of shared memory (hexadecimal)

fff

Type of service using static shared memory blocks (3 characters) (internal information)

gg

"class" value for static shared memory block management (hexadecimal)
(internal information)

hhhhhhh

Location of the static shared memory blocks from the beginning of the static shared memory pool (hexadecimal)

iiiiiii

Size of static shared memory blocks (hexadecimal)

jjjjjjj

"subclass" value for static shared memory block management (hexadecimal)
(internal information)

kk...kk

Process ID of the process for which dynamic shared memory blocks

lllllll

Location from the beginning of shared memory pool of dynamic shared memory blocks (hexadecimal)

mmmmmmmm

Location from the beginning of dynamic memory pool of dynamic shared memory blocks (hexadecimal)

nnnnnnnn

Size of dynamic shared memory blocks (hexadecimal)

oo...oo

Type of dynamic shared memory blocks

EMPTY: Unused

USE: In use

pp...pp

Shared memory identifier

qqqqqqqq

Size of shared memory pool which OS is requested to allocate (hexadecimal)

The size of the shared memory pool that the OS is requested to allocate is as follows:

size-of-shared-memory-pool-OS-is-requested-to-allocate =

*size-of-shared-memory-pool-for-OpenTP1-control +
size-of-static-shared-memory-pool + size-of-dynamic-shared-memory-pool*

rrrrrrrr

Approximate size of shared memory in current use (hexadecimal)

For the shared memory of TP1/Server Base, this indicates the size from the beginning of the shared memory pool for controlling OpenTP1 to the maximum usage level in the static shared memory pool being used as the shared memory. The shared memory of TP1/Server Base consists of (from the smallest address) the shared memory pool for controlling OpenTP1, the dynamic shared memory pool, and the static shared memory pool. Therefore, the value of *rrrrrrrr* is determined by adding the maximum usage size of the static shared memory pool to the size of the shared memory pool for controlling OpenTP1 and the size of the dynamic shared memory pool.

ss...s

Type of source requesting shared memory pool allocation

btn: TP1/Server Base

dam: TP1/FS/Direct Access

tam: TP1/FS/Table Access

ism: ISAM

mXX: MQA (where XX is a hexadecimal number between 00 and ff)

ist: TP1/Shared Table Access

rts: Real-time statistics service

Output messages

Message ID	Description	Output destination
KFCA00110-I	Help message	Standard output
KFCA00111-E	Option flag specification is invalid.	Standard error output
KFCA00112-E	No OpenTP1 execution environment.	Standard error output
KFCA00113-E	OpenTP1 shared memory cannot be referenced.	Standard error output
KFCA00114-E	Retrieval of utilization status failed, because shared memory is destroyed.	Standard error output
KFCA00123-E	There is a shared memory block whose usage status cannot be checked.	Standard error output

Note

To minimize the influence of the information output by this command on processing performance, the command references information without locking. Therefore, depending on the timing, the correct information may not be obtained. If OpenTP1 outputs message KFCA00123-W, re-execute the command. If the same message appears after executing the command again, the shared memory may be damaged. In this case, use `dcstop -df` to forcibly stop the system. Then, record the contents of message KFCA00123-W, save the core file and the shared memory dump, and contact the maintenance personnel.

dcstart (Start OpenTP1)

Format

```
dcstart [-ngU]
```

Function

The `dcstart` command starts OpenTP1 normally or restarts OpenTP1 in the computer.

Options

■ `-n`

Specifies that OpenTP1 is to be started normally, regardless of the previous termination mode.

When this option is omitted, the startup mode is determined from the previous termination mode. If the previous termination mode was normal termination, OpenTP1 is started normally; otherwise, OpenTP1 is restarted in order to inherit the previous processing status.

■ `-g`

Suppresses inquiries to the operator in the event of failure to start a user server or forced normal startup of OpenTP1.

■ `-U`

If you specify this option, the user server is not started when rerunning OpenTP1. Use this option to recover the OpenTP1 in which an error occurred by keeping OpenTP1 in the online mode without starting the user server.

You can also use this option to start OpenTP1 in the standby system in the system switchover configuration. By using this option, you can use OpenTP1 in the standby system for postprocessing, such as determination of undecided transactions and management of the integrity of the database for OpenTP1 in the running system. For details about starting OpenTP1 in the standby system by using this option, see the manual *OpenTP1 Description*.

Notes

- The `-g` option must be specified when executing the `dcstart` command in the background.
- Note the following when using the `dcstart -U` option:
 - `-U` is ignored for normal startup.

- This option's specification at the archive-journal target node is ignored.
- Use the `dcsvstart` command to start the user server by this option after starting the online operation.
- If you want to recover the user server configuration by this option after starting the online operation, shut down the system by means other than normal termination and forced normal termination and then start the online operation by the `dcstart` command without the `-U` option specified.
- If any messages remain in the disk queue ITQ when rerunning OpenTP1, the messages are discarded because the AP is not started. At this time, the AP which processes `ERREVT2` is also not started, so `ERREVT2` is not reported. This is the same when the application program is started on the user server and the messages remain in the disk queue ITQ or OTQ.
- If you specify this option when rerunning OpenTP1 after using the remote API facility at the previous online operation, the remote API configuration is not recovered. If you want to use the remote API facility after specifying this option, start the online operation and then start the RAP-processing listener manually by the `dcsvstart` command. Alternatively, shut down the system by means other than normal termination or forced normal termination and then start the online operation by the `dcstart` command without the `-U` option specified.
- The communication event processing SPP for the client/server mode communication using OSI TP is assumed as a UAP. If you use this facility when the communication event processing SSP is set to start at the start of OpenTP1, the communication event processing SPP does not start so that the association fault detection or reconnection cannot be done. To avoid this, be sure to start the communication event processing SPP after starting OpenTP1 by the `-U` option.
- Use of the server shutdown inheritance facility causes the following problems:
 1. The schedule shutdown status is not inherited in the online operation which is rerun by `dcstart -U`.
 2. If an SPP in the schedule shutdown status exists and you perform planned shutdown or forced shutdown of the online operation which was rerun by `dcstart -U` and you use `dcstart` to restart the online operation, the schedule shutdown status in the previous online operation before using `dcstart -U` is inherited. To avoid this, take the following countermeasures:
 - Perform the normal shutdown or forced normal startup of the online operation which was rerun by `dcstart -U`.
 - Specify `set hold_recovery=N` in all the user service definitions.

- Specify `set scd_hold_recovery_count=0` in the scheduling service definition.

- OpenTP1 may terminate abnormally if you use an OS command to forcibly stop the `dcstart` command.

- When the `dcstart` command is used to start OpenTP1, the process server is notified that OpenTP1 has started. If the notification fails, the `dcstart` command automatically retries notification.

When you use the function for retrying startup notification, the command outputs a KFCA01801-E message or a KFCA01861-E message if the notification fails and the command retries notification.

For details about the definition for retrying startup notification, see the manual *OpenTP1 System Definition*.

- The following messages may be output when you use the `dcstart` command to start OpenTP1:
 - KFCA01801-E (Cause: SETUP or PAUSE)
 - KFCA01861-E (Cause: COMMUNICATION or TIMEOUT)

If this occurs, take the following action:

- Execute the `dcstart` command again.
- Specify the retry count and retry interval for OpenTP1 startup in the `dcstart_wakeup_retry_count` and `dcstart_wakeup_retry_interval` operands, respectively, in the system common definition.

dcstats (Start and terminate collecting system statistics)

Format

```
dcstats { [-k statistics-type [, statistics-type] ...] [-m interval]
          [-{a|s}] | [server-name [  $\Delta$  server-name] ... ] | -r }
```

Function

The `dcstats` command outputs system statistics for specified servers, the system servers, or all servers to the system journal file at a specified interval.

Output of system statistics can be terminated with the `dcstats` command with the `-r` option specified. If the `dcstats` command with the `-r` option specified is not entered, output continues until OpenTP1 terminates.

The `dcstats` command does not acquire the user server-specific system statistics for a user server that is started after the `dcstats` command is executed.

The `dcstats` command cannot be used at archive-journal target nodes.

Options

■ `-k statistics-type`

Specifies the type of system statistics output. For more than one type of statistics, a comma (,) must be placed between types.

`rpc`: RPC information

`lck`: Lock information

`prc`: Process information

`nam`: Name information

`que`: Message queue information

`scd`: Scheduling information

`mcf`: MCF information

`dam`: DAM information

`tam`: TAM information

`trn`: Transaction information

`cpd`: Checkpoint dump information

`jnl`: Journal information

`osl`: Shared memory management information

dcstats (Start and terminate collecting system statistics)

ist: Node-to-node shared table information

mqa: MQA service information

xat: XATMI service information

When this option is omitted, all system statistics are output.

- *-m interval*~<unsigned integer> ((1-1440)) <<10>>

Specifies in minutes the interval at which the system statistics are to be output to the system journal file.

- *-a*

Specifies that system statistics for the system services and all user servers are to be output.

This option cannot be specified simultaneously with the *-s* option.

- *-s*

Specifies that system statistics for the entire system are to be output.

This option cannot be specified simultaneously with the *-a* option.

When neither the *-a* option nor the *-s* option is specified and no server name is specified, system statistics for the entire system are output.

- *-r*

Specifies that output of system statistics is to be terminated.

If this option is specified together with any other options, the *-r* option is effective, and all the other specified options are ignored.

Command arguments

- *server-name*~<1-8 alphanumeric characters>

Specifies the name of a user server for which system statistics are to be output.

System statistics will be output for each server whose name is specified.

If the *-a* option is specified, specification of a server name is ignored.

Notes

- The specification of the *dcstats* command is not inherited during an OpenTP1 restart. To output system statistics after an OpenTP1 restart, the *dcstats* command must be reentered.
- Once a user server is terminated, specification of output of its system statistics is not inherited during normal startup. To output system statistics for a user server that has been terminated, the *dcstats* command must be reentered after the user

server's normal startup processing has been completed.

dcstop (Terminate OpenTP1)

Format

```
dcstop [{-n|-a|-b|-f} [-d]]
```

Function

The `dcstop` command terminates OpenTP1 in the computer.

Options

■ `-n`

Specifies that OpenTP1 is to be terminated forcibly even if:

- A user server has forcibly been stopped by the `dcsvstop -f` command.
- The abnormally terminated user server is an SUP or an SPP that receives requests from the socket.

The difference from normal termination is that forced termination checks the forcibly stopped or abnormally terminated servers and continues normal termination of the system.

■ `-a`

Specifies that OpenTP1 is to be terminated by planned termination A.

■ `-b`

Specifies that OpenTP1 is to be terminated by planned termination B.

■ `-d`

Collects the core files from all active processes and the shared memory information when OpenTP1 is forced to terminate. This option can be specified to collect core files and shared memory information as troubleshooting information when the `dcstop` command has not been terminated or when a system error has occurred.

The `-d` option must be specified with the `-f` option.

■ `-f`

Specifies that OpenTP1 is to be terminated forcibly.

When none of these options are specified, OpenTP1 is terminated normally.

The `-n`, `-a`, `-b`, and `-f` options are mutually exclusive.

Output messages

Message ID	Description	Output destination
KFCA01801-E	DCDIR environment variable has not been set.	Standard error output
KFCA01821-E	OpenTP1 cannot be resumed.	Message log file
KFCA01840-I	OpenTP1 is terminating.	Message log file
KFCA01841-I	OpenTP1 has been terminated.	Standard output
KFCA01842-I	A server is terminating.	Message log file
KFCA01844-E	Server termination processing failed.	Standard error output
KFCA01846-E	System cannot be terminated normally.	Message log file, standard error output
KFCA01847-E	System cannot be terminated.	Message log file, standard error output
KFCA01848-I	All user servers have been terminated.	Message log file
KFCA01849-W	Waiting for server termination.	Message log file, standard output
KFCA01860-E	dcstop command input format is invalid.	Standard error output
KFCA01861-E	dcstop command processing failed.	Standard error output
KFCA01863-E	dcstop command cannot be executed.	Message log file, standard error output
KFCA01864-E	Overtime occurred.	Message log file, standard error output
KFCA01865-E	dcstop command has been cancelled.	Message log file, standard error output
KFCA01867-E	Shared memory cannot be allocated.	Standard error output
KFCA01869-E	dcstop command cannot be executed.	Standard error output

Notes

- If the `dcstop` command is entered when an error occurs with the system service, OpenTP1 may abnormally terminate at some timing.
- A partially recovered OpenTP1 cannot be terminated.
- A server is inhibited from restarting during termination process.
- When the `dcstop` command fails to terminate or a system error occurs, the cause

can be investigated by forcing to stop OpenTP1 with the `dcstop -fd` command and collecting core files and shared memory information.

- When the `dcstop -fd` command is entered, the machine load increases because of the operation to collect error information. Where two or more OpenTP1s are operating on the same machine, a timeout may occur at another OpenTP1 for a multiOpenTP1 configuration, or a slowdown or timeout for an HA configuration (such as a mutually hot-standby configuration).
- For HP-UX, the linkage bind mode of the user server started by the `dcsvstart` command must be `immediate`. When an executable file linked in any other mode is used, operations of OpenTP1 cannot be guaranteed. If the user server terminated abnormally for unknown reason, use the OS's `chatr` command to check that the bind mode is `immediate`.
- OpenTP1 may terminate abnormally if you use an OS command to forcibly stop the `dcstop` command.

dcsvstart (Start servers)

Format

```
dcsvstart -u user-server-name [, user-server-name] . . . [-a argument]
```

Function

The `dcsvstart` command starts specified servers.

The `dcsvstart` command cannot be used at archive-journal target nodes.

Options

- `-u user-server-name~<identifier, 1-8 characters>`

Specifies the name of a user server to be started.

A maximum of 50 user server names can be specified in a single `dcsvstart` command.

To specify multiple user server names, delimit them using commas (,), but do not insert any spaces between a user server name and the delimiter.

When this option is omitted, no user servers are started.

- `-a argument~<1-127 character string, excluding commas (,) and double-quotes (")>`

Specifies the first argument to start the SUP of the server(s) specified by the `-u` option.

If more than one server is specified by the `-u` option, the characters specified in the `-a` option are passed for the first argument of all the servers specified.

Output messages

Message ID	Description	Output destination
KFCA01801-E	DCDIR environment variable has not been set.	Standard error output
KFCA01811-I	Server is being started.	Message log file
KFCA01812-E	Server startup processing failed.	Message log file, standard error output
KFCA01813-I	Server is in online mode.	Message log file
KFCA01815-E	Error occurred in the operating system.	Standard error output
KFCA01818-E	Server startup is being cancelled.	Standard error output
KFCA01860-E	dcsvstart command input format is invalid.	Standard error output

Message ID	Description	Output destination
KFCA01861-E	dcsvstart command processing failed.	Standard error output
KFCA01863-E	Shared cannot be executed.	Message log file, standard error output
KFCA01866-E	Server name duplication error occurred.	Standard error output
KFCA01867-E	Shared memory cannot be allocated.	Standard error output
KFCA01868-E	Server name is invalid.	Standard error output
KFCA01869-E	dcsvstart command cannot be accepted.	Standard error output
KFCA01870-E	Memory shortage occurred.	Message log file, standard error output

Notes

- If the `dcsvstart` command does not end, the cause can be identified by forcibly terminating the UAP with the `dcsvstop -df` command and collecting the core file. When the `dcsvstop -df` command is executed, the `dcsvstart` command also outputs troubleshooting information to the core file and terminates. Abnormal termination of the `dcsvstart` command does not lead to abnormal termination of OpenTP1. However, OpenTP1 may terminate abnormally in the following cases:
 - When `critical=Y` is specified in the user service definition for the user server specified in the `dcsvstop -df` command
 - When the `dcsvstart` command is forcibly stopped by using an OS command
- To start an MHP, you need to release all the services for the MHP from shutdown status or hold status and then execute the `dcsvstart` command. You can check the status of services by using the `mcfalseap` command, the `mcftlssv` command, and the `mcftlssg` command.

dcsvstop (Terminate server)

Format

```
dcsvstop [-f [-d]] user-server-name [ user-server-name ] . . .
```

Function

The `dcsvstop` command terminates a server.

The `dcsvstop` command cannot be used at archive-journal target nodes.

Options

- `-d`

Collects the user server core file when a user server is forcibly terminated. This option can be specified to collect core files as troubleshooting information when the `dcsvstart` or `dcsvstop` command has not been terminated. The option must always be used with the `-f` option.

- `-f`

Specifies that the specified user server is to be terminated forcibly.

When this option is omitted, the specified server is terminated normally.

Command arguments

- `user-server-name~<identifier, 1-8 characters>`

Specifies the name of the user server to be terminated. You can specify a maximum of 50 user server names in the `dcsvstop` command.

Output messages

Message ID	Description	Output destination
KFCA01801-E	DCDIR environment variable has not been set.	Standard error output
KFCA01821-E	OpenTP1 cannot be continued.	Message log file
KFCA01842-I	Server is terminating.	Message log file
KFCA01843-I	Server has been terminated.	Message log file
KFCA01844-E	Server termination processing failed.	Message log file, standard error output
KFCA01849-W	Waiting for server termination.	Message log file, standard output

Message ID	Description	Output destination
KFCA01860-E	dcsvstop command input format is invalid.	Standard error output
KFCA01861-E	dcsvstop command processing failed.	Message log file, standard error output
KFCA01862-E	Enter a server name.	Message log file, standard error output
KFCA01863-E	dcsvstop command cannot be executed.	Message log file, standard error output
KFCA01864-E	Overtime occurred.	Message log file, standard error output
KFCA01866-E	Server name duplication error occurred.	Standard error output
KFCA01867-E	Shared memory cannot be allocated.	Standard error output
KFCA01868-E	Server name is invalid.	Standard error output
KFCA01869-E	dcsvstop command cannot be accepted.	Standard error output

Notes

- If the `dcsvstop` command does not end, the cause can be identified by forcibly terminating the UAP with the `dcsvstop -df` command and collecting the core file. When the `dcsvstop -df` command is executed, the `dcsvstop` command also outputs troubleshooting information to the core file and terminates. Abnormal termination of the `dcsvstop` command does not lead to abnormal termination of OpenTP1. However, OpenTP1 may terminate abnormally in the following cases:
 - When `critical=Y` is specified in the user service definition for the user server specified in the `dcsvstop -df` command
 - When the `dcsvstop` command is forcibly stopped by using an OS command
- When you execute the `dcsvstop` command for an MHP, the MHP may not end since messages are retained in the input queue. In this case, determine whether to shut down the MCF (by using the `mcfstdctsg` command and the `mcfthldiq`) taking into consideration the message processing after the MHP is ended by execution of the `dcsvstop` command.

filbkup (Back up an OpenTP1 file system)

Format

```
filbkup [-y] [-r] [-i] [-{c|l|f}]
        OpenTP1-file-system-area-name[/OpenTP1-file-name]
        backup-file-name
```

Function

The `filbkup` command backs up a specified OpenTP1 file system. The attributes of the OpenTP1 files are also backed up. Only one OpenTP1 file system can be backed up into a backup file. To back up multiple OpenTP1 file systems, as many files are required as there are OpenTP1 file systems.

The `filbkup` command requests confirmation from the user before backing up an OpenTP1 file system, but there is an option that enables backup processing to be started immediately without requesting user confirmation.

Only the OpenTP1 administrator can execute the `filbkup` command.

An OpenTP1 file system that has been backed up can be restored with the `filrstr` command.

It is also possible to back up a specified OpenTP1 file, but this method is usually used to collect maintenance information.

Options

- `-y`

Specifies that the OpenTP1 file system is to be backed up immediately without requesting user confirmation.

When this option is omitted, the OpenTP1 file system is backed up after the user confirms the backup request.

- `-r`

Specifies that if the backup destination file already exists, it is to be deleted before the backup file is created.

When this option is omitted and the backup destination file already exists, the `filbkup` command terminates in an error.

- `-i`

Specifies that if a read error occurs in an OpenTP1 file, backup of that file is to be cancelled and the next file is to be read. The remaining area of the file resulting in the error is cleared.

When this option is omitted and a read error occurs in an OpenTP1 file, the `filbkup` command terminates in an error and an error message is output.

■ `-{c|l|f}`

`-c`

Specifies that if an OpenTP1 file in the specified OpenTP1 file system is being used by another process or another user, the OpenTP1 file system is not to be backed up.

`-l`

Specifies that if an OpenTP1 file in the specified OpenTP1 file system is being used by another process or another user, that OpenTP1 file (only) is not to be backed up.

`-f`

Specifies that even if an OpenTP1 file in the specified OpenTP1 file system is being used by another process or another user, the OpenTP1 file system is to be backed up forcibly.

Command arguments

■ `OpenTP1-file-system-area-name~<path name>`

Specifies the name of the character special file or ordinary file containing the OpenTP1 file system to be backed up.

■ `OpenTP1-file-name~<OpenTP1 file name>`

Specifies the name of an OpenTP1 file to be backed up.

When this command argument is specified, only the specified OpenTP1 file is backed up.

■ `backup-file-name~<path name>`

Specifies the name of the backup destination file.

Notes

- If an OpenTP1 file in the specified OpenTP1 file system is being used by another process or another user, backup of the OpenTP1 file system will not be executed unless the `-l` or `-f` option is specified for forcible backup.
- If an error occurs during OpenTP1 file system backup processing, the contents of the backup destination file cannot be guaranteed, because backup processing terminates immediately; the backup destination file is not deleted automatically.

filchgrp (Change an OpenTP1 file group)

Format

```
filchgrp group OpenTP1-file-system-area-name/OpenTP1-file-name
```

Function

The `filchgrp` command changes a specified OpenTP1 file group to a specified group.

Only the OpenTP1 administrator and the OpenTP1 file owner can execute the `filchgrp` command.

Command arguments

- *group*
Specifies a group ID (as decimal digits) or a group name in the group file (`/etc/group`).
If the specified character string consists of numeric digits only and the matching group name is not found among the `/etc/group` entries, the specified character string is treated as a group ID expressed in decimal digits.
- *OpenTP1-file-system-area-name~<path name>*
Specifies the name of the character special file or ordinary file containing the OpenTP1 file system.
- *OpenTP1-file-name~<OpenTP1 file name>*
Specifies the name of an OpenTP1 file.

Output messages

Message ID	Description	Output destination
KFCA01507-I	Help message	Standard output, standard error output
KFCA01533-E	Specified OpenTP1 file was not found.	Standard error output
KFCA01536-E	Specification of a command argument is invalid.	Standard error output
KFCA01541-E	Specified group name was not found.	Standard error output
KFCA01543-E	Specified group ID value is too large.	Standard error output
KFCA01570-E	File versions do not match.	Standard error output
KFCA01571-E	User does not have access right to the file.	Standard error output

filchgrp (Change an OpenTP1 file group)

Message ID	Description	Output destination
KFCA01572-E	An upper bound was exceeded during open processing.	Standard error output
KFCA01573-E	Specified OpenTP1 file system area was not found.	Standard error output
KFCA01574-E	Specified file name is too long.	Standard error output
KFCA01575-E	Specified file is not a character special file or ordinary file; or, specified file was found, but there is no corresponding device.	Standard error output
KFCA01576-E	Specified file is not an OpenTP1 file system.	Standard error output
KFCA01577-E	User does not have right to execute <code>filchgrp</code> command.	Standard error output
KFCA01578-E	Locking the OpenTP1 file system failed.	Standard error output
KFCA01582-E	Memory is inadequate.	Standard error output
KFCA01583-E	I/O error occurred.	Standard error output
KFCA01599-E	Error occurred.	Standard error output

filchmod (Change an OpenTP1 file access authorization mode)

Format

```
filchmod mode OpenTP1-file-system-area-name/OpenTP1-file-name
```

Function

The `filchmod` command changes the access authorization for a specified OpenTP1 file to a specified mode.

Only the OpenTP1 administrator and the OpenTP1 file owner can execute the `filchmod` command.

Command arguments

- *mode*

Specifies the new mode in absolute format or symbolic format.

The absolute format is explained below:

Absolute format	Description
0400	Right to read by the owner
0200	Right to write by the owner
0040	Right to read by the group
0020	Right to write by the group
0004	Right to read by others
0002	Right to write by others

The symbolic format must be specified as follows:

[to-whom] operation [access-right]

to-whom: Specifies who is authorized:

u: Owner

g: Group

o: Others

a: Owner, group, and others

When this specification is omitted, a is assumed.

operation: Specifies the operation to be performed:

+: Add an access right

-: Remove an access right

=: Remove all access rights other than those specified

access-right: Specifies an access right

r: Authorized to read

w: Authorized to write

The access-right can be omitted only when "=" is specified as the operation. When the access-right is omitted, neither read nor write can be performed.

■ *OpenTP1-file-system-area-name*~<path name>

Specifies the name of the character special file or ordinary file containing the OpenTP1 file system.

■ *OpenTP1-file-name*~<OpenTP1 file name>

Specifies the name of the OpenTP1 file.

Output messages

Message ID	Description	Output destination
KFCA01508-I	Help message	Standard output, standard error output
KFCA01533-E	Specified OpenTP1 file was not found.	Standard error output
KFCA01536-E	Specification of a command argument is invalid.	Standard error output
KFCA01538-E	Specification of mode argument is invalid.	Standard error output
KFCA01570-E	File versions do not match.	Standard error output
KFCA01571-E	User does not have access right to the file.	Standard error output
KFCA01572-E	An upper bound was exceeded during open processing.	Standard error output
KFCA01573-E	Specified OpenTP1 file system area was not found.	Standard error output
KFCA01574-E	Specified file name is too long.	Standard error output
KFCA01575-E	Specified file is not a character special file or ordinary file; or, specified file was found, but there is no corresponding device.	Standard error output
KFCA01576-E	Specified file is not an OpenTP1 file system.	Standard error output
KFCA01577-E	User does not have right to execute <code>filchmod</code> command.	Standard error output

Message ID	Description	Output destination
KFCA01578-E	Locking the OpenTP1 file system failed.	Standard error output
KFCA01582-E	Memory is inadequate.	Standard error output
KFCA01583-E	I/O error occurred.	Standard error output
KFCA01599-E	Error occurred.	Standard error output

Examples

Following are three examples of adding the read right to all users using the symbolic format, where the special file name is "/dev/rhp2a" and the OpenTP1 file name is "file":

```
filchmod ugo+r /dev/rdisk/rhd2a/file
filchmod a+r /dev/rdisk/rhd2a/file
filchmod +r /dev/rdisk/rhd2a/file
```

filchown (Change an OpenTP1 file owner)

Format

```
filchown owner OpenTP1-file-system-area-name/OpenTP1-file-name
```

Function

The `filchown` command changes the owner of a specified OpenTP1 file.

Only the OpenTP1 administrator and the OpenTP1 file owner can execute the `filchown` command.

Command arguments

- *owner*

Specifies the new owner's user ID (as decimal digits) or login name in the password file (`/etc/passwd`).

If the specified character string consists of numeric digits only and the matching login name is not found among the `/etc/passwd` entries, the specified character string is treated as a user ID expressed in decimal digits.

- *OpenTP1-file-system-area-name~<path name>*

Specifies the name of a character special file or ordinary file containing the OpenTP1 file system.

- *OpenTP1-file-name~<OpenTP1 file name>*

Specifies the name of the OpenTP1 file.

Output messages

Message ID	Description	Output destination
KFCA01506-I	Help message	Standard output, standard error output
KFCA01533-E	Specified OpenTP1 file was not found.	Standard error output
KFCA01536-E	Specification of a command argument is invalid.	Standard error output
KFCA01540-E	Specified login name was not found.	Standard error output
KFCA01542-E	Specified user ID value is too large.	Standard error output
KFCA01570-E	File versions do not match.	Standard error output
KFCA01571-E	User does not have access right to the file.	Standard error output

Message ID	Description	Output destination
KFCA01572-E	An upper bound was exceeded during open processing.	Standard error output
KFCA01573-E	Specified OpenTP1 file system area was not found.	Standard error output
KFCA01574-E	Specified file name is too long.	Standard error output
KFCA01575-E	Specified file is not a character special file or ordinary file; or, specified file was found, but there is no corresponding device.	Standard error output
KFCA01576-E	Specified file is not an OpenTP1 file system.	Standard error output
KFCA01577-E	User does not have right to execute <code>filchown</code> command.	Standard error output
KFCA01578-E	Locking the OpenTP1 file system failed.	Standard error output
KFCA01582-E	Memory is inadequate.	Standard error output
KFCA01583-E	I/O error occurred.	Standard error output
KFCA01599-E	Error occurred.	Standard error output

fills (Display contents of an OpenTP1 file system)

Format

```
fills { [-H] [-L] [{t|u}] ] OpenTP1-file-system-area-name  
      [/OpenTP1-file-name] | [-x] OpenTP1-file-system-area-name  
      [/OpenTP1-file-name] }
```

Function

The `fills` command outputs to the standard output the contents of an OpenTP1 file system.

If only an OpenTP1 file system area name is specified in the command arguments, all OpenTP1 files in the specified OpenTP1 file system are output in alphabetical order. If an OpenTP1 file name is also specified, only the specified OpenTP1 file is output.

Options

■ `-H`

Specifies the header for the information to be displayed. The information is displayed vertically in alphabetical order of the file names.

■ `-L`

Specifies that the lock status of the files is to be displayed vertically in alphabetical order of the file names.

■ `-{t|u}`

`-t`

Specifies that the OpenTP1 file system contents are to be displayed sequentially beginning with the file with the most recent update date and time.

`-u`

Specifies that the OpenTP1 file system contents are to be displayed sequentially beginning with the most recent access date and time.

The `-t` and `-u` options cannot both be specified together with the `-H` and `-L` options. The order of displaying the information is governed by the `-t` or the `-u` option specification.

■ `-x`

Specifies that only the file names are to be displayed, listed horizontally in alphabetical order.

When this option is omitted, the OpenTP1 file system contents are displayed vertically

in alphabetical order of the file names.

Command arguments

- *OpenTP1-file-system-area-name*~<path name>

Specifies the name of the character special file or ordinary file containing the OpenTP1 file system.

- *OpenTP1-file-name*~<OpenTP1 file name>

Specifies the name of an OpenTP1 file.

Output format

- *(l)-H option specified*

MODE	UID	GID	RSIZE	RNUM	TIME	FILE
<i>aabbcc</i>	<i>dd...dd</i>	<i>ee...ee</i>	<i>ffff</i>	<i>ggggg</i>	<i>hh...hh</i>	<i>ii...ii</i>

aa

Owner's access right:

r: Has the read right

w: Has the write right

-: Does not have the read or write right

bb

Group's access right:

r: Has the read right

w: Has the write right

-: Does not have the read or write right

cc

Others' access right:

r: Have the read right

w: Have the write right

-: Do not have the read or write right

dd...dd

Owner name (up to 9 characters)

ee...ee

Owner's group name (up to 9 characters)

fills (Display contents of an OpenTP1 file system)

ffff

Record length

ggggg

Number of records

hh...hh

Last update date and time (in the format hour: minute Δ month-day-year)

ii...ii

OpenTP1 file name

■ (2)-H and -L options specified

MODE	UID	GID	PID	L	TIME	FILE
<i>aabbcc</i>	<i>dd...dd</i>	<i>ee...ee</i>	<i>pppp</i>	<i>q</i>	<i>hh...hh</i>	<i>ii...ii</i>

aa

Owner's access right:

r: Has the read right

w: Has the write right

-: Does not have the read or write right

bb

Group's access right:

r: Has the read right

w: Has the write right

-: Does not have the read or write right

cc

Others' access right:

r: Have the read right

w: Have the write right

-: Do not have the read or write right

dd...dd

Owner name (up to 9 characters)

ee...ee

Owner's group name (up to 9 characters)

pppp

ID of the locked process (if no process has been locked, "-" is displayed)

q

Lock status identifier flag:

E: Exclusive lock

S: Shared lock

-: No lock

hh...hh

Last update date and time (in the format hour: minute Δ month-day-year)

ii...ii

OpenTP1 file name

Output messages

Message ID	Description	Output destination
KFCA01503-I	Help message	Standard output, standard error output
KFCA01533-E	File was not found.	Standard error output
KFCA01536-E	Specification of a command argument is invalid.	Standard error output
KFCA01537-E	No option flag is specified or the combination of option flags is invalid.	Standard error output
KFCA01570-E	File versions do not match.	Standard error output
KFCA01571-E	User does not have access right to the file.	Standard error output
KFCA01572-E	An upper bound was exceeded during open processing.	Standard error output
KFCA01573-E	Specified OpenTP1 file system area was not found.	Standard error output
KFCA01574-E	Specified file name is too long.	Standard error output
KFCA01575-E	Specified file is not a character special file or ordinary file; or, specified file was found, but there is no corresponding device.	Standard error output
KFCA01576-E	Specified file is not an OpenTP1 file system.	Standard error output
KFCA01578-E	Locking of OpenTP1 file system failed.	Standard error output
KFCA01582-E	Memory is inadequate.	Standard error output
KFCA01583-E	I/O error occurred.	Standard error output

fills (Display contents of an OpenTP1 file system)

Message ID	Description	Output destination
KFCA01599-E	Error occurred.	Standard error output

filmkfs (Initialize an OpenTP1 file system)

Format

- (1) *Character special file*

```
filmkfs -s sector-length -n size -l maximum-files-count
        [-v OpenTP1-file-system-name] special-file-name
```

- (2) *Normal file*

```
filmkfs [-r] -n size -l maximum-files-count
        [-v OpenTP1-file-system-name] ordinary-file-name
```

Function

The `filmkfs` command initializes a specified hard disk partition or ordinary file for an OpenTP1 file system. Initialization is normally performed only once when an ordinary file or a partition is allocated to an OpenTP1 file system.

For a character special file, only a superuser or an owner of the character special file can execute the `filmkfs` command. For an ordinary file, only a superuser or an authorized user who can write into the directory in which the ordinary file is to be created can execute this command. If the ordinary file already exists, a superuser or an owner of the ordinary file can execute this command.

Options

- `-r`

Specifies that initialization is to be performed without requesting user confirmation. If the specified ordinary file already exists, its contents will be lost.

When this option is omitted and the specified ordinary file already exists, the command requests confirmation from the user that the file is to be initialized.

- `-s sector-length`

Specifies the sector length of the hard disk on which the OpenTP1 file system is to be constructed.

For the `-s` option, specify the sector length corresponding to the specified character special file. If you specify an invalid sector length, `filmkfs` command processing will fail with an error message indicating an I/O error, invalid argument, or another problem.

- `-n size~((1-4095))` (unit: MB)

Specifies in Mbytes the size of the space to be allocated to the OpenTP1 file system.

For the `-n` option, specify a value less than or equal to the capacity of the character special file. If you specify a value that is larger than the size of the disk volume or partition, `filmkfs` command processing will fail with an error message indicating insufficient disk space, an I/O error, or another problem.

- `-l maximum-files-count~((1-4096))`

Specifies the maximum number of files to be created in the OpenTP1 file system.

- `-v OpenTP1-file-system-name~<OpenTP1 file name, 1-8 characters>`

Specifies the name to be assigned to the OpenTP1 file system.

When this option is omitted, no name is assigned to the OpenTP1 file system.

Command arguments

- `special-file-name~<path name>`

Specifies the name of a special file to be initialized.

The specified file must be a character special file.

- `ordinary-file-name~<path name>`

Specifies the name of the ordinary file to be initialized.

Notes

- If `filmkfs` command processing fails with an error message indicating insufficient disk space, an I/O error, or some other problem, the OpenTP1 system administrator must accurately determine the size of the OpenTP1 file system and then allocate the disk space accordingly.
- When specifying a character special file as the OpenTP1 file system, do not specify the partition used for the UNIX file system.
- You can specify a maximum of 49 characters for an OpenTP1 file system area name. If you specify more than 49 characters, OpenTP1 outputs message KFCA01574-E.
- If you want to ensure the reliability of OpenTP1 file systems, you must use character special files.

filrstr (Restore an OpenTP1 file system)

Format

```
filrstr [-y] [-q] [-{t|o|r}] [-{c|1}]
        backup-file-name [/OpenTP1-file-name]
        OpenTP1-file-system-area-name
```

Function

The `filrstr` command restores to an OpenTP1 file system area an OpenTP1 file system backed up by the `filbkup` command.

The `filrstr` command requests confirmation from the user before restoring the OpenTP1 file system, but there is an option that enables restoration processing to be performed immediately without requesting user confirmation.

Only the superuser and a user authorized to initialize the OpenTP1 file system can execute the `filrstr` command.

It is also possible to restore a specified OpenTP1 file by specifying its OpenTP1 file name, but this method is usually used to collect maintenance information.

Options

- -y

Specifies that the OpenTP1 file system is to be restored immediately without requesting user confirmation.

When this option is omitted, the OpenTP1 file system is restored after the user confirms restoration.

- -q

Specifies that file management information on the OpenTP1 file system is to be written in a batch and then the OpenTP1 file system is to be restored at high speed.

- -{t|o|r}

Specifies the handling of restoration when the restoration source file and the restoration destination OpenTP1 file system contain the same OpenTP1 file name:

- t

Compare their most recent update dates and times and restore the OpenTP1 file from the restoration source file only if it has the more recent date and time;

- o

Do not restore an OpenTP1 file with the same name;

-r

Restore the OpenTP1 file regardless of the most recent update date and time.

■ {c|l}

-c

Specifies that if an OpenTP1 file in the OpenTP1 file system is being used by another process or another user, the OpenTP1 file system is not to be restored.

-l

Specifies that if an OpenTP1 file in the OpenTP1 file system is being used by another process or another user, that OpenTP1 file only is not to be restored.

Command arguments

■ *backup-file-name*~<path name>

Specifies the name of the file containing the backup of the OpenTP1 file system.

■ *OpenTP1-file-name*~<OpenTP1 file name>

Specifies the name of an OpenTP1 file to be restored.

When this command argument is specified, only the specified OpenTP1 file is restored.

■ *OpenTP1-file-system-area-name*~<path name>

Specifies the name of the character special file or ordinary file into which the OpenTP1 file system is to be restored. The specified character special file or ordinary file must have already been initialized as an OpenTP1 file system.

Notes

- The `filbkup` command backs up an invalid file whose file management table has been isolated due to process cancellation. If the `-r` option is specified and a file is found with the same name as such an invalid file, that file will be replaced with the invalid file. For this reason, the `-t` or `-o` option should usually be specified.
- If an OpenTP1 file in the specified OpenTP1 file system is being used by another process or another user, the OpenTP1 file system will not be restored unless the `-l` option is specified for forcible restoration.
- If the `filrstr` command is executed without the restoration destination OpenTP1 file system having been initialized, the previous OpenTP1 files are retained after the `filrstr` command has executed, because the OpenTP1 file system is restored in units of OpenTP1 files.
- If the record length of a file in the backup file is less than the sector length of the restoration destination special file, or if it is not a multiple of the sector length of

the restoration destination special file, that file will not be restored and restoration processing will be performed on the next file.

- If the OpenTP1 file system was not backed up correctly, a warning message is issued and restoration processing on the OpenTP1 file system is resumed.
- If you create a backup file using the `filbkup` command that can handle an OpenTP1 file system up to 4 GB or OpenTP1 files of up to 4 GB, you cannot restore the file using the `filrstr` command, since it can only handle up to 4 GB. If you use the `filbkup` command, which cannot handle 4 GB for an OpenTP1 file system of 4 GB, operation is not guaranteed.
- If the `filrstr` command terminates abnormally due to an error or in response to the kill command provided with the operating system, the integrity of the OpenTP1 file system will be compromised. If this occurs, remove the cause of the error and restore the file system.

If the `filrstr` command with the `-q` option specified terminates abnormally, the management information for the OpenTP1 file system is restored at the end of the restoration process, and the file system will no longer be recognized as the OpenTP1 file system. If this occurs, initialize the OpenTP1 file system.

filstatfs (Display status of an OpenTP1 file system)

Format

```
filstatfs [-w] [-S] OpenTP1-file-system-area-name
```

Function

The `filstatfs` command outputs to the standard output the status of a specified OpenTP1 file system.

Option

- `-w`

Displays the start location of OpenTP1 file management area. This option can be specified only when the OS is AIX.

- `-S`

Displays user area information, including listings of the used areas and the unused (free) areas.

Command arguments

- *OpenTP1-file-system-area-name*~<path name>

Specifies the name of a character special file or ordinary file containing the OpenTP1 file system.

Output format

```

sector-length [B]                : aa...aa
user-area-total-capacity [KB]    : bb...bb
remaining-user-area-capacity [KB]: cc...cc
maximum-creatable-file-size [KB]: dd...dd
maximum-number-of-creatable-files: ee...ee
number-of-created-files         : ff...ff
number-of-remaining-creatable-files: gg...gg
total-number-of-free-areas      : hh...hh
file-system-initialization-user-ID: ii...ii
file-system-initialization-time : jj...jj
file-system-name               : kk...kk
OpenTP1-file-management-area-start-position : ll...ll           1

OpenTP1 file system user area information
  OFFSET [KB]  SIZE [KB]  USED/UNUSED  FILE
  mm...mm    nn...nn    oo...oo    pp...pp
  :           :         :           :
  } 2

```


- 1: Displayed when the `-w` option is specified.
- 2: Displayed when the `-s` option is specified.

*aa...aa*Sector length specified with the `-s` option in the `filmkfs` command*bb...bb*

Total size of area allocated to the user in the OpenTP1 file system

cc...cc

Size of free space (not allocated as OpenTP1 files) in the area allocated to the user

dd...dd

Maximum size currently allocatable to a single file (maximum size of contiguous free area, excluding the management information (2 sectors in length))

*ee...ee*Maximum number of creatable files specified with the `-l` option in the `filmkfs` command*ff..ff*

Number of files already created

gg...gg

Number of remaining creatable files (maximum number of creatable files - number of files already created)

hh...hh

Number of noncontiguous file areas

ii...ii

User's login name used to initialize the OpenTP1 file system

*jj..jj*Time at which the OpenTP1 file system was initialized (in the format day-of-week Δ month Δ day Δ hour:minute:second Δ year)*kk...kk*OpenTP1 file system name specified in the `filmkfs` command*ll...ll*

Start location of OpenTP1 file management area

mm...mm

Offset from the user area start position

nn...nn

Size of used or unused area in the user area

oo...oo

Whether display is of used or unused area in the user area:

used: Used area

unused: Unused area

pp...pp

OpenTP1 file name corresponding to an area being used ("-" is displayed in the case of an unused area).

Output messages

Message ID	Description	Output destination
KFCA01536-E	Specification of a command argument is invalid.	Standard error output
KFCA01551-I	Help message	Standard output, standard error output
KFCA01570-E	File versions do not match.	Standard error output
KFCA01571-E	User does not have access right to the file.	Standard error output
KFCA01572-E	An upper bound was exceeded during open processing.	Standard error output
KFCA01574-E	Specified file name is too long.	Standard error output
KFCA01575-E	Specified file is not a character special file or ordinary file; or, the specified file was found, but there is no corresponding device.	Standard error output
KFCA01576-E	Specified file is not an OpenTP1 file system.	Standard error output
KFCA01582-E	Memory shortage occurred.	Standard error output
KFCA01583-E	I/O error occurred.	Standard error output
KFCA01599-E	Error occurred.	Standard error output

Notes

- Creation of an OpenTP1 file requires a contiguous free area that at least equals the size needed for the file. You can use the *dd...dd* option to check for the largest available contiguous free area. By executing the *filstats* command with the *-s* option specified, you can check the condition of the contiguous areas in the

OpenTP1 file system. Using the results of this check, when you delete an OpenTP1 file that is no longer needed, the preceding and succeeding free areas can be linked, securing a larger contiguous free area.

- Digits following the decimal point are truncated from the values shown in kilobytes [KB] in the output format. Consequently, in the list displayed with the `-s` option, the value of *OFFSET* + *SIZE* may not match exactly the value of *OFFSET* for the next area.

jnladdpf (Allocate journal physical files)

Format

```
jnladdpf -j cpd -g file-group-name
        {-a physical-file-name | -b physical-file-name |
        -a physical-file-name -b physical-file-name}
```

Function

The `jnladdpf` command allocates a physical file to the file group specified in the checkpoint dump service definition in the online mode. The file group becomes ready for use when it is opened by the `jnlopnfg` command after input of the `jnladdpf` command. Executing the `jnladdpf` command for a file group to which physical files have already been allocated results in a command error. The `jnladdpf` command can be input only while both OpenTP1 and the journal service are active.

Options

- `-j cpd`
Allocates a physical file to a checkpoint dump file.
- `-g file-group-name~<1-8 character identifier>`
Specifies the name of the file group to which the physical file is to be allocated. The name must be the same file group name as specified in the checkpoint dump service definition.
- `-a physical-file-name~<1-63 character path name>`
Specifies the name of the physical file to be allocated for the file group, with the full path name.
For a duplicated checkpoint dump file configuration, a physical file is allocated to system A.
- `-b physical-file-name~<1-63 character path name>`
Specifies the name of the physical file to be allocated for the file group, with the full path name.
For a duplicated checkpoint dump file configuration, a physical file is allocated to system B.
Specifying this option with a non-duplicated checkpoint dump file configuration results in a command error.

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTPI is running in journal fileless mode.	Standard error output
KFCA02191-E	-a option specification is invalid.	Standard error output
KFCA02192-W	A physical file has been allocated to the specified file group.	Standard error output
KFCA02193-E	Specified physical file name is invalid	Standard error output
KFCA02194-I	A physical file was allocated.	Message log file
KFCA04128-I	Help message	Standard output, standard error output
KFCA26001-E	-a or -b option specification is invalid.	Standard error output

Notes

With a physical file allocated to either system A or system B in a duplicated checkpoint dump file configuration, the file group cannot be opened. Physical files must be allocated to both systems.

jnlardis (Forces to release connection to resource group)

Format

```
jnlardis -a archive-resource-group-name  
        -t journal-resource-group-name-@-node-identifier [-p]
```

Function

The `jnlardis` command forces to release a resource group for global archive journal service and a resource group for journal service.

Options

- `-a archive-resource-group-name`
Specifies the name of a resource group for global archive journal service.
- `-t journal-resource-group-name-@-node-identifier`
Specifies the name of the resource group for journal service whose connection to a resource group for global archive journal service is to be released. The name must be followed by a node identifier, with an at mark (@) placed between the two.
- `-p`
Releases a connection, assuming that the journal service has terminated as planned.
If this option is not specified, a connection is released as system assumes that the journal service has terminated normally.

Notes

While the specified resource group for journal service is in online mode, OpenTP1 may reconnect the resource group for global archive journal service with the resource group for journal service.

jnlrls (Display archive status)

Format

```
jnlrls [-z]
```

Function

The `jnlrls` command displays the relationship between resource groups for the global archive journal service and those for the journal service, and archive status.

Options

- `-z`

Deletes the header line and displays status. If this option is omitted, the header line is displayed.

Output format

Node	Type	Resource	Node	Type	Resource	Connec- tion	Run ID number	Generation	Block	<i>l</i> .
<i>aaaaaaaa</i>	<i>bbb</i>	<i>ccccccc</i>	<i>d</i>	<i>aaaaaaaa</i>	<i>bbb</i>	<i>ccccccc</i>	<i>ef--</i>	<i>ggggggg</i>	<i>hhhhhhh</i>	<i>iiiiiii</i>
<i>aaaaaaaa</i>	<i>bbb</i>	<i>ccccccc</i>	<i>d</i>	<i>aaaaaaaa</i>	<i>bbb</i>	<i>ccccccc</i>	<i>ef--</i>	<i>ggggggg</i>	<i>hhhhhhh</i>	<i>iiiiiii</i>
:	:	:	:	:	:	:	:	:	:	:

1. The header line is not displayed if the `-z` option is specified.

aaaaaaaa

Name of an archive-journal target node or archive-journal source node

bbb

Type of a journal service or global archive journal service

`sys`: System journal file

`jar`: Archive journal file

ccccccc

Resource group name (definition file name of the system journal service definition or archive journal service definition)

d

Archive destination identifier

>: Indicates that the archive journal is allocated on the right side of this mark.

<: Indicates that the archive journal is allocated on the left side of this mark.

e

Connection status (which is recognized by the service indicated on the right side of the archive destination identifier)

c: Connected or in service

-: Not connected (for the global archive journal service; connected once or more in the past) or initialized status

d: Waiting for output completion after normal disconnection or being disconnected

f: Forced or waiting for output completion after service stop

i: Being connected

n: Non-service

f

Status of processing for a transfer-delayed journal

w: Being buffered

r: Reading

s: To be stopped

-: Displayed for archiver

gggggggg

Archive journal run ID (run ID on the journal server side which is recognized by the service indicated on the right side of the archive destination identifier)

Archive identifier is >:

Run ID on the journal server side for which the global archive journal service has been archived (output)

Archive identifier is <:

Run ID on the journal server side for which the journal service is being archived (transferred)

hhhhhhh

Archive journal generation number (generation number on the journal server side which is recognized by the service indicated on the right side of the archive destination identifier)

Archive identifier is >:

Generation number on the journal server side for which the global archive journal service has been archived (output)

Archive identifier is <:

Generation number on the journal server side for which the journal service is being archived (transferred)

iiiiiii

Archive journal block number (block number on the journal server side which is recognized by the service indicated on the right side of the archive destination identifier)

Archive identifier is >:

Block number on the journal server side for which the global archive journal service has been archived (output)

Archive identifier is <:

Block number on the journal server side for which the journal service is being archived (transferred)

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01280-E	<code>jnlarls</code> command processing failed.	Standard error output
KFCA01281-E	<code>jnlarls</code> command input format is invalid.	Standard error output
KFCA04127-I	Help message.	Standard output, standard error output

Notes

If the `jnlarls` command is executed, a correct status may not be displayed in the following case:

- The journal data is being transferred from a system journal file to an archive journal file whose status has been changed.

jnlaturl (Control the automatic unloading facility)

Format

```
jnlaturl -j sys [{-i|-b|-t[-w]}]
```

Function

The `jnlaturl` command controls the automatic unloading facility. You can use the command to do the following:

- Display the operating status of the automatic unloading facility.
- Restart the inactive automatic unloading facility while OpenTP1 is running.
- Cancel the automatic unloading facility that is being executed.

Options

- `-j sys`

Indicates that the automatic unloading facility is executed for system journal files. This option is mandatory.

- `-i`

Specify this option to display the operating status of the automatic unloading facility.

- `-b`

Specify this option to restart the automatic unloading facility while OpenTP1 is running. Immediately after the restart of the automatic unloading facility, the command uses the directory that was used last in the previous operation as the directory for storing unload journal files.

- `-t`

Specify this option to cancel the automatic unloading facility. The command forcibly ends the unload being executed.

- `-w`

When you have specified the `-t` option to cancel the automatic unloading facility, specify this option to delay the end of the command until unloading of the generation currently being unloaded is completed. The `-w` option is valid only when the `-t` option is specified. If you specify the `-t` option and omit the `-w` option, the automatic unloading facility is canceled when the `jnlaturl` command is executed and the unloading being executed is forcibly terminated.

When you specify the `-w` option, the maximum time for waiting for a response for the completion of unloading is about 9 hours (32,768 seconds). If the unloading is not

completed within this time, the automatic unloading facility is canceled at that point and the unloading being executed is forcibly terminated.

Output format

- When the `-i` option is specified

Status	Now unloading the file group	unload-destination-directory
<i>aa...aa</i>	<i>bb...bb</i>	<i>cc...cc</i>
current generation information		
file-group	generation-no	run-ID unload-file-name
<i>dd...dd</i>	<i>ee...ee</i>	<i>ff...ff</i> <i>gg...gg</i>

aa...aa

Operating status of the automatic unloading facility

ACTIVE: Active

STOP: Inactive

STOPPING: The command is stopped, or OpenTP1 is waiting for a response indicating the end of automatic unloading.

-: The automatic unloading facility is not available.

bb...bb

File group name of the system journal file being unloaded. `*****` appears if no system journal file is being unloaded.

cc...cc

Name of the unload journal file storage directory being used

dd...dd

File group name of the system journal file of the current generation

ee...ee

Generation number of the system journal file of the current generation

ff...ff

Run ID of the system journal file of the current generation

gg...gg

Name of the file that is created when the current generation is automatically unloaded

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01160-I	The journal utility service is started.	Message log file
KFCA01161-I	The journal utility service is terminated.	Message log file
KFCA01162-E	The journal utility service cannot be started.	Standard error output
KFCA01163-W	Processing continues even though an error occurred during termination processing of the journal utility service.	Message log file
KFCA01168-W	Unloading is canceled.	Message log file
KFCA01169-W	A definition clause is incorrect. OpenTP1 starts in a mode that does not use the automatic unloading facility for journal files.	Message log file
KFCA01170-I	Automatic unloading of a journal file started.	Standard error output
KFCA01171-I	A journal file will be automatically unloaded.	Message log file
KFCA01172-I	Automatic unloading of a journal file is complete.	Message log file
KFCA01173-W	The automatic unloading facility for journal files is stopped.	Message log file
KFCA01174-W	Automatic unload processing is canceled.	Message log file
KFCA01175-I	OpenTP1 is waiting for the end of automatic unload processing.	Message log file
KFCA01176-I	The specified directory is set as the automatic unloading destination directory.	Message log file
KFCA01177-E	A file error occurred during automatic unloading.	Message log file
KFCA01178-E	An error occurred during automatic unloading.	Message log file
KFCA01179-W	The automatic unloading destination directory cannot be changed.	Message log file
KFCA01299-I	Help message	Standard output

Note

- If the automatic unloading facility is not being used in your system execution environment, you cannot specify the `-b` option and the `-t` option.
- If the `jnlcatunl` command is executed with the `-b` option to restart the automatic unloading facility, the facility may not restart properly despite the `jnlcatunl` command terminating normally, due to a problem such as an error in the

automatic unloading process. Specify the `-i` option to check whether the automatic unloading facility started properly.

- If you execute the `jnlataunl` command with the `-t` or `-w` option specified, and OpenTP1 is forcibly terminated or terminates abnormally while still waiting for the automatic unloading facility to shut down, the `jnlataunl` command does not terminate normally, and instead aborts with the error code `olkcrt2`. However, this has no effect on online operation, and you do not need to restart OpenTP1.

jnlchgfg (Change status of journal files)

Format

```
jnlchgfg -j sys|jar [-r resource-group-name]
                -g file-group-name [-d service-definition-name] [-s]
```

Function

The `jnlchgfg` command places forcibly a specified file group on unloaded status. The status of the current file group cannot be changed.

The status of the archive journal file can forcibly be changed to archived status from archiving wait status.

Options

- `-j sys|jar`

Specifies the file group whose status is to be changed.

`sys`: System journal file

`jar`: Archive journal file

- `-r resource-group-name~<1-8 character identifier>`

Specifies the name of the target resource group. If `sys` is specified in the `-j` option, the name must be the same file name as defined in the system journal service definition. If `jar` is specified in the `-j` option, the name must be the definition file name specified in the archive journal service definition whose status is to be changed. When this option is omitted, the resource group name specified in the journal service definition of `$DCCONFPATH/jnl` is assumed if `sys` is specified in the `-j` option. If `jar` is specified in the `-j` option, the system assumes the definition file name specified in the archive journal service definition that has been defined at the beginning of the definition command `jnldefsv -a` in the global archive journal service definition of `$DCCONFPATH/jar`.

- `-g file-group-name~<identifier, 1-8 characters>`

Specifies the file group name of the system journal files whose status is to be changed.

- `-d service-definition-name~<path name> or <file name>`

Specifies the name of the system journal service definition file that defines the file group whose status is to be changed, in one of the following formats:

- Complete path name beginning with / (root)
- Definition file name only

If only a definition file name is specified, the directory containing the definition file must be set in `$DCCONFPATH`.

If this option is omitted, the specification of the `-r` option predominates.

■ `-s`

Changes the status of the archive journal file from archiving wait status to archived status.

When `jnl_arc_check_level` is set to 2 in the system journal service definition, the file group cannot be reused unless unloading and archiving the archive journal file. If this option is specified when archiving cannot be performed due to an error on the archive journal service, the archive journal file can forcibly be placed in archived status. Therefore, the system operation can be continued.

However, if the file group forcibly placed in archived status becomes a swappable file group, the file group cannot be archived even when the archive journal service returns normally.

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output

Notes

- Do not execute the `jnlchgfg` command during startup of the journal service. Startup spans the time from KFCA01100-I to KFCA01102-I in the case of a normal start, or from KFCA01101-I to KFCA01102-I in the case of a restart. If the `jnlchgfg` command is executed while this processing is underway, the status of the system journal file may not be recognized by OpenTP1. If this occurs, use the `jnlclsfg` command to close the corresponding file group after the journal service has completed startup, or restart processing, and then open the file group again using the `jnlpnfg` command. The status of the system journal file will then be recognized by OpenTP1.
- The `jnlchgfg` command references the definition file. For this reason, statuses cannot be changed properly if the definition file in effect at execution of the `jnlchgfg` command does not match the definition file that was in effect during the online session when the journal was collected.
- Specify the `-d` or `-r` option if the definition file in effect at execution of the `jnlchgfg` command does not match the definition file in effect at journal acquisition. Specify the `-d` option if the backup destination directory is not the directory specified by `$DCCONFPATH`, and specify the `-r` option if the file has been renamed. However, a backup of the system journal service definition or

jnlchgfg (Change status of journal files)

archive journal service definition that was in effect when the journal was acquired must be available.

jnlclsfg (Close journal files)

Format

```
jnlclsfg -j sys|cpd|jar [-s server-name]
          [-r resource-group-name] -g file-group-name
          [-e element-file-name] [-a] [-b]
```

Function

The `jnlclsfg` command closes the open physical files comprising a specified file group, so the files will not be used in the online mode. Physical files that are constituents of the following file groups cannot be closed:

- System journal file or archive journal file on current or nonwritable status
- Checkpoint dump file involved in a write operation or on nonwritable status

Additionally, the `jnlclsfg` command can be entered only when OpenTP1 and the journal service are both operating.

Options

- `-j sys|cpd|jar`

Specifies the type of journal files to be closed:

`sys`: System journal files

`cpd`: Checkpoint dump files

`jar`: Archive journal file

- `-s server-name~<identifier, 1-8 characters>`

Specifies the name of the server subject to this processing when checkpoint dump files are to be closed. Specifying the server name limits the search range of file group names.

When this option is omitted, the first (among all servers) checkpoint dump files with the matching file group name are closed.

If the `-j` option specifies `sys` or `jar`, the specification of the `-s` option is discarded.

- `-r resource-group-name~<1-8 character identifier>`

If `jar` is specified in the `-j` option, the name of the resource group for the file to be closed must be specified. It is the definition file name of the archive journal service definition specified in the definition command `jnldefsv -a` command of the global archive journal service definition.

If this option is omitted, the name of an online-mode archive journal file is assumed which is defined at the beginning of the definition command `jnlclfsv -a` of the archive journal service definition. If other than `jar` is specified in the `-j` option, the specification of the `-r` option is discarded.

- `-g file-group-name~<identifier, 1-8 characters>`

Specifies the file group name of the journal files to be closed.

- `-e element-file-name`

Specifies the name of an element file in the journal file group to be closed.

- `-a`

Specifies that physical files in system A are to be closed.

- `-b`

Specifies that physical files in system B are to be closed.

If this option is specified when dual files are not being used, a command error occurs.

When neither the `-a` nor `-b` option is specified, or when `cpd` is specified in the `-j` option, the command performs as follows:

- If dual files are not being used, physical files in system A are closed in the same manner as when the `-a` option is specified.
- If dual files are being used, physical files in both systems A and B are closed in the same manner as when the `-a` and `-b` options are both specified.
- When one-system operation is unavailable in dual system configuration, a command error occurs if adding either the `-a` or `-b` option to `-j cpd`.

If all physical files subject to close processing are already closed, a warning message (KFCA01283-W or KFCA02168-W) is issued, although no command error occurs (command termination code: 0).

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01216-I	File group has been closed successfully.	Message log file
KFCA01280-E	<code>jnlclfsfg</code> command processing failed.	Standard error output
KFCA01281-E	<code>jnlclfsfg</code> command input format is invalid.	Standard error output
KFCA01283-W	Insignificant error occurred during <code>jnlclfsfg</code> command execution.	Standard error output

Message ID	Description	Output destination
KFCA01285-E	Swap processing is underway.	Standard error output
KFCA01295-I	Help message	Standard error output
KFCA02128-E	Specified server name is invalid.	Standard error output
KFCA02129-E	Specified file group name is invalid.	Standard error output
KFCA02130-E	-s option specification is invalid.	Standard error output
KFCA02131-E	-g option specification is invalid.	Standard error output
KFCA02132-E	jnlclsfg command processing was cancelled due to an error.	Standard error output
KFCA02168-W	Specified file group is already closed.	Standard error output
KFCA02170-E	File close processing failed.	Standard error output
KFCA02184-E	Close processing cannot be performed, because there are not enough file groups.	Standard error output

Notes

The `jnlclsfg` command may result in an error if it is entered during swap processing or the processing of another operation command.

jnlcolc (Integrate file recovery journals)

Format

```
jnlcolc -k recovery-type [-f] [-l] [-n] [-m]
        [-i online-backup-information-name] [-c key]
        [file name] [Δfile-name] . . .
```

Function

The `jnlcolc` command extracts journal records required for DAM file, TAM file or ISAM file recovery from specified unload journal files and then outputs them to the standard output. The integrated result can also be redirected and output to a file.

Options

- `-k recovery-type`

Specifies the type of file to be subject to recovery:

d: DAM file

t: TAM file

i: ISAM file

- `-f`

Specifies that the journals required for file recovery are to be extracted from the first unload journal files. The command checks whether or not the journal file generation numbers are n ascending order beginning with 1. If not, processing terminates.

When this option is omitted, the command assumes that information is to be inherited from the previous journal record extraction processing and checks whether or not the inheritance file generation numbers are arranged in ascending order beginning with the last generation number + 1. If not, processing terminates.

- `-l`

Specifies that the journals required for file recovery are to be extracted from the last unload journal files. The command then deletes the inheritance file.

If file recovery journals are integrated by specifying this option, the `-f` option must be specified when another online journal is integrated.

When this option is omitted, inheritance files are not deleted.

- `-n`

Specifies that all FJ records are to be output, even if there is update information for the same logical file or the same relative block number in the transaction (information

about FJ before and after update processing is not checked, and duplicate information is not deleted).

When this option is omitted and there is update information for the same logical file or the same relative block number in the transaction, only the final update information is output.

This option is applicable only when the recovery type indicates a DAM file.

■ **-m**

Integrates journal records on the file to be recovered. If this option is specified, it is assumed that the **-n** option has been specified. If the option is omitted, a buffer is allocated on memory for integration of journal records.

■ **-i *online-backup-information-name***

Collects integration startup information of the file recovery journal from the specified file, and skips the records preceding the startup location. This option is applicable only when **d** or **t** (DAM or TAM file as the recovery type) is specified in the **-k** option. If the **-i** option is omitted, file recovery journals are integrated from the beginning of the file.

■ **-c *key*~((001-999)) <<001>>**

Specifies the last three characters of the inheritance file name. OpenTP1 creates an inheritance file under the name `jnlcolc***`, replacing `***` with the value specified with this option.

Command arguments

■ ***file-name*~<path name>**

Specifies the name of the file containing journal records needed for file recovery. When the multinode facility is used, the name of the file whose journal records have been sorted and merged by the `jnlsort` command must be specified. If the multinode function is not used, the name of the unload journal file must be specified. When more than one file is specified, a space must be placed between files. You can specify a maximum of 256 files.

When this command argument is omitted, the standard input is assumed.

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output

Message ID	Description	Output destination
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02606-E	The inheritance information file is invalid.	Standard error output
KFCA02607-E	The relationship between the inheritance information file and the unload journal file is invalid.	Standard error output
KFCA02621-W	A record, required for file recovery, that was to be output does not exist.	Standard error output
KFCA02650-I	Help message	Standard output, standard error output
KFCA02651-E	A parameter in the <code>jnlcolc</code> command is invalid or its value exceeds the limit.	Standard error output
KFCA02652-E	The combination of options in the <code>jnlcolc</code> command is invalid.	Standard error output
KFCA02653-E	An option in the <code>jnlcolc</code> command is specified incorrectly.	Standard error output
KFCA02654-E	A file name in the <code>jnlcolc</code> command is specified incorrectly.	Standard error output
KFCA02655-E	The information before and after update specified in the <code>jnlcolc</code> command is inconsistent.	Standard error output
KFCA02657-E	The <code>-k</code> option is not specified in the <code>jnlcolc</code> command.	Standard error output
KFCA02658-E	A record, required for file recovery, that is specified in the <code>jnlcolc</code> command is invalid.	Standard error output

Notes

- If the inheritance information file specified in the `jnlcolc` command already exists, the existing file is retained under the name `jnlcolc***.bak`.
- If multiple journal files are to be integrated, they must all have been output during the same online operation. If even one of them is different, processing terminates.
- The following types of journal records are output:

For DAM file recovery:

FJ: File (DAM) update information

HJ: Transaction temporary synchronous point information

PJ: Transaction commit processing startup information

DJ: Heuristic decision information

BJ: Transaction rollback information

For TAM file recovery:

CJ: Update information for a table subject to recovery

HJ: Transaction temporary synchronous point information

PJ: Transaction commit processing startup information

DJ: Heuristic decision information

BJ: Transaction rollback information

For ISAM file recovery:

FJ: File (ISAM) update information

HJ: Transaction temporary synchronous point information

PJ: Transaction commit processing startup information

DJ: Heuristic decision information

BJ: Transaction rollback information

- If journal records are to be integrated from multiple unload journal files by executing the `jnlcolc` command more than once, the same values must be specified for the `-k`, `-n`, and `-c` options in all the `jnlcolc` commands.
- If no unload journal file name or global archive journal file name is specified, the standard input is assumed, and the input file must be specified by piping, redirecting, etc.
- If a memory allocation error occurs, the `jnlcolc` command with its `-m` option specified must be entered again. No fast processing speed can be expected because journal records are integrated on a file. When the file name is omitted, the `-m` option must not be specified.
- The `jnlcolc` command is used with a file (DAM, TAM or ISAM) recovery command and the `jnlstort` command to recover files. For the ISAM file recovery command, see the manual *Indexed Sequential Access Method ISAM*.
- If you integrate multiple journal files into one file, specify the file names in the chronological order from the earliest acquired file. If file names are not chronologically specified, the system outputs the error message KFCA02601-E and terminates processing due to an invalid journal file.
- Do not specify an unload journal file acquired by specifying the `-t` option in the `jnlunlfg` command. If you do so, message KFCA02601-E is output.
- The number of characters you can enter in this command is determined by your OS. If the number of entered characters exceeds the upper limit, an error occurs.

- An error occurs when the number of opened files exceeds the maximum number of files that can be opened for one process, including when the specified number of files is less than 256.

Examples

In the examples below, the generation numbers are 1,2,...

Unload journal files: sysjnl001, sysjnl002,...

- (1) *Integrate journal records for DAM FRC from an unload journal file that has not been split*

```
jnlcolc -k d -f -l /OpenTP1/jnl/sysjnl001
```

- (2) *Integrate journal records for TAM FRC in one step from an unload journal file that has been split into three files*

```
jnlcolc -k t -f -l /OpenTP1/jnl/sysjnl001  
/OpenTP1/jnl/sysjnl002  
/OpenTP1/jnl/sysjnl003
```

- (3) *Integrate journal records for DAM FRC in three steps from an unload journal file that has been split into four files*

```
Step 1:  
jnlcolc -k d -f /OpenTP1/jnl/sysjnl001 /OpenTP1/jnl/sysjnl002  
Step 2:  
jnlcolc -k d /OpenTP1/jnl/sysjnl003  
Step 3:  
jnlcolc -k d -l /OpenTP1/jnl/sysjnl004
```

jnlcopy (Copy unload journal files)

Format

```
jnlcopy [-t begin [, end] [-j record-type [record-type...]]
        [-o journal-collection-mode [journal collection mode] ...]
        [{"unload-journal-file-name"} [{"unload-journal-file-name"} ...]
```

Function

The `jnlcopy` command outputs information from specified unload journal files to the standard output.

The `jnlcopy` command cannot be used under the multinode facility.

Options

- `-t` [*begin*] [, *end*]

Specifies the copy range in terms of the journal block output begin and end times.

In "begin", specify the copy start time. In "end", specify the copy end time.

For begin, specify a time between 00:00:00 on January 1, 1970 and the current year, month, day, and time.

The start time and end time cannot both be omitted. When the start time is omitted, the copy range is from the beginning of the unload journal file to the specified end time. When the end time is omitted, the copy range is from the specified start time to the end of the unload journal file.

The begin and end times are expressed in the format `hhmmss[MMDD[YYYY]]`:

hh

Hour (00 ≤ hh ≤ 23); cannot be omitted

mm

Minute (00 ≤ mm ≤ 59); cannot be omitted

ss

Second (00 ≤ ss ≤ 59): cannot be omitted

MM

Month (01 ≤ MM ≤ 12); can be omitted[#]

DD

Day (01 ≤ DD ≤ 31); can be omitted[#]

YYYY

Year (4-digit year from 1970); can be omitted[#]

[#]: If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If the `-t` option is omitted, all information in the specific file is edited and output.

■ `-j record-type`

Specifies the type of journal records to be copied.

Multiple record types can be specified.

Following are the specification values for the record types:

Record type	Specification	Description	Remarks
PJ	p	Transaction commit processing startup information	Synchronous point journal
HJ	h	Transaction temporary synchronous point information	
BJ	b	Transaction rollback information	
TJ	t	Transaction synchronous point processing termination information	
DJ	d	Heuristic decision information	
FJ	f	DAM file update information	Recovery journal
CJ	c	Update information for a table subject to recovery	
XJ	x	System service-specific recovery information	
SJ	s	System statistics	
AJ	a	Send completion information	Statistical journal
IJ	l	Messages cataloged into input queue	
OJ	o	Messages cataloged into output queue	
MJ	m	Message journal	
GJ	g	Receive information	
UJ	u	User-specific information	User journal

If the `-t` option is specified and the `-j` option is omitted, all journal records in the specified range are copied. If the `-t` option is not specified but the `-j` option is specified, all unload journal files whose journal record types are the same as the specified record types will be copied.

If the `-t` and `-j` options are all omitted, all information is output from the unload journal files.

■ `-o journal-collection-mode`

Specifies a journal collection mode. Multiple modes can be specified.

`c`

Copies journals in transaction test mode whose attribute is commit, or journals in MCF nontransaction test mode.

`r`

Copies journals in transaction test mode whose attribute is rollback.

`s`

Copies journals in other than online test mode.

If this option is omitted, journals in all journal collection modes are copied.

Command arguments

■ `unload-journal-file-name~<path name>`

Specifies the name of an unload journal file.

When more than one unload journal file is specified, a space must be placed between files. You can specify a maximum of 256 files.

When this command argument is omitted, the standard input is assumed.

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02620-W	The target record does not exist.	Standard error output

Message ID	Description	Output destination
KFCA02630-I	Help message	Standard error output
KFCA02631-E	A parameter in the <code>jnlcopy</code> command is invalid or its value exceeds the limit.	Standard error output
KFCA02632-E	The combination of options in the <code>jnlcopy</code> command is invalid.	Standard error output
KFCA02633-E	An option in the <code>jnlcopy</code> command is specified incorrectly.	Standard error output
KFCA02634-E	A file name in the <code>jnlcopy</code> command is specified incorrectly.	Standard error output

Notes

- If there are no journal records that match the start time or end time, the journal records between the start time and the end time are to be copied.
- If a journal record subject to copy processing does not exist, an error message is output and processing terminates.
- If the copy range is specified in terms of time and the start time value is greater than the end time value, the month and day or the month, day, and year must be specified so that the end time is after the start time. If the requisite information is omitted, an error occurs. If the date is omitted, the date the `jnlcopy` command is entered is assumed; therefore, if the copy range does not fit within the current day, either the month and day or the month, day, and year must be specified.
- If multiple unload journal files are specified, they must all have been output during the same online operation, and their generation numbers must be arranged consecutively.
- When no unload journal file name is specified, the standard input is assumed. Therefore, the input file must be specified by piping, redirecting, etc.
- The `jnlcopy` command's output information becomes input information for other operation commands. For this reason, the `jnlcopy` command also outputs various management information concerning files, blocks, and records.
- The number of characters you can enter in this command is determined by your OS. If the number of entered characters exceeds the upper limit, an error occurs.
- An error occurs when the number of opened files exceeds the maximum number of files that can be opened for one process, including when the specified number of files is less than 256.

Examples

In this example, the command outputs information from the beginning of the unload journal files up to 17:30:00.999999 on May 29, 1993:

Unload journal files: sysjnl001, sysjnl002

```
jnlcopy -t ,17300005291993 /OpenTP1/jnl/sysjnl001  
/OpenTP1/jnl/sysjnl002
```

jnlde1pf (Delete journal physical files)

Format

```
jnlde1pf -j cpd -g file-group-name [-a] [-b]
```

Function

The `jnlde1pf` command deletes a physical file from the specified file group in the online mode. Only physical files that were allocated with the `jnladdpf` command in the online mode and that have not been opened or are placed in the shutdown status can be deleted by the `jnlde1pf` command. An attempt to delete any other file results in a command error.

The `jnlde1pf` command can be entered only while both OpenTP1 and journal service are active.

Options

- `-j cpd`
Deletes a physical file from a checkpoint dump file group.
- `-g file-group-name~<identifier, 1-8 characters>`
Specifies the name of the file group from which a physical file is to be deleted.
- `-a`
Deletes a physical file from system A.
- `-b`
Deletes a physical file from system B.

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA26001-E	-a or -b specification is invalid.	Standard error output
KFCA26002-E	The physical file has not been allocated to specified file group.	Standard error output
KFCA26003-I	Physical file has been disconnected from file group.	Message log file
KFCA26010-E	Physical file cannot be disconnected from specified file group.	Standard error output

jnledit (Edit and output unload journal files or global archive unload journal files)

Format

```

jnledit [-e edit-type] [-t [begin] [, end]]
        [-j record-type [record-type..]] [-s server-name]
        [-v service-name] [-u transaction-global-identifier
        [, transaction-branch-identifier]]
        [-o journal-collection-mode [journal collection mode] ...]
        [-m logical-terminal-name [, logical-terminal-name ...]
        [-w characters] [-l lines] [-c]
        [file-name [Δfile-name] ...]

```

Function

The `jnledit` command edits the information in the specified unload journal file or global archive unload journal file, and outputs it to the standard output.

Options

- `-e edit-type`~<<f>>

Specifies the editing type:

r: Edit in units of records

b: Edit in units of blocks

f: Output a file listing

- `-t [begin] [, end]`

Specifies the editing range in terms of the journal block output begin and end times. In "begin", specify the edit start time. In "end", specify the edit end time. When r is specified for the `-e` option, journal record output times must be specified; when b is specified, journal block output times must be specified.

For *begin*, specify a time between 00:00:00 on January 1, 1970 and the current year, month, day, and time.

The start time and end time cannot both be omitted. When the start time is omitted, the editing range is from the beginning of the file to the specified end time. When the end time is omitted, the editing range is from the specified start time to the end of the file.

The begin and end times are expressed in the format hhmmss[MMDD[YYYY]]:

hh

Hour ($00 \leq hh \leq 23$); cannot be omitted

mm

Minute ($00 \leq mm \leq 59$); cannot be omitted

ss

Second ($00 \leq ss \leq 59$); cannot be omitted

MM

Month ($01 \leq MM \leq 12$); can be omitted[#]

DD

Day ($01 \leq DD \leq 31$); can be omitted[#]

YYYY

Year (4-digit year from 1970); can be omitted[#]

#: If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If the `-t` option is omitted, all information in the specified file is edited and output.

■ `-j record-type`

Specifies the type of journal records to be edited.

Following are the specification values for the record types:

Record type	Specification	Description	Remarks
PJ	p	Transaction commit processing startup information	Synchronous point journal
HJ	h	Transaction temporary synchronous point information	
BJ	b	Transaction rollback information	
TJ	t	Transaction synchronous point processing termination information	
DJ	d	Heuristic decision information	
FJ	f	DAM file update information	Recovery journal
CJ	c	Update information for a table subject to recovery	
XJ	x	System service-specific recovery information	

Record type	Specification	Description	Remarks
SJ	s	System statistics	Statistical journal
AJ	a	Send completion information	
IJ	ī	Messages cataloged into the input queue	
OJ	o	Messages cataloged into the output queue	
MJ	m	Message journal	
GJ	g	Receive information	
UJ	u	User-specific information	User journal

When this option is omitted, all journal record types become subject to editing.

- `-s server-name~<1-8 alphanumeric characters>`

Specifies the name of a server whose contents are to be edited.

When this option is omitted, all servers are subject to editing.

- `-v service-name~<1-31 alphanumeric characters>`

Specifies the name of a service whose contents are to be edited.

When this option is omitted, all services become subject to editing.

- `-u transaction-global-identifier [, transaction-branch-identifier] ~<33 alphanumeric characters, special character, and hexadecimal digits>`

Specifies the transaction identifier of a transaction whose contents are to be edited. The transaction identifier must be specified in the format "transaction-global-identifier, transaction-branch-identifier".

When specification of a transaction identifier is omitted, all transaction identifiers become subject to editing.

- `-o journal-collection-mode`

Specifies a journal collection mode. Multiple modes can be specified.

c

Edits and outputs journals in transaction test mode whose attribute is commit, or journals in MCF nontransaction test mode.

r

Edits and outputs journals in transaction test mode whose attribute is rollback.

s

Edits and outputs journals in other than online test mode.

If this option is omitted, journals in all journal collection modes are edited and output.

- `-m logical-terminal-name`~<1-16 alphanumeric characters>

Specifies the name of the logical terminal for which the information is to be edited and output. This option takes effect when the record type is a, g, i, m, or o.

When this option is omitted, edited data is output for all the logical terminals.

- `-w characters`~<unsigned integer> ((80-132)) <<80>>

Specifies the number of characters per line.

- `-l lines`~<unsigned integer> ((12-256)) <<24>>

Specifies the number of output lines per page.

- `-c`

Specifies that editing is to be in both hexadecimal and character formats.

When this option is omitted, information is edited in hexadecimal format only.

Command arguments

- `unload-journal-file-name`~<path name>

Specifies the name of an unload journal file or global archive unload journal file.

Multiple file names can be specified only when `f` is specified in the `-e` option. You can specify a maximum of 256 files.

When this command argument is omitted, the standard input is assumed.

Following are the valid and invalid combinations of options:

Option	-e			-t	-j	-s	-v	-u	-m	-w	-l	-c	-o	File		
	b	r	f											Single	Multiple	
-e	b	--	N	N	Y	N	N	N	N	N	Y	Y	Y	N	Y	N
	r	N	--	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
	f	N	N	--	N	N	N	N	N	N	N	N	N	N	Y	Y
-t	Y	Y	N	--	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
-j	N	Y	N	Y	--	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
-s	N	Y	N	Y	Y	--	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
-v	N	Y	N	Y	Y	Y	--	Y	Y	Y	Y	Y	Y	Y	Y	N
-u	N	Y	N	Y	Y	Y	Y	--	Y	Y	Y	Y	Y	Y	Y	N
-m	N	Y	N	Y	Y	Y	Y	Y	--	Y	Y	Y	Y	Y	Y	N
-w	Y	Y	N	Y	Y	Y	Y	Y	Y	--	Y	Y	Y	Y	Y	N
-l	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	Y	Y	Y	N	
-c	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	--	Y	Y	N	
-o	N	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	Y	N	
File	Single	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	--
	Multiple	N	N	Y	N	N	N	N	N	N	N	N	N	N	--	--

Output format

■ (1)-c option specified

```

jnledit vv-rr ***** Unload journal file editing ***** Page yyyy

File name [aa...aa]
File type [bbb]           File creation date and time [cc...cc]

Block number [ddddddd]
Record number [eeeeeee] Record acquisition date and time [ff...ff]
Journal record kind [gg] Journal record type [h]
Journal acquisition mode [ii...ii]
Transaction global ID [jj...jj]
Transaction branch ID [kk...kk]
MCF record creation date and time [mm...mm]
Transaction branch startup date and time [nn...nn]

*** Journal record ***
uuu uuuuu uuuuu uuuuu uuuuu uuu uuu uuu uuu
uuu uuuuu uuuuu uuuuu uuuuu uuu uuu uuu uuu
uuu uuuuu uuuuu uuuuu uuuuu uuu uuu uuu uuu
: : : : : : : : :
    
```

1. This information is displayed as many times as there are specified records.

aa...aa

Unload journal file name

bbb

File type:

sys: System journal file

jar: Global archive unload journal file

cc...cc

Unload journal file creation date and time, in the format YY/MM/DD
hh:mm:ss

dddddddd

Block number (8 hexadecimal digits)

eeeeeeee

Record number (8 hexadecimal digits)

ff..ff

Date and time of editing in units of records

The date and time is output in the format of year-month-day hour: minutes: seconds.microseconds (microseconds are displayed as a decimal number from 0 through 999999). However, if you edit a journal before TP1/Server Base 06-01, 0 is output as the microsecond value.

gg

Record type (2 alphabetic characters)

h

Record type: (1 alphanumeric character)

ii...ii

Journal collection mode (1-13 alphabetic characters)

jj..jj

Transaction global identifier (8 alphanumeric and special characters plus 8 hexadecimal digits)

kk...kk

Transaction branch identifier (8 alphanumeric and special characters plus 8 hexadecimal digits)

ll...ll

Journal record (32-96 hexadecimal digits and 16-32 characters)

If a journal record contains only management information, only the title is output.

mm...mm

Date and time that an MCF record is created

The date and time when the message control facility created a journal record are output.

The date and time are output in the *year-month-day hour: minutes: seconds.microseconds* format (*microseconds* is a decimal number from 0 to 999999).

nn...nn

Date and time that a transaction branch is started

The date and time are output in the *year-month-day hour: minutes: seconds.microseconds* format (*microseconds* is a decimal number from 0 to 999999).

vv-rr

Version number-revision number

yyyy

Page

■ (2) *jnledit -ef specified*

```

jnledit vv-rr ***** Unload Journal file information ***** Page yyyy
-----
          File name      [aa...aa]
-----
File creation date and time [bb...bb]
File type                   [ccc]
Node identifier             [dd...dd]
Journal server run ID      [0xeeeeeeee]
Usage startup date and time [ff...ff]
First generation number#1  [0xgggggggg]
Last generation number#1   [0xhhhhhhhh]
First block number         [0xiiiiiii]
First record number        [0xjjjjjjj]
Partial start date and time [oo...oo]
Partial end date and time   [pp...pp] } 3

*** Connect information < Node identifier [dd...dd] >

File type                   [ccc]
Resource name               [kkkkkkkkk]
--- Last information of former generation#2
  Journal server runID      [0xeeeeeeee]
  File generation Number    [0xlllllll]
  Block Number              [0xmmmmmmmmm]
  Acquisition time of server side [nn...nn]
--- Last information of present generation#2
  Journal server runID      [0xeeeeeeee]
  File generation Number    [0xlllllll]
  Block Number              [0xmmmmmmmmm]
  Acquisition time of server side [nn...nn]
    
```

#1: When the file type is jar, 0x***** is displayed for the first block number and the first record number.

#2: If each generation end information is invalid or has not been set up, * . . * is displayed in each of its entry.

1. As many files as specified are displayed.
2. The information in this section is displayed when the specified file is the global archive unload journal file.
3. The information in this section is displayed when the specified file is acquired by specifying the -t option in the jnlunlfg command.

aa...aa

File name

bb...bb

File creation date and time

ccc

Journal file type:

sys: System journal file

jar: Global archive unload journal file

dd...dd

Node identifier

eeeeeeee

Journal server run ID (8 hexadecimal digits, prefixed by 0x)

ff...ff

Usage startup date and time

gggggggg

Begin generation number (8 hexadecimal digits, prefixed by 0x)

hhhhhhhh

End generation number (8 hexadecimal digits, prefixed by 0x)

iiiiiii

Begin block number (8 hexadecimal digits, prefixed by 0x)

jjjjjjj

Begin record number (8 hexadecimal digits, prefixed by 0x)

kkkkkkkk

Resource name (1-8 alphanumeric characters)

lllllll

file generation number (8 hexadecimal digits, prefixed by 0x)

mmmmmmm

Block number (8 hexadecimal digits, prefixed by 0x)

nn...nn

server-side journal data collection date and time

oo...oo

Start time and date that are specified in the *-t* option in the *jnlunlfg* command (*...* is displayed when the *-t* option is omitted)

pp...pp

End time and date that are specified in the `-t` option in the `jnlunlfg` command (*...* is displayed when the `-t` option is omitted)

vv-rr

Version number-revision number

yyy

Page number

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02640-I	Help message	Standard output, standard error output
KFCA02641-E	A parameter in the <code>jnledit</code> command is invalid or its value exceeds the limit.	Standard error output
KFCA02642-E	The combination of options in the <code>jnledit</code> command is invalid.	Standard error output
KFCA02643-E	An option in the <code>jnledit</code> command is specified incorrectly.	Standard error output
KFCA02644-E	A file name in the <code>jnledit</code> command is specified incorrectly.	Standard error output

Notes

- If an editing range is specified but no journal record corresponds to the specified begin or end time, the journal records between the begin and end positions are to be edited.
- If an editing range is specified and no journal record corresponds to the specified begin or end time, the journal records between the begin and end positions are to be edited.
- If the editing range is specified in terms of time and the start time value is greater than the end time value, the month and day or the month, day, and year must be

specified so that the end time is after the start time. If the requisite information is omitted, an error occurs. If the date is omitted, the date the `jnledit` command is entered is assumed; therefore, if the editing range does not fit within the current day, either the month and day or the month, day, and year must be specified.

- When no file name is specified, the standard input is assumed. Therefore, the input file must be specified by piping, redirecting, etc.
- Editing is performed so that the total length fits within the length specified with the `-w` option by setting hexadecimal digits to a multiple of (8 digits x 4 blocks) and characters to a multiple of (4 characters x 4 blocks).
- When editing a user journal (UJ) in units of records or blocks, the user-specified value plus 8 bytes is adjusted to fit in the 4-byte boundary, and then edited and output. If the user-specified value does not fit in the 4-byte boundary, null data follows the output user-specified data.

The data actually output by the user is 8 bytes shorter than the data length specified in the first 4 bytes of the editing data, and begins with the 8th byte of the editing data.

- The number of characters you can enter in this command is determined by your OS. If the number of entered characters exceeds the upper limit, an error occurs.
- An error occurs when the number of opened files exceeds the maximum number of files that can be opened for one process, including when the specified number of files is less than 256.

Examples

In this example, the command outputs information from the beginning of an unload journal file up to 17:30:00.999999 on May 29, 1993 in hexadecimal and character formats in units of records:

Unload journal file: `sysjnl001`

```
jnledit -e r -t ,17300005291993 -c /OpenTP1/jnl/sysjnl001
```

jnlinit (Initialize journal files)

Format

```
jnlinit -j jnl|cpd|srf -f physical-file-name
        -n OpenTP1-records-count
```

Function

The `jnlinit` command creates a journal file under the OpenTP1 file system and then initializes it so that the file can be used in the online mode.

Options

- `-j jnl|cpd|srf`

Specifies the type of journal file to be initialized:

`jnl`: System journal file or archive journal file

`cpd`: Checkpoint dump file

`srf`: Server recovery journal file created from an OpenTP1 file

- `-f physical-file-name~<path name>`

Specifies a name for the physical file to be initialized, expressed as the full path name.

For the system journal file for which "jnl" is specified with the `-j` option, specify the same name as that specified as the physical file name with the definition command `jnladdpf` in the system journal service definition.

For the archive journal file for which "jnl" is specified with the `-j` option, specify the same name as that specified as the physical file name with the definition command `jnladdfg` in the archive journal service definition.

When "cpd" or "srf" is specified with the `-j` option, specify the same name as that specified as the physical file name in the definition command `jnladdfg` in the checkpoint dump service definition.

If an existing physical file is specified, an error occurs.

- `-n OpenTP1-records-count~<unsigned integer> ((12-524287))`

Specifies the number of records in the OpenTP1 file to be initialized. A record means a 4096-byte area in the OpenTP1 file system managed by the journal service.

When `jnl` is specified in the `-j` option, the `-n` option must have a value greater than the value shown below:

- System journal file:

$$\uparrow \frac{(\text{jnl-max-datasize value in system service definition})+336}{4096} \uparrow + 12$$

↑ ↑: Fractions to be rounded up

- Archive journal file: 260

Notes

- If the following environment variable is not set to the same value as the OpenTP1 directory's environment variable during `jnlinit` command execution, no message will be issued in the event of a command error:

DCDIR

- If the OpenTP1 file system has been created in a character special file, the multiple of the sector length must be 4096 for the hard disk partition in which the journal file is to be created.
- Among the records in the file that is initialized using the `jnlinit` command, the number of records which you can acquire information from is the number of records specified in the `-n` option minus four records. This is because the `jnlinit` command initializes the target file after subtracting the management records (three records) from the number of records (value specified in the `-n` option) from which OpenTP1 acquires information.

jnlIs (Display journal files information)

Format

```
jnlIs -j sys|cpd|jar|srf [-s server-name]
      [-r resource-group-name] [-g file-group-name] [-d]
```

Function

The `jnlIs` command outputs information about journal files to the standard output.

Options

- `-j sys|cpd|jar|srf`

Specifies the type of journal files:

`sys`: System journal files

`cpd`: Checkpoint dump files

`jar`: Archive journal file

`srf`: Server recovery journal file as OpenTP1 file

When "cpd" or "srf" is specified, this command can be executed only when the journal service is operating.

- `-s server-name~<identifier, 1-8 characters>`

Specifies the name of a server that is to be subject to this processing when information about checkpoint dump files is being displayed.

When this option is omitted, information for all servers is output.

If a value other than `cpd` is specified in the `-j` option, the specification of the `-s` option is discarded.

- `-r resource-group-name~<1-8 character identifier>`

Specifies the name of a resource group for the file whose information is to be displayed when `jar` is specified in the `-j` option. It must be the definition file name in the archive journal service definition specified by the definition command `jnldefsv -a` in the global archive journal service definition. If this option is omitted, information about all file groups is displayed. If a value other than `jar` is specified in the `-j` option, the specification of the `-r` option is discarded.

- `-g file-group-name~<identifier, 1-8 characters>`

Specifies that information about a specific file group is to be displayed when information about system journal files is to be displayed.

When this option is omitted, information about all file groups is displayed.

■ -d

Displays element file information, together with system journal file, archive journal file or checkpoint dump file information.

When this option is omitted, element file information is not displayed.

If `serf` is specified with the `-j` option, specification of the `-d` option is ignored.

Output format

■ (1) "jnlls -j sys -d" or "jnlls -j jar -d" specified

Group	Type	Resource	Generation number	Status	Run ID	Block number
aaaaaaaa	bbb	cccccccc	dddddddd	efghijk	lllllll	mmmmmmmm nnnnnnnn
	Element	Element status	Number of records		Status of system A	Status of system B ^{#1}
oooooooo	pqrstu	vvvvvvv	wwwwwww	ABCDEF	ABCDEF	

#1: The status of system B is displayed only when dual system journal files are used.

1. File group status

2. Element file status

aaaaaaaa

File group name

bbb

File type:

sys: System journal file

jar: Archive journal file

ccccccc

Resource group name (definition file name in the archive journal service definition specified in the global archive journal service definition)

ddddddd

Generation number (hexadecimal digits)

e

File group open/close status:

o
Open (element file that is a constituent of the corresponding file group is open)

c
Close (element file that is a constituent of the corresponding file group is closed)

f

File group status:

c
Current (file group consists of valid element file and is currently subject to journal output processing)

s
Standby (file group consists of valid element file, but is not currently subject to journal output processing)

n
Reserve (file group does not consist of valid element file)

g

File group unload status:^{#1}

u
Waiting to be unloaded (contains a journal that was previously used in the current status, but has been swapped and needs to be unloaded)

-
Already unloaded

h

File group overwritable status:

d
Not overwritable (contains journal required for system recovery)

-
Overwritable (does not include journal required for system recovery)

i

File group status in OpenTP1:

b

Currently in use by the `jnlunlfg` or `jnlchgfg` command or involved in recovery processing

-

Currently not in use by `jnlunlfg` or `jnlchgfg` command or involved in recovery processing

j

File group compatibility status:

c

Element file whose management information is still on current status exists in the journal file, because an error occurred when it was previously being used on current status (e.g., a journal output error occurred in the journal group while on current status).

-

File was previously used on current status and has been processed correctly

k

File group archived status:^{#1}

u

Waiting to be archived. The file group uses the multinode facility and contains journals that have not been output to an archive journal file.

-

Already archived

lllllll

Run ID (Run ID for the journal service or global archive journal service when a file is used; hexadecimal)

mmmmmmmm

Begin block number (hexadecimal)

nnnnnnnn

End block number (hexadecimal)

oooooooo

Element file name

p

Element file open/close status:

o

Open (physical file that is a constituent of the corresponding element file is open)

c

Close (physical file that is a constituent of the corresponding element file is closed)

q

Element file usable status:

n

Element file does not consist of a physical file that can be used in online mode

u

Element file consists of a physical file that can be used in online mode

r

Element file unload status:^{#2}

u

Waiting to be unloaded (contains a journal that needs to be unloaded)

-

Already unloaded

s

Element file archived status:^{#1}

u

Waiting to be archived. The element file uses the multinode function and contains journals that have not been output to an archive journal file.

-

Already archived

t

Element file shutdown status:

h

Shutdown (corresponding element file contains a physical file whose journal

output has been suspended due to a journal output error in the physical file or due to a journal file header I/O error)

-

Not shut down (normal element file)

u

Element file readable status:

r

Physical file that is a constituent of the corresponding element file cannot be read

(Full recovery or partial recovery cannot be performed, because the journal has been erased (initialized) and cannot be read. This status occurs if an error occurs in the system journal file and the physical file is re-created and opened after eliminating the cause of the error)

-

Physical file that is a constituent of the corresponding element file can be read

vvvvvvvv

Number of records already used in the corresponding file (in hexadecimal)^{#3, #4}

This is the number of records in the OpenTP1 file system to which the journal can be output.

wwwwwwww

Total number of records in the corresponding file (in hexadecimal)^{#4, #5}

This is the number of records in the OpenTP1 file system to which the journal can be output.

A

Physical file open/close status:

o

Open (corresponding physical file has been opened by the journal service)

c

Close (corresponding physical file has not been closed by the journal service)

B

Physical file status:

c

Current (if the file group status is current, it is subject to journal output operation; if the file group status is not current, an error occurred while it was on current status, and the status is retained in the journal file header, because the error has not been recovered. The physical file cannot be used unless its status is recovered by the `jnlunlfg` or `jnlchgfg` command or it is initialized by the `jnlinit` command)

s

Standby

C

Physical file unload status:^{#2}

u

Waiting to be unloaded (corresponding physical file contains a journal that needs to be unloaded)

-

Already unloaded (corresponding physical file does not contain a journal that needs to be unloaded)

D

Physical file archived status:^{#1}

u

Waiting to be archived. The physical file uses the multinode facility and contains journals that have not been output to an archive journal file.

-

Already archived

E

Physical file shutdown status:

h

Shutdown (journal output error or journal file header I/O error occurred, and no journal has been output thereafter)

-

Not shut down (journal can be output)

F

Physical file readable status:

r

Cannot be read (full recovery or partial recovery cannot be performed, because the journal has been erased (initialized) and cannot be read. This status occurs if an error occurs in the system journal file and the physical file is re-created and opened after eliminating the cause of the error)

-

Can be read

#1: *u* is displayed if the file group is:

- A system journal file not using the multinode facility, or
- An archive journal file.

When the file group uses the multinode facility and its archived status is '-' (already archived), the file group is not archived even if the archived status of the element file and physical file is 'u' (waiting to be archived). Status mismatch would be caused by an error with the physical file leading to failure to update the status.

#2: If the file group is already unloaded, there is no need to execute the `jnlunlfg` or `jnlchgfg` command even if the element file or physical file is on unload wait status. If an error occurred during execution of the `jnlunlfg` or `jnlchgfg` command, their statuses will not match because the status will not have been rewritten.

#3: If 0 is displayed, no journal has been output. If the status is not current after completion of full recovery, the number of used records may not be correct.

#4: The total number of records in a file is the number of records in the OpenTP1 file to which the journal service or global archive journal service can output journal blocks, and is equal to the number of records specified with the `-n` option of the `jnlinit` command minus 4.

#5: One record is a 4096-byte area in the OpenTP1 file system managed by the journal service or global archive journal service.

■ (2)"jnlls -j cpd" specified

Server name	Group	Generation number	Status	Journal group	Journal block
aaaaaaaa	bbbbbbbb	ccccccc	[d]	eeeeeee	ffffff

aaaaaaaa

jnlIs (Display journal files information)

Server name

bbbbbbbb

File group name

cccccccc

Generation number (hexadecimal digits)

d

Generation status:

a: Non-overwritable

u: Overwritable or involved in a write operation

r: Reserve status

eeeeeeee

Journal file group name for the overwrite pointer

ffffff

Journal block number for the overwrite pointer (hexadecimal digits)

■ (3) "jnlIs -j cpd -d" specified

Server name	Group	Generation number	Status	System a	System b	Journal group	Journal block
<i>aaaaaaaa</i>	<i>bbbbbbbb</i>	<i>cccccccc</i>	[<i>d</i>]	[<i>g</i>]	[<i>g</i>]	<i>eeeeeeee</i>	<i>ffffff</i>

For *aaaaaaaa-ffffff*, see (2) above.

g

System A or B file status

o: Open

c: Close

h: Error shutdown

-: Physical file not allocated

■ (4) "jnlls -j srf" specified

Server name	File name	Number of srf blocks	File utilization percentage
aaaaaaaa	bbbbbbbbbbbbbbbb	cccc	ddd(%)

aaaaaaaa

Server name

bbbbbbbbbbbbbbbb

File name (file group name of a server recovery journal file specified in the checkpoint dump service definition)

cccc

Number of server recovery journal file blocks

ddd

File utilization percentage (percent ratio of server recovery journals as OpenTP1 files to the file size initialized by the jnllinit command)

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01280-E	jnlls command processing failed.	Standard error output
KFCA01281-E	jnlls command input format is invalid.	Standard error output
KFCA01282-E	Specified item was not found.	Standard error output
KFCA02128-E	Specified server name is invalid.	Standard error output
KFCA02130-E	-s option specification is invalid.	Standard error output
KFCA02132-E	jnlls command processing was cancelled due to an error.	Standard error output
KFCA02171-E	Work area cannot be allocated.	Standard error output
KFCA02172-I	There is no checkpoint dump file information to be displayed.	Standard error output
KFCA01291-I	Help message	Standard output, standard error output

Notes

- If the `jnlls` command is entered while the status of a file group is being changed (during swap processing or during execution of an operation command that manipulates the file group), the correct status may not be displayed.
- When `N` is specified in `jn1_unload_check` in the system journal service definition, the file group not on unload completion status can be used as a swappable file group.
- The `jnlls` command displays the file information based on the journal management information on the shared memory when OpenTP1 is operating. However, when OpenTP1 is inactive, there is not journal management information on the shared memory. The `jnlls` command displays the file information, reading the physical file. If the `jnlls` command is executed when OpenTP1 is inactive, the information managed by OpenTP1 online is not displayed.
- For the system journal file and archive journal file, the display information differs depending on whether OpenTP1 is active or inactive. The following shows the OpenTP1 status and display information.

Display item		OpenTP1's status	
Display	Display symbol	Active	Inactive
File group name	<i>aaaaaaaa</i>	Y	Y
File type	<i>bbb</i>	Y	Y
Resource group name	<i>ccccccc</i>	Y	Y
Generation number	<i>ddddddd</i>	Y	Y
Display item for file group	Open	<i>e</i>	N
	Usage status	<i>f</i>	(Y) ^{#1}
	Unload status	<i>g</i>	Y
	Overwritability	<i>h</i>	N
	Status in OpenTP1	<i>i</i>	N
	Inconsistent	<i>j</i>	Y
	Archived	<i>k</i>	Y
Run ID	<i>lllllll</i>	Y	Y
First block number	<i>mmmmmmm</i>	Y	Y

Display item		OpenTP1's status		
Display	Display symbol	Active	Inactive	
Last block number		<i>nnnnnnnn</i>	Y	(Y) ^{#2}
Display item for element file	Element file	<i>oooooooo</i>	Y	Y
	Open	<i>p</i>	Y	N
	Usage status	<i>q</i>	Y	N
	Unload status	<i>r</i>	Y	Y
	Archived	<i>s</i>	Y	Y
	Shut down	<i>t</i>	Y	N
	Readability	<i>u</i>	Y	Y
Number of used records		<i>vvvvvvv</i>	Y	N
Total number of records in file		<i>wwwwwwww</i>	Y	Y
Display item for physical file	Open	<i>A</i>	Y	N
	Usage status	<i>B</i>	Y	Y
	Unload status	<i>C</i>	Y	Y
	Archived	<i>D</i>	Y	Y
	Shut down	<i>E</i>	Y	N
	Readability	<i>F</i>	Y	Y

Y: Displayed

N: Not displayed

(Y): Displayed but different from while OpenTP1 is active

#1: ? is displayed for the file groups other than the current one.

#2: This information is not displayed for the current file group. This information may not be displayed for a file group where an error occurred.

- The `jnlls` command reads a physical file and displays the journal status when OpenTP1 is inactive. The `jnlls` command analyzes the journal-related definition files to determine which physical file it should read. Therefore, do not change the following definitions:

jnlIs (Display journal files information)

- Journal service definition
- System journal service definition
- Global archive journal service definition
- Archive journal service definition

When a definition is changed, the result cannot be assured. However, a definition can be changed at restart.

- Take care when displaying the following journal files when OpenTP1 is inactive:

1. System journal file and archive journal file

When OpenTP1 is inactive, the `jnlIs` command displays file information, reading the journal-related definition files and physical files. Therefore, it must be executed by the OpenTP1 administrator.

2. Checkpoint dump file and server recovery journal file converted to OpenTP1 file

Executing the `jnlIs` command while OpenTP1 is inactive results in a command error.

jnlmcst (Output MCF uptime statistics)

Format

```
jnlmcst [-e edit-type] [-l line] [-i time-interval]
        [-t [begin] [, end] [-m logical-terminal-name]
        [-a application-name]
        [-o journal-collection-mode [journal-collection-mode] ...]
        [file-name]
```

Function

The `jnlmcst` command collects MCF uptime statistics from specified unload journal files or global archive journal files, then edits and outputs it to the standard output. The information to be output includes MCF uptime statistics with respect to receiving and sending messages.

Options

- `-e edit-type`~<<all>>

Specifies the type of editing.

let: Edits and outputs MCF uptime statistics on a logical terminal name basis.

uap: Edits and outputs MCF uptime statistics on an application name basis.

all: Edits and outputs all MCF uptime statistics.

- `-l line`~((12-256))<<24>>

Specifies the number of output lines per page.

- `-i time-interval`

Specifies the time interval (in hours or minutes) at which MCF uptime statistics are edited and output.

h [*HH*]

Information is output at intervals of hours ($01 \leq HH \leq 24$). If this option is omitted, information is output at intervals of one hour.

m [*MM*]

Information is output at intervals of one minute ($01 \leq MM \leq 09$) or 10 minutes ($10 \leq MM \leq 30$). If this option is omitted, information is output at intervals of 10 minutes.

If both the hour-based and minute-based intervals are specified, an error occurs. If the `-i` option is omitted, information is output at intervals of one hour.

■ `-t [begin] [, end]`

Specifies the output range of MCF uptime statistics in terms of journal output begin and end times.

In *begin*, specify the output start time. In *end*, specify the output end time.

For *begin*, specify a time between 00:00:00 on January 1, 1970 and the current year, month, day, and time.

The start time and end time cannot both be omitted. When the start time is omitted, the output range is from the beginning of the unload journal file to the specified end time. When the end time is omitted, the output range is from the specified start time to the end of the unload journal file.

The begin and end times are expressed in the format *hhmmss* [*MMDD* [*YYYY*]] :

hh: Hour (00 ≤ *hh* ≤ 23); cannot be omitted

mm: Minute (00 ≤ *mm* ≤ 59); cannot be omitted

ss: Second (00 ≤ *ss* ≤ 59); cannot be omitted

MM: Month (01 ≤ *MM* ≤ 12); can be omitted[#]

DD: Day (01 ≤ *DD* ≤ 31); can be omitted[#]

YYYY: Year (4-digit year from 1970); can be omitted[#]

[#]: If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If the `-t` option is omitted, all information in the specified file is edited and output.

■ `-m logical-terminal-name~<1-10 alphanumeric characters>`

Specifies the name of a logical terminal subject to editing. If this option is omitted, all logical terminals are subject to editing. The option is applicable only when `let` is specified in the `-e` option.

■ `-a application-name~<1-10 alphanumeric characters>`

Specifies the name of an application subject to editing. If this option is omitted, all applications are subject to editing. The option is applicable only when `uap` is specified in the `-e` option.

■ `-o journal-collection-mode`

Specifies a journal collection mode. Multiple modes can be specified.

c

Outputs MCF uptime statistics in transaction test mode whose attribute is commit or in MCF nontransaction test mode.

r

Outputs MCF uptime statistics in transaction test mode whose attribute is rollback.

s

Outputs MCF uptime statistics in other than online test mode.

If this option is omitted, MCF uptime statistics in all journal collection modes are output.

The combinations of options that can be specified are shown below:

Option		-e			-l	-i	-t	-w	-a	-o
		let	uap	all						
-e	let	--	N	N	Y	Y	Y	Y	N	Y
	uap	N	--	N	Y	Y	Y	N	Y	Y
	all	N	N	--	Y	Y	Y	--	--	Y
	-l	Y	Y	Y	--	Y	Y	Y	Y	Y
	-i	Y	Y	Y	Y	--	Y	Y	Y	Y
	-t	Y	Y	Y	Y	Y	--	Y	Y	Y
	-m	Y	N	--	Y	Y	Y	--	N	Y
	-a	N	Y	--	Y	Y	Y	N	--	Y
	-o	Y	Y	Y	Y	Y	Y	Y	Y	--

Command arguments

- file-name~<path name>

Specifies the name of an unload journal file or global archive unload journal file. If the multinode facility is used, the name of the file sorted and merged by the `jnlst` command must be specified. If this command argument is omitted, standard input is assumed.

Output format

- `jnlmcst -e let specified`

```

jnlmcst vv-rr          ***** MCF Statistics *****          page: yyyy
File name: aa...aa
File creation date and time: bb..bb
Output specification date and time from cc...cc to dd...dd
Time interval: ee...ee
-----
Node identifier          = ff..ff
Logical entity terminal name = gg...gg#1
-----
< receive message information >
                                     Minimum Maximum Average Total (Unit)
                                     value  value  value  value
Number of inquiry
message input                    xx...xx  xx...xx  xx...xx  xx...xx (Events)
Number of one-way
receiving message input          xx...xx  xx...xx  xx...xx  xx...xx (Events)
Number of receiving
requirement issue                 xx...xx  xx...xx  xx...xx  xx...xx (Times)
Receiving message size           xx...xx  xx...xx  xx...xx  - (Bytes)
< send message information >
                                     Minimum Maximum Average Total (Unit)
                                     value  value  value  value
Number of priority branch
send requirement issue           xx...xx  xx...xx  xx...xx  xx...xx (Times)
Number of general branch
send requirement issue           xx...xx  xx...xx  xx...xx  xx...xx (Times)
Number of response send
requirement issue                 xx...xx  xx...xx  xx...xx  xx...xx (Times)
Number of outputs (segment)      xx...xx  xx...xx  xx...xx  xx...xx (Events)
-----

```

#: If `uap` is specified in the `-e` option, application names are displayed. If all is specified, all MCF information is displayed.

aa...aa

Name of the specified file

bb...bb

File creation date and time

cc...cc

Start time specified in the `-t` option ('*' is output if this option is omitted)

dd...dd

End time specified in the `-t` option ('*' is output if this option is omitted)

ee...ee

output interval specified in the *-i* option (' Δ 1 hour' is output if this option is omitted)

The output time interval and unit are displayed as follows:

1-hour interval => Δ 1 hour

1-minute interval => Δ 1 minute

ff..ff

Node identifier

gg...gg

Logical terminal name

xx...xx

Statistics (1-10 decimal digits)

If the minimum, maximum, average or total overflows, '*' is output.

The average is rounded to the nearest whole number. '-' is output as the total size of receive messages.

vv-rr

Version number-revision number

yyyy

Page

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02670-I	Help message	Standard output, standard error output
KFCA02671-E	A parameter in the <code>jnlmcst</code> command is invalid or its value exceeds the limit.	Standard error output

Message ID	Description	Output destination
KFCA02672-E	The combination of options in the <code>jnlmcst</code> command is invalid.	Standard error output
KFCA02673-E	An option in the <code>jnlmcst</code> command is specified incorrectly.	Standard error output
KFCA02674-E	A file name in the <code>jnlmcst</code> command is specified incorrectly.	Standard error output

Notes

- If the output range of MCF uptime statistics is specified and no journal record corresponds to the specified begin or end time, the journal records between the begin and end positions are to be edited.
- If no journal record corresponds to the `jnlmcst` command, processing terminates.
- If the output range of MCF uptime statistics is specified in terms of time and the start time value is greater than the end time value, the month and day or the month, day, and year must be specified so that the end time is after the start time. If the requisite information is omitted, an error occurs. If the date is omitted, the date the `jnlmcst` command is entered is assumed; therefore, if the output range does not fit within the current day, either the month and day or the month, day, and year must be specified.
- The editing and output start time for MCF uptime statistics depends on whether or not the start time is specified in the `-t` option. If the start time is specified, information is output from the time zone in which the specified start time is included. If the start time is not specified, information is output from the time zone including the IJ, GJ, OJ, and AJ record time at the beginning of the applicable file. The term "time zone" means the time interval divided by the value specified in the `-i` option on the basis of the start time (hh: 00).
- If the output end time for MCF uptime statistics is specified, information is output until the time zone including the specified end time.
- When output, the edited average is rounded to one decimal place.
- '-' is output as the total size of receive messages.
- If the minimum, maximum, average or total overflows, '*' is output.
- If no file name is specified, the standard input is assumed. Therefore, the input file must be specified by piping, redirecting, etc.
- When the `-t` option is specified and there are many portions matching the specified range in the specified file, all of them are displayed.
- For the `-t` option, if you specify a value beyond 2038 as the year of the output

end time, in some cases CPU load and memory usage may increase to the extent that it takes some time for the command to terminate.

Examples

In this example, the command edits the MCF uptime statistics from 17:30:00.0 on May 29, 2003 to the end of an unload journal file, and outputs them:

Unload journal file: `sysjnl001`

```
jnlmcst -e all -t 17300005292003 /tmp/jnl/sysjnl001
```

jnlmkrf (Recover journal files)

Format

```
jnlmkrf {-j trf|-j srf -s server-name} unload-journal-file-name
        [[ $\Delta$ unload-journal-file-name] ...]
```

Function

The `jnlmkrf` command recovers journal files with which an error occurred during OpenTP1 restart processing.

Options

- `-j trf`
Recovers a transaction recovery journal file.
- `-j srf`
Recovers a server recovery journal file.
- `-s server-name~<1-8 character identifier>`
Specifies the name of the applicable server when a server recovery journal file is to be recovered. This option is mandatory when `srf` is specified in the `-j` option. If it is omitted, a command error occurs. With `trf` specified in the `-j` option, the specification of the `-s` option is discarded.

Command arguments

- `unload-journal-file-name~<path name>`
Specifies the name of an unload journal file that was created by executing the `jnlunlfg` command for the system journal file containing journals needed for recovery of journal files.

If there are more than one unload journal file, all must be specified. Up to 64 unload journal files can be specified. When multiple unload journal files are specified, a space must be placed between files. Unload journal files can be specified non-chronologically with respect to their generations. If any missing serial number is found with the specified unload journal files, recovery processing for them is suspended.

jnlopfnfg (Open journal files)

Format

```
jnlopfnfg -j sys|cpd|jar [-s server-name]
           [-r resource-group-name] -g file-group-name
           [-e element-file-name] [-a] [-b]
```

Function

The `jnlopfnfg` command opens a closed physical file that is a constituent of a specified file group so that the files can be used in the online mode.

The `jnlopfnfg` can be entered only when OpenTP1 and the journal service are both operating.

Options

- `-j sys|cpd|jar`

Specifies the type of journal files to be opened:

`sys`: System journal files

`cpd`: Checkpoint dump files

`jar`: Archive journal file

- `-s server-name~<identifier, 1-8 characters>`

Specifies the name of a server that is to be subject to this processing when checkpoint dump files are to be opened.

When this option is omitted, the first (among all servers) checkpoint dump files with the matching file group name are opened.

If `sys` is specified for `-j` option, specification of the `-s` option is ignored.

- `-r resource-group-name~<1-8 character identifier>`

Specifies the name of the target resource group. If `jar` is specified in the `-j` option, the name of the resource group in the file to be opened must be specified. It must be the definition file name specified in the archive journal service definition specified in the definition command `jnldfs -a` of the global archive journal service definition.

When this option is omitted, the name of an online archive journal file defined at the beginning of the definition command `jnldfs -a` of the archive journal service definition is assumed.

If other than `jar` is specified in the `-j` option, the specification of the `-r` option is discarded.

- **-g** *file-group-name*~<identifier, 1-8 characters>
Specifies the file group name of the journal files to be opened.
- **-e** *element-file-name*
Specifies the name of an element file in the journal file group to be opened.
- **-a**
Specifies that physical files in system A are to be opened.
- **-b**
Specifies that physical files in system B are to be opened.

If this option is specified when dual files are not being used, a command error occurs.

When the **-a** and **-b** options are both omitted, or when **cpd** is specified in the **-j** option, the command performs processing as follows:

- If dual files are not being used, the physical file in system A is closed in the same manner as when the **-a** option is specified.
- If dual files are being used, the physical files in both systems A and B are opened in the same manner as when the **-a** and **-b** options are both specified.
- When one-system operation is unavailable in dual system configuration, a command error occurs if adding either the **-a** or **-b** option to "**-j cpd**".

If all physical files subject to open processing are already open, a warning message (KFCA01283-W or KFCA02167-W) is issued, although no command error occurs (command termination code: 0).

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01215-I	File group has been opened successfully.	Message log file
KFCA01280-E	<code>jnlopnfg</code> command processing failed.	Standard error output
KFCA01281-E	<code>jnlopnfg</code> command input format is invalid.	Standard error output
KFCA01283-W	Insignificant error occurred during <code>jnlopnfg</code> command execution.	Standard error output
KFCA01285-E	Swap processing is underway.	Standard error output
KFCA01294-I	Help message	Standard output, standard error output

Message ID	Description	Output destination
KFCA02128-E	Specified server name is invalid.	Standard error output
KFCA02129-E	Specified file group name is invalid.	Standard error output
KFCA02130-E	-s option specification is invalid.	Standard error output
KFCA02131-E	-g option specification is invalid.	Standard error output
KFCA02132-E	jnlopnfg command processing was cancelled due to an error.	Standard error output
KFCA02167-W	Specified file group is already open.	Standard error output
KFCA02169-E	File open processing failed.	Standard error output

Notes

The `jnlopnfg` command may result in an error if it is entered during swap processing or during the processing of another operation command.

jnlrinf (Display journal information that is read during restart)

Format

```
jnlrinf -j sys|jar
```

The `jnlrinf` command outputs the journal information that is read during restart.

Options

- `-j sys|jar`

Specifies the journal file containing the information to be displayed.

`sys`: System journal file

`jar`: Archive journal file

Output format

Type	Resource	Group	Generation No.	Block No.	Time of reading	Block information of the current generation
<i>aaa</i>	<i>bbbbbbb</i>	<i>ccccccc</i>	<i>ddddddd</i>	<i>eeeeeee</i>	<i>yyyy/mm/dd hh:mm:ss</i>	<i>ffffff - ggggggg</i>

- *aaa*
File type
`sys`: System journal file
`jar`: Archive journal file
- *bbbbbbb*
Resource group name (definition file name of the archive journal service definition specified in the global archive journal service definition)
- *ccccccc*
File group name
- *ddddddd*
Generation number (hexadecimal)
- *eeeeeee*
Number of the block that is already read (hexadecimal)
- *yyyy/mm/dd hh:mm:ss*.

Time when the block is read

- *ffffff*

Beginning block number of the generation currently processed (hexadecimal)

- *gggggggg*

Last block number of the generation currently processed (hexadecimal)

The current file group and the failed file group are displayed as *********.

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01280-E	The <code>jnlrinf</code> command cannot be executed.	Standard error output
KFCA01281-E	The format of the <code>jnlrinf</code> command is incorrect.	Standard error output
KFCA01282-E	The specified item is not found.	Standard error output
KFCA01284-I	Help message	Standard output, standard error output

Notes

- Message KFCA01280-E may be output even when the OpenTP1 system is running if the journal service is not being restarted.
- When the start status of the OpenTP1 system is "Restart" and the journal service is online, this command outputs the journal information that was read last.

jnlrm (Delete journal files)

Format

```
jnlrm -f physical-file-name [-u]
```

Function

The `jnlrm` command deletes journal files under the OpenTP1 file system.

If a specified file is open, it cannot be deleted.

Options

- `-f physical-file-name~<path name>`

Specifies the name of the physical file to be deleted, expressed as the full path name.

- `-u`

When the file specified for the `-f` option is a system journal file or archive journal file, it is forcibly deleted even if it is on unload wait status. For a checkpoint dump file or server recovery journal file, the specification of the `-u` option is discarded.

If this option is omitted and the file specified for the `-f` option is a system journal file or archive journal file, the command checks the file's unload status is checked. If the file is on unload wait status, a command error occurs.

jnlrput (Output records of unload journal files or global archive unload journal files)

Format

```
jnlrput [-t start-time [, end-time] [-e] [-f] [-l] [-c key]
        [-u transaction-global-identifier [, transaction-branch-identifier]]
        [-o journal-collection-mode [journal-collection-mode] ...]
        [-j journal-type [journal-type] ...] [-q _trn] [-q _rpc]
        [-d user's-extraction-condition(s)] [-x] [file-name
        ( $\Delta$ file-name) ...]
```

Function

The `jnlrput` command outputs to the standard output user journal record information within the specified unload journal file or global archive unload journal file, CPU usage time information for a transaction branch, and statistics about response.

Options

- `-t` [*start-time*] [, *end-time*]

Specifies the output range of uptime statistics in terms of journal output begin and end times.

In "begin", specify the output start time. In "end", specify the output end time.

For begin, specify a time between 00:00:00 on January 1, 1970 and the current year, month, day, and time.

The start time and end time cannot both be omitted. When the start time is omitted, the output range is from the beginning of the unload journal file to the specified end time. When the end time is omitted, the output range is from the specified start time to the end of the unload journal file.

The begin and end times are expressed in the format `hhmmss[MMDD[YYYY]]`:

hh: Hour (01 ≤ hh ≤ 23); cannot be omitted

mm: Minute (00 ≤ mm ≤ 59); cannot be omitted

ss: Second (00 ≤ ss ≤ 59); cannot be omitted

MM: Month (01 ≤ MM ≤ 12); can be omitted[#]

DD: Day (01 ≤ DD ≤ 31); can be omitted[#]

YYYY: Year (4-digit year from 1970); can be omitted[#]

#: If the year is omitted from the begin or end time, the specified month, day, and time

of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If the `-t` option is omitted, all information in the specified file is output.

■ `-e`

Outputs only commit-determined user journal records (UJ).

When this option is not specified, all user journal records (UJ) are output.

■ `-f`

Outputs commit-determined user journal records from the first file. Also, checks whether the journal file's block numbers are sequential ones in ascending order, beginning with 1. If an error is found, processing terminates.

The `-f` option is valid only when the `-e` option is specified.

If this option is not specified, system assumes inheritance from the previous record output process, checking whether the block numbers are sequential ones in ascending order, beginning with the last block number + 1. If an error is found, processing terminates.

■ `-l`

Outputs commit-determined user journal records from the last file. Then the inheritance file is deleted.

The `-l` option is valid only when the `-e` option is specified.

If this option is not specified, system assumes inheritance to the next record output process, creating an inheritance file.

■ `-c key~((001-999))<<001>>`

Specifies part of the name of an inheritance file. Actually, OpenTP1 creates a file named `jnlrput ***` (`***` = the value specified in this option), which in turn becomes the name of the inheritance file.

■ `-u transaction-global-identifier [,transaction-branch-identifier~<33 alphanumeric characters, special characters, and hexadecimal digits>`

Specifies the contents to be edited by specifying the transaction identifier. The transaction identifier must be in "transaction-global-identifier, transaction-branch-identifier." If this option is omitted, all transaction IDs are subject to editing.

■ `-o journal-collection-mode`

Specifies a journal collection mode. Multiple modes can be specified.

c

Outputs journals in transaction test mode whose attribute is commit or in MCF nontransaction test mode.

r

Outputs journals in transaction test mode whose attribute is rollback.

s

Outputs journals in other than online test mode.

If this option is omitted, journals in all journal collection modes are output.

■ -j ~<<u>>

Specifies the record type that identifies the contents to be output. Following are the values to be specified as record type and the contents to be output.

Record type	Value to be specified	Contents	Remarks
SJ	s	System statistics [#]	Statistical journal
AJ	a	Send completion information	
IJ	I	Input queue registration information	
OJ	o	Messages in the output queue	
MJ	m	Message journal	
GJ	g	Receive information	
UJ	u	User's own information	User journal

[#]: With `_trn` specified in the `-q` option, the information about the CPU usage time by a transaction branch is output. With `_rpc` specified in the `-q` option, the response statistics are output.

■ -q `_trn`

Outputs the information about the CPU usage time by a transaction branch within the specified file.

Specification of this option is valid only with "s" (system statistics) specified in the `-j` option.

■ -q `_rpc`

Outputs the response statistics within the specified file.

Specification of this option is valid only with "s" (system statistics) is specified in the

-j option.

■ -d *user's-extraction-conditions*

Specifies the conditions of extracting the user journal records (UJ). When this option is specified, the journal records are compared to the comparison data and the journal records that satisfy the conditions specified with comparison symbols. The number of characters in the specified extraction conditions must not exceed 255.

The following shows the description format of extraction conditions:

comparison-position , [*comparison-symbol*] , [*comparison-format*] , *comparison-data*
[, *comparison-position* , [*comparison-symbol*] , [*comparison-format*] , *comparison-data*]

Up to two extraction conditions can be specified.

When more than one extraction condition is specified, the journal records that satisfy all the conditions are extracted.

■ *comparison-position*~<hexadecimal>((0 to 7faf8))

Specifies the position of the comparison data to be compared to the user journal records. Specify the position from the beginning of the UAP journal. For UJ code, specify "CODE".

■ *comparison-symbol*~<<eq>>

Specifies a comparison symbol as the comparison condition for the data to be compared to the user journal records.

eq

The contents of UJ are equal to the comparison data. (contents of UJ = comparison data)

ne

The contents of UJ are not equal to the comparison data. (contents of UJ ≠ comparison data)

gt

The contents of UJ are greater than the comparison data. (contents of UJ > comparison data)

ge

The contents of UJ are greater than or equal to the comparison data. (contents of UJ ≥ comparison data)

lt

The contents of UJ are less than the comparison data. (contents of UJ <

comparison data)

le

The contents of UJ are less than or equal to the comparison data. (contents of UJ \geq comparison data)

■ *comparison-format*~<<x>>

Specifies the format of the data to be compared. The user journal records are compared to the comparison data, based on the format specified by the comparison format.

x: Hexadecimal format

c: Character format

■ *comparison-data*

Specifies the data to be compared. When specifying the comparison data in hexadecimal format, specify the data with 0 to 9 and a to f (or A to F) in even number of digits.

When specifying the UJ code, use a 2-, 4-, 6-, or 8-digit hexadecimal number.

A comma (,), space, and metacharacters of the shell cannot be specified.

■ -x

Outputs only the user journal records (UJ) that are out of the scope transaction.

If you omit the -x option, all the user journal records (UJ) are output.

Command arguments

■ *file-name*~<path name>

Specifies the name of an unload journal file or global archive unload journal file. If the multinode facility is used, the name of the file sorted and merged by the `jnl-sort` command must be specified.

The -e option allows more than one file to be specified. When more than one is specified, a space must be placed between files. You can specify a maximum of 256 files.

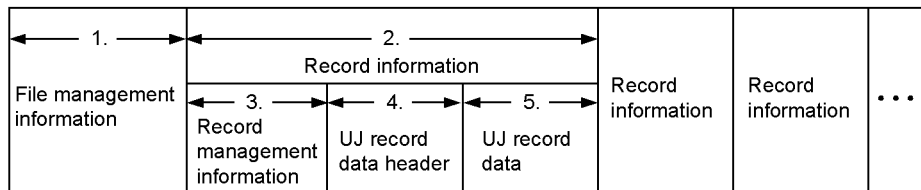
If this command argument is omitted, standard input is assumed.

The following shows the possible combinations of command arguments and options that can be specified:

Option	-t	-e	-f	-l	-c	-u	-o	-j			-q	-d	-x	File	
								s	u	Other				Single	Multiple
-t	--	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
-e	Y	--	Y	Y	Y	N	N	N	Y	N	N	N	N	Y	Y
-f	Y	Y	--	Y	Y	N	N	N	Y	N	N	N	N	Y	Y
-l	Y	Y	Y	--	Y	N	N	N	Y	N	N	N	N	Y	Y
-c	Y	Y	Y	Y	--	N	N	N	Y	N	N	N	N	Y	Y
-u	Y	N	N	N	N	--	Y	Y	Y	Y	Y	Y	N	Y	N
-o	Y	N	N	N	N	Y	--	Y	Y	Y	Y	Y	Y	Y	N
-j	s	Y	N	N	N	Y	Y	--	Y	Y	Y	--	Y	Y	N
	u	Y	Y	Y	Y	Y	Y	Y	--	Y	--	Y	N	Y	Y
	Other	Y	N	N	N	Y	Y	Y	Y	--	--	--	Y	Y	N
-q	Y	N	N	N	N	Y	Y	Y	--	--	--	--	--	Y	N
-d	Y	N	N	N	N	Y	Y	--	Y	--	--	--	Y	Y	N
-x	Y	N	N	N	N	N	Y	Y	N	Y	--	Y	--	Y	Y
File	Single	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	--
	Multiple	Y	Y	Y	Y	Y	N	N	N	Y	N	N	N	Y	--

Output format

- (1) Output format of all the journal records when UJ is output



1. The size of the file management information is 128 bytes.
2. The size of the record information is stored in the "record information size" in the record management information (3.).
3. The size of the record management information is stored in the "record management information size" in the file management information (1.).
4. The size of the record data header is stored in the "record data header size" in the file management information (1.).
5. The size of the UJ record data is as follows:

$$[\text{Size of UJ record data}] = [\text{Size of record information}] - [\text{Size of record}]$$

management information + Size of UJ record data header]

■ (2) Detailed output format of each information item

```

/*****
/* File Management Information */
/*****
struct dcjup_filmng_t { /*File management information */
    DCULONG    filmng_leng    /*File management information size (128 bytes) */
    char       eye_catcher[4] /*File management information identifier "JUP" */
    char       nodeid_inf[8]  /*Node identifier */
    DCULONG    recmng_leng    /*Record management information size */
    DCULONG    reched_leng    /*Record data header size */
    DCULONG    buff_leng     /*Maximum record length */
    DCULONG    format_ver    /*Format version */
    char       reserve[96]   /*Reserved */
}
typedef struct dcjup_filmng_t DCJUP_FILMNG_T ;/*
/*****
/* Record Management Information */
/*****
struct dcjup_recmng_t { /* record management information */
    DCULONG    record_leng    /* record length */
    char       jnl_rec_kind   /* journal record kind */
    char       trnid_flag     /* transaction_id existence flag
                               /* 0x00 : no_exist
                               /* 0x80 : exist
    char       reserve1[2]    /* reserve1
    DCULONG    buffering_time /* acquisition time
    struct request_data {
        char    nodeid_inf[8] /* node identifier
        char    serv_name[9]  /* server name
        char    reserve2[3]   /* reserve2
        DCULONG run_id       /* run_id
    } request_serv_data
    char       usr_inf[8]     /* user information
    char       reserve3[12]  /* reserve3
    struct tran_id {
        unsigned char global_tran_id[12] /* global transaction_id
        unsigned char branch_tran_id[12] /* branch transaction_id
    } tran_id_inf
    char       hed2_flag     /* second record data header
                               /* existence flag
    char       reserve4[3]   /* reserve4
    DCULONG    hed2_size    /* second record data header size
    DCULONG    buffering_utm /* acquisition time
    char       reserve5[36] /* reserve5
}
typedef struct dcjup_recmng_t DCJUP_REC_MNG_T ;/*

```

```

/*****
/* User Journal Record Data Header */
/*****
struct dcjup_ujhed_t {          /* */
    DCULONG   data_leng      ; /*Journal record data size  */
    DCULONG   uj_code       ; /*uj-code              */
    char      reserve[120]   ; /*Reserved            */
}                               /* */
typedef struct dcjup_ujhed_t DCJUP_UJHED_T ; /* */
/*****
/* System Statistics Record Data Header */
/*****
struct dcjup_admhed_t {
    DCULONG   data_ent_cnt   ; /* number of data entry */
    char      sj_serv_name[9] ; /* server name          */
    char      reserve[115]   ; /* reserve1             */
}
typedef struct dcjup_admhed_t DCJUP_ADMHED_T ;
/*****
/* System Statistics Record Data*/
/*****
struct dcjup_adment_t {
    DCULONG   data_ent_size  ; /* data entry size     */
    DCULONG   event_id       ; /* event id            */
    DCULONG   sj_sum_data    ; /* total value         */
    DCULONG   sj_max_data    ; /* maximum value       */
    DCULONG   sj_min_data    ; /* minimum value       */
    DCULONG   sj_event_cnt   ; /* event count         */
}
typedef struct dcjup_adment_t DCJUP_ADMENT_T ;

```

```

/*****
/* Transaction Branch CPU Usage Time Information Data Header */
/*****
struct dcjup_sjhed_t {
char      trn_brc_cnt[2] /*Determination, including subbranches
                        /*'c' :Commit determination
                        /*'r' :Rollback determination
                        /*'hc' :Commit determination by commands
                        /*'hr' :Rollback determination by commands
                        /*'hh' :Hazard determination by commands
                        /*'hm' :Mix determination by commands
char      trn_prc_cnt /*Branch determination process type
                        /*'u' :User server process
                        /*'r' :Recovery process
char      trn_brc_man /*Branch body determination
                        /*'c' :Commit determination
                        /*'r' :Rollback determination
DCULONG  min_data /*Branch execution time data in seconds
DCULONG  miq_data /*Branch execution time data in less than seconds
DCULONG  syn_min_data /*Branch synchronous point processing execution
                        time data in seconds
DCULONG  syn-miq_data /*Branch synchronous point processing execution
                        time data in less than seconds
char      serv_name[9] /*User server name
char      reserve1[3] /*Reserved, system
char      svc_name[32] /*Service name
DCULONG  trnall_cputime /*Transaction all CPU time (μsec.)
DCULONG  betran_cputime /*OpenTP1 CPU time (μsec.)
DCULONG  uap_cputime /*UAP CPU time (μsec.)
DCULONG  dam_cputime /*OpenTP1_DAM CPU time (μsec.)
DCULONG  tam_cputime /*OpenTP1_TAM CPU time (μsec.)
DCULONG  ism_cputime /*OpenTP1_ISM CPU time (μsec.)
char      reserve2[40] /*Reserved, system
}
typedef struct dcjup_sjhed_t DCJUP_SJHED_T ;

```

5

```

/*****
/* ij record data header */
/*****
struct dcjup_ijhed_t {
    char      input_le_name[16]    /* Input logical terminal name */
    char      app_name[10]         /* Application name */
    char      reserve1[2]          /* Reserved 1 */
    unsigned char input_no[12]     /* Message input serial number */
    char      map_name[9]          /* Map name */
    char      reserve2[3]          /* Reserved 2 */
    char      msg_type             /* Input message type */
    char      seg_type             /* Sequence ID */
    char      reserve3[2]          /* Reserved 3 */
    DCULONG   input_msg_sz         /* Input message size */
    char      reserve4[68]         /* Reserved 4 */
}
typedef struct dcjup_ijhed_t DCJUP_IJHED_T;

/*****
/* mj record data header */
/*****
struct dcjup_mjhed_t {
    char      le_name[16]          /* Logical terminal name */
    char      cn_name[16]          /* Connection name */
    char      mj_type             /* mj type */
    char      seg_type            /* Sequence ID */
    char      reserve1[2]          /* Reserved 1 */
    DCULONG   msg_sz              /* Message size */
    char      reserve2[88]         /* Reserved 2 */
}
typedef struct dcjup_mjhed_t DCJUP_MJHED_T;

/*****
/* oj record data header */
/*****
struct dcjup_ojhed_t {
    char      output_le_name[16]   /* Output logical terminal name */
    char      app_name[10]         /* Application name */
    char      reserve1[2]          /* Reserved 1 */
    char      msg_type             /* Output message type */
    unsigned char output_no_flag  /* Presence of output serial no. */
    char      reserve2[2]          /* Reserved 2 */
    DCULONG   output_no           /* Message output serial no. */
    DCULONG   output_msg_sz       /* Output message size */
    char      reserve3[88]         /* Reserved 3 */
}
typedef struct dcjup_ojhed_t DCJUP_OJHED_T;

```



```

/*****
/* aj record data header */
/*****/
struct dcjup_ajhed_t {
    char        output_le_name[16]    /* Output logical terminal name */
    char        app_name[10]         /* Application name */
    char        reserve1[2]          /* Reserved 1 */
    char        msg_type             /* Output message type */
    unsigned char output_no_flag     /* Presence of output serial no. */
    char        reserve2[2]          /* Reserved 2 */
    DCULONG    output_no            /* Message output serial no. */
    char        reserve3[92]         /* Reserved 3 */
}
;

typedef struct dcjup_ajhed_t DCJUP_AJHED_T ;

/*****
/* gj record data header */
/*****/
struct dcjup_gjhed_t {
    char        output_le_name[16]    /* Input logical terminal name */
    char        app_name[10]         /* Application name */
    char        reserve1[2]          /* Reserved 1 */
    unsigned char input_no[12]       /* Message input serial no. */
    DCULONG    input_msg_sz         /* Input message size */
    char        reserve2[84]         /* Reserved 2 */
}
;

typedef struct dcjup_gjhed_t DCJUP_GJHED_T ;

/*****
/** Response statistical information record data header **/
/*****/
struct dcjup_rspshed_t {
    char        node_id_sv[8]        /* Node identifier (for server) */
    char        serv_name_sv[9]      /* Server name (for server) */
    char        reserve1[3]          /* Reserved area 1 */
    char        node_id_cl[8]        /* Node identifier (for client) */
    char        serv_name_cl[9]      /* Server name (for client) */
    char        reserve2[3]          /* Reserved area 2 */
    char        svcgr_name[32]       /* Service group name*/
    char        svc_name[32]         /* Service name */
    DCLONG     rpc_flags             /* RPC type */
    DCULONG    nodeaddr_cl          /* IP address (for client) */
    unsigned short port_cl          /* Port number */
    char        reserve3[14]         /* Reserved area 3 */
}
;

typedef struct dcjup_rspshed_t DCJUP_RSPHED_T ;

```

```

/*****
/* Response statistical information second record data header */
/*****
struct dcjup_rspshed2_t {
    DCULONG    res_min_data        /*Response time (seconds)      */
    DCULONG    res_miq_data        /*Response time (mseconds)     */
    DCULONG    svc_min_data        /*Service execution time (seconds)
    DCULONG    svc_miq_data        /*Service execution time (mseconds)
    DCULONG    wait_min_data       /*Service waiting time (seconds)
    DCULONG    wait_miq_data       /*Service waiting time (mseconds)
    DCULONG    ucpu_min_data       /*CPU usage time by user (seconds)
    DCULONG    ucpu_miq_data       /*CPU usage time by user (mseconds)
    DCULONG    scpu_min_data       /*CPU usage time by system (seconds)
    DCULONG    scpu_miq_data       /*CPU usage time by system (mseconds)
    char        reserve1[88]       /*Reserved area 1
}
typedef struct dcjup_rspshed2_t  DCJUP_RSPHED2_T ;

```

Note

When the transaction ID existence flag is 0x00, 0x00 is set for the global_tran_id[12] entry and the branch_tran_id[12] entry.

1. Displays file management information (the status and contents of output results as a whole).
 - 2., 3., 4., 5., 6., 7., 8., 9., 10., 11., and 12. Indicates Record information.
 2. Displays record management information (the status and contents of applicable record information as a whole)
 - 3., 4., 5., 6., 7., 8., 9., 10., 11., and 12. Displays record data (the record data header indicating fixed record data, and record data indicating floating record data). For transaction branch CPU usage time information and aj record, only record data headers indicating fixed record data are displayed.
- For response statistics, record data headers and second record data headers indicating fixed record data are displayed.

The output format of 1. to 12., is defined by dcjup.h.

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02605-E	An error occurred while the inheritance file was being accessed.	Standard error output
KFCA02606-E	The inheritance file is invalid.	Standard error output
KFCA02607-E	The relationship between the inheritance file and the unload journal file is invalid.	Standard error output
KFCA02622-W	The target record to be output does not exist.	Standard error output
KFCA02680-I	Help message	Standard output, standard error output
KFCA02681-E	A parameter in the <code>jnlrput</code> command is invalid or its value exceeds the limit.	Standard error output
KFCA02682-E	The combination of options in the <code>jnlrput</code> command is invalid.	Standard error output
KFCA02683-E	An option in the <code>jnlrput</code> command is specified incorrectly.	Standard error output
KFCA02684-E	A file name in the <code>jnlrput</code> command is specified incorrectly.	Standard error output
KFCA02685-E	The record to be extracted that is specified in the <code>jnlrput</code> command is invalid.	Standard error output

Notes

- If the output range is specified and no journal record corresponds to the specified begin or end time, the journal records between the begin and end positions are to be output.
- If the output range is specified in terms of time and the start time value is greater than the end time value, the month and day or the month, day, and year must be specified so that the end time is after the start time. If the requisite information is omitted, an error occurs. If the date is omitted, the date the `jnlmcst` command is entered is assumed; therefore, if the output range does not fit within the current day, either the month and day or the month, day, and year must be specified.

- If no file name is specified, the standard input is assumed. Therefore, the input file must be specified by piping, redirecting, etc.
- When the inheritance file specified in the `-e` option already exists, the file remains under the name `jnlrput***.bak`.
- When the `jnlrput` command with its `-e` option specified is executed several times to output commit-determined user journal records (UJ) from more than one unloaded-journals file, the same value must be specified in the `-c` option of each `jnlrput` command.
- When the `-e` option is specified to extract records from more than one unloaded-journals file or global archive unloaded-journals file, all of the journal files must be those which were output during the same online operation.
- If a timeout error occurs during the RPC, `-l` is set for the response time (`res_min_data` and `res_max_data`) in the response statistics second record header while other information items are set to 0.
- When the `-e` option is specified, do not specify the unload journal file that is acquired by specifying the `-t` option in the `jnlunlfg` command. If you do so, message KFCA02601-E is output.
- The number of characters you can enter in this command is determined by your OS. If the number of entered characters exceeds the upper limit, an error occurs.
- An error occurs when the number of opened files exceeds the maximum number of files that can be opened for one process, including when the specified number of files is less than 256.
- The last four bytes of the "transaction global identifier" and "transaction branch identifier" regions (unsigned DCLONG type numerals) of the record management information differ from the last four bytes of the global transaction IDs that appear in messages and other forms of system output.

Examples

1. Output CPU usage time information from an unloaded-journals file.

```
jnlrput -j s -q_trn/tp1/jnl/sysjnl001
```

2. Output commit-determined user journal records (UJ) from an unloaded-journals file consisting of three files, in one step.

```
jnlrput -e -f -l /tp1/jnl/sysjnl001 /tp1/jnl/sysjnl002  
/tp1/jnl/sysjnl003
```

3. Output commit-determined user journal records (UJ) from an unloaded-journals file consisting of four files, in three steps.

```
<First>  
jnlrput -e -f /tpl/jnl/sysjnl001 /tpl/jnl/sysjnl002  
<Second>  
jnlrput -e /tpl/jnl/sysjnl003  
<Third>  
jnlrput -e -l /tpl/jnl/sysjnl004
```

4. Output system statistics (SJ), which are in an unload journal file, from 00:00:00.0 on January 1, 2003 up to 23:59:59.999999 on December 31, 2003:

Unload journal file: sysjnl001

```
jnlrput -t 00000001012003,23595912312003 -j s /tpl/jnl/sysjnl001
```

jnl-sort (Sort and merge unload journal files or global archive unload journal files chronologically)

Format

```
jnl-sort [-n node-identifier] [-g]
          [-i [begin-journal-server-run-ID]
             [, end-journal-server-run-ID]]
          [file-name [file-name] ...]
```

Function

The `jnl-sort` command sorts and merges the contents, block to block, of the specified unload journal file or global archive unload journal file chronologically. The result of processing is output to the standard output in the form of a global archive unload journal file.

Options

- `-n node-identifier` (~<4 alphanumeric characters and symbols)

Sorts and merges only the journal blocks having the specified node identifier.

The node identifier must begin with an alphabetic character, followed by alphanumeric characters. If this option is omitted, the file is not sorted and merged on a node identifier basis.

- `-g`

Specify this option only to merge journal blocks, not to sort them.

When multiple unload journal files and global archive unload journal files are specified, the files are merged in ascending order of run ID (generation number if the files are executed during the same online operation). Note that journal blocks are not sorted. When you specify this option, work files are not created.

- `-i [begin-journal-server-run-ID] [, end-journal-server-run-ID] ~<1-8 hexadecimal digits>((0-ffffff))`

Sorts and merges only the journal blocks having the specified journal server run ID.

If the `-i` option is specified, either a begin or end journal server run ID must be specified. If both are specified, the begin journal server run ID must be equal to or smaller than the end journal server run ID.

If the begin journal server run ID is omitted, the journal blocks ranging from 0 to the specified end journal server run ID are sorted and merged.

If the end journal server run ID is omitted, the journal blocks ranging from the

specified begin journal server run ID to ffffffff are sorted and merged.

When this option is omitted, journal blocks are not sorted and merged on a journal server run ID basis.

Command arguments

- *file-name*~<path name>

Specifies the name of an unload journal file and a global archive unload journal file. You can specify a maximum of 256 files.

When this command argument is omitted, the standard input is input. Specify this command argument when the -n option is omitted.

The combinations of options and command arguments that can be specified are shown below:

Option	-n	-g	-i	Unload journal file	Global archive unload journal file
-n	--	Y	Y	Y	Y
-g	Y	--	Y	Y	Y
-i	Y	Y	--	Y	Y
Unload journal file	Y	Y	Y	--	Y
Global archive unload journal file	Y	Y	Y	Y	--

Y: Can be specified

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02690-I	Help message	Standard output, standard error output
KFCA02691-E	A parameter in the jnl-sort command is invalid or its value exceeds the limit.	Standard error output

jnl`sort` (Sort and merge unload journal files or global archive unload journal files chronologically)

Message ID	Description	Output destination
KFCA02692-E	The combination of options in the <code>jnl<code>sort</code></code> command is invalid.	Standard error output
KFCA02693-E	An option in the <code>jnl<code>sort</code></code> command is specified incorrectly.	Standard error output
KFCA02694-E	A file name in the <code>jnl<code>sort</code></code> command is specified incorrectly.	Standard error output
KFCA02695-E	An access error occurred in the work file to be used by the <code>jnl<code>sort</code></code> command.	Standard error output
KFCA02696-E	The target block for the <code>jnl<code>sort</code></code> command does not exist.	Standard error output

Notes

- The `jnlsort` command creates temporary work files for using the OS sort command. Following are the work files that are created by the `jnlsort` command:

File name	Directory	File size
<code>sort_i</code> + process ID	Current directory	256 bytes x number of blocks to be sorted
<code>sort_o</code> + process ID	Current directory	256 bytes x number of blocks to be sorted

- When you specify the standard input (when you omit a file name as a command argument), only a global archive unload journal file can be entered.
- The number of characters you can enter in this command is determined by your OS. If the number of entered characters exceeds the upper limit, an error occurs.
- An error occurs when the number of opened files exceeds the maximum number of files that can be opened for one process, including when the specified number of files is less than 256.

jnlstts (Output uptime statistics)

Format

```
jnlstts [-e edit-item] [-u edit-unit] [-l lines] [-c] [-i interval]
        [-t [begin] [, end]] [{-s server-name|-s service-name}]
        [file-name]
```

Function

The `jnlstts` command collects statistics from a specified unload journal file or global archive unload journal file. This command edits the statistics and outputs it to the standard output.

The following information is output:

- System statistics (information about each component in a system statistical record)
- Transaction statistics (information about each transaction in a system statistical record)
- Statistics about response (information about each response time in a system statistical record)
- Statistics about communication delay (information about communication delay in a system statistical record)

Options

- `-e edit-item~<<all>>`

Specifies the item to be edited:

`sys`: Output system statistics

`trn`: Output transaction statistics

`rsp`: Output statistics about response

`dly`: Output statistics about communication delay

`all`: Both of the above

- `-u edit-unit~<<srv>>`

Specifies the unit of editing:

`srv`

Edit in units of server names.

If `srv` is specified, it is assumed that a server name has been specified in the `-s` option.

`svc`

Edit in units of service names.

If `svc` is specified, it is assumed that a service name has been specified in the `-s` option.

This option is applicable only when `trn` or all is specified for the `-e` option.

- `-l lines~<unsigned integer>((12~256))<<24>>`

Specifies the number of output lines per page.

- `-c`

Displays the edited results of uptime statistics, in CSV format.

- `-i interval`

Specifies in hours or minutes the interval at which uptime statistics are to be edited and output:

`h[HH]`

Output at an interval of the specified number of hours ($01 \leq HH \leq 24$) (when specification of HH is omitted, the output interval is one hour).

`m[MM]`

Output at an interval of the specified number of minutes ($01 \leq MM \leq 09$) or tens of minutes ($10 \leq MM \leq 30$) (when specification of MM is omitted, the output interval is 10 minutes).

If intervals in both hours and minutes are specified, an error occurs.

When the `-i` option is omitted, the output interval is one hour.

- `-t [begin] [, end]`

Specifies the uptime statistics output range in terms of the journal block output begin and end times.

In "begin", specify the output start time. In "end", specify the output end time.

For begin, specify a time between 00:00:00 on January 1, 1970 and the current year, month, day, and time.

The start time and end time cannot both be omitted. When the start time is omitted, the output range is from the beginning of the file to the specified end time. When the end time is omitted, the output range is from the specified start time to the end of the file.

The begin and end times are expressed in the format `hhmmss[MMDD[YYYY]]`:

hh: Hour (00 ≤ hh ≤ 23); cannot be omitted

mm: Minute (00 ≤ mm ≤ 59); cannot be omitted

ss: Second (00 ≤ ss ≤ 59); cannot be omitted

MM: Month (01 ≤ MM ≤ 12); can be omitted[#]

DD: Day (01 ≤ DD ≤ 31); can be omitted[#]

YYYY: Year (4-digit year from 1970); can be omitted[#]

[#]: If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

When the `-t` option is omitted, all information in the specified file is edited and output.

- `-s server-name~<1-8 alphanumeric characters>`

Specifies the name of a server that is to be subject to editing.

If `sys` is specified in the `-e` option and no server name is specified, the statistics for all servers are output. If `srv` is specified in the `-u` option, it is assumed that a server name has been specified in the `-s` option. If `sys` is specified in the `-e` option and `SYSTEM` as the server name, the statistics for the entire system are output.

- `-s service-name~<1-31 alphanumeric characters>`

Specifies the name of the service whose statistics are to be edited and output. If `trn` is specified in the `-e` option, `svc` in the `-u` option and no service name is specified, the statistics for all services are output. If `svc` is specified in the `-u` option, it is assumed that a service name has been specified in the `-s` option.

Following are the combinations of the `-e`, `-u`, and `-s` options:

-e option	-u option	-s option	Output information	Remarks
sys	--	Server name	System statistics for specified server	None
		SYSTEM	Statistics for entire system	
trn	srv	Server name	Transaction statistics for specified server	
	svc	Service name	Transaction statistics for specified service	

jnlstts (Output uptime statistics)

-e option	-u option	-s option	Output information	Remarks
	Omitted	Server name	Transaction statistics for specified server	
rsp	--	Service name	Statistics about response for specified service	
dly	--	--	All statistics about communication delay	
all	srv	Server name	All statistics for specified server	All data will be output for statistics about response or communication delay.
	svc	Service name	All statistics for specified service	All data will be output for statistics about the system or communication delay.
	Omitted	Server name	All statistics for specified server	All data will be output for statistics about response or communication delay.

--: Cannot be specified

Following are the combinations of options that can be specified by the `jnlstts` command:

Option	-e					-u		-l	-c	-i		-t	-s		
	sys	trn	rsp	dly	all	srv	svc			h	m		Server	Service	
-e	sys	--	N	N	N	N	N	Y	Y	Y	Y	Y	Y	N	
	trn	N	--	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	
	rsp	N	N	--	N	N	N	N	Y	Y	Y	Y	N	Y	
	dly	N	N	N	--	N	N	N	Y	Y	Y	Y	N	N	
	all	N	N	N	N	--	Y	Y	Y	N	Y	Y	Y	Y	
-u	srv	N	Y	N	N	Y	--	N	Y	Y	Y	Y	Y	N ^{#1}	
	svc	N	Y	N	N	Y	N	--	Y	Y	Y	Y	N ^{#2}	Y	
-l		Y	Y	Y	Y	Y	Y	--	N	Y	Y	Y	Y	Y	
-c		Y	Y	Y	Y	N	Y	Y	N	--	Y	Y	Y	Y	
-i	h	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	N	Y	Y	
	m	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	--	Y	Y	
-t		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	--	Y	
-s	Server	Y	Y	N	N	Y	Y	N ^{#2}	Y	Y	Y	Y	Y	--	N
	Service	N	Y	Y	N	Y	N ^{#1}	Y	Y	Y	Y	Y	Y	N	--

#1: If `svc` is specified in the `-u` option, it is assumed that a service name has been specified in the `-s` option.

#2: If `srv` is specified in the `-u` option, it is assumed that a server name has been specified in the `-s` option.

Command arguments

- `file-name~<path name>`

Specifies the name of an unload journal file or global archive unload journal file.

When the multinode facility is used, the name of the file sorted and merged by the `jnlstts` command must be specified.

When this command argument is omitted, the standard input is assumed.

Output format

- (1) *System statistics*

The output format is as follows:

```

jnlstts vv-rr          ***** System Statistics *****          Page: yyyy
File name              :aa...aa
File creation date/time :bb...bb
Output specification date/time :cc...cc ~ dd...dd
-----
Node identifier        = ee...ee
Server name           = fffffff
-----
          ***** Editing period          gg...gg ~ hh...hh *****
Event                Event count  Maximum  Minimum  Average  (unit)
(acquired value)      value        value    value
<RPC service information>
RPC call (response time)  xx...xx    xx...xx    xx...xx    xx...xx (microseconds)
User service execution (execution time) xx...xx    xx...xx    xx...xx    xx...xx (microseconds)
RPC timeout              xx...xx    -          -          -
RPC error                xx...xx    -          -          -
<Schedule service information>
Waiting to be scheduled  xx...xx    xx...xx    xx...xx    xx...xx (events)
Schedule <message size>  xx...xx    xx...xx    xx...xx    xx...xx (bytes)
Size of buffer being used xx...xx    xx...xx    xx...xx    xx...xx (bytes)
:                      :          :          :          :
    
```

The output format is as follows when the -c option is specified (CSV format):

```

"jnlstts vv-rr          ***** System Statistics *****"
"File name              : aa...aa"
"File creation date/time : bb...bb"
"Output specification date/time : cc...cc ~ dd...dd"
"Node identifier", "Server name", "Edit start time", "Edit end time", "Statistics type",
"Event (acquired value)", "Event count", "Maximum value", "Minimum value", "Average value", "(unit)"
ee...ee,ffffff,gg...gg,hh...hh,"RPC service information", "RPC call (response time)"
",xx...xx,xx...xx,xx...xx,xx...xx,"(microseconds)"
ee...ee,ffffff,gg...gg,hh...hh,"RPC service information", "user service execution (execution time)",
xx...xx,xx...xx,xx...xx,xx...xx,"(microseconds)""
ee...ee,ffffff,gg...gg,hh...hh,"RPC service information", "RPC timeout",xx...xx,"-",
"-","-"
ee...ee,ffffff,gg...gg,hh...hh,"RPC service information", "RPC error",xx...xx,"-","-","-",
ee...ee,ffffff,gg...gg,hh...hh,"schedule service information", "waiting to be scheduled",
xx...xx,xx...xx,xx...xx,xx...xx,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,"schedule service information",
"schedule (message size)",xx...xx,xx...xx,xx...xx,xx...xx,"(bytes)"
ee...ee,ffffff,gg...gg,hh...hh,"schedule service information",
"Size of buffer being used",xx...xx,xx...xx,xx...xx,xx...xx,"(bytes)"
:
    
```

1

Statistics about each collected component are edited and displayed for each

specified server name.

2 through 8

Displayed on a single line each.

aa...aa

File name

bb...bb

File creation date and time

cc...cc

Start time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

dd...dd

End time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

ee...ee

Node identifier

ffffff

Server name

When displayed, SYSTEM means the entire system statistics.

gg...gg

Edit start time for system statistics

hh...hh

Edit end time for system statistics

xx...xx

Statistics (decimal number of 10 digits or less. If the number of events or the average value overflows, * is displayed. If there is no edited data for the maximum value, the minimum value, or the average value, - is displayed.)

vv-rr

Version number-revision number

yyyy

Page

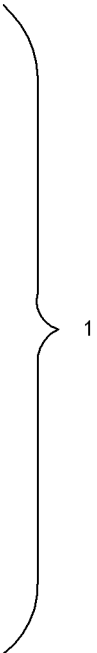
■ (2) Transaction statistics (editing in units of service names)

When `srv` is specified in the `-u` option or the `-u` option is omitted, statistics are edited in units of server names. When `svc` is specified in the `-u` option, statistics are edited in units of service names.

The output format is as follows:

```

jnlstts vv-rr      ***** Transaction Statistics *****   Page: yyyy
File name          :aa...aa
File creation date/time      :bb...bb
Output specification date/time :cc...cc ~ dd...dd
-----
Node identifier    = ee...ee
User server name#1 = fffffff
-----
          ***** Editing period      gg...gg ~ hh...hh *****
<Service name#2:iiiiiii>
          Average value   Maximum value   Minimum value (unit)
Branch execution time      xxxxxxxxxxx   xxxxxxxxxxx   xxxxxxxxxxx (microseconds)
Synchronization point branch
processing execution time  xxxxxxxxxxx   xxxxxxxxxxx   xxxxxxxxxxx (microseconds)
-----
Branch determination
method                    'c' (Commits)                :xxxxxxxxxxx (events)
                          'r' (Rollbacks)                :xxxxxxxxxxx (events)
                          Total                            :xxxxxxxxxxx (events)
-----
Main and child branch
determination method     'c' (Commits)                :xxxxxxxxxxx (events)
                          'r' (Rollbacks)                :xxxxxxxxxxx (events)
                          'hc' (Commit settlements by command) :xxxxxxxxxxx (events)
                          'hr' (Rollback settlements by command) :xxxxxxxxxxx (events)
                          'hh' (Hazard settlements by command)  :xxxxxxxxxxx (events)
                          'hm' (Mixed settlements by command)   :xxxxxxxxxxx (events)
                          Total                            :xxxxxxxxxxx (events)
-----
Branch determination
process type              'u' (User server process)    :xxxxxxxxxxx (events)
                          'r' (Recovery process)            :xxxxxxxxxxx (events)
                          Total                            :xxxxxxxxxxx (events)
    
```



#1: When `svc` is specified in the `-u` option, a service name is output.

#2: When `svc` is specified in the `-u` option, a user server name is output.

The output format is as follows when the `-c` option is specified (CSV format):


```

"jnlstts vv-rr          ***** Transaction Statistics *****
"File name              : aa...aa"
"File creation date/time : bb...bb"
"Output specification date/time : cc...cc ~ dd...dd"
"Node identifier", "User server name", "Edit start time", "Edit end time", "Service name", "Event",
"Event count", "Maximum value", "Minimum value", "Average value", "(unit)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch execution time",-,"xx...xx,xx...xx,xx...xx,"(microseconds)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch synchronization point processing execution time",-,"xx...xx,xx...xx,xx...xx,"(microseconds)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch determination: 'c'(Commit)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch determination: 'r'(Rollback)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch determination: Total",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'c'(Commits)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'r'(Rollbacks)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'hc'(Commit settlements by command)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'hr'(Rollback settlements by command)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'hh'(Hazard settlements by command)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination: 'hm'(Mixed settlements by command)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Main & child branch determination method Total",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch decision process type 'u'(User server process)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch decision process type 'r'(Recovery process)",xx...xx,-,-,-,-,"(events)"
ee...ee,ffffff,gg...gg,hh...hh,iiiiiii,
"Branch decision process type Total",xx...xx,-,-,-,-,"(events)"

```

1

This information is displayed for each service name.

When `svc` is specified in the `-u` option, this information is repeatedly displayed for each server name.

2 through 17

Displayed on a single line each.

aa...aa

File name

bb...bb

File creation date and time

cc...cc

start time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

dd...dd

End time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

ee...ee

Node identifier

ffffff

User server name

When `SVC` is specified in the `-u` option, a service name is output.

gg...gg

Edit start time for transaction statistics

hh...hh

Edit end time for transaction statistics

iiiiiii

Service name (if there is no service name, * is displayed)

When `SVC` is specified in the `-u` option, a user server name is output.

xx...xx

Statistics (displayed by counting fractions of .5 and over of the average as a unit and disregarding the rest. If the maximum, minimum, or average overflows, * is set.)

vv-rr

Version number-revision number

yyyy

Page

■ (3) *Statistics about response*

The output format is as follows:

```

jnlstts  vv-rr  ***** Response Time Statistical Information ***** Page: yyyy

File name: aa...aa
File creation date and time: bb...bb
Output specification date and time from cc...cc to dd...dd
-----
Node identifier      = ee...ee
Service group name = fffffff

***** Editing object time from gg...gg to hh...hh *****
< service name : ii...ii >
RPC type
  Reply              : xxxxxxxx (events)
  No-reply           : xxxxxxxx (events)
  No-wait            : xxxxxxxx (events)
  Chained            : xxxxxxxx (events)
  Total              : xxxxxxxx (events)
-----
Event                Number of  Maximum  Minimum  Average
(collected value)   events    value    value    value
                    (times)
Response time        xx...xx   xx...xx  xx...xx  xx...xx
                    (microsec.)
Execution of user    xx...xx   xx...xx  xx...xx  xx...xx
service (execution time)
                    (microsec.)
Service wait         xx...xx   xx...xx  xx...xx  xx...xx
                    (microsec.)
-----

```

The output format is as follows when the -c option is specified (CSV format):

```

"jnlstts vv-rr                ***** Response Statistics *****"
"File name : aa...aa"
"File creation date/time : bb...bb"
"Output specification date/time : cc...cc ~ dd...dd"
"Node identifier", "Service group name", "Edit start time", "Edit end time", "Service name", "Event
(acquired value)", "Event count", "Maximum value", "Minimum value", "Average value", "(unit)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"RPC type Synchronous response type",xx...xx,"-","-","-","(events)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"RPC type Non-response type",xx...xx,"-","-","-","(events)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"RPC type Asynchronous response type",xx...xx,"-","-","-
","(events)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"RPC type Chained",xx...xx,"-","-","-","(events)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"RPC type Total",xx...xx,"-","-","-","(events)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"Response time",xx...xx,xx...xx,xx...xx,xx...
xx,"(microseconds)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"Service execution time",xx...xx,xx...xx,xx...xx,xx...
xx,"(microseconds)"
ee...ee,ffffff,gg...gg,hh...hh,ii...ii,"Service wait time",xx...xx,xx...xx,xx...xx,xx...
xx,"(microseconds)"

```

#: The service execution time and the service wait time of the MHP are not acquired.
/

This information is displayed for each service name.

2 through 6

Displayed on a single line each.

aa...aa

File name

bb...bb

File creation date and time

cc...cc

Start time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

dd...dd

End time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

ee...ee

Node identifier

ff...ff

Service group name

gg...gg

Edit start time for statistics about response

hh...hh

Edit end time for statistics about response

ii...ii

Service name

xx...xx

Statistics (decimal number of 10 digits or less. The average is determined by counting fractions of .5 and over as a unit and disregarding fractions under. If the maximum, minimum, or average overflows, * is set. If there is no edited data, - is set.)

vv-rr

Version number-revision number

yyyy

Page

■ (4) Statistics about communication delay

The output format is as follows:

```

jnlstts vv-rr      ***** Delay Time Statistical Information ***** Page: yyyy
File name: aa...aa
File creation date and time: bb...bb
Output specification date and time from cc...cc to dd...dd
-----
Node identifier           = ee...ee
Remote node identifier = ff...ff
-----
***** Editing object time from gg...gg to hh...hh *****
Event                    Number of   Maximum   Minimum   Average
(collected value)       events     value     value     value
                        (times)
Delay time                xx...xx   xx...xx   xx...xx   xx...xx
                        (microsec.)
-----

```

The output format is as follows when the `-c` option is specified (CSV format):

```

"jnlstts vv-rr      ***** Communication Delay Time Statistics *****"
"File name : aa...aa"
"File creation date/time : bb...bb"
"Output specification date/time : cc...cc ~ dd...dd"
"Node identifier", "Send destination node identifier", "Edit start time", "Edit end time", "Event
(acquired value)", "Event count", "Maximum value", "Minimum value", "Average value", "(unit)"
ee...ee,ff...ff,gg...gg,hh...hh,"Communication delay time",xx...xx,xx...xx,xx...xx,xx...xx,
"(microseconds)"
ee...ee,ff...ff,gg...gg,hh...hh,"Communication delay time",xx...xx,xx...xx,xx...xx,xx...xx,
"(microseconds)"

```

1

This information is displayed for each send destination node identifier.

2 through 4

Displayed on a single line each.

aa...aa

File name

bb...bb

File creation date and time

cc...cc

Start time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

dd...dd

End time specified with the `-t` option (when the `-t` option is omitted, * is displayed)

ee...ee

Node identifier

ff...ff

Send destination node identifier

gg...gg

Edit start time for statistics about communication delay

hh...hh

Edit end time for statistics about communication delay

xx...xx

Statistics (decimal number of 10 digits or less. The average is determined by counting fractions of .5 and over as a unit and disregarding fractions under. If the maximum, minimum, or average overflows, * is set.)

vv-rr

Version number-revision number

yyy

Page

Output messages

Message ID	Description	Output destination
KFCA02600-E	An error occurred while the unload journal file was being accessed.	Standard error output
KFCA02601-E	The unload journal file is invalid.	Standard error output
KFCA02602-E	The journal block is invalid.	Standard error output
KFCA02603-E	The journal record is invalid.	Standard error output
KFCA02604-E	Memory cannot be allocated.	Standard error output
KFCA02660-I	Help message	Standard output, standard error output
KFCA02661-E	A parameter in the <code>jnlstts</code> command is invalid or its value exceeds the limit.	Standard error output

Message ID	Description	Output destination
KFCA02662-E	The combination of options in the <code>jnlstts</code> command is invalid.	Standard error output
KFCA02663-E	An option in the <code>jnlstts</code> command is specified incorrectly.	Standard error output
KFCA02664-E	A file name in the <code>jnlstts</code> command is specified incorrectly.	Standard error output
KFCA02665-E	Statistics cannot be output because the value specified for the <code>-l</code> option in the <code>jnlstts</code> command is too small.	Standard error output

Notes

- If the output range of uptime statistics is specified and no journal record corresponds to the specified begin or end time, the journal records between the begin and end positions are to be output.
- If the uptime statistics output range is specified in terms of time and the start time value is greater than the end time value, the month and day or the month, day, and year must be specified so that the end time is after the start time. If the requisite information is omitted, an error occurs. If the date is omitted, the date the `jnlstts` command is entered is assumed; therefore, if the editing range does not fit within the current day, either the month and day or the month, day, and year must be specified.
- The time at which editing and output of uptime statistics begins depends on whether or not a start time is specified with the `-t` option. If a start time is specified with the `-t` option, output begins within the time frame that begins at the specified start time. If no start time is specified, output begins within the time frame that begins when the first SJ record for a subject file is created. A time frame is a span of time obtained by dividing the time into intervals of the interval length specified with the `-i` option on the basis of the start time (hh: 00).

The following shows the relationships between the specifications of the `-t` and `-i` options of the `jnlstts` command and the time at which uptime statistics (SJ) are output to the unload journal file:

	Start time specified with <code>-t</code> option	Time of SJ output to unload journal file	Interval specified with <code>-i</code> option	Output start time frame	Remarks
Example 1	09:00:00	09:10:00	10 minutes	9:00-9:10	The time frame corresponding to 9:00-9:10 is from 09:00:00 to 09:09:59, but no SJ is included in this time frame; therefore, SJ will be output in the next time frame.

	Start time specified with -t option	Time of SJ output to unload journal file	Interval specified with -i option	Output start time frame	Remarks
Example 2	09:00:00	08:50:00	7 minutes	9:00-9:07	SJ was output before the start time specified with the -t option; the SJ is therefore not subject to output and no SJ can be output by executing the jnlstts command.
Example 3	Not specified	09:05:34	4 minutes	9:04-9:08	SJ from 09:04:00 to 09:07:59 is output.
Example 4	Not specified	09:37:00	30 minutes	9:30-10:00	SJ from 09:30:00 to 09:59:59 is output.
Example 5	09:29:59	09:30:00	30 minutes	9:00-9:30	The time frame corresponding to 9:00-9:30 is from 09:00:00 to 09:29:59, but no SJ is included in this time frame; therefore, SJ will be output in the next time frame.

- If an end time for uptime statistics output is specified, information is output up to the time frame that includes the specified end time. For example, if 09:00:00 is specified as the end time with the -t option and 30 minutes is specified as the interval with the -i option, the final time frame is 9:00-9:30, but the uptime statistics at 09:00:00 are actually output.
- In the system statistics, the first decimal place is rounded down for the edited average value.
- In the transaction statistics, the first decimal place is rounded off for the edited average value.
- Even if an overflow occurs in the number of events or the average value and * is displayed, the maximum and minimum values are guaranteed.
- If there is no data in a specified collection and editing time frame, "No applicable data" is displayed for that time frame.
- When no file name is specified, the standard input is assumed. Therefore, the input file must be specified by piping, redirecting, etc.
- In the response statistics, if statistics about response time collection failure are only collected, zeros are output for all data.
- For the -t option, if you specify a value beyond 2038 as the year of the output end time, in some cases CPU load and memory usage may increase to the extent

that it takes some time for the command to terminate.

- If, for example, the user clock is reset backward by the OS's time synchronization function, and the data for the acquired edit time period shows negative values, '*' is output for the maximum value, the minimum value, and the average value for the time period.
- The memory requirements for the `jnlstts` command vary depending on the operation statistics items to be edited, the output range, and the time interval. Use the following formulas to estimate the memory requirements (in bytes):

System statistics memory requirement

$$460000 + (16 + 8 \times \uparrow(\text{number-of-node-identifiers} / 128) \uparrow \times 128) \\ + \text{maximum-record-length}^{\#1} \\ + (\uparrow((24 + 204 \times \text{server-count}) / 4096) \uparrow \times 4096) \\ + ((6640 \times \text{time-interval-count}^{\#2}) \times \text{server-count}) \times 6092$$

Transaction statistics memory requirement

$$460000 + (16 + 8 \times \uparrow(\text{number-of-node-identifiers} / 128) \uparrow \times 128) \\ + \text{maximum-record-length}^{\#1} \\ + (\uparrow((24 + 204 \times \text{server-count}^{\#3}) / 4096) \uparrow \times 4096) \\ + \sum_{i=1}^A (128 + 176 \times \text{service-count-at-each-server}^{\#4} \times \text{time-interval-count}^{\#2})$$

Response statistics memory requirement

$$460000 + (16 + 8 \times \uparrow(\text{number-of-node-identifiers} / 128) \uparrow \times 128) \\ + \text{maximum-record-length}^{\#1} \\ + (\uparrow((24 + 224 \times \text{service-group-count}) / 4096) \uparrow \times 4096) \\ + \sum_{i=1}^B (128 + 192 \times \text{service-count-at-each-server} \times \text{time-interval-count}^{\#2})$$

Communication delay statistics memory requirement

$$460000 + (16 + 8 \times \uparrow(\text{number-of-node-identifiers} / 128) \uparrow \times 128) \\ + \text{maximum-record-length}^{\#1} \\ + (\uparrow((24 + 200 \times \text{number-of-send-destination-node-identifiers}) / 4096) \uparrow \times 4096) \\ + ((122 \times \text{time-interval-count}^{\#2}) \times \text{number-of-send-destination-node-identifiers})$$

Legend:

↑ ↑: Decimal values are truncated.

A: Server count. Service count when `svc` is specified for the `-u` option.

jnlstts (Output uptime statistics)

B: Service group count

#1

Value specified in the `jnl_max_datasize` operand of the system journal service definition.

#2

Value obtained by dividing the time period from the edit start time to the edit termination time, specified by `-t`, by the time interval specified by the `-i` option. Time periods containing no data are not counted.

#3

Service count when `svc` is specified for the `-u` option.

#4

Server count of each service when `svc` is specified for the `-u` option.

Examples

In this example, the command edits system statistics from the beginning of an unload journal file up to 17:30:00.999999 on May 29, 1993, and outputs them every 30 minutes:

Unload journal file: `sysjnl001`

```
jnlstts -e sys -i m30 -t ,17300005291993  
/OpenTP1/jnl/sysjnl001
```

In this example, the last time period is from 17:30 to 18:00. However, system statistics from 17:30:00.0 on May 29, 1993 up to 17:30:00.999999 on May 29, 1993 are output.

jnlswpfg (Swap journal files)

Format

```
jnlswpfg -j sys|jar [-r resource-group-name]
```

Function

The `jnlswpfg` command switches immediately the current file group. If there is no file group to be swapped in, a command error occurs.

The `jnlswpfg` command can be entered only when OpenTP1 and the journal service are both operating.

Options

- `-j sys|jar`
Swaps file groups.
`sys`: System journal file
`jar`: Archive journal file
- `-r resource-group-name~<1-8 character identifier>`

If `jar` is specified in the `-j` option, the name of the resource group in the file to be swapped must be specified. It must be the definition file name specified in the archive journal service definition specified in the definition command `jnldfs -a` of the global archive journal service definition.

When this option is omitted, the name of an online archive journal file defined at the beginning of the definition command `jnldfs -a` of the archive journal service definition is assumed.

If other than `jar` is specified in the `-j` option, specification of the `-r` option is discarded.

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output
KFCA01280-E	<code>jnlswpfg</code> command processing failed.	Standard error output
KFCA01281-E	<code>jnlswpfg</code> command input format is invalid.	Standard error output
KFCA01285-E	Swap processing is underway.	Standard error output

jnlswpfg (Swap journal files)

Message ID	Description	Output destination
KFCA01286-E	There is no file to be swapped in.	Standard error output
KFCA01290-I	Help message	Standard output, standard error output

jnlunlfg (Unload journal files)

Format

```
jnlunlfg -j sys|jar [[-f] | [-t start-time [, end-time]]]
                [-r resource-group-name] [-R run-ID]
                [-g file-group-name] [-o output-destination-file-name]
                [-d service-definition-name] [-n]
```

Function

The `jnlunlfg` command unloads a specified file group into a specified output destination file or the standard output and places the file group on unloaded status.

When the KFCA01222-I message is output, the user must execute the `jnlunlfg` command to unload the file group.

The current file group or an unloaded file group cannot be unloaded unless the `-f` option is specified.

When unloading the archive journal, the unload information is output after the unloading is completed.

Options

- `-j sys|jar`

Unloads a file group.

`sys`: System journal file

`jar`: Archive journal file

- `-f`

Unloads a file group without checking its status. Accordingly, the current file group or an unloaded file group can be unloaded, however, its status does not change.

When a file group on swappable status (that is, the file has been unloaded and is on overwritable status) is unloaded with the `-f` option specified, it may become the current file group during unloading. If it is unloaded before collecting journals in the online mode, an error is detected and a command error occurs. The user must close an unloaded file group with the `jnlclsfg` command before unloading it with the `-f` option specified.

The journal file unloaded in the running OpenTP1 system is not inherited by the standby OpenTP1 system. The standby system can unload the journal file by executing the `jnlunlfg` command with its `-f` option specified. Note that the `jnlunlfg` command must be executed only when the unloaded journal is not used by another

OpenTP1.

- `-t [start-time] [, end-time]`

Specifies the period of time for which the journal block is to be unloaded.

For *start-time*, specify a time and date to start unloading the target journal block. For *end-time*, specify a time and date to end unloading the target journal block.

For *start-time* and *end-time*, you can specify a time and date after 00:00:00 on January 1, 1970.

You must specify either *start-time* or *end-time*, or both. If you omit *start-time*, unloading is performed from the beginning of the file until the specified end time. If you omit *end-time*, unloading is performed starting from the specified start time until the end of the journal file.

Specify *start-time* and *end-time* in the *hhmmss[MMDD[YYYY]]* format.

hh: Hour (00 ≤ *hh* ≤ 23)

You cannot omit the specification.

mm: Minutes (00 ≤ *mm* ≤ 59)

You cannot omit the specification.

ss: Seconds (00 ≤ *ss* ≤ 59)

You cannot omit the specification.

MM: Month (01 ≤ *MM* ≤ 12)

You can omit the specification.[#]

DD: Day (01 ≤ *DD* ≤ 31)

You can omit the specification.[#]

YYYY: Year (4-digit year from 1970)

You can omit the specification.[#]

[#]: The specifiable period depends on the time zone set for the system. For example, when `JST-9` is set for the `TZ` environment variable, the earliest time you can specify starts at 9:00:00 on January 1, 1970. If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If you omit the `-t` option, all the information in the specified file is unloaded.

When you specify the `-t` option, you cannot specify the `-f` option. Note that the status of the file group does not change when you specify the `-t` option for unloading.

When you specify the `-t` option and the `-R` option, and omit the `-g` option, the range for unloading is the one specified by using the `-t` option in the file group of the oldest generation among the file groups of the run ID specified by using the `-R` option.

When you specify the `-t` option, and omit the `-R` and the `-g` options, the range for unloading is the one specified by using the `-t` option in the file group of the oldest generation among the file groups of the latest run ID.

- `-r resource-group-name~<1-8 character identifier>`

Specifies the name of a resource group of the file to be unloaded.

If `sys` is specified in the `-j` option, the definition file name in the system journal service definition must be specified. If `jar` is specified, the name of the definition file in the archive journal service definition to be unloaded must be specified.

When this option is omitted, the system assumes the resource group name specified in the journal service definition of `$DCCONFPATH/jnl` if `sys` is specified in the `-j` option. With `jar` specified in the `-j` option, the system assumes the file name defined at the beginning of the definition command `jnldefsv -a` of the global archive journal service definition of `$DCCONFPATH/jar`.

- `-R run-ID`

Specifies the run-ID of the journal file group to be unloaded. When this option is omitted, the specification of the `-g` option predominates.

- `-g file-group-name~<identifier, 1-8 characters>`

Specifies the file group name of the journal files to be unloaded.

When this option is omitted, the oldest generation of the file groups waiting to be unloaded will be unloaded. The `-f` option cannot be specified.

The following shows the specification combinations with the `-R` option and the journal file group that will be unloaded.

Option		-R	
		Specified	Not specified
-g	Specified	The file group specified by <code>-g</code> is unloaded. A command error occurs if the run ID specified by <code>-R</code> does not match that specified by <code>-g</code> .	The file group specified by <code>-g</code> is unloaded.
	Not specified	Of the file groups with the run ID specified by <code>-R</code> , the file group of the oldest generation is unloaded.	Of the file groups with the latest run ID, the file group of the oldest generation is unloaded.

- -o *output-destination-file-name*~<path name>

Specifies the name of the output destination file.

When this option is omitted, output to the standard output is assumed.

- -d *service-definition-name*~<path name> or <file name>

Specifies the name of the system journal service definition file that defines the file group to be unloaded, in one of the following formats:

- Full path name beginning with / (root)
- Definition file name only

If only the definition file name is specified, the directory containing the definition file must be set in \$DCCONFPATH.

When this option is omitted, the specification of the -r option predominates.

- -n

Specifies that unload wait status is to be maintained after unloading without changing the status of the file group.

This option enables the same generation to be unloaded repeatedly, thereby creating multiple unload files.

When the standby OpenTP1 system is operating under the system switchover facility, a file group can be unloaded with the `jnlunlfg` command with its `-n` option specified. The status of the unloaded file group remains unchanged. This requires the running OpenTP1 system to set the file group to unloaded status by executing the `jnlchgfg` command. Thus, the load of unloading on the running OpenTP1 system can be dispersed onto the standby OpenTP1 system.

Output format

The output format is as follows when the archive journal file is unloaded:

Node identifier	Type	Resource group name	Run ID
<i>aa...aa</i>	<i>bbb</i>	<i>ccccccc</i>	<i>ddddddd</i>
	First block information	Generation no.	Block no. Collection time
		<i>eeeeeee</i>	<i>ffffff</i> <i>gg..gg</i>
	Last block information	Generation no.	Block no. Collection time
		<i>eeeeeee</i>	<i>ffffff</i> <i>gg..gg</i>

This information is displayed for each run ID in the file group.

aa...a

Node identifier (4 characters)

bbb

File type

sys: System journal file

ccccccc

Resource group name

ddddddd

Run ID (0x + 6-digit hexadecimal number).

eeeeeee

Generation number (0x + 6-digit hexadecimal number).

ffffff

Block number (0x + 6-digit hexadecimal number).

gg...gg

Journal block collection time (displayed in the format *year-month-day hh:mm:dd*)

Output messages

Message ID	Description	Output destination
KFCA01141-E	The command cannot be executed because OpenTP1 is running in journal fileless mode.	Standard error output

Notes

- The `jnlunlfg` command must not be executed during journal service startup processing, which is from the time the KFCA01100-I message is issued until the time the KFCA01102-I message is issued in the case of normal start, or from the time of KFCA01101-I message issuance until the time of KFCA01102-I message issuance in the case of restart. If the command is executed while such startup processing is underway, the status of the system journal files may not be incorporated in OpenTP1. When this happens, the status of the system journal files can be incorporated into OpenTP1 by closing the corresponding file group with the `jnlclsfg` command after completion of restart processing and then opening the file group with the `jnlpnfg` command.
- The `jnlunlfg` command is executed with referencing the definition file. It cannot be correctly unloaded unless the definition file at journal collection in the online mode coincides with the definition file at execution of the `jnlunlfg` command. If they mismatch, specify the `-d` option if the backup target directory is not `$DCCONFPATH`, and the `-r` option if the file name is changed. In this case,

however, the definition at journal collection in the online mode must be backed up in the system journal service definition or the archive journal service definition.

- Do not use the following commands for the unloaded files that are output by specifying the `-t` option:
 - `damfrc`
 - `tamfrc`
 - `jnlcolc`
 - `jnlrput` (unavailable only when the `-e` option is specified)

lckls (Display lock information)

Format

```
lckls { [-a] [server-name] | [-r resource-name] }
```

Function

The `lckls` command outputs lock information to the standard output.

Options

- `-a`

Specifies that occupancy and wait information about a lock resource is to be displayed.

When this option is omitted, only wait information is displayed.

- `-r resource-name`

Specifies the resource whose occupancy and wait information is to be displayed. When you specify this option, OpenTP1 attempts to find a forward match for the specified character string (resource name) and displays the information about all the matching resources.

Command arguments

- `server-name~<identifier, 1-8 characters>`

Specifies the name of a server for which information is to be output to the standard output. OpenTP1 attempts to find a forward match for the specified character string (user server name) and outputs the information about all the matching user servers to the standard output.

When this command argument is omitted, information about all user servers related to the lock information is displayed.

Output format

Server name	PID	Resource name	Mode	Request type	Wait time	D priority	W priority	...
aa...aa	bbbb	cc...cc(ddd) . . . ee		ffffff	ggggg	hhh	iii	...1.
								...2.

#1. and #2. are displayed on one line. If there are two or more pieces of lock or wait information, line 2. is displayed repeatedly by the number of the pieces.

aa...aa

Server name (up to 8 characters)

lckls (Display lock information)

bbbb

Process ID (decimal)

cc...cc

Resource name (1-16 characters)

ddd

Server ID (3 characters)

ee

Lock control mode (2 characters)

EX: Referencing or updating other UAPs is not permitted.

PR: Only referencing other UAPs is permitted.

ffffff

Request type (1-7 characters)

MIGRATE: TAM service lock request or user lock request

BRANCH: DAM service lock request

ggggg

Wait time (decimal, in seconds)

(***** is displayed for the occupancy information)

hhh

Deadlock priority value (decimal)

iii

Lock wait priority value (decimal)

Output messages

Message ID	Description	Output destination
KFCA00430-E	lckls command cannot be executed.	Standard error output
KFCA00431-I	Help message	Standard output, standard error output
KFCA00433-E	No lckls command execution environment has been set up.	Standard error output
KFCA00434-E	lckls command cannot be executed due to incompatible versions.	Standard error output
KFCA00437-E	lckls command format is invalid.	Standard error output

lckpool (Display lock control table pool information)

Format

```
lckpool
```

Function

The `lckpool` command outputs the utilization factor of the lock control table pool to the standard output.

Output format

Definition name	Defined value	Used value	Utilization factor
<code>lck_limit_forusr</code>	<i>aaaaa</i>	<i>bbbbbb</i>	<i>ccc%</i>
<code>lck_limit_fordam</code>	<i>dddddd</i>	<i>eeeeee</i>	<i>fff%</i>
<code>lck_limit_fortam</code>	<i>ggggg</i>	<i>hhhhh</i>	<i>iii%</i>
<code>lck_limit_formqa</code>	<i>jjjjj</i>	<i>kkkkk</i>	<i>lll%</i>

aaaaa

Maximum number of concurrent lock requests defined for the user server (decimal)

bbbbbb

User server's current number of lock requests (decimal)

ccc

User server's current lock control table pool utilization factor (decimal)

dddddd

Maximum number of concurrent lock requests defined for DAM service (decimal)

eeeeee

DAM service's current number of lock requests (decimal)

fff

DAM service's current lock control table pool utilization factor (decimal)

ggggg

Maximum number of concurrent lock requests defined for TAM service (decimal)

hhhhh

lckpool (Display lock control table pool information)

TAM service's current number of lock requests (decimal)

iii

TAM service's current lock control table pool utilization factor (decimal)

jjjj

Maximum number of concurrent lock requests defined for MQA service (decimal)

kkkkk

MQA service's current number of lock requests (decimal)

lll

MQA service's current lock control table pool utilization factor (decimal)

Output messages

Message ID	Description	Output destination
KFCA00430-E	lckpool command cannot be executed.	Standard error output
KFCA00432-I	Help message	Standard output, standard error output
KFCA00433-E	No lckpool command execution environment has been set up.	Standard error output
KFCA00434-E	lckpool command cannot be executed due to incompatible versions.	Standard error output
KFCA00437-E	lckpool command format is invalid.	Standard error output

lckrminf (Delete deadlock information file and timeout information file)

Format

```
lckrminf -d number-of-days
```

Function

The `lckrminf` command deletes deadlock information files and timeout information files.

Options

- `-d number-of-days~((1-24855))`

Deletes deadlock information files and timeout information files that were created before "24 hours x (number of days specified in the `lckrminf` command)", counted from the time when the `lckrminf` command was executed.

Output messages

Message ID	Description	Output destination
KFCA00430-E	<code>lckrminf</code> command cannot be executed.	Standard error output
KFCA00433-E	No <code>lckrminf</code> command execution environment has been set up.	Standard error output
KFCA00434-E	<code>lckrminf</code> command cannot be executed due to incompatible versions.	Standard error output
KFCA00437-E	<code>lckrminf</code> command format is invalid.	Standard error output
KFCA00439-I	Help message	Standard output, standard error output

logcat (Display message log file)

Format

```
logcat [-niNIdtHpab] [-f message-log-file-name]
```

Function

The `logcat` command outputs the message contents of a message log file to the standard output.

The output operation merges `$DCDIR/spool/dcllog1` and `$DCDIR/spool/dcllog2` in chronological order.

The information to be displayed is specified with options.

Options

- `-n`
Display the message serial numbers.
- `-i`
Display the ID of each request source process.
- `-N`
Display the message log output serial numbers within each process.
- `-I`
Display the system ID (the user-specified OpenTP1 identifier).
- `-d`
Display the date of the message log output request.
- `-t`
Display the time of the message log output request.
- `-H`
Display the name of the message log output request source host.
- `-p`
Display the request source program ID.
- `-a`
Display all of the above. If this option is specified together with any of the options

above, the specification of the other options is ignored.

■ **-b**

Display only the message IDs and message texts without any of the above information. If this option is specified together with any other options, the specification of the other options is ignored. If this option is specified together with the **-a** option, the **-a** option is effective, and the **-b** option is ignored.

■ **-f *message-log-file-name*~<path name>**

Specifies the name of the message log file.

Information from the specified message log file only is displayed.

When this option is omitted, `$DCDIR/spool/dcllog1` and `$DCDIR/spool/dcllog2` are merged, and the information is displayed in chronological order.

Output format

The following shows the output format when "logcat -a" is specified:

```
aaaaaaa bbbbb cccccc dd ee...ee ff..ff gg..gg hhh ii..ii jj..jj
```

aaaaaaa

Message serial number (7 numeric characters)

bbbbbb

Process ID (5 numeric characters)

ccccc

Message serial number within the process (7 numeric characters)

dd

System ID (2 alphanumeric characters)

ee...ee

Date (10 numeric characters), in the format year/month/day

ff..ff

Time (8 numeric characters), in the format hour:minute:second

gg..gg

Request source host name (8 alphanumeric characters) (first eight characters of the host name)

hhh

Request source program ID (3 alphanumeric characters) (ID beginning with *

logcat (Display message log file)

indicates a message issued from a UAP; an ID that does not begin with * indicates a system message)

ii...ii

Message ID (11 alphanumeric characters)

jj...jj

Message text

Output messages

Message ID	Description	Output destination
KFCA01950-I	Help message	Standard output, standard error output
KFCA01951-E	Memory is inadequate.	Standard error output
KFCA01952-E	Error occurred in log service.	Standard error output
KFCA01953-E	No log file storage directory has been set up.	Standard error output
KFCA01954-E	Log file open error occurred.	Standard error output
KFCA01955-E	Collection of log file's last update time failed.	Standard error output
KFCA01956-E	I/O error occurred.	Standard error output
KFCA01957-E	Log file format is invalid.	Standard error output
KFCA01958-E	Log file version is invalid.	Standard error output
KFCA01978-E	Specification of a command argument is invalid.	Standard error output

logcon (Switch message log realtime output facility)

Format

```
logcon [-{y|n}]
```

Function

The `logcon` command switches the realtime output specification (`log_msg_console`) for the log service definition in the online mode.

The `logcon` command is applicable only when the log service is operating.

Options

■ `-{y|n}`

`-y`

Specifies that the `log_msg_console` specification is to be switched to Y. When the specification is switched to Y, the message log is output to the standard output in realtime.

`-n`

Specifies that the `log_msg_console` specification is to be switched to N. When the specification is switched to N, the message log is not output to the standard output in realtime.

Output messages

Message ID	Description	Output destination
KFCA01951-E	Memory is inadequate.	Standard error output
KFCA01952-E	Error occurred in log service.	Standard error output
KFCA01970-I	Help message	Standard output, standard error output
KFCA01971-E	Communication error occurred.	Standard error output
KFCA01972-E	Log service has not been started.	Standard error output
KFCA01973-E	Version of <code>logcon</code> command is invalid.	Standard error output
KFCA01974-E	On active status.	Standard error output
KFCA01975-E	On inactive status.	Standard error output
KFCA01976-E	Environment variable has not been set.	Standard error output

logcon (Switch message log realtime output facility)

Message ID	Description	Output destination
KFCA01977-E	Name service has not been started.	Standard error output
KFCA01978-E	Specification of a command argument is invalid.	Standard error output
KFCA01979-E	An option flag is missing, or the combination of option flags is invalid.	Standard error output

mcfactap (Release shutdown of applications)

Format

```
mcfactap -s MCF-communication-process-identifier
          -a application-name [-k application-type]
```

Function

The `mcfactap` command releases applications from shutdown status; the applications' abnormal terminations counts are initialized (set to 0).

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

- `-a application-name<identifier, 1-8 characters>`

Specifies the name of an application to be released from shutdown status.

A maximum of eight application names can be specified. When entering large amounts of inputs for the operation, specify more than one application or specify a group of applications to increase the number of applications to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. The name of a group of applications cannot coexist with the names of applications for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the shutdown status of all applications is to be released.

*character-string**

Specifies leading characters of an application name when all applications with names beginning with the specified character string are to be released from shutdown status.

<Example of specifying more than one application>

To specify `ap11`, `ap12`, and `ap13`, code: `-a "ap11 Δ ap12 Δ ap13 "`

<Example of specifying a group of applications>

To specify all applications beginning with apl, code: -a "apl*"

- -k *application-type*~<<user>>

Specifies the type of application.

user: User application

mcf: MCF event

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfactap command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfactap command input source failed.	Message log file or standard error output
KFCA10371-I	mcfactap command has been accepted normally.	Standard output
KFCA10373-E	mcfactap command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of a remote process failed.	Standard error output
KFCA10386-E	Specified application has not been cataloged.	Standard error output
KFCA10516-I	Help message	Standard output
KFCA11063-W	Shutdown status of a specified application has already been released.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output

Message ID	Description	Output destination
KFCA11099-W	The application cannot be closed or the close cannot be canceled because of an SPP resource.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfacldcap (Initialize abnormal terminations counts for applications)

Format

```
mcfacldcap -s MCF-communication-process-identifier
           -a application-name [-k application-type]
```

Function

The `mcfacldcap` command initializes (to 0) the abnormal terminations count for applications.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

- `-a application-name~<identifier, 1-8 characters>`

Specifies the name of an application whose abnormal terminations count is to be initialized (to 0).

A maximum of eight application names can be specified. When entering large amounts of inputs for the operation, specify more than one application or specify a group of applications to increase the number of applications to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. The name of a group of applications cannot coexist with the names of applications for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the abnormal terminations count is to be initialized for all applications.

*character-string**

Specifies leading characters of an application name when the abnormal terminations count is to be initialized for all applications with names beginning with the specified character string.

<Example of specifying more than one application>

To specify `apl1`, `apl2`, and `apl3`, code: `-a "apl1 Δ apl2 Δ apl3"`

<Example of specifying a group of applications>

To specify all applications beginning with `apl`, code: `-a "apl*"`

■ `-k application-type~<<user>>`

Specifies the type of application.

`user`: User application

`mcf`: MCF event

Output messages

Message ID	Description	Output Destination
KFCA10350-I	<code>mcfac1cap</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcfac1cap</code> command input source failed.	Message log file or standard error output
KFCA10371-I	<code>mcfac1cap</code> command has been accepted normally.	Standard output
KFCA10373-E	<code>mcfac1cap</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10386-E	A specified application has not been cataloged.	Standard error output
KFCA10520-I	Help message	Standard output
KFCA11070-W	Abnormal terminations count of a specified application has already been cleared.	Standard error output
KFCA11801-W	Number of abnormal terminations cannot be initialized due to SPP resources.	Standard error output

mcfacldcap (Initialize abnormal terminations counts for applications)

Message ID	Description	Output Destination
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfadctap (Shut down applications)

Format

```
mcfadctap -s MCF-communication-process-identifier
          -a application-name [-t shutdown-type]
          [-k application-type]
```

Function

The `mcfadctap` command shuts down specified applications.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

- `-a application-name~<identifier, 1-8 characters>`

Specifies the name of an application to be shut down.

A maximum of eight application names can be specified. When entering large amounts of inputs for the operation, specify more than one application or specify a group of applications to increase the number of applications to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. The name of a group of applications cannot coexist with the names of applications for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that all applications are to be shut down.

*character-string**

Specifies leading characters of an application name when all applications with names beginning with the specified character string are to be shut down.

<Example of specifying more than one application>

To specify `ap11`, `ap12`, and `ap13`, code: `-a "ap11 Δap12ap13"`

<Example of specifying a group of applications>

To specify all applications beginning with `apl`, code: `-a "apl*"`

- `-t shutdown-type~<<insc>>`

Specifies the shutdown type for the applications:

`insc`: Shut down input to input queue and input queue scheduling

`in`: Shut down input to input queue only

- `-k application-type~<<user>>`

Specifies the type of application.

`user`: User application

`mcf`: MCF event

Output messages

Message ID	Description	Output destination
KFCA10350-I	<code>mcfadctap</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcfadctap</code> command input source failed.	Message log file or standard error output
KFCA10371-I	<code>mcfadctap</code> command has been accepted normally.	Standard output
KFCA10373-E	<code>mcfadctap</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10386-E	A specified application has not been cataloged.	Standard error output
KFCA10517-I	Help message	Standard output
KFCA11062-W	A specified application has already been shut down.	Standard error output

Message ID	Description	Output destination
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11099-W	The application cannot be closed or the close cannot be canceled because of an SPP resource.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfadltap (Delete timer activation requests for applications)

Format

```
mcfadltap [-s Application-startup-process-identifier]
          -a application-name
```

Function

The `mcfadltap` command deletes timer requests for specified applications and terminates startup of the applications. Timer requests for the `ans` or `cont` type applications cannot be deleted.

Options

- `-s application-startup-process-identifier~<numerals (0-9), a-f>((01-ef))`

Specifies an application startup process identifier. No more than one application startup process identifier can be specified.

If this option is omitted, the `mcfadltap` command is executed on all MCFs. This means that the overhead required to search for the application startup processes is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of application startup processes or if you want to use many operation commands, use the `-s` option to specify an Application startup process identifier.

- `-a application-name~<identifier, 1-8 characters>`

Specifies the name of an application whose startup is to be terminated.

A maximum of eight application names can be specified. When entering large amounts of inputs for the operation, specify more than one application or specify a group of applications to increase the number of applications to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. The name of a group of applications cannot coexist with the names of applications for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that startup of all applications is to be terminated.

*character-string**

Specifies leading characters of an application name when startup of all applications with names beginning with the specified character string is to be terminated.

<Example of specifying more than one application>

To specify `apl1`, `apl2`, and `apl3`, code: `-a "apl1 Δ apl2 Δ apl3"`

<Example of specifying a group of applications>

To specify all applications beginning with `apl`, code: `-a "apl*"`

Output messages

Message ID	Description	Output Destination
KFCA10350-I	<code>mcfadltap</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcfadltap</code> command input source failed.	Message log file or standard error output
KFCA10371-I	<code>mcfadltap</code> command has been accepted normally.	Standard output
KFCA10373-E	<code>mcfadltap</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10391-E	<code>mcfadltap</code> command is not supported.	Standard error output
KFCA10529-I	Help message	Standard output
KFCA10704-I	Timer startup has been deleted from an application engaged in timer startup processing.	Message log file
KFCA10705-E	Error was detected during timer startup deletion processing.	Standard error output, message log file

mcfadltap (Delete timer activation requests for applications)

Message ID	Description	Output Destination
KFCA10706-W	No timer startup request has been issued for a specified application, or timer startup has already executed.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfalsap (Display status of applications)

Format

```
mcfalsap -s MCF-communication-process-identifier -a application-name
          [-k application-type]
```

Function

The `mcfalsap` command displays the input and scheduling status of specified applications.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f>` ((01-ef))

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

- `-a application-name<identifier, 1-8 characters>`

Specifies the name of an application whose status is to be displayed.

A maximum of eight application names can be specified. When entering large amounts of inputs for the operation, specify more than one application or specify a group of applications to increase the number of applications to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of application names. The name of a group of applications cannot coexist with the names of applications for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the status of all applications is to be displayed.

*character-string**

Specifies leading characters of an application name when the status of all applications with names beginning with the specified character string is to be displayed.

<Example of specifying more than one application>

To specify `ap11`, `ap12`, and `ap13`, code: `-a "ap11 Δ ap12 Δ ap13"`

<Example of specifying a group of applications>

To specify all applications beginning with apl, code: -a "apl*"

- -k *application-type*~<<user>>

Specifies the type of application.

user: User application

mcf: MCF event

Output format

<i>aaaaaaaaaa bbb cccc dd...dd eee fff [tttt]</i>

aaaaaaaaaa

Message ID

bbb

MCF identifier

cccc

Application type

dd...dd

Application name

eee

Status of the application (input):

ACT: Shutdown release

DCT: Shutdown

***: Displayed in the case of SPP applications

fff

Status of the application (scheduling):

ACT: Shutdown release

DCT: Shutdown

***: Displayed in the case of SPP applications

tttt

Application test mode status (displayed only when TP1/Message Control or TP1/Message Control/Tester is used)

TEST: Test mode

(Blank): Non-test mode

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfalsap command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfalsap command input source failed.	Message log file
KFCA10360-I	Status display has started.	Standard output
KFCA10369-I	Status display has terminated.	Standard output
KFCA10373-E	mcfalsap command terminated abnormally.	Standard error output
KFCA10379-I	See the output format section above.	Standard output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10386-E	Specified application has not been cataloged.	Standard error output
KFCA10518-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfalstap (Display application timer startup requests)

Format

```
mcfalstap [-s application-startup-process-identifier] -a application-name
```

Function

Displays the timer startup requests related to a specified application.

Options

- **-s** *application-startup-process-identifier* ~<numerals (0-9), a-f> ((01-ef))

Specifies an application startup process identifier. No more than one application startup process identifier can be specified.

If this option is omitted, the `mcfalstap` command is executed on all MCFs. This means that the overhead required to search for the application startup processes is added to the processing load of the operation commands.

If you want to have a configuration that includes many application startup processes or if you want to use many operation commands, use the `-s` option to specify an application startup process identifier.

- **-a** *application-name*~<identifier, 1-8 characters>

Specifies the name of an application whose information is to be displayed.

A maximum of eight application names can be specified. When you enter large number of inputs for the operation, specify more than one application or specify a group of applications, as described below, to increase the number of applications processed in parallel by one command and to reduce the number of commands that are entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. Only one group can be specified. The name of a group of applications cannot coexist with the names of applications for which no group is specified. A group name must be enclosed in double quotation marks (").

- *****: Specifies that the timer startup requests related to all applications are displayed.
- *character-string******: Specifies leading characters of an application name when the display is limited to the timer startup requests related to the applications whose names begin with the specified character string.

<Example of specifying more than one application>

To specify `apl1`, `apl2`, and `apl3`:

```
-a "apl1 Δapl2 Δapl3"
```

<Example of specifying a group of applications>

To specify all applications with names beginning with `apl`:

```
-a "apl*"
```

Output format

```
mmm aa....aa hh1:mm1:ss1 bbbb hhh2:mm2:ss2
```

- *mmm*: MCF identifier
- *aa....aa*: Application name
- *hh1:mm1:ss1*: Application startup request acceptance time (*hh1*: hour, *mm1*: minute, *ss1*: second)
- *bbbb*: Type of time specification
 - INTV: Elapsed time specification
 - TIME: Exact time specification
- *hhh2:mm2:ss2*: Elapsed time or exact time specified by the UAP for starting the timer-specified application (*hh2*: hour, *mm2*: minute, *ss2*: second)

If 360000 was specified as the time when a request was made to start an application with an elapsed time specification, 100:00:00 is displayed.

Display examples

```
KFCA10360-I Starts TAP status display.
KFCA16436-I A01 ap001      19:06:33 INTV  00:45:00
KFCA16436-I A01 ap2       19:07:17 TIME  19:20:00
KFCA16436-I A01 ap000003 19:08:23 INTV 100:00:00
KFCA10369-I Ends TAP status display.
```

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcfalstap</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway. Wait for the startup processing to be completed, and then re-enter the command.	Standard error output

Message ID	Description	Output destination
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid. Correct the input format and re-enter the command.	Standard error output
KFCA10354-E	Memory is inadequate. If the same result occurs when the command is re-entered, check the system environment.	Standard error output
KFCA10355-W	Specification of an argument is invalid. Check the specification of the application startup process identifier and application names, make corrections, and then re-enter the command.	Standard error output
KFCA10356-E	Timeout occurred during a process. See the message that was output to a log before this message.	Standard error output
KFCA10357-E	Timeout occurred in the MCF. See the message that was output to a log before this message.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcfalstap</code> command input source failed. If the reason code is <code>-19178</code> , either do not use a group specification for specifying application names or use a group specification with a leading character string added to narrow the display targets, then re-enter the command.	Message log file or standard error output
KFCA10360-I	Status display is starting.	Standard output
KFCA10369-I	Status display is terminating.	Standard output
KFCA10372-E	The <code>mcfalstap</code> command terminated abnormally. See the message that was output to a log before this message.	Standard error output
KFCA10380-E	Retrieval of remote a process failed. Check the application startup process identifier and re-enter the command.	Standard error output
KFCA10391-E	The service for which command execution was attempted does not support the <code>mcfalstap</code> command. Specify the application startup process identifier in the <code>-s</code> option and re-enter the command.	Standard error output
KFCA10563-I	Help message	Standard output
KFCA10714-E	Memory is inadequate. If the same result occurs when the command is re-entered, check the system environment.	Message log file
KFCA16402-E	RPC error occurred during command processing. Take the corrective action indicated by the reason code.	Standard error output
KFCA16436-I	See the above output format.	Standard output

Message ID	Description	Output destination
KFCA16437-I	No timer startup request has been made, a timer is already active, or an application name is invalid. If this message is output when a timer request exists, make sure that the application name is valid or use a group specification, and then re-enter the command.	Standard output

mcfreport (Edit MCF statistics)

Format

```
mcfreport [-k edit-type] [-u edit-target] [-i input-file-name]
          [-d start-date:end-date|target-date] [-t start-time:end-time]
```

Function

The `mcfreport` command inputs and edits the MCF statistics file that is output by the `mcfstats` command. This command then outputs the edited result to the standard output.

Options

- `-k edit-type`~<<all>>

Specifies an edit type.

`all`: Edits all the statistics.

`in`: Edits the statistics of receive messages.

`out`: Edits the statistics of send messages.

`cn`: Edits the connection wait statistics.

- `-u edit-target`

Specifies a service group name, a logical terminal name, or an MCF communication process identifier for editing.

Specifies the following items depending on the specification of the `-k` option.

`all` specified in the `-k` option: Omits this option.

`in` specified in the `-k` option: Service group name

`out` specified in the `-k` option: Logical terminal name

`cn` specified in the `-k` option: MCF communication process identifier

If you specify `in`, `out`, or `cn` in the `-k` option, make sure to specify this option.

Specifying all in the `-k` option invalidates the specification of this option.

You cannot specify more than one service group name, logical terminal name, or MCF communication process identifier.

You can make a batch specification using an asterisk (*) in this option. You can make only one batch specification. You cannot make a batch specification together with an item which is included in the batch specification. For the batch specification, enclose the edit target in double quotation marks (").

*

Edits the statistics of all the service groups, logical terminals or MCF communication processes.

*prefix-character-string**

Edits the statistics of all the service groups or logical terminals whose names start with the prefix character string. You cannot make this specification for the MCF communication processes.

<Example of the batch specification>

To specify service groups starting with "abc"

```
-u "abc*"
```

- **-i** *input-file-name*~<path name, 1-35 characters> <<mcfstc>>

Specifies the file containing the execution result of the `mcfstats` command. If you specify only a file name, the file under the current directory is edited. If you specify an absolute path name, the file under the specified directory is edited.

If you omit this option, `mcfstc` under the current directory is edited.

- **-d** *start-date:end-date|target-date*~<decimal digits> <<s:e>>

Specifies a target date range or the target date for editing.

If you omit this option, `-d s: e` is assumed.

`s`: Default start date (00000101)

`e`: Default end date (99991231)

The start date, end date and target date are represented in the *YYYYMMDD* format.

YYYY: Year (Desired year)

MM: Month (01-12)

DD: Day (01-31)

- **-t** *start-time:end-time*~<decimal digits> <<s:e>>

Specifies the time range for editing.

If you omit this option, `-t s: e` is assumed.

`s`: Default start time (0000)

`e`: Default end time (2359)

The start time and end time are represented in the *HHMM* format.

HH: Hour (00-23)

MM: Minute (00-59)

All the information within the time range specified by this option in the date range specified by the *-d* option is edited. For example, specifying " *d* Δ 19970101: 19970103 Δ -t Δ 0900:1700" will edit all the statistics from 9:00 to 17:00 on January 1, 2, and 3 in 1997. You cannot specify the time from one day over the next day (for example, -t Δ 2300: 0100).

Output format

```

mcfreport  *** MCF Uptime Statistics ***  file name:aaaaaaa
-----
Date of output file:bbb-bb-bb bb:bb:bb
-----
<Information of send messages>
Logical Terminal[ccccccc] MCFID[ddd]
[SYNC] [eeeeeeeee]      ffffffff      ffffffff      ffffffff      ffffffff
                        g.ggggggg      g.ggggggg      g.ggggggg      g.ggggggg
[RESP] [hhhhhhhhhh]      iiiiii      iiiiii      iiiiii      iiiiii
                        j.iiiii      j.iiiii      j.iiiii      j.iiiii
[PRIO] [kkkkkkkkkk]      uuuuuu      uuuuuu      uuuuuu      uuuuuu
                        m.mmmmmmm      m.mmmmmmm      m.mmmmmmm      m.mmmmmmm
[NORM] [nnnnnnnnnn]      oooooooooo  oooooooooo  oooooooooo  oooooooooo
                        p.ppppppp      p.ppppppp      p.ppppppp      p.ppppppp
-----
<Information of waiting connections for starting transaction>
MCFID[ddd]
[qqqqqqqqqq]      rrrrrrrr      rrrrrrrr      rrrrrrrr      rrrrrrrr
                        s.ssssss      s.ssssss      s.ssssss      s.ssssss
-----
<Information of receive messages>
Service Group[tttttt] MCFIF[ddd]
[uuuuuuuuuu]      vvvvvvvv      vvvvvvvv      vvvvvvvv      vvvvvvvv
                        w.wwwwww      w.wwwwww      w.wwwwww      w.wwwwww

```

aaaaaaa

Input file name

bbb-bb-bb bb: bb: bb

File output date and time

ccccccc

Logical terminal name

ddd

MCF identifier

eeeeeeeee

Number of synchronous send messages that have been processed[#]

fffffff

Minimum, maximum, average and total number of synchronous send messages which are in the processing wait status

g.gggggg

Minimum, maximum, average and total processing wait time for synchronous send messages

hhhhhhhhh

Number of inquiry-type send messages that have been processed[#]

iiiiiii

Minimum, maximum, average and total number of inquiry type send messages that are in the processing wait status

j.jjjj

Minimum, maximum, average and total processing wait time for inquiry type send messages

kkkkkkkkk

Number of priority branch-type send messages that have been processed[#]

llllllll

Minimum, maximum, average and total number of priority branch-type send messages that are in the processing wait status

m.mmmmmm

Minimum, maximum, average and total processing wait time for priority branch-type send messages

nnnnnnnnn

Number of normal branch-type send messages that have been processed[#]

oooooooo

Minimum, maximum, average and total number of normal branch-type send messages that are in the processing wait status

p.pppppp

Minimum, maximum, average and total processing wait time for normal branch type send messages

qqqqqqqqq

Number of connections that have been processed[#]

rrrrrrrrr

Minimum, maximum, average and total number of connections that are in the processing wait status

s.ssssss

Minimum, maximum, average and total processing wait time for connections

ttttttt

Service group name

uuuuuuuuuuuu

Number of receive messages that have been processed[#]

vvvvvvvvvv

Minimum, maximum, average and total number of receive messages that are in the processing wait status

w.wwwwwww

Minimum, maximum, average and total processing wait time for receive messages

#

Displayable to the upper limit of int.

Output messages

Message ID	Description	Output destination
KFCA19751-I	Starts editing the MCF statistics.	Standard output
KFCA19752-I	Terminates editing the MCF statistics.	Standard output
KFCA19753-E	Memory required for editing the MCF statistics cannot be allocated.	Standard error output
KFCA19754-E	An invalid option exists.	Standard error output
KFCA19755-E	The length of the input file name exceeds the specified value.	Standard error output
KFCA19757-E	An invalid input file name is specified.	Standard error output
KFCA19758-E	An invalid edit type is specified.	Standard error output
KFCA19759-E	The edit target is not specified.	Standard error output
KFCA19760-E	An invalid edit target is specified.	Standard error output
KFCA19761-E	An invalid date is specified.	Standard error output
KFCA19762-E	An invalid time is specified.	Standard error output

Message ID	Description	Output destination
KFCA19763-E	An error occurred while the input file is being processed.	Standard error output

Notes

- If you specify an invalid edit target, the system assumes that no statistics of the target are output so that no error message is output.
- If you specify the file for which the `mcfstats` command is being executed, its operation is not guaranteed.

mcfstats (Output MCF statistics)

Format

```
mcfstats [-k output-type] [-u output-target] [-o output-file-name]
         [-f output-file-size] [-s time-interval] [-t output-count] [-r]
```

Function

Outputs the MCF statistics to a file at the specified time interval.

Specifying the `-r` option terminates output of the MCF statistics.

Options

- `-k output-type~<<all>>`

Specifies an output type.

`all`: Outputs all the statistics.

`in`: Outputs the statistics of receive messages.

`out`: Outputs the statistics of send messages.

`cn`: Outputs the connection wait statistics.

- `-u output-target`

Specifies a service group name, a logical terminal name, or an MCF communication process identifier to output the statistics.

Specifies the following items depending on the specification of the `-k` option.

`all` specified in the `-k` option: Omits this option.

`in` specified in the `-k` option: Service group name

`out` specified in the `-k` option: Logical terminal name

`cn` specified in the `-k` option: MCF communication process identifier

If you specify `in`, `out`, or `cn` in the `-k` option, be sure to specify this option. Specifying `all` in the `-k` option invalidates the specification of this option.

You cannot specify more than one service group name, logical terminal name, or MCF communication process identifier.

You can make a batch specification using an asterisk (*) in this option. You can make only one batch specification. You cannot make a batch specification together with an item which is included in the batch specification. For the batch specification, enclose the output target in double quotation marks (").

*

Outputs the statistics of all the service groups, logical terminals or MCF communication processes.

*prefix-character-string**

Outputs the statistics of all the service groups or logical terminals whose names start with the prefix character string. You cannot make this specification for the MCF communication processes.

<Example of the batch specification>

To specify service groups starting with "abc"

```
-u "abc*"
```

- -o *output-file-name*~<path name, 1-35 characters> <<mcfstc>>

Specifies the output file name. If you specify only a file name, the file is created under `$DCDIR/spool`. If you specify an absolute path name, the file is created under the specified directory. However, if you specify a non-existing directory, the directory is not created. You cannot specify an existing file name.

If you omit this option, the statistics are output to `$DCDIR/spool/mcfstc`.

- -f *output-file-size*~<unsigned integer> ((1-1000)) <<10>> (unit: megabytes)

Specifies the size (in megabytes) of the file to which the MCF statistics are output. Use the following expressions to calculate the file size.

For a service group: $\uparrow(64 + A) \times \text{output count} / (1024 \times 1024) \uparrow$

For a logical terminal: $\uparrow(64 + B) \times \text{output count} / (1024 \times 1024) \uparrow$

For an MCF communication process: $\uparrow(64 + C) \times \text{output count} / (1024 \times 1024) \uparrow$

For all MCF statistics: $(64 + (A + B + C)) \times \text{output count} / (1024 \times 1024)$

$\uparrow \uparrow$: Rounded up to an integer.

A: 96 x total number of service groups

B: 64 x 4 x total number of logical terminals

C: 64 x total number of MCF communication processes

Note

The maximum value with an asterisk (*) specified in the -u option is assumed for these expressions.

If the specified file size is exceeded, output of the statistics terminates.

- `-s time-interval~<unsigned integer> ((1-43200)) <<60>>` (unit: seconds)

Specifies the time interval (in seconds) to output the MCF statistics to a file.

If you omit this specification, the time interval is 60 seconds.

- `-t output-count~<unsigned integer> ((1-65535))` (unit: count)

Specifies the output count.

If you omit this option, the statistics are output until "mcfstats -r" is executed, MCF stops or the file size specified by the `-f` option is exceeded. When the specified output count is reached, output of the statistics terminates.

If the file size specified by the `-f` option is exceeded, output of the MCF statistics terminates even if the specified output count is not reached.

- `-r`

Terminates output of the MCF statistics.

Specifying this option invalidates all other options.

Output messages

Message ID	Description	Output destination
KFCA19700-E	An error occurred while the internal function is being processed.	Standard output, message log file
KFCA19701-W	The MCF statistics cannot be acquired due to insufficient shared memory.	Standard error output, message log file
KFCA19702-I	Starts outputting the MCF statistics to a file.	Standard output, message log file
KFCA19703-I	Terminates outputting the MCF statistics to a file.	Standard output, message log file
KFCA19704-W	Acquisition of the MCF statistics is not started.	Standard error output
KFCA19705-W	Output of the MCF statistics is not started.	Standard error output
KFCA19706-I	Terminates outputting the MCF statistics to a file because the file size is exceeded.	Standard output, message log file
KFCA19707-E	An error occurred while the MCF statistics is being output to a file.	Standard error output, message log file
KFCA19708-W	The command cannot be accepted because output of the MCF statistics to a file has been started.	Standard error output
KFCA19709-W	The specified file has already exist.	Standard error output
KFCA19710-E	Preparation to output the MCF statistics to a file failed.	Standard error output

Message ID	Description	Output destination
KFCA19711-E	The file could not be opened.	Standard error output

Notes

- If you specify an invalid output target, the system assumes that no statistics of the target are acquired. Therefore, the system does not output an error message.
- If you delete the file to which the statistics are being output, the subsequent operation of OpenTP1 is not guaranteed.
- If you specify a too short output time interval, the interval may extend depending on the MCF operation status.

mcfactcn (Establish connections)

Format

```
mcfactcn [-s MCF-communication-process-identifier]
          {-c connection-ID|-g connection-group-name}
          [-u subconnection-ID] [-s XP-service-name@host-name]
```

Function

The `mcfactcn` command establishes connections.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target connection. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfactcn` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-c connection-ID`~<identifier, 1-8 characters>

Specifies the ID of a connection to be established.

A maximum of eight connection IDs can be specified. When entering large amounts of inputs for the operation, specify more than one connection or specify a group of connections to increase the number of connections to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection IDs, separate them by a space and enclose the entire string of connection IDs in double quotation marks ("). The same connection ID cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection IDs. The ID of a group of connections cannot coexist with the IDs of connections for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that all connections are to be established.

*character-string**

Specifies leading characters of a connection ID when all connections with IDs beginning with the specified character string are to be established.

<Example of specifying more than one connection>

To specify *cn1*, *cn2*, and *cn3*, code: `-c "cn1 Δ cn2 Δ cn3 "`

<Example of specifying a group of connections>

To specify all connections beginning with *cn*, code: `-c "cn* "`

- `-g connection-group-name~<identifier, 1-8 characters>`

Specifies the name of a connection group to be established.

A maximum of eight connection group names can be specified. When entering large amounts of inputs for the operation, specify more than one connection group or specify a group of connection groups to increase the number of connection groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection group names, separate them by a space and enclose the entire string of connection group names in double quotation marks ("). The same connection group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection group names. The name of a group of connection groups cannot coexist with the names of connection groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Establishes all connection groups.

*character-string**

Specifies leading characters of a connection group name when all connection groups with names beginning with the specified character string are to be established.

<Example of specifying more than one connection group>

To specify *cng1*, *cng2*, and *cng3*, code: `-g "cng1 Δ cng2 Δ cng3 "`

<Example of specifying a group of connection groups>

To specify all connection groups beginning with *cng*, code: `-g "cng* "`

Note that the `-g` option is valid only when the TP1/NET/OSI-TP protocol is used.

- `-u subconnection-ID~<1-8 character identifier>`

Specifies the subconnection ID of a subconnection to be established. It can be specified

when only one connection ID is used and no grouping is specified. Neither more than one subconnection ID nor ID of a group of subconnections can be specified. When this option is omitted, all subconnections under the specified connection are established.

The `-u` option can be used only under the TP1/NET/HNA-NIF protocol.

■ `-s XP-service-name@host-name`

Specifies the XP service to be connected. Specify this when the protocol is TP1/NET/XMAP3.

XP-service-name~<1-to-14-character identifier>

Specifies the name of the XP service to be connected.

host-name~<1-to-127-character identifier>

Specifies the name of the host where the management daemon for the XP service to be connected operates. When this option is specified, only one connection ID can be specified. The `-g` option cannot be specified with this option.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfactcn command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfactcn command input source failed.	Message log file
KFCA10371-I	mcfactcn command has been accepted normally.	Standard output
KFCA10373-E	mcfactcn command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10391-E	mcfactcn command is not supported.	Standard error output

Message ID	Description	Output destination
KFCA10500-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error
KFCA16424-E	Specified connection group has not been cataloged.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcfactle (Release shutdown of logical terminals)

Format

```
mcfactle [-s MCF-communication-process-identifier]
          [-c connection-ID] -l logical-terminal-name
          [-t shutdown-release-type]
```

Function

The `mcfactle` command releases logical terminals from shutdown status.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfactle` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

When the configuration includes lots of MCF communication services or large amounts of operation commands are entered for the operation, use the `-s` option to specify an MCF communication process identifier.
- `-c connection-ID`~<identifier, 1-8 characters>

Specifies the connection ID of a logical terminal whose shutdown status is to be released.

Neither multiple connection IDs nor the ID of a group of connections can be specified.
- `-l logical-terminal-name`~<identifier, 1-8 characters>

Specifies the name of a logical terminal whose shutdown status is to be released.

If the `-c` option is specified, the logical terminal name corresponding to the specified connection ID must be specified.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the shutdown status of all logical terminals is to be released.

*character-string**

Specifies leading characters of a logical terminal name when the shutdown status of all logical terminals with names beginning with the specified character string is to be released.

<Example of specifying more than one logical terminal>

To specify *len1*, *len2*, and *len3*, code: -1 "*len1* Δ *len2* Δ *len3*"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with *len*, code: -1 "*len**"

■ -t *shutdown-release-type*~<1-8 character identifier><<all>>

Specifies the type of releasing shutdown.

all

Releases a logical terminal from shut-down terminal status and queue status.

term

Releases a logical terminal from shut-down terminal status.

queue

Releases a logical terminal from shut-down queue status.

It may not be specified under some protocols; when specified, the option is ignored.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfactle command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output

Message ID	Description	Output destination
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mftactle</code> command input source failed.	Message log file or standard error output
KFCA10371-I	<code>mftactle</code> command has been accepted normally.	Standard output
KFCA10373-E	<code>mftactle</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output
KFCA10390-E	Correspondence between specified connection ID and logical terminal name is invalid.	Standard error output
KFCA10391-E	<code>mftactle</code> command is not supported.	Standard error output
KFCA10395-E	The name of an unconnected logical terminal is specified in specified connection.	Standard error output
KFCA10503-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcfactmj (Start message journal collection for logical terminals)

Format

```
mcfactmj [-s MCF-communication-process-identifier]
          -l logical-terminal-name
```

Function

The `mcfactmj` command starts collection of message journals for specified logical terminals.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfactmj` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-l logical-terminal-name<identifier, 1-8 characters>`

Specifies the name of a logical terminal for which message journal collection is to be started.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that message journal collection for all logical terminals is to be started.

*character-string**

Specifies leading characters of a logical terminal name when message journal collection for all logical terminals with names beginning with the specified character string is to be started.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3, code: -1 "len1 Δ len2 Δ len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -1 "len*"

Output messages

Message ID	Description	Output destination
KFCA10107-W	The MJ of the specified logical terminal is being acquired.	Standard error output
KFCA10350-I	mftactmj command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mftactmj command input source failed.	Message log file or standard error output
KFCA10371-I	mftactmj command has been accepted normally.	Standard output
KFCA10373-E	mftactmj command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output
KFCA10527-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftactsg (Release service groups from shutdown)

Format

```
mcftactsg -g service-group-name [-t shutdown-release-type]
```

Function

The `mcftactsg` command releases specified service groups from shutdown status.

Options

- `-g service-group-name~<identifier, 1-31 characters>`

Specifies the name of a service group to be released from shutdown status.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service group names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that shutdown status of all service groups is to be released.

*character-string**

Specifies leading characters of a service group name when all service groups with names beginning with the specified character string are to be released from shutdown status.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`, code: `-1 "seg1 Δseg2 Δseg3"`

<Example of specifying a group of services>

To specify all services beginning with `seg`, code: `-1 "seg*"`

- `-t shutdown-release-type~<<insc>>`

Specifies the type of shutdown release:

mcftactsg (Release service groups from shutdown)

in sc

Release input to input queue and input queue scheduling from shutdown status

in

Release input to input queue only from shutdown status

sc

Release input queue scheduling only from shutdown status

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftactsg command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftactsg command input source failed.	Message log file or standard error output
KFCA10371-I	mcftactsg command has been accepted normally.	Standard output
KFCA10373-E	mcftactsg command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	A specified service group has not been cataloged.	Standard error output
KFCA10513-I	Help message	Standard output
KFCA11038-W	A service group has already been released from shutdown status.	Standard error output
KFCA11042-W	Shutdown cannot be released because the MHP user server is terminating.	Standard error output
KFCA11043-W	Shutdown cannot be released because the MHP user server has not been activated.	Standard error output
KFCA11053-E	Shutdown cannot be released because the user server is on shutdown status.	Standard error output

Message ID	Description	Output destination
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11098-W	An application cannot be shut down or released from shutdown status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfactss (Start a session)

Format

```
mcfactss [-s MCF-communication-process-identifier]
          [-c connection-ID] -l logical-terminal-name
```

Function

The `mcfactss` command starts the session that corresponds to the specified logical terminal. It can be used only under the TP1/NET/HNA-560/20 protocol.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfactss` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.
- `-c connection-ID~<1-8 character identifier>`

Specifies the connection ID for the connection through which a session starts. Neither multiple connection IDs nor the ID of a group of connections can be specified.

When this option is omitted, the `mcfactss` command is executed on all connections.
- `-l logical-terminal-name ~<1-8 character identifier>`

Specifies the name of the logical terminal at which a session starts.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. Only one group can be specified. The name of a group of logical terminals

cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks ("").

*

Starts a session for all logical terminals.

*leading-string**

Starts a session for all logical terminals beginning with the leading string.

<Example of specifying more than one logical terminal>

To specify *len1*, *len2*, and *len3*, code: -1 "*len1* Δ*len2* Δ*len3*"

<Example of specifying a group of logical terminals>

To specify all services beginning with *len*, code: -1 "*len**"

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfactss command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10359-W	Response to mcfactss command input source failed.	Message log file
KFCA10371-I	mcfactss command has been accepted normally.	Standard output
KFCA10373-E	mcfactss command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10390-E	Correspondence between specified connection ID and logical terminal name is invalid.	Standard error output
KFCA10395-E	The name of an unconnected logical terminal is specified in specified connection.	Standard error output

mcftactss (Start a session)

Message ID	Description	Output destination
KFCA10539-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcfactsv (Release services from shutdown)

Format

```
mcfactsv -v service-name
```

Function

The `mcfactsv` command releases services from shutdown status.

Options

- `-v service-name~<identifier, 1-31 characters>`

Specifies the name of a service to be released from shutdown status.

A maximum of eight service names can be specified. When entering large amounts of inputs for the operation, specify more than one service or specify a group of services to increase the number of services to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service names, separate them by a space and enclose the entire string of service names in double quotation marks ("). The same service name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that shutdown status of all services is to be released.

*character-string**

Specifies leading characters of a service name when all services with names beginning with the specified character string are to be released from shutdown status.

<Example of specifying more than one service>

To specify `ser1`, `ser2`, and `ser3`, code: `-l "ser1 Δser2 Δser3"`

<Example of specifying a group of services>

To specify all services beginning with `ser`, code: `-l "ser*"`

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfactsv command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfactsv command input source failed.	Message log file or standard error output
KFCA10371-I	mcfactsv command has been accepted normally.	Standard output
KFCA10373-E	mcfactsv command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10384-E	A specified service has not been cataloged.	Standard error output
KFCA10510-I	Help message	Standard output
KFCA11037-W	A service has already been released from shutdown status.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11098-W	An application cannot be shut down or released from shutdown status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftchn (Switch connections)

Format

```
mcftchn [-s MCF-communication-process-identifier]
        -f origin-connection-ID -t destination-connection-ID
```

Function

The `mcftchn` command invalidates the link from origin connection to a logical terminal and links the destination connection with the logical terminal.

Before executing the `mcftchn` command, the logical terminal on the origin connection must be shut down.

The `mcftchn` command can be used under the TP1/NET/TCP/IP protocol.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target connection. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftchn` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-f origin-connection-ID`~<1-8 character identifier>

Specifies the connection ID of the origin connection. No more than one origin connection ID can be specified.

- `-t destination-connection-ID`~<1-8 character identifier>

Specifies the connection ID of the destination connection. No more than one destination connection ID cannot be specified.

Output messages

Message ID	Description	Output destination
KFCA10350-I	<code>mcftchn</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output

Message ID	Description	Output destination
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfthcn command input source failed.	Message log file
KFCA10371-I	mcfthcn command has been accepted normally.	Standard output
KFCA10373-E	mcfthcn command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10391-E	mcfthcn command is not supported.	Standard error output
KFCA10392-E	A necessary program product has not been installed.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

Notes

Before executing the mcfthcn command, TP1/NET/High Availability must be installed.

mcftdctn (Release connections)

Format

```
mcftdctn [-s MCF-communication-process-identifier]
          {-c connection-ID|-g connection-group-name}
          [-u subconnection-ID] [-f]
```

Function

The `mcftdctn` command releases connections.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target connection. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftdctn` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-c` *connection-ID*~<identifier, 1-8 characters>

Specifies the ID of a connection to be released.

A maximum of eight connection IDs can be specified. When entering large amounts of inputs for the operation, specify more than one connection or specify a group of connections to increase the number of connections to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection IDs, separate them by a space and enclose the entire string of connection IDs in double quotation marks ("). The same connection ID cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection IDs. The ID of a group of connections cannot coexist with the IDs of connections for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that all connections are to be released.

*character-string**

Specifies leading characters of a connection ID when all connections with IDs beginning with the specified character string are to be released.

<Example of specifying more than one connection>

To specify *cnn1*, *cnn2*, and *cnn3*, code: `-c "cnn1 Δ cnn2 Δ cnn3 "`

<Example of specifying a group of connections>

To specify all connections beginning with *cnn*, code: `-c "cnn* "`

- `-g connection-group-name~<identifier, 1-8 characters>`

Specifies the name of a connection group to be released.

A maximum of eight connection group names can be specified. When entering large amounts of inputs for the operation, specify more than one connection group or specify a group of connection groups to increase the number of connection groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection group names, separate them by a space and enclose the entire string of connection group names in double quotation marks ("). The same connection group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection group names. The name of a group of connection groups cannot coexist with the names of connection groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Releases all connection groups.

*character-string**

Specifies leading characters of a connection group name when all connection groups with names beginning with the specified character string are to be released.

<Example of specifying more than one connection group>

To specify *cng1*, *cng2*, and *cng3*, code: `-g "cng1 Δ cng2 Δ cng3 "`

<Example of specifying a group of connection groups>

To specify all connection groups beginning with *cng*, code: `-g "cng* "`

Note that the `-g` option is valid only when the TP1/NET/OSI-TP protocol is used.

- `-u subconnection-ID~<1-8 character identifier>`

Specifies the subconnection ID of a subconnection to be released. It can be specified

when only one connection ID is used and no grouping is specified. Neither more than one subconnection ID nor ID of a group of subconnections can be specified. When this option is omitted, all subconnections under the specified connection are released.

This option can be specified only when the TP1/NET/HNA-NIF protocol is used.

■ -f

Specifies that connection release is to be forced.

When this option is omitted, processing depends on the protocol being used; for details, see the applicable protocol manual.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfldctcn command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfldctcn command input source failed.	Message log file
KFCA10371-I	mcfldctcn command has been accepted normally.	Standard output
KFCA10373-E	mcfldctcn command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	A specified connection has not been cataloged.	Standard error output
KFCA10391-E	mcfldctcn command is not supported.	Standard error output
KFCA10501-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output
KFCA16424-E	Specified connection group has not been cataloged.	Standard error output

In addition these messages, messages for the protocol being used are output; for

mcftdctn (Release connections)

details, see the applicable protocol manual.

mcftdctl (Shut down logical terminals)

Format

```
mcftdctl [-s MCF-communication-process-identifier]
          [-c connection-ID] -l logical-terminal-name
          [-t shutdown-type]
```

Function

The `mcftdctl` command shuts down logical terminals forcibly.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftdctl` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-c` *connection-ID*~<identifier, 1-8 characters>

Specifies the connection ID of a logical terminal to be shut down.

Neither more than one connection ID nor ID of a group of connections can be specified.

- `-l` *logical-terminal-name*~<identifier, 1-8 characters>

Specifies the name of a logical terminal to be shut down.

If the `-c` option is specified, the logical terminal name corresponding to the specified connection ID must be specified.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks ("").

*

Specifies that all logical terminals are to be shut down.

*character-string**

Specifies leading characters of a logical terminal name when all logical terminals with names beginning with the specified character string are to be shut down.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3, code: -l "len1 Δ len2 Δ len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -l "len*"

- -t *shutdown-type*~<1-8 character identifier><<all>>

Specifies the type of shutdown.

all

Shuts down the terminal status and queue status of logical terminals.

term

Shuts down the terminal status of logical terminals.

queue

Shuts down the queue status of logical terminals.

It may not be specified under some protocols; when specified, the option is ignored.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftdctl command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output

Message ID	Description	Output destination
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfstdctle command input source failed.	Message log file or standard error output
KFCA10371-I	mcfstdctle command has been accepted normally.	Standard output
KFCA10373-E	mcfstdctle command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output
KFCA10390-E	Correspondence between specified connection ID and logical terminal name is invalid.	Standard error output
KFCA10391-E	mcfstdctle command is not supported.	Standard error output
KFCA10395-E	The name of an unconnected logical terminal is specified in specified connection.	Standard error output
KFCA10504-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcftdctmj (Terminate message journal collection for logical terminals)

Format

```
mcftdctmj [-s MCF-communication-process-identifier]  
          -l logical-terminal-name
```

Function

The `mcftdctmj` command terminates collection of message journals for specified logical terminals.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))
Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftdctmj` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.
- `-l logical-terminal-name`~<identifier, 1-8 characters>
Specifies the name of a logical terminal for which message journal collection is to be terminated.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that message journal collection for all logical terminals is to be terminated.

*character-string**

Specifies leading characters of a logical terminal name when message journal collection is to be terminated for all logical terminals with names beginning with the specified character string.

<Example of specifying more than one logical terminal>

To specify *len1*, *len2*, and *len3*, code: -1 "*len1* Δ *len2* Δ *len3*"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with *len*, code: -1 "*len**"

Output messages

Message ID	Description	Output destination
KFCA10108-W	The acquisition of the MJ of the specified logical terminal is stopped.	Standard error output
KFCA10350-I	mcfddctmj command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfddctmj command input source failed.	Message log file or standard error output
KFCA10371-I	mcfddctmj command has been accepted normally.	Standard output
KFCA10373-E	mcfddctmj command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output

mcftdctmj (Terminate message journal collection for logical terminals)

Message ID	Description	Output destination
KFCA10528-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftdctsg (Shut down service groups)

Format

```
mcftdctsg -g service-group-name [-t shutdown-type] [-r]
```

Function

The `mcftdctsg` command shuts down service groups.

Options

- `-g service-group-name`~<identifier, 1-31 characters>

Specifies the name of a service group to be shut down.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service group names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that all service groups are to be shut down.

*character-string**

Specifies leading characters of a service group name when all service groups with names beginning with the specified character string are to be shut down.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`, code: `-l "seg1 Δseg2 Δseg3"`

<Example of specifying a group of services>

To specify all services beginning with `seg`, code: `-l "seg*"`

- `-t shutdown-type`~<<inse>>

Specifies the type of shutdown:

mcftdctsg (Shut down service groups)

in sc

Shut down input to input queue and input queue scheduling.

in

Shut down input to input queue only.

sc

Shut down input queue scheduling only.

■ -r

Specifies that the shutdown status of the service group not be inherited after full recovery.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftdctsg command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftdctsg command input source failed.	Message log file or standard error output
KFCA10371-I	mcftdctsg command has been accepted normally.	Standard output
KFCA10373-E	mcftdctsg command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	A specified service group has not been cataloged.	Standard error output
KFCA10514-I	Help message	Standard output
KFCA11064-W	A service group has already been shut down.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output

Message ID	Description	Output destination
KFCA11098-W	An application cannot be shut down or released from shutdown status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftdctss (Terminate a session)

Format

```
mcftdctss [-s MCF-communication-process-identifier]
          [-c connection-ID] -l logical-terminal-name
```

Function

The `mcftdctss` command terminates the session that corresponds to the specified logical terminal. It can be used under the TP1/NET/HNA-560/20 or TP1/NET/C/S560 protocol.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftdctss` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.
- `-c connection-ID~<1-8 character identifier>`

Specifies the connection ID for the connection through which a session terminates. Neither more than one connection ID nor ID of a group of connections can be specified.

When this option is omitted, the `mcftdctss` command is executed on all connections.
- `-l logical-terminal-name~<1-8 character identifier>`

Specifies the name of the logical terminal at which a session terminates.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. Only one group can be specified. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Terminates a session for all logical terminals.

*leading-string**

Terminates a session for all logical terminals beginning with the leading string.

<Example of specifying more than one logical terminal>

To specify *len1*, *len2*, and *len3*, code: -1 "*len1* Δ*len2* Δ*len3*"

<Example of specifying a group of logical terminals>

To specify all services beginning with *len*, code: -1 "*len**"

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfctdctss command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfctdctss command input source failed.	Message log file or standard error output
KFCA10371-I	mcfctdctss command has been accepted normally.	Standard output
KFCA10373-E	mcfctdctss command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	Specified connection has not been cataloged.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output

mcftdctss (Terminate a session)

Message ID	Description	Output destination
KFCA10390-E	Correspondence between specified connection ID and logical terminal name is invalid.	Standard error output
KFCA10395-E	The name of an unconnected logical terminal is specified in specified connection.	Standard error output
KFCA10540-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcftdctsv (Shut down services)

Format

```
mcftdctsv -v service-name [-t shutdown-type]
```

Function

The `mcftdctsv` command shuts down services.

Options

- `-v service-name~<identifier, 1-31 characters>`

Specifies the name of a service to be shut down.

A maximum of eight service names can be specified. When entering large amounts of inputs for the operation, specify more than one service or specify a group of services to increase the number of services to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service names, separate them by a space and enclose the entire string of service names in double quotation marks ("). The same service name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that all services are to be shut down.

*character-string**

Specifies leading characters of a service name when all services with names beginning with the specified character string are to be shut down.

<Example of specifying more than one service>

To specify `ser1`, `ser2`, and `ser3`, code: `-l "ser1 Δser2 Δser3"`

<Example of specifying a group of services>

To specify all services beginning with `ser`, code: `-l "ser*"`

- `-t shutdown-type~<<inse>>`

Specifies the type of shutdown:

mcftdctsv (Shut down services)

in sc

Shut down input to input queue and input queue scheduling.

in

Shut down input to input queue only.

sc

Shut down input queue scheduling only.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftdctsv command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftdctsv command input source failed.	Message log file or standard error output
KFCA10371-I	mcftdctsv command has been accepted normally.	Standard output
KFCA10373-E	mcftdctsv command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10384-E	A specified service has not been cataloged.	Standard error output
KFCA10511-I	Help message	Standard output
KFCA11036-W	A service has already been shut down.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11098-W	An application cannot be shut down or released from shutdown status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftdlqle (Delete output queues for logical terminals)

Format

```
mcftdlqle [-s MCF-communication-process-identifier]
          -l logical-terminal-name [-d deletion-type]
```

Function

The `mcftdlqle` command deletes the entire output queues for specified logical terminals.

When the `mcftdlqle` command is to be used independently, the applicable logical terminal must first be shut down using the `mcftdctle` command or the `dc_mcf_tdctle` function. When the TP1/NET/HNA-560/20 protocol is used, the `mcftdctss` command must also be executed to terminate the session. The logical terminal must not be a terminal of alternate origin.

If the `mcftdlqle` command is used with the `mcftdmpqu` command, the `mcftthldoq` command must first be executed to hold the output queue process of the logical terminal before executing the `mcftdlqle` command.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftdlqle` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-l logical-terminal-name<identifier, 1-8 characters>`

Specifies the name of a logical terminal whose output queue is to be deleted.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical

terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks ("").

*

Specifies that the output queues of all logical terminals are to be deleted.

*character-string**

Specifies leading characters of a logical terminal name when the output queues of all logical terminals with names beginning with the specified character string are to be deleted.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3: -l "len1 Δ len2 Δ len3"

<Example of specifying a group of logical groups>

To specify all logical terminals beginning with len, code: -l "len*"

■ -d *deletion-type*

Specifies the type of deletion.

disk

Only disk queues are deleted.

memory

Only memory queues are deleted.

all

Both disk and memory queues are deleted.

When this option is specified, events that report the discarding of an unprocessed message (ERREVT) are not reported.

When this option is omitted, the MHP for MCF events that report the discarding of an unprocessed message is activated to delete the output queues (disk queue and memory queue).

Output messages

Message ID	Description	Output destination
KFCA10103-E	A specified logical terminal has not been shut down.	Standard error output
KFCA10350-I	mcftdlqle command has been entered.	Standard output

Message ID	Description	Output destination
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfddlqle command input source failed.	Message log file or standard error output
KFCA10371-I	mcfddlqle command has been accepted normally.	Standard output
KFCA10373-E	mcfddlqle command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output
KFCA10391-E	mcfddlqle command is not supported.	Standard error output
KFCA10507-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftdlqsg (Delete input queues of service groups)

Format

```
mcftdlqsg -g service-group-name -d deletion-type
```

Function

The `mcftdlqsg` command deletes input queues for the specified service group.

If the `mcftdlqsg` command is used independently, the `mcftdctsg` command must be executed to shut down the applicable service group before executing the `mcftdlqsg` command.

If the `mcftdlqsg` command is used with the `mcftdmpqu` command, the `mcfthldiq` command must first be executed to hold the input queue process of the service group before executing the `mcftdlqsg` command.

Options

- `-g service-group-name`~<1-31 character identifier>

Specifies the name of a service group from which input queues are to be deleted.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. Only one group can be specified. The name of a group of service groups cannot coexist with the names of service groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Deletes input queues for all service groups.

*leading-string**

Deletes input queues for all service groups beginning with the leading string.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`: `-g "seg1 Δ seg2 Δ seg3"`

<Example of specifying a group of service groups>

To specify all services beginning with *seg*, code: `-g "seg*"`

■ `-d` *deletion-type*

Specifies the type of deletion.

`disk`

Only disk queues are deleted.

`memory`

Only memory queues are deleted.

`all`

Both disk and memory queues are deleted.

Output messages

Message ID	Description	Output destination
KFCA10350-I	<code>mcftdlqsg</code> command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA19357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcftdlqsg</code> command input source failed.	Message log file or standard error output
KFCA10371-I	<code>mcftdlqsg</code> command has been accepted normally.	Standard output
KFCA10373-E	<code>mcftdlqsg</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	Specified service group has not been cataloged.	Standard error output
KFCA10522-I	Help message	Standard output
KFCA11077-E	Specified service group is not on shutdown status (or hold status).	Standard error output
KFCA11097-W	Input queue cannot be deleted due to SPP resources.	Standard error output

mcftdlqsg (Delete input queues of service groups)

Message ID	Description	Output destination
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfstdmpqu (Copy contents of input/output queues)

Format

```
mcfstdmpqu -k output-unit-type [-u output-unit-name]
           -f dump-file-name [-a]
```

Function

The `mcfstdmpqu` command copies the contents of input/output queues to a file. The contents to be copied are disk messages.

Before copying the contents of an input queue, the applicable service group must be shut down with the `mcfstdctsg` command. When the `mcfstdmpqu` command is used with the `mcfstdlqsg` command (delete input queues), the `mcfsthldiq` command must be executed to hold the input queue process of the service group before executing the `mcfstdmpqu` command.

When the `mcfstdmpqu` command is to be used independently to copy the contents of the output queue, the applicable logical terminal must be shut down first using the `mcfstdctle` command or the `dc_mcf_tdctle` function. When you use the `mcfstdmpqu` command in conjunction with the `mcfstdlqle` command or the `dc_mcf_tdlqle` function (to delete the output queue), you must first use the `mcfsthldoq` command to place the output queue of the applicable logical terminal on hold status.

Options

- `-k output-unit-type`

Specifies the units of dump output:

`all`

Contents of all input and output queues

`svg`

Queue contents in units of service groups

`le`

Queue contents in units of logical terminal names

- `-u output-unit-name`

Specifies the name of a service group or logical terminal that is to be subject to dump output processing.

If `svg` is specified for the `-k` option, a service group name must be specified; if `le` is

specified, a logical terminal name must be specified.

If `svg` or `le` is specified for the `-k` option, this option must be specified.

If `all` is specified for the `-k` option, specification of this option is ignored.

The user can specify a maximum of eight service group names or logical terminal names separated by spaces. Enclose the entire string of logical terminal names or service group names in double quotation marks (`"`). The same service group name or logical terminal name cannot be duplicated when specified.

The asterisk (`*`) can be used in the specification to indicate a group of service group names or logical terminal names. The name of a group of service groups cannot coincide with the names of service groups for which grouping is not specified. A group name must be enclosed in double quotation marks (`"`).

Specifies that the queue contents of all service groups or logical terminals are to be dumped.

*character-string**

Specifies leading characters of a service group name or a logical terminal name when the queue contents of all service groups or logical terminals with names beginning with the specified character string are to be dumped.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`: `-g "seg1 Δ seg2 Δ seg3"`

<Example of specifying a group of service groups>

To specify all service groups beginning with `seg`, code: `-u "seg*"`

■ **-f** *dump-file-name*~<path name, 1-35 characters>

Specifies the name of the output dump file.

Specifying the file name alone creates a file under the `$DCDIR/spool` directory. Specifying the full path name creates a file under the specified directory. However, the command does not create a directory.

In Windows, you cannot specify the dump file name by an absolute path you can only the name of the file to be created under `$DCDIR/spool`.

■ **-a**

Specifies that the output queue contents are to be added to an existing file.

When this option is omitted, a new file is opened, and the queue contents are written into it.

Output messages

Message ID	Description	Output destination
KFCA10103-E	The specified logical terminal has not been shut down (or suspended).	Standard error output
KFCA10350-I	mcftdmpqu command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftdmpqu command input source failed.	Message log file
KFCA10372-E	mcftdmpqu command terminated abnormally.	Standard error output
KFCA10374-I	Unsent message is copied to the file.	Standard output
KFCA10375-E	Dump file name	Standard output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	The service group name is not registered.	Standard error output
KFCA10519-I	Help message	Standard error output
KFCA11096-W	Input queue cannot acquire the dump due to SPP resources.	Standard error output
KFCA11811-E	Error occurred during output of the file.	Standard error output
KFCA11812-E	Error was detected during read of the input/output queue.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftealt (Terminate alternate sending)

Format

```
mcftealt [-s MCF-communication-process-identifier]
          -l logical-terminal-name
```

Function

The `mcftealt` command terminates alternate sending.

This command can be used under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftealt` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.
- `-l logical-terminal-name`~<1-8 character identifier>

Specifies the name of the logical terminal which originates alternate sending. It must be the logical terminal name that is specified in the `-f` option of the `mcfstalt` command. Neither more than one name nor name of a group of service groups can be specified.

Output messages

Message ID	Description	Output destination
KFCA10117-I	Alternate sending of the message will terminate.	Standard output
KFCA10350-I	<code>mcftealt</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output

Message ID	Description	Output destination
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftealt command input source failed.	Message log file
KFCA10371-I	mcftealt command has been accepted normally.	Standard output
KFCA10373-E	mcftealt command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10542-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcfendct (Terminate forcibly continuous inquiry-response processing for logical terminals)

Format

```
mcfendct [-f] [-s MCF-communication-process-identifier]  
          -l logical-terminal-name
```

Function

The `mcfendct` command terminates forcibly continuous inquiry response processing for specified logical terminals.

The `mcfendct` command can be used under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

Options

■ `-f`

Specifies that continuous inquiry-response processing is to be terminated forcibly. When this option is specified, the temporary storage area for continuous inquiry is released immediately, and the MHP processing an inquiry (including an error event) from a corresponding logical terminal is terminated abnormally (the error event will not be started).

When this option is omitted and MHP is currently processing an inquiry from a corresponding logical terminal, an error message is output.

■ `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfendct` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

■ `-l logical-terminal-name~<identifier, 1-8 characters>`

Specifies the name of a logical terminal whose continuous inquiry-response processing is to be terminated.

A maximum of eight logical terminal names can be specified. When entering large

amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that continuous inquiry-response processing is to be terminated forcibly for all logical terminals.

*character-string**

Specifies leading characters of a logical terminal name when continuous inquiry response processing is to be terminated for all logical terminals with names beginning with the specified character string.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3, code: -1 "len1 Δ len2 Δ len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -1 "len*"

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfendct command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output

mcftendct (Terminate forcibly continuous inquiry-response processing for logical terminals)

Message ID	Description	Output destination
KFCA10359-W	Response to mcftendct command input source failed.	Message log file or standard error output
KFCA10371-I	mcftendct command has been accepted normally.	Standard output
KFCA10373-E	mcftendct command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output
KFCA10391-E	mcftendct command is not supported.	Standard error output
KFCA10521-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

Notes

MHP terminates abnormally in the following cases if the `-f` option is specified:

- During access to the temporary storage area for continuous inquiry-response processing (when `dc_mcf_tempget` function or `dc_mcf_tempput` function is issued)
- During a rollback request (when `dc_mcf_rollback` function is issued (action: DCMCFRTRY or DCMCFNRTN))

mcfthldiq (Hold process of input queue for service groups)

Format

```
mcfthldiq -g service-group-name [-k hold-type] [-r]
```

Function

The `mcfthldiq` command holds the process of input queues for the specified service group.

Using this command affects the OpenTP1 system as described in the notes below. Once this command is executed and the target processing terminates, be sure to execute the `mcftrlsiq` command to release the held process.

Options

- `-g service-group-name` ~<1-31 character identifier>

Specifies the name of a service group whose input queue process is to be held.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. Only one group can be specified. The name of a group of service groups cannot coexist with the names of service groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Holds input queues for all service groups.

*leading-string**

Holds input queues for all service groups beginning with the leading string.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`: `-g "seg1 Δ seg2 Δ seg3"`

<Example of specifying a group of service groups>

To specify all services beginning with `seg`, code: `-g "seg*"`

mcftldiq (Hold process of input queue for service groups)

- -k *hold-type*~<<insec>>

Specifies the type of holding.

insec

Both input and scheduling of input queues are held.

in

Only input of input queues is held.

sc

Only scheduling of input queues is held.

- -r

Specifies that the hold status of the service group not be inherited after full recovery.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftldiq command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftldiq command input source failed.	Message log file or standard error output
KFCA10371-I	mcftldiq command has been accepted normally.	Standard output
KFCA10373-E	mcftldiq command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	Specified service group has not been cataloged.	Standard error output
KFCA10523-I	Help message	Standard output
KFCA11072-W	A specified service group has been held.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output

Message ID	Description	Output destination
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11800-W	Input queue cannot be held or released from hold status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

Notes

- If you hold the input of the input queue for a service group which is started up with the MCF application start service, the MCF application start service holds the startup of all the service groups.
- If you hold the input of the input queue for a service group which the MCF communication service starts, the messages to be sent to the logical terminal of the input source also remain in the OTQ until the held process is released.
- If you hold the input of the input queue for a service group which the MCF communication service starts, the processing performance of the MCF communication service may be degraded.
- When there is at least one held service group, an attempt to perform normal termination, planned termination A, or planned termination B causes OpenTP1 to terminate abnormally. In this case, execute the `dcstop` command after releasing all the service groups from hold.
- If an attempt is made to input a message to the service group for which the input of the input queue is held, the operation command to the logical terminal of the input source and the connection to which the logical terminal belongs may result in a timeout. The operation command is accepted even if a timeout occurs. The operation command is executed after releasing the service group from hold.

mcfthldoq (Hold process of a logical terminal output queue)

Format

```
mcfthldoq [-s MCF-communication-process-identifier]
          [-l logical-terminal-name [-k hold-type]]
```

Function

The `mcfthldoq` command holds the process of a logical terminal output queue.

Using this command affects the OpenTP1 system as described in the notes below.

Once this command is executed and the target processing terminates, be sure to execute the `mcftrlsdq` command to release the held process.

Options

- `-s` *MCF-communication-process-identifier*

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfthldoq` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-l` *logical-terminal-name*~<1-8 character identifier>

Specifies the name of the logical terminal whose output queue process is to be held.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. Only one group can be specified. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Holds the output queue process for all logical terminals.

*leading-string**

Holds the output queue process for all logical terminals beginning with the leading string.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3: -1 "len1 Δ len2 Δ len3 "

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -1 "len* "

■ -k *hold-type*~<<insc>>

Specifies the type of holding.

insc

Both input and scheduling of output queues are held.

in

Only input of output queues is held.

sc

Only scheduling of output queues is held.

Output messages

Message ID	Description	Output destination
KFCA10104-W	Specified logical terminal has been held.	Standard error output
KFCA10122-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA10123-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA10350-I	mcfthldoq command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output

Message ID	Description	Output destination
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfthldoq command input source failed.	Message log file or standard error output
KFCA10371-I	mcfthldoq command has been accepted normally.	Standard output
KFCA10373-E	mcfthldoq command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10391-E	mcfthldoq command is not supported.	Standard error output
KFCA10525-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

Note

- If a UAP sends a message to a logical terminal for which the input of the output queue is held, it stops until the held process is released.
- For a UAP sending messages to multiple logical terminals, if the input of the output queue for one of the logical terminals is held, the messages to the other logical terminals may also be stopped until the held process is released.
- When there is at least one held service group, an attempt to perform normal termination, planned termination A, or planned termination B causes OpenTP1 to terminate abnormally. In this case, execute the `dcstop` command after releasing all the service groups from hold.

mcfllsbuf (Display utilization status of buffer groups)

Format

```
mcfllsbuf -s MCF-communication-process-identifier
          -b buffer-group-number
```

Function

The `mcfllsbuf` command outputs to the standard output the utilization status of buffer groups.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f>` ((01-ef))
Specifies an MCF communication process identifier.
No more than one MCF communication process identifier can be specified.
- `-b buffer-group-number<unsigned integer>` ((1-512))
Specifies the number of a buffer group whose utilization status is to be displayed.
No more than one buffer group number can be specified.
If all is specified, the utilization status of all buffer groups corresponding to the specified MCF communication process is displayed.

Output format

```
mmm ggg NNN XXX YYY ZZZ
```

mmm

MCF identifier

ggg

Buffer group number

NNN

Number of buffers

XXX

Number of buffers in use

YYY

Number of extended buffers

ZZZ

Number of extended buffers in use

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfllsbuf command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfllsbuf command input source failed.	Message log file
KFCA10360-I	Status display has started.	Standard output
KFCA10366-I	See the output format section above.	Standard output
KFCA10369-I	Status display has terminated.	Standard output
KFCA10372-E	mcfllsbuf command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10385-E	A specified buffer group number has not been cataloged.	Standard error output
KFCA10509-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfctlscn (Display status of connections)

Format

```
mcfctlscn [-s MCF-communication-process-identifier]
           {-c connection-ID|-g connection-group-name} [-d]
```

Function

The `mcfctlscn` command outputs to the standard output the status of connections.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target connection. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfctlscn` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-c connection-ID`~<identifier, 1-8 characters>

Specifies the ID of a connection whose status is to be displayed.

A maximum of eight connection IDs can be specified. When entering large amounts of inputs for the operation, specify more than one connection or specify a group of connections to increase the number of connections to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection IDs, separate them by a space and enclose the entire string of connection IDs in double quotation marks ("). The same connection ID cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection IDs. The ID of a group of connections cannot coexist with the IDs of connections for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the status of all connections is to be displayed.

*character-string**

Specifies leading characters of a connection ID when the status of all connections with IDs beginning with the specified character string is to be displayed.

<Example of specifying more than one connection>

To specify *cnn1*, *cnn2*, and *cnn3*, code: `-c "cnn1 Δ cnn2 Δ cnn3 "`

<Example of specifying a group of connections>

To specify all connections beginning with *cnn*, code: `-c "cnn* "`

■ `-g` *connection-group-name*~<identifier, 1-8 characters>

Specifies the name of a connection group whose status is to be displayed.

A maximum of eight connection group names can be specified. When entering large amounts of inputs for the operation, specify more than one connection group or specify a group of connection groups to increase the number of connection groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple connection group names, separate them by a space and enclose the entire string of connection group names in double quotation marks ("). The same connection group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of connection group names. The name of a group of connection groups cannot coexist with the names of connection groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Displays the status of all connection groups.

*character-string**

Specifies leading characters of a connection group name when the status of all connection groups with names beginning with the specified character string are to be displayed.

<Example of specifying more than one connection group>

To specify *cng1*, *cng2*, and *cng3*, code: `-g "cng1 Δ cng2 Δ cng3 "`

<Example of specifying a group of connection groups>

To specify all connection groups beginning with *cng*, code: `-g "cng* "`

Note that the `-g` option is valid only when the TP1/NET/OSI-TP protocol is used.

■ `-d`

Displays the connection status and the additional information about the logical terminal corresponding to each specified connection.

When this option is omitted, only the connection status is displayed.

Output format

<i>mmm</i>	<i>ccccccc</i>	<i>ppp</i>	<i>sssss</i>	<i>ddd</i>	<i>bbbbbbb</i>	...1.
	<i>lllllll</i>	<i>ttt</i>	<i>uuuu</i>	<i>uuuu</i>		...2.
	:	:	:	:		...2.
	<i>lllllll</i>	<i>ttt</i>	<i>uuuu</i>	<i>uuuu</i>		...2.

1. Only this line is output when the `-d` option is omitted.
2. The additional information about the logical terminal corresponding to the connection.

mmm

MCF identifier

ccccccc

Connection ID

ppp

Protocol type

sssss

Connection status:

ACT: Shutdown released status

ACT/B: Under shutdown release processing

DCT: Shutdown status

DCT/B: Under shutdown processing

ddd

Detailed status (maintenance information)

bbbbbbb

The contents of display differ depending on the protocol being used. For the contents of display, see the applicable protocol manual.

lllllll, ttt, uuuu

Displayed only when the `-d` option is specified. The display differs depending on the protocol. For the display, see the manual of the protocol.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcfctlscn command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfctlscn command input source failed.	Message log file
KFCA10360-I	Status display has started.	Standard output
KFCA10361-I	Standard information	Standard output
KFCA10362-I	Detailed information	Standard output
KFCA10369-I	Status display has terminated.	Standard output
KFCA10373-E	mcfctlscn command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	A specified connection has not been cataloged.	Standard error output
KFCA10391-E	mcfctlscn command is not supported.	Standard error output
KFCA10502-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output
KFCA16424-E	Specified connection group has not been cataloged.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

mcftlscom (Reference MCF communication service status)

Format

```
mcftlscom [-s MCF-communication-process-identifier]
```

Function

The `mcftlscom` command references the status of MCF communication services.

When you replace some files of an MCF communication service, partially stop the MCF communication service and execute this command to check the completion of partial stop of the MCF communication service.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specify the MCF communication process identifier of the MCF communication service whose status you want to reference. You cannot specify multiple MCF communication process identifiers.

If you do not specify this option, the status of all MCF communication services is output.

Output format

```
KFCA16431-I bb cc...cc dd...dd ee...ee ff..ff gggg
```

bb

MCF communication process identifier

cc...cc

MCF communication server name

dd...dd

Process ID of the MCF communication server

ee...ee

Protocol type

ff..ff

Status of the MCF communication service

OFFLINE: The service is inactive.

STARTING: The service is preparing to start.

ONLINE: The service is active or preparing to end.

PREENDING: The service is preparing to end (this status is output only for a partial stop).

ENDING: The service is ending.

gggg

Maintenance information

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcftlscom</code> command has been entered.	Standard output
KFCA10351-E	The MCF is being started.	Standard error output
KFCA10352-E	The MCF is being ended.	Standard error output
KFCA10353-W	The input format is incorrect.	Standard error output
KFCA10354-E	Memory is insufficient.	Standard error output
KFCA10355-W	The specified argument is incorrect.	Standard error output
KFCA10356-E	A timeout error occurred between processes.	Standard error output
KFCA10357-E	A timeout error occurred in the MCF.	Standard error output
KFCA10358-E	An error occurred in an internal function.	Standard error output
KFCA10359-W	A response to the source of the <code>mcftlscom</code> command failed.	Message log file
KFCA10360-I	The status display for the MCF begins.	Standard output
KFCA10369-I	The status display for the MCF ends.	Standard output
KFCA10370-I	The <code>mcftlscom</code> command has been accepted successfully.	Standard output
KFCA10373-E	The <code>mcftlscom</code> command ended abnormally.	Standard error output
KFCA10380-E	A search for the target process failed.	Standard error output
KFCA10391-E	The <code>mcftlscom</code> command is not supported.	Standard error output
KFCA10558-I	Help message	Standard output
KFCA16402-E	An RPC error occurred during command processing.	Standard error output
KFCA16431-I	See the output format section above.	Standard output

mcftlsle (Display status of logical terminals)

Format

```
mcftlsle [-s MCF-communication-process-identifier]
          [-c connection-ID] -l logical-terminal-name [-q]
```

Function

The `mcftlsle` command outputs to the standard output the status of logical terminals.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftlsle` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-c` *connection-ID*~<identifier, 1-8 characters>

Specifies the connection ID corresponding to a logical terminal whose status is to be displayed.

Neither more than one connection ID nor ID of a group of connections can be specified.

- `-l` *logical-terminal-name*~<identifier, 1-8 characters>

Specifies the name of a logical terminal whose status is to be displayed.

If the `-c` option is specified, the logical terminal name corresponding to the specified connection ID must be specified.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the status of all logical terminals is to be displayed.

*character-string**

Specifies leading characters of a logical terminal name when the status of all logical terminals with names beginning with the specified character string is to be displayed.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3, code: -l "len1 Δ len2 Δ len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -l "len*"

■ -q

Displays the held status of output queues for the specified logical terminal. When this option is omitted, the held status of output queues for the specified logical terminal is not displayed.

Output format

```
mmm llllll sss [ttt]
aaaaa bbbbbbbb
SYNC xxxxxxxxx yyyyyyyyyy zzzzzzzzzz
  IO      :          :          :
  PRIO    :          :          :
  NORM    :          :          :
iii ooo
```

mmm

MCF identifier

llllll

Logical terminal name

sss

Status of the logical terminal:

ACT: Shutdown released status

DCT: Shutdown status

ttt

Logical terminal test mode (displayed only when TP1/Message Control/Tester is used)

TEST: Test mode

(Blank) Non-test mode

aaaaa

Type of alternate sending (displayed when alternate sending is used)

ALT_F: Logical terminal of alternate origin

ALT_T: Logical terminal of alternate destination

bbbbbbb

Logical terminal name (displayed only when alternate sending is used)

If the applicable logical terminal is a terminal of alternate origin, the name of the logical terminal of alternate destination is displayed. In this case, ALT_T is displayed in aaaaa.

SYNC

Synchronous-type message

IO

Asynchronous-type inquiry-response message

PRIO

Asynchronous-type one-way sent message (priority)

NORM

Asynchronous-type one-way sent message (normal)

xxxxxxxxx

Number of unsent messages (displayable up to the upper limit of `unsigned short`)

yyyyyyyyy

Lowest serial number of unsent messages (displayable up to the upper limit of `int`)

zzzzzzzzz

Highest serial number of unsent messages (displayable up to the upper limit of `int`)

iii

mcftlsle (Display status of logical terminals)

Held status of the input of output queues (displayed only when the -q option is specified)

NOH: Held status is released.

HLD: Held.

ooo

Held status of the scheduling of output queues (displayed only when the -q option is specified)

NOH: Held status is released.

HLD: Held.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftlsle command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftlsle command input source failed.	Message log file
KFCA10360-I	Status display has started.	Standard output
KFCA10364-I	See the output format section above.	Standard output
KFCA10365-I	See the output format section above.	Standard output
KFCA10369-I	Status display has terminated.	Standard output
KFCA10373-E	mcftlsle command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10381-E	A specified connection ID has not been cataloged.	Standard error output
KFCA10382-E	A specified logical terminal has not been cataloged.	Standard error output

Message ID	Description	Output destination
KFCA10390-E	Correspondence between specified connection ID and logical terminal name is invalid.	Standard error output
KFCA10391-E	mcfctlslsle command is not supported.	Standard error output
KFCA10395-E	The name of an unconnected logical terminal is specified in specified connection.	Standard error output
KFCA10505-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfctlsln (Display network status)

Format

```
mcfctlsln [-s MCF-communication-process-identifier] [-t]
```

Function

The `mcfctlsln` command displays the status of a network.

Options

- `-s` *MCF-communication-process-identifier* ~<hexadecimal>((01-ef))

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfctlsln` command is executed for all MCFs.

- `-t`

Specifies that the status of the network related to exchange of messages with a remote system is displayed.

When this option is omitted, the display is of the acceptance status of requests to establish a server-type connection.

Output format

```
mmm uuuuuu
```

- *mmm*: MCF identifier
- *uuuuuu*: Network information.

The display depends on the protocol; for details, see the applicable *OpenTPI Protocol* manual.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcfctlsln</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output

Message ID	Description	Output destination
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Timeout occurred during a process.	Standard error output
KFCA10357-E	Timeout occurred in the MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcfctlsln</code> command input source failed.	Message log file or standard error output
KFCA10360-I	Status display is starting.	Standard output
KFCA10369-I	Status display is terminating.	Standard output
KFCA10372-E	The <code>mcfctlsln</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of a remote process failed.	Standard error output
KFCA10391-E	The <code>mcfctlsln</code> command is not supported.	Standard error output
KFCA10561-I	Help message	Standard output
KFCA16402-E	RPC error occurred.	Standard error output
KFCA16432-I	Network information is being displayed.	Standard output

mcftlssg (Display status of service groups)

Format

```
mcftlssg -g service-group-name [-q] [-m]
```

Function

The `mcftlssg` command displays the status of service groups and the numbers of received messages currently queued.

Options

- `-g service-group-name`~<identifier, 1-31 characters>

Specifies the name of a service group whose status is to be displayed.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service group names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the status of all service groups is to be displayed.

*character-string**

Specifies leading characters of a service group name when the status of all service groups with names beginning with the specified character string is to be displayed.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`, code: `-1 "seg1 Δ seg2 Δ seg3"`

<Example of specifying a group of services>

To specify all services beginning with `seg`, code: `-1 "seg*"`

- -q

Displays the held status of input queues for the specified service group. When this option is omitted, the held status of input queues for the service group is not displayed.

- -m

Displays the maximum number of unprocessed receive messages from when OpenTP1 was started normally or restarted until the current time.

Output format

<pre> aaaaaaaaaa bbb cc...cc ddd eee fff[(mmm)] [ttt] aaaaaaaaaa iii ooo </pre>
--

aaaaaaaaaa

Message ID

bbb

MCF identifier

cc...cc

Service group name

ddd

Status of the service group (input):

ACT: Shutdown released status

DCT: Shutdown status

***: Displayed for SPP service groups.

eee

Status of the service group (scheduling):

ACT: Shutdown released status

DCT: Shutdown status

fff

Number of unprocessed receive messages in the input queue corresponding to the applicable service group (displayable up to the upper limit of `int`)

mmm

Maximum number of unprocessed receive messages in the input queue corresponding to the applicable service group (displayed only when the `-m` option is specified; displayable up to the upper limit of `int`)

ttt

Test mode status of service group (displayed only when TP1/Message Control/Tester is used)

TEST: Test mode

(Blank): Non-test mode

iii

Held status of the input of input queues (displayed only when the -q option is specified)

NOH: Held status is released.

HLD: Held.

***: Displayed for application service groups

ooo

Held status of the scheduling of input queues (displayed only when the -q option is specified)

NOH: Held status is released.

HLD: Held.

***: Displayed for application service groups

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftlssg command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftlssg command input source failed.	Message log file or standard error output
KFCA10360-I	Status display has started.	Standard output
KFCA10367-I	See the output format section above.	Standard output

Message ID	Description	Output destination
KFCA10369-I	Status display has terminated.	Standard output
KFCA10373-E	mcftrlssg command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	Specified service group has not been cataloged.	Standard error output
KFCA10515-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

Note

When a disk queue is used for the input queue and unprocessed receive messages remain prior to an OpenTP1 restart, the number of unprocessed receive messages becomes the maximum number of unprocessed receive messages during the OpenTP1 restart.

mcfsslsv (Display status of services)

Format

```
mcfsslsv -v service-name
```

Function

The `mcfsslsv` command displays the input and scheduling status of specified services.

Options

- `-v service-name~<identifier, 1-31 characters>`

Specifies the name of a service whose status is to be displayed.

A maximum of eight service names can be specified. When entering large amounts of inputs for the operation, specify more than one service or specify a group of services to increase the number of services to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service names, separate them by a space and enclose the entire string of service names in double quotation marks ("). The same service name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. The name of a group of services cannot coexist with the names of services for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Specifies that the input and scheduling status of all services is to be displayed.

*character-string**

Specifies leading characters of a service name when the input and scheduling status of all services with names beginning with the specified character string is to be displayed.

<Example of specifying more than one service>

To specify `ser1`, `ser2`, and `ser3`, code: `-l "ser1 Δ ser2 Δ ser3"`

<Example of specifying a group of services>

To specify all services beginning with `ser`, code: `-l "ser*"`

Output format

<i>aaaaaaaaaa bbb cc...cc ddd eee</i>

aaaaaaaaaa

Message ID

bbb

MCF identifier

cc...cc

Service name

ddd

Status of the service (input):

ACT: Shutdown released status

DCT: Shutdown status

***: Application service

eee

Status of the service (scheduling):

ACT: Shutdown released status

DCT: Shutdown status

***: Application service

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftrlssv command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftrlssv command input source failed.	Message log file or standard error output

mcftlssv (Display status of services)

Message ID	Description	Output destination
KFCA10360-I	Status display has started.	Standard output
KFCA10368-I	See the output format section above.	Standard output
KFCA10369-I	Status display has terminated.	Standard output
KFCA10373-E	mcftlssv command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10384-E	Specified service has not been cataloged.	Standard error output
KFCA10512-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftlstrd (Display the status of multi-processing of messages)

Format

```
mcftlstrd -s MCF-communication-process-identifier
```

Function

The `mcftlstrd` command displays the status of the multi-processing of messages.

The MCF communication process can simultaneously handle messages it receives from multiple connections at the same time and messages that a user application program requests to send to multiple connections at the same time. You can change the limit of multiplicity, which allows parallel processing of messages, in the maximum processing multiplicity definition (`mcfttred -m`). However, even if a high traffic level is enabled, parallel processing may not be executed up to the value specified in the definition depending on the system environment. In this case, local memory is used inefficiently, negatively influencing the performance (as you increase the value specified in the maximum processing multiplicity definition by 1, the MCF communication process uses additional 32 kilobytes of local memory).

Use the `mcftlstrd` command to check the status of multi-processing and tune the value in the maximum processing multiplicity definition. For example, if the result of executing this command several times during peak traffic indicates:

1. "running" is always smaller than the current value specified in the processing multiplicity definition (number of connections if the value specified in the processing multiplicity definition is greater than the number of connections) and,
2. "waiting" is always greater than 0, you cannot expect multi-processing exceeding "running".

Change the value specified in the processing multiplicity definition to "running + α ". (α is a safety margin. The recommended value is 3.) The result of the command can indicate that "running" equals the current value specified in the processing multiplicity definition (number of connections if the value specified in the processing multiplicity definition is greater than the number of connections). If that occurs and "waiting" is greater than 0, you may be able to increase the multiplicity. Consider a new multiplicity value based on the amount of memory and your performance requirements, such as the current throughput and the turnaround time.

To determine the processing status, obtain the MCF statistics.

Options

- `-s` *MCF-communication-process-identifier*

Specifies the MCF communication process identifier which is the target of the command.

This option is mandatory.

You can specify only one MCF communication process identifier. You cannot specify a group name using `*`.

Output format

```
KFCA16430-I aaa running=bbbb waiting=cccc limit=dddd
```

- *aaaa*: MCF identifier
- *bbbb*: Connection with the message being processed
- *cccc*: Number of connections with the messages waiting to be processed (displayable up to the upper limit of `int`)
- *dddd*: Value specified in the maximum processing multiplicity definition (`mcfttred -m`)
(number of connections defined if that number is smaller than the value specified in the maximum processing multiplicity definition)

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcftlstrd</code> command is entered.	Standard output
KFCA10351-E	The MCF is being started.	Standard error output
KFCA10352-E	The MCF is being terminated.	Standard error output
KFCA10353-W	The input format is incorrect.	Standard error output
KFCA10354-E	Memory is insufficient.	Standard error output
KFCA10355-W	The argument is specified incorrectly.	Standard error output
KFCA10356-E	A timeout occurred between processes.	Standard error output
KFCA10357-E	A timeout occurred in the MCF.	Standard error output
KFCA10358-E	An error occurred in the internal function.	Standard error output
KFCA10372-E	The <code>mcftlstrd</code> command is terminated abnormally.	Standard error output
KFCA10380-E	Searching for the destination process failed.	Standard error output

Message ID	Description	Output destination
KFCA10555-I	Help message	Standard output
KFCA16402-E	An RPC error occurred during command processing.	Standard error output
KFCA16430-I	See the output format section above.	Standard output

mcfctlsum (Display user timer monitoring status)

Format

```
mcfctlsum [-s MCF-communication-process-identifier]
           { -a application-name | -l logical-terminal-name }
```

Function

The `mcfctlsum` command displays the status of user timer monitoring.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies an MCF communication process identifier. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfctlsum` command is executed on all MCFs. This means that the overhead required to search for the MCF communication processes is added to the processing load of the operation commands.

If you want to have a configuration that includes many MCF communication processes or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-a application-name~<identifier, 1-8 characters>`

Specifies the name of an application whose information is to be displayed.

A maximum of eight application names can be specified. When you enter large numbers of inputs for the operation, specify more than one application or specify a group of applications, as described below, to increase the number of applications processed in parallel by one command and to reduce the number of commands to be entered.

To specify multiple application names, separate them by a space and enclose the entire string of application names in double quotation marks ("). The same application name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of applications. Only one group can be specified. The name of a group of applications cannot coexist with the names of applications for which no group is specified. A group name must be enclosed in double quotation marks (").

- `*`: Specifies that the user timer monitoring status of all applications is displayed.
- `character-string*`: Specifies leading characters of an application name when the display is limited to the user timer monitoring status of applications whose names

begin with the specified character string.

<Example of specifying more than one application>

To specify `apl1`, `apl2`, and `apl3`:

```
-a "apl1 Δapl2 Δapl3 "
```

<Example of specifying a group of applications>

To specify all applications beginning with `apl`:

```
-a "apl*"
```

■ -l *logical-terminal-name*~<identifier, 1-8 characters>

Specifies the name of a logical terminal whose information is to be displayed.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals, as described below, to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. Only one group can be specified. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

- *: Specifies that the user timer monitoring status of all logical terminals is to be displayed.
- *character-string**: Specifies leading characters of a logical terminal name when the display is to be limited to the user timer monitoring status of logical terminals whose names begin with the specified character string.

<Example of specifying more than one logical terminal>

To specify `len1`, `len2`, and `len3`:

```
-l "len1 Δlen2 Δlen3 "
```

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with `len`:

```
-l "len*"
```

Output format

KFCA10360-I Starts status display of UTM.

mcftlsutm (Display user timer monitoring status)

```
KFCA16433-I mmm aaaaa (bbbb) cccc  
KFCA16434-I mmm dd...dd ee...ee ff..ff gg...gg hh...hh  
KFCA16434-I mmm dd...dd ee...ee ff..ff gg...gg hh...hh  
KFCA10369-I Terminates status display of UTM.
```

- UTM: User timer monitoring
- *mmm*: MCF identifier
- *aaaaa*: Number of registered user timers (5 digits)
- *bbbb*: Maximum number of registered user timers from when the communication process was started normally or restarted until the current time (5 digits)
- *cccc*: Maximum number of user timers that can be monitored concurrently (5 digits)
- *dd...dd*: Application name specified during user timer registration (8 digits)
- *ee...ee*: Time at which the user timer was registered (*hh:mm:ss*)
- *ff..ff*: Logical terminal name specified during the user timer registration (8 digits)
- *gg...gg*: Timer request identifier specified during the user timer registration (8 digits (hexadecimal))
- *hh...hh*: Start time specified during the user timer registration (*hh:mm:ss*)
If 360000 is specified, 100:00:00 is displayed.

Display examples

- When user timers are registered in the communication process:

```
KFCA10360-I Starts status display of UTM.  
KFCA16433-I A01 30(100) 10000  
KFCA16434-I A01 ap001 05:00:59 lec00101 00000001 00:00:50  
KFCA16434-I A01 ap001 23:59:59 lec001 00000002 99:00:00  
KFCA16434-I A01 ap001001 00:01:20 lec00101 00003600 100:00:00  
KFCA10369-I Terminates status display of UTM.
```

- When user timers are not registered in the communication process:

```
KFCA10360-I Starts status display of UTM.  
KFCA16433-I A01 30(100) 10000  
KFCA16435-I A01 User timer specified by MCF operation command is  
not registered. command-name = mcftlsutm  
KFCA10369-I Terminates status display of UTM.
```

- When user timers are not registered in the communication process:

KFCA10360-I Starts status display of UTM.
 KFCA16438-E A01 MCF operation command cannot be used.
command-name = mcftlsutm
 KFCA10369-I Terminates status display of UTM.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The mcftlsutm command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway. Wait for the startup processing to be completed, and then re-enter the command.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid. Correct the input format and re-enter the command.	Standard error output
KFCA10354-E	Memory is inadequate. If the same result occurs when the command is re-entered, check the system environment.	Standard error output
KFCA10355-W	Specification of an argument is invalid. Check the specification of the MCF communication process identifier, the application names, and the logical terminal names, make corrections, and then re-enter the command.	Standard error output
KFCA10356-E	Timeout occurred during a process. See the message that was output to a log before this message.	Standard error output
KFCA10357-E	Timeout occurred in the MCF. See the message that was output to a log before this message.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftlsutm command input source failed.	Message log file
KFCA10360-I	Status display is starting.	Standard output
KFCA10369-I	Status display is terminating.	Standard output
KFCA10372-E	The mcftlsutm command terminated abnormally. See the message that was output to a log before this message.	Standard error output
KFCA10380-E	Retrieval of a remote process failed. Check the MCF communication process identifier and re-enter the command.	Standard error output
KFCA10562-I	Help message	Standard output

mcftlsum (Display user timer monitoring status)

Message ID	Description	Output destination
KFCA16402-E	RPC error occurred during command processing. Take the corrective action indicated by the reason code.	Standard error output
KFCA16433-I	See the above output format.	Standard output
KFCA16434-I	See the above output format.	Standard output
KFCA16435-I	Either no user timers are registered for the application names or logical terminal names specified in the <code>mcftlsum</code> command, or a specified application name or logical terminal name is invalid. Check that user timers are registered or that the specified application names or logical terminal names are valid.	Standard output
KFCA16438-E	The <code>mcftlsum</code> command cannot be used in this environment. The setting that does not permit use of the user timer monitoring facility (<code>mcfttim usertime=no</code>) is specified in the timer definition of the MCF communication configuration definition. Check that the communication process specified in the command is valid. If it is valid, change to the setting that permits use of the user timer monitoring facility (<code>mcfttim usertime=yes</code>).	Standard error output

mcftofln (Stop accepting server-type connection establishment requests)

Format

```
mcftofln -s MCF-communication-process-identifier
```

Function

The `mcftofln` command stops acceptance of requests to establish a server-type connection.

Option

- `-s MCF-communication-process-identifier~<hexadecimal>` ((01-ef))

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcftofln</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Timeout occurred during a process.	Standard error output
KFCA10357-E	Timeout occurred in the MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcftofln</code> command input source failed.	Message log file
KFCA10370-I	The <code>mcftofln</code> command was accepted normally.	Standard output
KFCA10372-E	The <code>mcftofln</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of a remote process failed.	Standard error output
KFCA10391-E	The <code>mcftofln</code> command is not supported.	Standard error output

mcftofln (Stop accepting server-type connection establishment requests)

Message ID	Description	Output destination
KFCA10560-I	Help message	Standard output
KFCA16402-E	RPC error occurred.	Standard error output

In addition to the messages listed above, other messages might be output depending on the protocol being used. For details, see the manual for each protocol.

mcftonln (Start accepting server-type connection establishment requests)

Format

```
mcftonln -s MCF-communication-process-identifier
```

Function

The `mcftonln` command starts acceptance of requests to establish a server-type connection.

Option

- `-s MCF-communication-process-identifier~<hexadecimal> ((01-ef))`

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcftonln</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Timeout occurred during a process.	Standard error output
KFCA10357-E	Timeout occurred in the MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to <code>mcftonln</code> command input source failed.	Message log file
KFCA10370-I	The <code>mcftonln</code> command was accepted normally.	Standard output
KFCA10372-E	The <code>mcftonln</code> command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of a remote process failed.	Standard error output
KFCA10391-E	The <code>mcftonln</code> command is not supported.	Standard error output

mcftonln (Start accepting server-type connection establishment requests)

Message ID	Description	Output destination
KFCA10559-I	Help message	Standard output
KFCA16402-E	RPC error occurred.	Standard error output

In addition to the messages listed above, other messages might be output depending on the protocol being used. For details, see the manual for each protocol.

mcftrlsiq (Release held process of a service group input queue)

Format

```
mcftrlsiq -g service-group-name [-k hold-release-type]
```

Function

The `mcftrlsiq` command releases the held process of input queues for the specified service group.

Options

- `-g service-group-name`~<1-31 character identifier>

Specifies the name of a service group whose held input queue process is to be released.

A maximum of eight service group names can be specified. When entering large amounts of inputs for the operation, specify more than one service group or specify a group of service groups to increase the number of service groups to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple service group names, separate them by a space and enclose the entire string of service group names in double quotation marks ("). The same service group name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of service names. Only one group can be specified. The name of a group of service groups cannot coexist with the names of service groups for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Releases the held input queue process for all service groups.

*leading-string**

Releases the held input queue process for all service groups beginning with the leading string.

<Example of specifying more than one service>

To specify `seg1`, `seg2`, and `seg3`: `-g "seg1 Δ seg2 Δ seg3"`

<Example of specifying a group of service groups>

To specify all services beginning with `seg`, code: `-g "seg*"`

- `-k hold-release-type`~<<in>>

Specifies the type of releasing the held process.

mcftlsiq (Release held process of a service group input queue)

in sc

Both the held input and scheduling of input queues are released.

in

Only the held input of input queues is released.

sc

Only the held scheduling of input queues is released.

Output messages

Message ID	Description	Output destination
KFCA10350-I	mcftlsiq command has been entered.	Standard output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftlsiq command input source failed.	Message log file or standard error output
KFCA10371-I	mcftlsiq command has been accepted normally.	Standard output
KFCA10373-E	mcftlsiq command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10383-E	Specified service group has not been cataloged.	Standard error output
KFCA10524-I	Help message	Standard output
KFCA11073-W	A specified service group has been released from hold status.	Standard error output
KFCA11074-W	Service group will be shut down because the MHP user server is in completion process.	Standard error output
KFCA11075-W	Service group will be shut down because the MHP user has not been activated.	Standard error output
KFCA11076-W	Service group will be shut down because the MHP user server is on shutdown status.	Standard error output
KFCA11088-W	Command was processed normally, but status inherit cataloging failed.	Standard error output

Message ID	Description	Output destination
KFCA11089-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA11800-W	Input queue cannot be held or released from hold status due to SPP resources.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

Notes

The input queue that is held will already have been released when the message KFCA11074-W, KFCA11075-W, or KFCA11076-W is output.

mcftlsoq (Release held process of a logical terminal output queue)

Format

```
mcftlsoq [-s MCF-communication-process-identifier]  
         -l logical-terminal-name [-k hold-release-type]
```

Function

The `mcftlsoq` command releases the held process of a logical terminal output queue.

Options

- `-s` *MCF-communication-process-identifier*

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftlsoq` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

- `-l` *logical-terminal-name*~<1-8 character identifier>

Specifies the name of the logical terminal whose held output queue process is to be released.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

The asterisk (*) can be used in the specification to indicate a group of logical terminal names. Only one group can be specified. The name of a group of logical terminals cannot coexist with the names of logical terminals for which no grouping is specified. A group name must be enclosed in double quotation marks (").

*

Releases the held output queue process for all logical terminals.

*leading-string**

Releases the held output queue process for all logical terminals beginning with the leading string.

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3: -l "len1 len2 len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len, code: -l "len*"

■ -k *hold-release-type*~<<insec>>

Specifies the type of releasing the held process.

insec

Both input and scheduling of output queues are released.

in

Only input of output queues is released.

sc

Only scheduling of output queues is released.

Output messages

Message ID	Description	Output destination
KFCA10105-W	Specified logical terminal has been released.	Standard error output
KFCA10122-W	Command was processed normally, but status inherit cataloging failed.	Standard error output
KFCA10123-W	Command was processed normally, but status inherit cataloging failed because number of inherit statuses exceeded the maximum specified in the definition.	Standard error output
KFCA10350-I	mcctrlsoq command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output

mcftlsoq (Release held process of a logical terminal output queue)

Message ID	Description	Output destination
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftlsoq command input source failed.	Message log file or standard error output
KFCA10371-I	mcftlsoq command has been accepted normally.	Standard output
KFCA10373-E	mcftlsoq command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10391-E	mcftlsoq command is not supported.	Standard error output
KFCA10526-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcftspqle (Skip first message in a logical terminal message queue)

Format

```
mcftspqle -s MCF-communication-process-identifier
          -l logical-terminal-name -t message-type [-g]
```

Function

The `mcftspqle` command deletes a specified logical terminal's first unsent message of a specified message type.

Before executing the `mcftspqle` command, you must shut down the logical terminal with the `mcftdctle` command or the `dc_mcf_tdctle` function. If the TP1/NET/HNA-560/20 protocol is used, the session must also be terminated with the `mcftdctss` command.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specifies an MCF communication process identifier.

No more than one MCF communication process identifier can be specified.

- `-l logical-terminal-name<identifier, 1-8 characters>`

Specifies the name of a logical terminal whose first unsent message is to be deleted.

A maximum of eight logical terminal names can be specified. When entering large amounts of inputs for the operation, specify more than one logical terminal or specify a group of logical terminals to increase the number of logical terminals to be processed in parallel by one command while reducing the number of commands to be entered.

To specify multiple logical terminal names, separate them by a space and enclose the entire string of logical terminal names in double quotation marks ("). The same logical terminal name cannot be duplicated when specified.

Use an asterisk (*) to specify a group of logical terminal names. Only one group can be specified. Do not specify the name of a group of logical terminals with any other logical terminal names. Enclose a group name in double quotation marks (").

*

Deletes the first message for all the logical terminals.

*leading-string**

Deletes the first message for all logical terminals beginning with the leading string.

mcftspqle (Skip first message in a logical terminal message queue)

<Example of specifying more than one logical terminal>

To specify len1, len2, and len3: -1 "len1 len2 len3"

<Example of specifying a group of logical terminals>

To specify all logical terminals beginning with len: -1 "len*"

■ -t *message-type*

Specifies the type of the first unsent message to be deleted.

IO: Inquiry-response message

PRIO: One-way send message (priority)

NORM: One-way send message (normal)

■ -g

Deletes an untransmitted message group of the type specified in the -t option.

If there is no message group or the -g option is omitted, only the first untransmitted message of the type specified in the -t option is deleted.

Note that the -g option is valid only when TP1/NET/XMAP3 protocol is used.

Output messages

Message ID	Description	Output destination
KFCA10101-E	Specified logical terminal has not been shut down.	Standard error output
KFCA10102-E	Specified type of queue contains no messages.	Standard error output
KFCA10350-I	mcftspqle command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcftspqle command input source failed.	Message log file or standard error output

Message ID	Description	Output destination
KFCA10371-I	mcf tspqle command has been accepted normally.	Standard output
KFCA10373-E	mcf tspqle command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10391-E	mcf tspqle command is not supported.	Standard error output
KFCA10506-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfstalt (Start alternate sending)

Format

```
mcfstalt [-s MCF-communication-process-identifier]
         -f alternate-origin-logical-terminal-name
         -t alternate-destination-logical-terminal-name
```

Function

The `mcfstalt` command sends a message to an alternate logical terminal if the destination logical terminal fails.

This command is applicable under the TP1/NET/XMAP3 or TP1/NET/HNA-560/20 protocol.

Options

- `-s MCF-communication-process-identifier`~<numerals (0-9), a-f> ((01-ef))

Specifies the MCF communication process identifier of the MCF communication service controlling the target logical terminal. It can be specified only when the terminal of alternate origin and that of alternate destination are under the same process. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcfstalt` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.
- `-f alternate-origin-logical-terminal-name`~<1-8 character identifier>

Specifies the name of the logical terminal of alternate origin. The terminal to be specified must be on shutdown status due to an error. If the TP1/NET/HNA-560/20 protocol is specified, the session for the specified terminal must have been terminated. No more than one alternate origin logical terminal name can be specified. No name of a group of terminals can be specified, either.
- `-t alternate-destination-logical-terminal-name`~<1-8 character identifier>

Specifies the name of the logical terminal of alternate destination. A logical terminal name already specified as an alternate destination cannot be used. No logical terminal on shutdown status can be specified, either. If the TP1/NET/HNA-560/20 protocol is used, the session for the specified logical terminal must have been started. No more than one alternate origin logical terminal name can be specified. No name of a group

of terminals can be specified, either.

Output messages

Message ID	Description	Output destination
KFCA10116-I	Alternate sending of a message will start.	Standard output
KFCA10350-I	mcfstalt command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfstalt command input source failed.	Message log file
KFCA10371-I	mcfstalt command has been accepted normally.	Standard output
KFCA10373-E	mcfstalt command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10382-E	Specified logical terminal has not been cataloged.	Standard error output
KFCA10541-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

In addition to these messages, messages for the protocol being used are output; for details, see the applicable protocol manual.

Notes

- If, under the TP1/NET/HNA-560/20 protocol, the session for the logical terminal of alternate origin is started during alternate sending, the command terminates the sending process after the logical message arrives at the destination.
- Under the TP1/NET/HNA-560/20 protocol, the command terminates the sending process when the session for the logical terminal of alternate destination terminates.

mcfstart (Partial start of an MCF communication service)

Format

```
mcfstart -s MCF-communication-process-identifier
```

Function

The `mcfstart` command starts part of an MCF communication service.

When you replace some files of an MCF communication service, replace the files and execute this command to start part of the target MCF communication service.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specify the MCF communication process identifier of the MCF communication service that you want to start partially. You cannot specify multiple MCF communication process identifiers.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcfstart</code> command has been entered.	Standard output
KFCA10351-E	The MCF is being started.	Standard error output
KFCA10352-E	The MCF is being ended.	Standard error output
KFCA10353-W	The input format is incorrect.	Standard error output
KFCA10354-E	Memory is insufficient.	Standard error output
KFCA10355-W	The specified argument is incorrect.	Standard error output
KFCA10356-E	A timeout error occurred.	Standard error output
KFCA10358-E	An error occurred in an internal function.	Standard error output
KFCA10359-W	A response to the source of the <code>mcfstart</code> command failed.	Message log file
KFCA10370-I	The <code>mcfstart</code> command has been accepted successfully.	Standard output
KFCA10372-E	The <code>mcfstart</code> command ended abnormally.	Standard error output
KFCA10380-E	A search for the target process failed.	Standard error output
KFCA10391-E	The <code>mcfstart</code> command is not supported.	Standard error output

Message ID	Description	Output destination
KFCA16523-I	Partial start of the MCF communication service is being prepared.	Message log file
KFCA16524-I	The MCF communication service is now partially started.	Message log file
KFCA16525-E	An OpenTP1 end command was executed during the partial start of the MCF communication service.	Message log file
KFCA16527-E	The MCF communication service is not inactive.	Standard error output
KFCA16529-E	A feature that cannot be used together with this feature is being used.	Standard error output

Notes

- This command ends without waiting for the completion of the partial start of the MCF communication service.
- Partial start is only available for the MCF communication services (including the application startup service) that use the following protocol products. If you execute partial start of the MCF communication services that use other protocol products, the KFCA10391-E message appears.
 - TP1/NET/TCP/IP
 - TP1/NET/Secondary Logical Unit - TypeP2
 - TP1/NET/OSAS-NIF
- MCF communication services cannot be partially started in a system that uses independent rerun of MCF services or a system that provides remote MCF services. If you execute partial start in such systems, the KFCA16529-E message appears.
- User timer monitoring requests before partial stop of an MCF communication service are valid even if the target MCF communication service becomes partially stopped. User timer monitoring requests that are issued to the MCF communication service in partial stop status are also registered. Timer monitoring is not performed while the target MCF communication service is partially stopped. However, the partial stop is included in the elapsed time. Therefore, when timer monitoring begins when the target MCF communication service partially starts after a partial stop, the system may detect a timeout error and UAPs may be started.

mcftstop (Partial stop of an MCF communication service)

Format

```
mcftstop -s MCF-communication-process-identifier
```

Function

The `mcftstop` command stops part of an MCF communication service.

When you replace some files of an MCF communication service, execute this command before you replace the files and partially stop the target MCF communication service.

If this command is used incorrectly, the OpenTP1 system may end abnormally. Read the notes carefully before you use this command.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specify the MCF communication process identifier of the MCF communication service that you want to stop partially. You cannot specify multiple MCF communication process identifiers.

Output messages

Message ID	Description	Output destination
KFCA10350-I	The <code>mcftstop</code> command has been entered.	Standard output
KFCA10351-E	The MCF is being started.	Standard error output
KFCA10352-E	The MCF is being ended.	Standard error output
KFCA10353-W	The input format is incorrect.	Standard error output
KFCA10354-E	Memory is insufficient.	Standard error output
KFCA10355-W	The specified argument is incorrect.	Standard error output
KFCA10356-E	A timeout error occurred.	Standard error output
KFCA10358-E	An error occurred in an internal function.	Standard error output
KFCA10359-W	A response to the source of the <code>mcftstop</code> command failed.	Message log file
KFCA10370-I	The <code>mcftstop</code> command has been accepted successfully.	Standard output
KFCA10372-E	The <code>mcftstop</code> command ended abnormally.	Standard error output

Message ID	Description	Output destination
KFCA10380-E	A search for the target process failed.	Standard error output
KFCA10391-E	The <code>mcfstop</code> command is not supported.	Standard error output
KFCA16521-I	Partial stop of the MCF communication service is being prepared.	Message log file
KFCA16522-I	The MCF communication service is now partially stopped.	Message log file
KFCA16526-E	The MCF communication service is not online.	Standard error output
KFCA16528-E	A feature that cannot be used together with this feature is being used.	Standard error output

Notes

If this command is used incorrectly, the OpenTP1 system may end abnormally. Read the following notes carefully before you use the command.

- Before you enter this command, shut down the logical terminal for the target MCF communication service and release the connection to the MCF communication service. If you enter this command while the MCF communication service is sending or receiving messages, the command may not end or the MCF communication service may end abnormally.
- This command ends without waiting for the completion of the partial stop of the target MCF communication service.
- Partial stop is only available for the MCF communication services (including the application startup service) that use the following protocol products. If you execute partial stop of the MCF communication services that use other protocol products, a KFCA10391-E message appears.
 - TP1/NET/TCP/IP
 - TP1/NET/Secondary Logical Unit - TypeP2
 - TP1/NET/OSAS-NIF
- MCF communication services cannot be partially stopped in a system that uses independent rerun of MCF services or a system that provides remote MCF services. If you execute partial stop in such systems, a KFCA16528-E message appears.
- User timer monitoring requests before partial stop of an MCF communication service are valid even if the target MCF communication service becomes partially stopped. User timer monitoring requests issued to the MCF communication service in partial stop status are also registered. Timer monitoring is not executed while the target MCF communication service is partially stopped. However, the

partial stop is included in the elapsed time. Therefore, when timer monitoring begins when the MCF communication service partially starts after partial stop, the system may detect a timeout error and UAPs may be started.

- If you partially stop an MCF communication service when there is a held logical terminal or service group, OpenTP1 ends abnormally. Release all held logical terminals and service groups before you partially stop an MCF communication service.

mcfstptr (End acquiring an MCF trace)

Format

```
mcfstptr -s MCF-communication-process-identifier
```

Function

The `mcfstptr` command ends the acquisition of an MCF trace and outputs the acquired trace data to the MCF trace file.

Options

- `-s MCF-communication-process-identifier<numerals (0-9), a-f> ((01-ef))`

Specify the identifier of the MCF communication process for which you want to end acquiring an MCF trace.

Output messages

Message ID	Description	Output destination
KFCA10261-I	Acquisition of trace data is ended.	Message log file
KFCA10264-W	The command is ignored since trace data is being output.	Message log file
KFCA10265-E	Command processing is canceled since the output to the trace file failed.	Message log file
KFCA10266-W	Acquisition of trace data is not started.	Message log file

Notes

- You must execute the `mcfstrtr` command (command for starting the acquisition of an MCF trace) before you execute the `mcfstptr` command. If you do not execute the `mcfstrtr` command, an error occurs and an error message is output. However, if you have specified the use of the disk output feature for MCF traces in the MCF communication configuration definition (`mcftrc -t "disk=yes"`), you do not need to execute the `mcfstrtr` command before you execute the `mcfstptr` command. In this case, the disk output feature stops after trace data is output to the MCF trace file. To enable the disk output feature again, execute `mcfstrtr`.
- If you repeatedly execute the `mcfstptr` command with the same MCF communication process identifier in the `-s` option, an error occurs and an error message is output.
- If a trace buffer swap occurs during acquisition of trace data after you execute this

`mcfstptr` (End acquiring an MCF trace)

command and an MCF trace end request is generated, the MCF trace data is not output to the trace file. To output the MCF trace data, use the command for starting the acquisition of an MCF trace (`mcfststrtr`).

mcfstrtr (Start acquiring an MCF trace)

Format

```
mcfstrtr -s MCF-communication-process-identifier
```

Function

The `mcfstrtr` command starts the acquisition of an MCF trace.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specify the identifier of the MCF communication process for which you want to start identifier of the acquiring an MCF trace.

Output messages

Message ID	Description	Output destination
KFCA10260-I	Acquisition of trace data is started.	Message log file
KFCA10262-E	Preparation for acquiring trace data failed.	Message log file
KFCA10263-W	A trace acquisition start request is accepted while trace data is being acquired.	Message log file
KFCA10269-E	Output of trace data to the trace file is canceled since an error occurred during I/O with the trace file.	Message log file
KFCA10273-E	An error occurred during I/O with the trace file.	Message log file

Notes

- The acquisition of an MCF trace starts a little later than the point the `mcfstrtr` command is executed since the start is set after control is transferred from the `mcfstrtr` command to the MCF communication process.
- If you repeatedly execute the `mcfstrtr` command with the same MCF communication process identifier specified in the `-s` option, an error occurs and an error message is output.
- Even if you have defined not to use the disk output feature for MCF traces in the MCF communication configuration definition (`mcftrc -t "disk=no"`), the disk output feature is used until the command for ending the acquisition of an MCF trace (`mcfstptr`) is executed if you execute the `mcfstrtr` command.

mcftswptr (Swap MCF trace files forcibly)

Format

```
mcftswptr [-s MCF-communication-process-identifier]
```

Function

The `mcftswptr` command swaps forcibly the MCF trace files.

Options

- `-s MCF-communication-process-identifier~<numerals (0-9), a-f> ((01-ef))`

Specifies the MCF communication process identifier of the MCF communication service controlling the target connection. No more than one MCF communication process identifier can be specified.

If this option is omitted, the `mcftswptr` command is executed on all MCFs. This means that the overhead required to search for the MCF communication services is added to the processing load of the operation commands.

If you want to have a configuration that includes lots of MCF communication services or if you want to use many operation commands, use the `-s` option to specify an MCF communication process identifier.

Output messages

Message ID	Description	Output destination
KFCA10264-W	The command is ignored because output to the trace file is in progress.	Standard error output
KFCA10266-W	Acquisition of trace data has not started.	Message log file
KFCA10267-W	Output from the trace file is not performed because output to disk is not specified in the trace environment definition.	Standard error output
KFCA10278-I	Swapping of trace files has ended.	Message log file
KFCA10279-W	Swapping of trace files failed.	Standard error output
KFCA10280-W	The command is ignored because output of the trace file is disabled.	Standard error output
KFCA10350-I	<code>mcftswptr</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output

Message ID	Description	Output destination
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output
KFCA10359-W	Response to mcfswptr command input source failed.	Message log file
KFCA10370-I	mcfswptr command has been accepted normally.	Standard output
KFCA10372-E	mcfswptr command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10508-I	Help message	Standard output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

mcfuevt (Start application programs)

Format

```
mcfuevt -s MCF-communication-process-identifier
        -t input-message
```

Function

The `mcfuevt` command starts the application (UCMDEVT) specified in the application attribute definition and passes the input message specified in the `-t` option.

Options

- `-s` *MCF-communication-process-identifier*~<numerals (0-9), a-f> ((01-ef))

Specifies an MCF communication process identifier. No more than one MCF communication process identifier can be specified.

Because UCMDEVT does not rely on a protocol, it is recommended to specify an MCF application startup process identifier. This identifier must be the same as that specified in the `mcftevt` definition command (MCF environment definition) of the MCF communication configuration definition.

- `-t` *input-message*~<string of 1-80 characters>

Specifies the input message to be passed to UCMDEVT, enclosed in double quotation marks ("). One double-byte character is counted as two characters.

Output messages

Message ID	Description	Output destination
KFCA10350-I	<code>mcfuevt</code> command has been entered.	Standard output
KFCA10351-E	MCF startup processing is underway.	Standard error output
KFCA10352-E	MCF termination processing is underway.	Standard error output
KFCA10353-W	Input format is invalid.	Standard error output
KFCA10354-E	Memory is inadequate.	Standard error output
KFCA10355-W	Specification of an argument is invalid.	Standard error output
KFCA10356-E	Overtime occurred during process.	Standard error output
KFCA10357-E	Overtime occurred in MCF.	Standard error output
KFCA10358-E	Error occurred in an internal function.	Standard error output

Message ID	Description	Output destination
KFCA10359-W	Response to mcfuevt command input source failed.	Message log file
KFCA10370-I	mcfuevt command has been accepted normally.	Standard output
KFCA10372-E	mcfuevt command terminated abnormally.	Standard error output
KFCA10380-E	Retrieval of remote process failed.	Standard error output
KFCA10393-E	No application (UCMDEVT) exists.	Standard error output
KFCA10394-E	Length of an input message is invalid.	Standard error output
KFCA10532-I	Help message	Standard output
KFCA11513-E	Specification of an operand in definition command is invalid.	Standard error output
KFCA16402-E	RPC error occurred during command processing.	Standard error output

namalivechk (Verify OpenTP1 startup and clear cache)

Format

```
namalivechk {-l|-c}  
[-t length-of-time-to-wait-until-connection-is-established]
```

Function

The `namalivechk` command checks whether OpenTP1 is active in the OpenTP1 nodes specified in the `all_node` operand or `all_node_ex` operand in the system common definition, as output by the `-l` option.

The name of the OpenTP1 node whose start was not detected is output to the error log file, the standard error output, and the `syslog` file. In addition, all the service information of the OpenTP1 node whose start was not detected is deleted from the cache and the connection with the node is ended.

This command can be executed only when OpenTP1 is active.

Options

■ `-l`

Outputs to the standard error output the service information for every OpenTP1 node to be checked. OpenTP1 nodes to be checked for startup are the nodes specified in the `all_node` operand in the system common definition, which OpenTP1 has not detected to be inactive, and the nodes specified in the `all_node_ex` operand in the system common definition.

This option cannot coexist with the `-c` option.

■ `-c`

Checks whether or not the OpenTP1 nodes output by the `-l` option are active. Information on the OpenTP1 nodes that are not determined to be active is output to the error log file and to the standard error output. Among the OpenTP1 nodes specified in the `all_node` operand in the system common definition, OpenTP1 nodes that were not detected to be active are registered in an RPC suppression list. The service information for these OpenTP1 nodes and the OpenTP1 nodes specified in the `all_node_ex` operand in the system common definition is deleted from the cache. An RPC suppression list is a list of information maintained by the OpenTP1 system about inactive OpenTP1 nodes.

This option cannot coexist with the `-l` option.

- `-t length-of-time-to-wait-until-connection-is-established-<unsigned integer>`
 ((1-65535))<<8>> (unit: seconds)

OpenTP1 nodes whose activation cannot be confirmed within the length of time specified in this option are determined to be inactive. If you specify a value outside the allowed range, 8 seconds is assumed.

Output format

- (1) When `namalivechk -l` is executed

No.	Node to be checked	Port number
<i>aaa</i>	<i>bb...bb</i>	<i>cccc</i>

aaa

OpenTP1 node item name (1- to 3-digit integer)

bb...bb

Name of OpenTP1 node subjected to activation check (1 to 32 alphanumeric characters)

cccc

Port number of the OpenTP1 node subjected to activation check (1 to 5 integer)

- (2) When `namalivechk -c` is executed

No.	Node not determined to be active	Port number
<i>aaa</i>	<i>bb...bb</i>	<i>cccc</i>

aaa

OpenTP1 node item name (1 to 3 integer)

bb...bb

Name of OpenTP1 node not determined to be active (1 to 32 alphanumeric characters)

cccc

Port number of the OpenTP1 node not determined to be active (1 to 5 integer)

Note

If you have specified nodes that have the same IP address in the `all_node` operand or the `all_node_ex` operand in the system common definition, do not execute the `namalivechk` command.

namblad (Manipulate RPC suppression list of OpenTP1 nodes)

Format

```
namblad {-a|-d} node-name[:port-number] [,node-name[:port-number] . . .]
```

Function

The `namblad` command adds the specified nodes to the RPC suppression list or deletes the specified nodes from the RPC suppression list.

Options

- `-a`
Adds the nodes specified in the command argument to the RPC suppression list.
- `-d`
Deletes the nodes specified in the command argument from the RPC suppression list.

Command arguments

- *node-name*~<1-to-255-character identifier>
Specify the name of a node to be added to or deleted from the RPC suppression list. Specify a node name specified in the `all_node` operand in the system common definition or the domain definition file under `$DCCONFPATH/dcnamnd`. If you specify a node name that is not specified in the `all_node` operand, an error occurs and a `KFCA00675-E` message appears.
- *port-number*~<unsigned integer> ((5001-65535))
Specify a port number corresponding to a node name specified in the `all_node` operand in the system common definition and the domain definition file created under `$DCCONFPATH/dcnamnd`. If you do not specify a port number, the value specified for the `name_port` operand in the system common definition is assumed.

Output messages

Message ID	Description	Output destination
KFCA00674-I	Help message	Standard output
KFCA00675-E	The specified node is not defined in <code>all_node</code> .	Standard error output
KFCA00676-E	The specified node is already specified.	Standard error output

Notes

- When you execute this command simultaneously with the RPC processing executed by a UAP, the performance of the RPC executed by the UAP may be affected.
- When you specify the `-a` option, the command adds the node specified in the command to the RPC suppression list. When the name service detects the startup of the node specified in the command, the name service may delete the detected node from the RPC suppression list. In that case, if the deleted node is no longer in the RPC suppression list, the `namsvinf` or another command may display the name of the node.
- When you execute this command, the service information for the service group running on the node specified in the command is deleted from the cache. When this happens, the following may occur:

When the same service group is running on the node on which the `namblad` command is executed and on other nodes

Even though the node has been deleted from the RPC suppression list by the `namblad` command or the name service, RPC processing might not be executed for the service group running on the applicable node for a maximum of 30 minutes.

When the same service group is not running on the node on which the `namblad` command is executed and other nodes

When the node is deleted from the RPC suppression list by the `namblad` command or the name service, the RPC that is executed after the deletion searches for service information, thereby possibly affecting the performance of the RPC.

namchgfl (Change domain configuration (by using the domain definition files))

Format

```
namchgfl [-n] [-d [-t time-for-monitoring-established-connection]] [-e]
```

Function

The `namchgfl` command changes the OpenTP1 domain configuration while OpenTP1 is running. Define the new domain configuration (arrangement of nodes) in the domain definition files for `all_node` and `all_node_ex`, and then execute the `namchgfl` command. The domain configuration in the domain definition files takes effect when the command successfully terminates. For details about the domain definition files, see *3.10.1(2) Using the `namchgfl` command to change the domain configuration*.

The `namchgfl` command is available only when `Y` is specified in the `name_domain_file_use` operand in the system common definition. If you specify `N` in the `name_domain_file_use` operand, the command results in an error. You can execute the command only when OpenTP1 is running.

Options

■ `-n`

When you specify this option, the shared memory is not changed.

Specify this option only to check whether the nodes specified in the domain definition file for `all_node` can be stored in the area managed by the name service in the shared memory without changing the shared memory. When you specify the `-e` option, OpenTP1 checks the nodes specified in the domain definition file for `all_node_ex` as well. If all the nodes cannot be stored in the managed area in the shared memory, OpenTP1 outputs a message to the standard error output.

You cannot check the nodes that are specified in the `all_node` operand and the `all_node_ex` operand in the system common definition.

■ `-d`

Specify this option to check the startup status of the nodes specified in the domain definition file for `all_node`. OpenTP1 changes the node configuration only when all the nodes are active and the nodes specified in the domain definition file for `all_node` can be stored in the area managed by the name service in the shared memory. When you specify the `-e` option, OpenTP1 checks the nodes specified in the domain definition file for `all_node_ex` and includes those nodes in the new node configuration.

If there is an inactive node, OpenTP1 outputs information about the node to the standard output. However, depending on the startup status, OpenTP1 does not delete the service information from the area storing the service information of each node nor does it register or cancel the RPC suppression list. If there is no inactive node in the domain definition file for `all_node` and the domain definition file for `all_node_ex` (no node is specified or startup of all the nodes has been confirmed), OpenTP1 outputs no node information, including the header.

You cannot check the nodes specified in the `all_node` operand and the `all_node_ex` operand in the system common definition or include those nodes in the new node configuration.

Do not specify the `-d` option if you want to change the shared memory even though there is an inactive node among the nodes specified in the domain definition files.

- `-t` *time-for-monitoring-established-connection*~<unsigned integer>((1-65535)) <<8>> (units: seconds)

Determines that an OpenTP1 node is inactive if the startup of the OpenTP1 node is not confirmed within the specified period of time.

The following table shows the options to be specified when you want to check the size of the area managed by the name service in the shared memory, check the startup status of the nodes, and change the shared memory.

Check the size of the area managed by the name service in the shared memory	Check the startup status of the nodes	Change the shared memory	Options to be specified
Y	N	N	<code>namchgf1 -n</code>
Y	Y	N	<code>namchgf1 -n -d</code>
Y	N	Y	<code>namchgf1</code>
Y	Y	Y#	<code>namchgf1 -d</code>

Legend:

Y: Performed.

N: Not performed.

#: If the size of the area managed by the name service in the shared memory does not permit the storage of all the nodes in the domain definition files, the shared memory is not changed.

- `-e`

Checks the startup of the nodes specified in the domain definition file for `all_node`

namchgfl (Change domain configuration (by using the domain definition files))

and the domain definition file for `all_node_ex`, and includes the started nodes in the new domain configuration. If you do not specify the `-e` option, OpenTP1 only checks the startup of the nodes specified in the domain definition file for `all_node` and includes those started nodes in the new domain configuration.

Output format

The output format is as follows when `namchgfl -d -e` is specified:

all_node information		
No.	Inactive node	Port number
aaa	bb...bb	cccc
aaa	bb...bb	cccc
all_node_ex information		
No.	Inactive node	Port number
aaa	bb...bb	cccc
aaa	bb...bb	cccc

aaa

Item number of the node (3-digit integer or smaller)

bb...bb

Name of the inactive node (maximum of 32 alphanumeric)

cccc

Port number of the inactive node (5-digit integer or smaller)

Output messages

Message ID	Description	Output destination
KFCA00654-E	The value of the <code>all_node</code> operand specified in the domain definition file exceeded the permitted limit.	Standard error output
KFCA00655-E	The value of the <code>all_node_ex</code> operand specified in the domain definition file exceeded the permitted limit.	Standard error output
KFCA00656-E	The specified node name is incorrect.	Standard error output
KFCA00657-E	The specified port number is incorrect.	Standard error output
KFCA00658-W	The same node information is defined more than once.	Standard error output
KFCA00659-I	The command (<code>namchgfl</code>) is started.	Standard output
KFCA00660-I	A startup check (<code>namchgfl</code>) has started.	Standard output
KFCA00661-I	A startup check (<code>namchgfl</code>) has ended.	Standard output
KFCA00662-I	Rewriting of the domain configuration (<code>namchgfl</code>) has ended.	Standard output

Message ID	Description	Output destination
KFCA00663-E	The command (<code>namchgfl</code>) cannot be executed because <code>N</code> is specified for <code>name_domain_file_use</code> .	Standard error output
KFCA00665-E	An error occurred during analysis processing.	Standard error output
KFCA00666-E	The specified node name is undefined.	Standard error output
KFCA00669-W	The directory does not exist. OpenTP1 continues processing assuming that no operand is specified.	Standard error output
KFCA00670-I	No node is specified. The OpenTP1 system continues processing as a system that consists only of the local node.	Standard output
KFCA00671-I	The checking of the nodes specified in the domain definition file (<code>namchgfl</code>) has ended.	Standard output
KFCA00672-E	The command (<code>namchgfl</code>) ended abnormally.	Standard error output

Notes

- The `namchgfl` command can increase the number of nodes in the OpenTP1 system to make a larger configuration. You can increase the following node counts after OpenTP1 has started:
 - `all_node`
Can be increased up to the value specified for the `all_node_extend_number` operand in the system common definition (64 if the operand is omitted).
 - `all_node_ex`
Can be increased up to the value specified for the `all_node_ex_extend_number` operand in the system common definition (64 if the operand is omitted).
- To delete nodes from the domain definition file of `all_node`, stop all the SPPs running on the nodes before executing the `namchgfl` command. If you delete nodes while SPPs are active, a transaction processing error or other error might occur.
- OpenTP1 outputs message KFCA00672-E and terminates the command abnormally in the following cases:
 - A command argument is specified incorrectly.
 - An item is specified incorrectly in a domain definition file.
 - There is not enough room in the area that the name service manages in shared memory to store all of the nodes specified in a domain definition file.

namchgfl (Change domain configuration (by using the domain definition files))

- There is at least one inactive node among the nodes specified in a domain definition file.

namdomainsetup (Register and delete domain representative schedule service)

Format

```
namdomainsetup [-d] domain-data-file-name host-name
```

Function

The `namdomainsetup` command registers up to three domain representative schedule services in the domain data file. Also, this command deletes a domain representative schedule service from the domain data file.

Options

- `-d`

Deletes, from the domain data file, the domain representative schedule service activated at the specified host name.

Command arguments

- *domain-data-file-name*~<path name>

Specifies the domain data file name with the full path name. The domain data file is a hosts information file managed by DNS.

- *host-name*~<1-to-255-character identifier>

Specifies the name of the host where the domain representative schedule service will be activated. Do not specify the alias of the host.

Notes

- The `namdomainsetup` command can be executed by only the superuser.
- After the `namdomainsetup` command is executed, restart DNS.

namndchg (Change domain configuration (by using the system common definition))

Format

```
namndchg [-l]
```

Function

The `namndchg` command changes the OpenTP1 domain configuration while OpenTP1 is active. Define the new domain configuration (a series of nodes) by the `all_node` operand in the system common definition. Then, execute this command. After the command terminates normally, the new domain configuration is applied.

Also, the `namndchg` command outputs the specification of the `all_node` operand to the standard output.

This command can be executed only when OpenTP1 is active.

Options

■ `-l`

Outputs, to the standard output, the node name and the port number of the node where the command is executed, and the node names and the port numbers of all the nodes in the current domain configuration specified by using the `all_node` operand in the system common definition.

Output format

■ *(l) Output format when `namndchg -l` is specified*

```
all_node information (aa...aa)
port number           node name
bbbb                 cc...cc
```

aa...aa

`all_node` operand registration time (1 to 19 integers). Displayed in the format `yyyy/mm/dd hh:mm:ss`.

bbbb

Port number (1 to 5 integers)

cc...cc

Node name (1 to 255 alphanumeric characters)

Output messages

Message ID	Description	Output destination
KFCA00601-E	The shared memory became insufficient during execution of the name service.	Standard error output
KFCA00604-W	The node name specified by the <code>all_node</code> operand is duplicated.	Standard error output
KFCA00606-E	The node name specified by the <code>all_node</code> operand is not defined.	Standard error output
KFCA00607-E	The port number specified by the <code>all_node</code> operand is incorrect.	Standard error output
KFCA00611-E	The specification of the command argument is incorrect.	Standard error output
KFCA00615-I	Help message	Standard output, standard error output
KFCA00616-E	The number of <code>all_node</code> operands exceeds the specified number.	Standard error output
KFCA00620-E	An error occurred during execution of the <code>namndchg</code> command. Reason code=??? Maintenance information=???	Standard error output
KFCA00663-E	The command (<code>namndchg</code>) cannot continue because <code>Y</code> is specified for <code>name_domain_file_use</code> .	Standard error output

Notes

- The `namndchg` command can increase the number of nodes and make the configuration larger.

After startup of OpenTP1, the number of nodes can be increased up to the value specified for the `all_node_extend_number` operand in the system common definition (or 64 if the operand is omitted).

- To delete nodes from the `all_node` operand in the system common definition, stop all the SPPs running on those nodes and then execute the `namndchg` command. When nodes have been deleted while SPPs are active, an error may occur on transaction processing.
- The specifications in the `all_node_ex` operand in the system common definition that take effect are those that you specify when OpenTP1 starts. Accordingly, you cannot use the `namndchg` command to change the `all_node_ex` operand while OpenTP1 is running.

However, if you use the `all_node` operand in the system common definition to perform the following changes, the applicable nodes are deleted from the `all_node_ex` operand after the `namndchg` command is executed:

namndchg (Change domain configuration (by using the system common definition))

- If you move the nodes specified in the `all_node_ex` operand in the system common definition to the `all_node` operand in the system common definition
- If you add the nodes specified in the `all_node_ex` operand in the system common definition to the `all_node` operand in the system common definition

The following table shows the changes you can make to the `all_node` operand and the status of the `all_node` and `all_node_ex` operands after the `namndchg` command is executed.

Table 13-5: Changes you can make to the `all_node` operand and the status of the `all_node` and `all_node_ex` operands after the `namndchg` command is executed

Type of operation for the <code>all_node</code> operand	Example of change you can make	Status after the command is executed	
		<code>all_node</code> operand	<code>all_node_ex</code> operand
Addition	Adding a node that is not specified in the <code>all_node_ex</code> operand to the <code>all_node</code> operand	The applicable node is added.	No change
	Moving a node specified in the <code>all_node_ex</code> operand to the <code>all_node</code> operand	The applicable node is added.	The applicable node is deleted.
	Adding a node specified in the <code>all_node_ex</code> operand to the <code>all_node</code> operand	The applicable node is added.	The applicable node is deleted.
Deletion	Deleting a node specified in the <code>all_node</code> operand	The applicable node is deleted.	No change
	Moving a node specified in the <code>all_node</code> operand to the <code>all_node_ex</code> operand	The applicable node is deleted.	No change

namsvinf (Display OpenTP1 server information)

Format

```
namsvinf { -b | [-r] [-w maximum-response-wait-time] }
```

Function

The `namsvinf` command checks whether the OpenTP1 nodes specified in the `all_node` operand and the `all_node_ex` operand in the system common definition have started. Then the command displays the result and information about the OpenTP1 nodes (IP addresses and the port number of the name service).

Options

- `-b`

Indicates whether OpenTP1 nodes are registered in the RPC suppression list in shared memory, but does not check whether OpenTP1 nodes have started. Either of the following characters is output depending on the registration status:

OpenTP1 nodes are registered in the RPC suppression list: `D`

OpenTP1 nodes are not registered in the RPC suppression list: `A`

This option cannot be specified together with the `-r` or `-w` option. If you do, an error occurs.

If you do not specify this option, the command checks whether OpenTP1 nodes have started. When `Y` is specified for the `name_global_lookup` operand in the name service definition, a global search is performed to check whether OpenTP1 nodes have started.

- `-r`

Outputs the startup status of OpenTP1 nodes in the sequence they are checked.

This option cannot be specified together with the `-b` option. If you do, an error occurs.

If you do not specify this option, the startup statuses of all OpenTP1 nodes are output in a batch after all the startup statuses are checked.

- `-w maximum-response-wait-time ~<unsigned integer>((0-3600))<<8>>(unit: seconds)`

Specify the maximum time to wait for a response which starts after an `all_node` information acquisition request is sent to OpenTP1 in the OpenTP1 nodes specified in the `all_node` operand, and the maximum time to wait for a response to a startup check request. If the `-w` option is specified, you must always specify the maximum

response wait time. The startup status of an OpenTP1 node is s (transmission to the node is possible) if a response is not returned within the specified maximum response wait time.

This option cannot be specified together with the `-b` option. If you do, an error occurs.

If you do not specify all the options, the command checks whether OpenTP1 nodes have started, and shows the startup statuses of all nodes in a list after all the startup statuses are checked.

Output format

- When all the options are omitted

```

name_global_lookup a
NODE INFORMATION
EX      NODE      PORT      NOTIFY-TIME      STAT HOST-NAME      ...1.
bb ccc.ccc.ccc.ccc      dddd yyyy/mm/dd hh:mm:ss.mmmmm e ff..ff      ...2.
No      NODE      PORT      STAT HOST-NAME      ...1.
gggg hhh.hhh.hhh.hhh      iiii      j kk..kk      ...2.
    
```

1 and 2: One line is displayed for each.

2: Displayed for the number of nodes when there are multiple nodes to be checked.

Information is output in a batch when the startup check of all the nodes finishes. Therefore, it may take some time depending on the status of the target nodes.

If you do not specify both the `-b` option and the `-r` option, a message indicating the progress of checking is output to the standard output until the startup check finishes. The procedure from entering the command to the end of node information output is shown below.

1. Enter the command.
2. `namsvinf:asking node information... is output.`
3. `namsvinf:checking alive... is output.`
4. The output of node information begins.
5. The output of node information ends.
 - *a*: Specification in the `name_global_lookup` operand in the name service definition (Y or N) (one character)
 - *bb*: Identifier of the displayed node information (two characters)
 - EX: Node information specified in the `all_node_ex` operand of the node at which the command was executed
 - (Blank): Node information specified in the `all_node` operand

- *zutri2.tif*: Node information in which a service information priority specification keyword is specified in the `all_node` operand (*zutri2.tif*: blank)
- *ccc.ccc.ccc.ccc*: IP address of *ff..ff* (dotted decimal notation)

If the conversion to dotted decimal notation fails, ****.***.***.**** is output.

- *dddd*: Port number of *ff..ff* (five characters)
- *yyyy/mm/dd hh:mm:ss.mmmmmm*: Time when startup is reported (year/month/day/hour: minutes: seconds.microseconds)
- *e*: Startup status of *ff..ff* (one character)
 - A: Running normally
 - D: Not started
 - S: Transmission to the node is possible

When the `-b` option is specified, A or D is displayed as the startup status of the nodes specified in the `all_node` operand. The startup status of the nodes specified in the `all_node_ex` operand is not displayed.

- *ff..ff*: Node name specified in the `all_node` operand of the node where the command is executed (up to 255 characters)
- *gggg*: Number of the OpenTP1 node specified in the `all_node` operand of *ff..ff* (four characters).

You can assign the number you wish to each IP address (*ccc.ccc.ccc.ccc*) displayed on the left.

- *hhh.hhh.hhh.hhh*: IP address that is specified in *ccc.ccc.ccc.ccc* (dotted decimal notation). The applicable node is specified in the `all_node` operand.
- If the conversion to dotted decimal notation fails, ****.***.***.**** is output.
- *iiii*: Port number of the node indicated by *ccc.ccc.ccc.ccc* (five characters). The applicable node is specified in the `all_node` operand.
 - *j*: Startup status of the node indicated by *ccc.ccc.ccc.ccc* (one character). The applicable node is specified in the `all_node` operand.
 - A: Running normally
 - D: Not started
 - S: Transmission to the node is possible
 - *kk..kk*: Name of the node indicated by *ccc.ccc.ccc.ccc* (maximum of 255 characters). The applicable node is specified in the `all_node` operand.

If acquiring the node name fails, 64 asterisks (*) are output.

If the node information displayed for *bb* to *ff...ff* is all the same, *bb* to *ff...ff* on the second line and later are blank.

gggg to *kk...kk* are blank in the following cases:

- If no node name is specified in the `all_node` operand of the node where the command is executed
- If `N` is specified in the `name_global_lookup` operand
- If the startup status[#] of the node displayed by using the `namsvinf` command is `D` (Not started.) or `S` (Transmission to the node is possible.)

#

For details about the node startup statuses displayed by the `namsvinf` command, see *Table 13-6*.

The system displays information about the nodes in the `all_node` operand of the node where the command is executed, and then information about the nodes in the `all_node_ex` operand.

About the startup status of the nodes to be displayed by using the `namsvinf` command

The meaning of the startup status displayed by the `namsvinf` command differs depending on whether the `-b` option is specified. The following table shows the cases mainly responsible for each type of startup status according to whether the `-b` option is specified.

Table 13-6: Node startup statuses displayed by the namsvinf command

Specification of the -b option	Startup status to be displayed	Main cause of displayed startup status
When the -b option is omitted	A (Normally running.)	OpenTP1 determined that the displayed node was operating normally because the version of the node supports the <code>namsvinf</code> command, the message for checking startup was sent successfully, and the response message was received successfully.
	D (Not started.)	OpenTP1 determined that the displayed node was not operating normally because the message to the node for checking the startup status of the node was not successfully sent due to a communications error or an inactive destination when the message was sent.
	S (Transmission to the node is possible.)	OpenTP1 determined that messages could be sent to the node because the message for checking startup was sent successfully. No response message was received, however, the possible causes are as follows: <ul style="list-style-type: none"> The version of OpenTP1 running on the displayed node does not support the <code>namsvinf</code> command. The response message from the displayed node could not be received successfully due to a communications error or some other reason or due to expiration of the response reception wait time.
When the -b option is specified (communications for checking startup are not performed)	A (Normally running.)	The displayed node is not registered in the RPC suppression list.
	D (Not started.)	The displayed node is registered in the RPC suppression list.
	Blank	The displayed node is specified in the <code>all_node_ex</code> operand.

The displayed startup status does not change even when the `-r` and `-w` options are specified. However, if the value specified in the `-w` option is too small, the displayed startup status may change when the reception time expires.

- When the `-r` option is specified

The format of output records is the same as the case when all options are omitted. However, since node information is displayed each time a startup check completes, the display sequence of records is not fixed. `bb` to `ff..ff` are displayed for all the records.

namunavl (Forcibly invalidate the OpenTP1 start notification information)

Format

```
namunavl { -l | [-g] -n "node-name" [, "node-name" . . . ] }
```

Function

The `namunavl` command forcibly invalidates start notification information from an instance of OpenTP1 that starts on the node specified in an option of this command. When you execute this command, a new connection is established and the system can communicate normally without receiving a startup notification from the communication destination.

Options

■ -l

Specifying this option outputs the IP address that could not receive a startup notification due to a communication error that occurred when OpenTP1 was started. Since startup is not notified to the output IP address, the instance of OpenTP1 that starts on that IP address may not be able to communicate normally. In that case, you need to execute the `namunavl` command in the instance of OpenTP1 that cannot communicate.

To check if a startup notification is prevented by a communication error, check whether the KFCA00642-W message is output in the system log.

■ -g

Specifying this option suppresses the output of a recheck message (KFCA00644-R).

■ -n "node-name" [, "node-name" . . .] ~<identifier of 1 to 255 characters>

Specify the name of the node where OpenTP1 that invalidates the notified information (notification time) is to be started. You can neither omit this option, nor can you specify it more than once.

Notes

- You can execute the `namunavl` command while OpenTP1 is operating regardless of the value in the `name_notify` operand in the system common definition.
- The node names you can specify in the `namunavl` command are those that are specified in the `all_node` operand or the `all_node_ex` operand in the system common definition of the nodes to be entered. If you specify the name of the node with an instance of OpenTP1 that is currently communicating, communication

may fail.

- Do not set a node that is running normally or a node that can perform communications normally in the argument of the `-n` option of the `namunavl` command. If you have set node B that is running normally in the argument of the `-n` option of the `namunavl` command on node A, specify node A in the `-n` option in the `namunavl` command on node B and execute the command.
- If you have specified nodes that have the same IP address in the `all_node` operand or the `all_node_ex` operand in the system common definition, do not specify those nodes in the `namunavl` command.

prckill (Forcibly terminate OpenTP1 process)

Format

```
prckill process-ID [process-ID] ...
```

Function

The `prckill` command forcibly terminates the process with the signal number specified in the `prc_abort_signal` operand in the user service definition or user service default definition.

Command arguments

- *process-ID*

Specifies the process ID of the process to be terminated.

Notes

The process may be terminated abnormally with other than the signal number specified in the `prc_abort_signal` operand.

Output messages

Message ID	Description	Output destination
KFCA00710-E	Shared memory cannot be used.	Standard error output
KFCA00712-E	Unexpected error occurred.	
KFCA00750-E	Memory is inadequate.	
KFCA00751-E	<code>prckill</code> command format is invalid.	
KFCA00760-E	Process ID or status is invalid.	
KFCA00794-I	Help message	Standard output

prcls (Display status of servers)

Format

```
prcls [{-a|-g service-group-name|-l executable-form-file-name
|-s server-name|-p process-ID|-x}]
```

Function

The `prcls` command displays to the standard output the status of specified servers.

Options

- `-a`
Specifies that the status of all user servers and the system service is to be displayed.
- `-g service-group-name~<identifier, 1-31 characters>`
Specifies the name of a service group, the status of all of whose servers is to be displayed.
- `-l executable-form-file-name~<identifier, 1-14 characters>`
Specifies the execution form file name of a server whose status is to be displayed.
- `-s server-name~<identifier, 1-8 characters>`
Specifies the name of a server whose status is to be displayed.
- `-p process-ID~<unsigned integer>`
Specifies the ID of a process whose server's status is to be displayed.
When all these options are omitted, the status of the user server managed by the process server is displayed.
- `-x`
Displays the service execution status of each process in the user server.

Output format

- When the `-x` option is specified

PID	Server	Most recent service start time	Most recent service end time	Count
aa...aa	bb...bb	cc...cc	dd...dd	ee...ee

aa...aa

Process ID (up to 10 characters)

prcls (Display status of servers)

bb...bb

Server name

cc...cc

Time when the service function is called

The time is updated each time the service function is executed. The time is the number of seconds from 00:00:00 January 1, 1970 for the TZ environment variable. The time is displayed as follows if the process has never executed the service function.

Thu Jan 1 09:00:00 1970 (when the TZ environment variable is JST-9)

dd...dd

Time when the service function is returned

The time is updated each time the service function is executed. The time is the number of seconds from 00:00:00 January 1, 1970 for the TZ environment variable. The time is displayed as follows if the process has never executed the service function.

Thu Jan 1 09:00:00 1970 (when the TZ environment variable is JST-9)

ee...ee

Total number of times the service function is executed after the process has started

The count resets to 0 when it exceeds 4294967295.

- When an option other than the -x option is specified

Status	PID	UID	GID	Server	Object	Service group
<i>a</i>	<i>bb...bb</i>	<i>cc...cc</i>	<i>dd...dd</i>	<i>ee...ee</i>	<i>ff...ff</i>	<i>gg...gg</i>
:	:	:	:	:	:	:

a

Status of the server:

D: Engaged in server startup or termination processing

L: Engaged in server execution

*: Debugger process

bb...bb

Process ID (up to 10 characters)

cc...cc

User ID (up to 10 characters)

* is displayed for any process other than the child process generated from the process service.

dd...dd

Group ID (up to 10 characters)

* is displayed for any process other than the child process generated from the process service.

ee...ee

Server name (up to 8 characters)

ff..ff

Execution form file name (up to 14 characters)

utodbgrp: Execution format file of debugger working with appropriate server

gg...gg

Service group name (up to 31 characters)

Notes

To specify the `-x` option, TP1/Extension 1 must be installed. The operation is not guaranteed if TP1/Extension 1 is not installed.

Output messages

Message ID	Description	Output destination
KFCA00710-E	Shared memory cannot be used.	Standard error output
KFCA00712-E	Unexpected error occurred.	Standard error output
KFCA00750-E	Memory is inadequate.	Standard error output
KFCA00751-E	<code>prcls</code> command format is invalid.	Standard error output
KFCA00752-E	Execution form file name is invalid.	Standard error output
KFCA00753-E	Server name is invalid.	Standard error output
KFCA00754-E	Process ID is invalid.	Standard error output
KFCA00759-E	Service group name is invalid.	Standard error output
KFCA00790-I	Output header	Standard output
KFCA00791-I	Help message	Standard output

prcpath (Change search path names for user server and for command activated from user server)

Format

```
prcpath changed-path-name [:changed-path-name] . . .
```

Function

The `prcpath` command changes to a specified path name the search path used when a user server process is started and the search path used for a command to be activated from the user server.

The user server before the search path change and the user server started by the search path change both exist after the `prcpath` command executes. If this is not desirable, the search path change must be executed after terminating one of the user servers.

Command arguments

- *changed-path-name*~<path name> <<\$DCDIR/aplib:\$DCDIR/bin>>

Specifies the changed search path name.

More than one search path name can be specified, with a colon (:) placed between names. If specified, additional search path names must include the existing paths. The allowable number of search path names that can be specified is 255 characters including colon (:).

Notes

The following explains the procedure for creating a new user server under a separate directory from the one containing the currently active user server:

1. Terminate the currently active user server with the `dcsvstop` command.
2. Define a user server under a separate directory.
3. Change the path to the new directory with the `prcpath` command.
4. Start the new user server with the `dcsvstart` command.

Output messages

Message ID	Description	Output destination
KFCA00710-E	Shared memory cannot be used.	Standard error output
KFCA00712-E	Unexpected error occurred.	Standard error output
KFCA00750-E	Memory is inadequate.	Standard error output

prcpath (Change search path names for user server and for command activated from user server)

Message ID	Description	Output destination
KFCA00755-E	Communication cannot be established between processes.	Standard error output
KFCA00756-E	An argument is invalid.	Standard error output
KFCA00757-E	File was not found.	Standard error output
KFCA00758-E	Specification information is not a directory.	Standard error output
KFCA00792-I	Help message	Standard output

prcpathls (Display search path names for user server and for command activated from user server)

Format

```
prcpathls
```

Function

The `prcpathls` command outputs to the standard output the search path used when a user server process is started and when a command is activated from the user server.

Output format

```
aa...aa
```

aa...aa

Search path name for user server

Output messages

Message ID	Description	Output destination
KFCA00710-E	Shared memory cannot be used.	Standard error output
KFCA00712-E	Unexpected error occurred.	Standard error output
KFCA00750-E	Memory is inadequate.	Standard error output
KFCA00755-E	Communication cannot be established between processes.	Standard error output

prctctrl (Stop or restart the prctee process)

Format

```
prctctrl {-e | -s} [-i checking-interval] [-c checking-count]
```

Function

The `prctctrl` command stops or restarts the `prctee` process. The `prctee` process is a resident process that redirects OpenTP1 to standard output or standard error output. The `prctee` process runs with the options defined in `$DCDIR/bin/prcout`.

When the `prctee` process is stopped or restarted, whether it has stopped or has restarted is checked the number of times specified in the `-c` option at the checking interval specified in the `-i` option. If no stop or restart is confirmed following the specified checking operations, the message KFCA00773-E (Cause: TIME_OUT) is output and checking is terminated.

For details about the `prctee` process, see *prctee* in this chapter.

Options

- `-e`
Stops the `prctee` process.
- `-s`
Restarts the `prctee` process.
- `-i: checking-interval ~<unsigned integer>((1-60))<<1>>` (unit: seconds)
Specifies the interval for checking whether the `prctee` process has stopped or restarted.
- `-c: checking-count ~<unsigned integer>((1-60))<<10>>`
Specifies the number of times that checking whether the `prctee` process has stopped or restarted is performed.

Notes

- Only a superuser can execute the `prctctrl` command. To use this command, the `DCDIR` and `DCCONFPATH` environment variables must be specified.
- You can execute this command during an online session to stop the `prctee` process, but the messages that are output while the `prctee` process is stopped will be erased. Therefore, you should execute this command at a time when messages are not being output frequently.

prctctrl (Stop or restart the prctee process)

- If one of the operations listed below is executed after the `prctctrl` command is used to stop the `prctee` process, the `prctee` process will restart:
 - Startup of OpenTP1
 - Termination of OpenTP1
 - `dcsetup` command
 - `dcreset` command
- If the `prctctrl` command is executed with the `-s` option specified, in an environment in which the `dcsetup` command with the `-d` option specified has been executed (either the `-n` option is specified or `n` is specified for the inquiry), the KFCA00773-E error message (Cause: `TIME_OUT`) will be output after the amount of time equal to *checking-interval* x *checking-count* has elapsed.

prctee (Redirect standard output and standard error output of OpenTP1)

Format

```
prctee [file-length output-file-name]
```

Function

The `prctee` command outputs the standard output and standard error output of OpenTP1 to a file and performs generation management of the file.

Command arguments

- *file-length*~<unsigned integer> ((10-65535)) <<1024>>

Specifies the length of a file in kilobytes to which the standard output and standard error output should be output.

Based on the specified file length, two-generation management is performed. The length of the output file may exceed the specified value depending on the contents of the output messages. When 0 is specified, two-generation management is not performed and the file monotonously increases.

- *output-file-name*~<path name>

Specifies the name of the file to which the standard output and standard error output should be output, expressed as the full path name. When generation management is to be performed, 1 or 2 is appended to the file name.

When the file name is omitted, `$DCDIR/spool/prclog1` and `$DCDIR/spool/prclog2` are created.

Notes

- Define and use this command in `$DCDIR/bin/prcout`.
- The process uses a pipe to receive the standard output and standard error output and therefore messages may be discarded if a heavy-load write is repeatedly performed.
- To correct the `prcout` file, directly access and edit the `prcout` file under `/BeTRAN/bin` before executing the `dcsetup` command. Even if the `prcout` file under OpenTP1 home directory `/bin` is edited after the `dcsetup` command, the changes will not become valid.
- When you want to omit the command arguments, omit both file length and output file name. You cannot omit only one of these arguments.

`prctee` (Redirect standard output and standard error output of OpenTP1)

- Executing this command starts the `prctee` process (a resident process), which redirects OpenTP1 to standard output or standard error output.

prfed (Edit and output trace information files)

Format

```
prfed [-d] [-m] [-v] [-T start-time [, end-time]] [-r run-ID]
      [-p process-ID [, process-ID] ...]
      [trace-data-file-name [trace-data-file-name] ...]
```

Function

The `prfed` command inputs `prf` trace information, edits it, and outputs the resulting trace information.

Options

■ `-d`

Outputs the detail information in dump format.

When this option is specified with the `-v` option also specified, the detail information is output as an ASCII character string and a hexadecimal number. The ASCII character string is enclosed in double quotation marks (") and characters that cannot be printed are displayed as periods (.). If a double quotation mark is included within detail information displayed as the ASCII character string, it is displayed in the output as two double quotation marks (" ").

■ `-m`

Outputs the information with the event IDs of TP1/Message Queue.

■ `-v`

Outputs the edited result in CSV format.

■ `-T` [*start-time*] [, *end-time*]

Outputs only the trace information obtained within a specified period.

Specify the start time between 00:00:00 on January 1, 1970, and the current time, the current day, and the current month of the current year.

You must specify either the start time or the end time, or both. If you omit the start time, the command outputs the trace information from the beginning until the specified end time. If you omit the end time, the command outputs the trace information from the specified start time until the end of the trace information.

The start and end times are expressed in the format *hhmmss* [*MMDD* [*YYYY*]] :

hh: hour (00 ≤ *hh* ≤ 23); cannot be omitted

mm: minutes (00 ≤ *mm* ≤ 59); cannot be omitted

ss: seconds (00 ≤ *ss* ≤ 59); cannot be omitted

MM: month (01 ≤ *MM* ≤ 12); can be omitted[#]

dd: day (01 ≤ *dd* ≤ 31); can be omitted[#]

yyyy: year (4-digit year from 1970); can be omitted[#]

[#]: If the year is omitted from the start or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If you omit the -T option, all the information in the specified file is edited and output.

■ -r *run-ID*

Edits and outputs only the information with the specified run ID.

■ -p *process-ID*

Edits and outputs only the information with the specified process IDs. More than one process ID can be specified.

Command arguments

■ *trace-data-file-name*~<path name>

Specifies the file in which the trace data obtained with the *prfget* command is stored. When you omit this trace data file name, data will be input from the standard input. If trace data files acquired from multiple nodes are specified simultaneously, you can sort the output results in the order of the acquisition times.

Output format

No option is specified:

```

PRF: aaa Node: bbbb Run-ID: ccccccccc Process: dddddddddd Trace: eeeee           } 1
Event: fffff Time: gggg/hh/ii jj:kk:ll mmm.nnn.ooo Server-name: AAAAAAAA     }
Rc: ppppppppppp Client: qqqq - rrrrrrrr Server: ssss Root: tttt - uuuuuuuuu     } 2
Svc-Grp: vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv }
Trn: xxx...xxx

```

For a description of each output item, see below.

The -d option is specified:

```

PRF: aaa Node: bbbb Run-ID: ccccccccc Process: dddddddddd Trace: eeeee } 1
Event: fffff Time: gggg/hh/ii jj:kk:ll mmm.nnn.ooo Server-name: AAAAAAAA }
Rc: pppppppppp Client: qqqq - rrrrrrrr Server: ssss Root: tttt - uuuuuuuuu } 2
Svc-Grp: vvvvvvvvvvvvvvvvvvvvvvvvvvvvv Svc: wwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
Trn: xxx...xxx

Offset +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f 0123456789abcdef
yyyy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yyyyyyyyyyyyyy
yyyy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yyyyyyyyyyyyyy
.
yyyy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yy yyyyyyyyyyyyyy
    
```

The output formats are as shown below when the LCK performance verification trace information file and the TRN event trace information file are specified as the trace data sources.

The output format when the LCK performance verification trace information file is specified follows:

```

PRF: aaa Node: bbbb Run-ID: ccccccccc Process: dddddddddd Trace: eeeee } 1
Event: fffff Time: gggg/hh/ee jj:kk:ll mmm.nnn.ooo Server-name: AAAAAAAA }
Rc: pppppppppp Client: qqqq - rrrrrrrr Server: ssss Root: tttt - uuuuuuuuu } 2
Svc-Grp: vvvvvvvvvvvvvvvvvvvvvvvvvvvvv Svc: wwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
Trn: xxx...xxx
    ID:lll Resource:JJJJJJJJJJJJJJJJ Mode:KK Ownerflag:LLLLLLLLLLLLLLLLLLLL
    Flags:MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
    Internal code1:NNNNNNNNNNNNNNNNNNNN Internal code2:OOOOOOOOOOOOOOOOOO
    Internal code3:PPPPPPPPPP
    
```

The output format when the TRN event trace information file is specified follows:

```

PRF: aaa Node: bbbb Run-ID: ccccccccc Process: dddddddddd Trace: eeeee } 1
Event: fffff Time: gggg/hh/ii jj:kk:ll mmm.nnn.ooo Server-name: AAAAAAAA }
Rc: pppppppppp Client: qqqq - rrrrrrrr Server: ssss Root: tttt - uuuuuuuuu } 2
Svc-Grp: vvvvvvvvvvvvvvvvvvvvvvvvvvvvv Svc: wwwwwwwwwwwwwwwwwwwwwwwwwwwwwww
Trn: xxx...xxx
    BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB(CCC)DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
    axid:EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
    EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
    Internal code1 : FFFFFF Internal code2 : GGGG
    Internal code3 : HHHHHHHHHH
    
```

- 1 and 2: Displayed as a single line each.

aaa

Record status (3 characters)

REC: Normal record

ERR: Error record

bbbb

Node ID (up to 4 characters)

cc...cc

Run ID (10 hexadecimal digits [including 0x at the beginning])

dd...dd

ID of process that obtained the trace information (up to 10 decimal digits)

eeee

Trace serial number of the process (up to 5 decimal digits)

fffff

Event ID (6 hexadecimal digits [including 0x at the beginning])

gggg

Time when the trace was obtained (year)

hh

Time when the trace was obtained (month)

ii

Time when the trace was obtained (day)

jj

Time when the trace was obtained (time)

kk

Time when the trace was obtained (minute)

ll

Time when the trace was obtained (second)

mmm

Time when the trace was obtained (millisecond)

nnn

Time when the trace was obtained (microsecond)

ooo

Time when the trace was obtained (nanosecond)

AA...AA

Name of the server where an event occurred (up to 8 characters)

pp...pp

Return code (up to 11 decimal digits [including the first symbol])

qqqq

Client OpenTP1 identifier (up to 4 characters)

rr...rr

Client communication number (10 hexadecimal digits [including 0x at the beginning])

ssss

Server OpenTP1 identifier (up to 4 characters)

tttt

Route OpenTP1 identifier (10 hexadecimal digits [including 0x at the beginning])

uu...uu

Route communication number (10 hexadecimal digits [including 0x at the beginning])

vv...vv

Service group name (up to 32 characters)

ww...ww

Service name (up to 32 characters)

*xx...xx*Global transaction ID (up to 128 characters)^{#1}

If no applicable event ID has been acquired, the asterisk (*) is displayed.

y

Dump information

BB...BB

Event type (up to 32 characters)

When you specify `xafunc` in the `trn_prf_event_trace_condition`

operand, an XA function name is output. When you specify `trnservice` in the `trn_prf_event_trace_condition` operand, one of the following event types is output.

Event	Timing	Event type
Transaction management service	Starting	trnd starting
	Completely started	trnd started
	Ending	trnd ending
	Completely ended	trnd ended
Transaction recovery service	Starting	trnrvd starting
	Completely started	trnrvd started
	Starting recovery	trnrvd recovery started
	Ending recovery	trnrvd recovery ended
	Ending	trnrvd ending
	Completely ended	trnrvd ended
Resource manager monitoring service	Starting	trnrmd starting
	Completely started	trnrmd started
	Ending	trnrmd ending
	Completely ended	trnrmd ended

CCC

Acquisition timing (up to three characters)

DD...DD

Resource manager name + resource manager extension (up to 33 characters)

EE...EE

Transaction identifier assigned to a resource manager by OpenTP1 (80 hexadecimal digits)

FF...FF

Internal code 1 (up to six characters)

GGGG

Internal code 2 (up to four characters)

HH...HH

Resource manager name (up to 10 characters)

III

Lock request source type (3 characters)

When the lock request is received from a UAP: `usr`

When the lock request was generated within OpenTP1: `dam`, `tam`, `mqa`, or `trn`

JJ...JJ

Name of resource to be locked (maximum of 16 characters)

KK

Lock control mode (2 characters)

LL...LL

Request type (maximum of 20 characters)

MM...MM

Option flag (maximum of 25 characters)

NN...NN

Internal information 1 (16 characters)

OO...OO

Internal information 2 (16 characters)

PP...PP

Internal information 3 (hexadecimal, 10 digits (including the leading 0x))

Note

The asterisk (*) is displayed for items not obtained in the trace.

The `-m` option is specified:

	Time when the trace is acquired (month)
<i>ii</i>	Time when the trace is acquired (day)
<i>jj</i>	Time when the trace is acquired (hour)
<i>kk</i>	Time when the trace is acquired (minutes)
<i>ll</i>	Time when the trace is acquired (seconds)
<i>mmm</i>	Time when the trace is acquired (milliseconds)
<i>nnn</i>	Time when the trace is acquired (microseconds)
<i>ooo</i>	Time when the trace is acquired (nanoseconds)
<i>AA...AA</i>	Name of the server where the event occurred (up to eight characters)
<i>pp...pp</i>	Return code (up to 11 decimal digits (including the beginning sign))
<i>qqqq</i>	Client OpenTP1 identifier (up to four characters)
<i>rr...rr</i>	Client communication number (10 hexadecimal digits (including the beginning 0x))
<i>ssss</i>	Server OpenTP1 identifier (up to four characters)
<i>tttt</i>	Root OpenTP1 identifier (up to four characters)
<i>uu...uu</i>	Root communication number (10 hexadecimal digits (including the beginning

prfed (Edit and output trace information files)

0x))
 vv...vv
 Service group name (up to 32 characters)
 ww...ww
 Service name (up to 32 characters)
 xx...xx
 Global transaction ID (up to 128 characters)#
 zz...zz
 Queue name (up to 48 characters)#
 BB...BB
 Message identifier (66 hexadecimal digits (including the beginning 0x))#
 CC...CC
 Message token (18 hexadecimal digits (including the beginning 0x))#
 y
 Dump information

Note

The asterisk (*) is displayed for items not obtained in the trace.

#: The asterisk (*) is displayed if applicable event IDs are not acquired.

When the -v option is specified:

PRF,Node,Run-ID,Process,Trace,Event,Year/Month/Day,Hour:Minute:Second,Nano-second, Server-name,Rc,Node(Client),Number(Client),Node(Server), Node(Root),Number(Root), Group, Service,Trn	}	1
aaa,bbb,ccccccccc,ddddddddd,eeee,fffff,gggg/hh/ii,jj:kk:ll,mmmmmmmmm, nnnnnnnn,ooooooooooo,pppp,qqqqqqqqq,rrrr,ssss,tttttttt, uu....uu,vv....vv,ww....ww	}	2

For explanations of the output items, see *When the -d, -m, and -v options are specified* below.

When the -d and -v options are specified:

```

PRF,Node,Run-ID,Process,Trace,Event,Year/Month/Day,Hour:Minute:Second,Nano-second,
Server-name,Rc,Node(Client),Number(Client),Node(Server), Node(Root),Number(Root),
Group, Service,Trn,Data(ASCII),Data(Hexadecimal)
aaa,bbbb,ccccccccc,ddddddddd,eeee,fffff,gggg/hh/ii,jj:kk:ll,mmmmmmmmm,
nnnnnnnn,ooooooooooo,pppp,qqqqqqqqq,rrrr,ssss,tttttttt,
uu....uu,vv....vv,ww....ww,"AA....AA",BBBBBBBBBΔBBBBBBBBBΔ..BBBBBBBBBΔBBBBBBBBB
    
```

For explanations of the output items, see *When the -d, -m, and -v options are specified* below.

When the -m and -v options are specified:

```

PRF,Node,Run-ID,Process,Trace,Event,Year/Month/Day,Hour:Minute:Second,Nano-second,
Server-name,Rc,Node(Client),Number(Client),Node(Server), Node(Root),Number(Root),
Group, Service,Trn,Que-Name,Msg-ID,Msg-Token
aaa,bbbb,ccccccccc,ddddddddd,eeee,fffff,gggg/hh/ii,jj:kk:ll,mmmmmmmmm,
nnnnnnnn,ooooooooooo,pppp,qqqqqqqqq,rrrr,ssss,tttttttt,
uu....uu,vv....vv,ww....ww,xx....xx,yy....yy,zz....zz
    
```

For explanations of the output items, see *When the -d, -m, and -v options are specified* below.

When the -d, -m, and -v options are specified:

```

PRF,Node,Run-ID,Process,Trace,Event,Year/Month/Day,Hour:Minute:Second,Nano-second,
Server-name,Rc,Node(Client),Number(Client),Node(Server), Node(Root),Number(Root),
Group, Service,Trn,Que-Name,Msg-ID,Msg-Token,Data(ASCII),
Data(Hexadecimal)
aaa,bbbb,ccccccccc,ddddddddd,eeee,fffff,gggg/hh/ii,jj:kk:ll,mmmmmmmmm,
nnnnnnnn,ooooooooooo,pppp,qqqqqqqqq,rrrr,ssss,tttttttt,
uu....uu,vv....vv,ww....ww,xx....xx,yy....yy,zz....zz,"AA....AA",
BBBBBBBBBΔBBBBBBBBBΔ..BBBBBBBBBΔBBBBBBBBB
    
```

- 1 and 2: Displayed as a single line each.
- 2: Displayed for each record for which a trace is acquired.
- *aaa*: Record status (3 characters)
 - *Rec*: Normal record
 - *Err*: Error record

- *bbbb*: Node ID (maximum of 4 characters)
 - *cc....cc*: Run ID (hexadecimal, 10 digits (including the leading 0x))
 - *dd....dd*: Process ID of the process from which trace information is acquired (hexadecimal, maximum of 10 digits)
 - *eeee*: Trace serial number in the applicable process (hexadecimal, maximum of 5 digits)
 - *fffff*: Event ID (hexadecimal, 10 digits (including the leading 0x))
 - *gggg*: Year during which trace is acquired
 - *hh*: Month during which trace is acquired
 - *ii*: Date on which trace is acquired
 - *jj*: Hour at which trace is acquired
 - *kk*: Minute at which trace is acquired
 - *ll*: Second at which trace is acquired
 - *mm....mm*: Nanosecond at which trace is acquired
 - *nn....nn*: Name of server where the event occurred (maximum of 8 characters)
 - *oo....oo*: Return code (hexadecimal, maximum of 11 digits (including the leading symbol))
 - *pppp*: Client OpenTP1 identifier (maximum of 4 characters)
 - *qq....qq*: Client serial number (hexadecimal, 10 digits (including the leading 0x))
 - *rrrr*: Server OpenTP1 identifier (maximum of 4 characters)
 - *ssss*: Route OpenTP1 identifier (maximum of 4 characters)
 - *tt....tt*: Route communication number (hexadecimal, 10 digits (including the leading 0x))
 - *uu....uu*: Service group name (maximum of 32 characters)
 - *vv....vv*: Service name (maximum of 32 characters)
 - *ww....ww*: Global transaction ID (maximum of 128 characters)
- If no applicable event ID is acquired, the asterisk (*) is displayed.
- *xx....xx*: Queue name (maximum of 48 characters)
 - *yy....yy*: Message identifier (hexadecimal, 66 digits (including the leading 0x))
 - *zz....zz*: Message token (hexadecimal, 18 digits (including the leading 0x))
 - *AA....AA*: ASCII character string of the detail information (maximum of 256

characters)

- *BB...BB*: Hexadecimal character string of the detail information (including *zutri2.tif* (single-byte space) which appears after each string of 8 characters; maximum of 575 characters)

The table below shows the configuration of the detail information when the TRN event trace information file is specified for the trace data.

Item	Position	Length (bytes)
Maintenance information 1	0	4
Maintenance information 2	4	4
Maintenance information 3	8	4
Maintenance information 4	12	4
Maintenance information 5	16	4
Resource manager name + resource manager extension	20	36
Transaction identifier	56	40
Event type [#]	96	2
Acquisition timing: 0x10: IN, 0x20 or 0x30: OUT	98	1
Maintenance information 6	99	9
Maintenance information 7	108	4
Maintenance information 8	112	16

#

The following table shows the event types. Event types not listed in the table are for OpenTP1 maintenance information.

Event type	Event name
0x1001	trnd starting
0x1002	trnd started
0x1003	trnrmd starting
0x1004	trnrmd started
0x1005	trnrvd starting
0x1006	trnrvd started

Event type	Event name
0x1007	trn preending
0x1008	trn preended
0x1009	trnd ending
0x100a	trnd ended
0x100b	trnrmd ending
0x100c	trnrmd ended
0x100d	trnrvd ending
0x100e	trnrvd ended
0x100f	trnrvd recover started
0x1010	trnrvd recover ended
0x2001	xa_open
0x2002	xa_close
0x2003	xa_start
0x2004	xa_end
0x2005	xa_rollback
0x2006	xa_prepare
0x2007	xa_commit
0x2008	xa_recover
0x2009	xa_forget
0x200a	xa_complete
0x200b	ax_reg
0x200c	ax_unreg
0x2021	xa_done
0x2022	xa_ready
0x2023	xa_start_2
0x2024	xa_wait
0x2025	xa_wait_recovery

Event type	Event name
0x2041	ax_add_branch
0x2042	ax_commit
0x2043	ax_done
0x2044	ax_end
0x2045	ax_forget_branch
0x2046	ax_get_branch_info
0x2047	ax_prepare
0x2048	ax_ready
0x2049	ax_recover
0x204a	ax_reg_2
0x204b	ax_rollback
0x204c	ax_set_branch_info
0x204d	ax_start
0x204e	ax_start_2
0x3000	dc_trn_begin
0x3001	dc_trn_chained_commit
0x3002	dc_trn_chained_rollback
0x3003	dc_trn_unchained_commit
0x3004	dc_trn_unchained_rollback
0x3005	dc_trn_get_xid
0x3006	dc_trn_info
0x4000	tx_begin
0x4001	tx_commit
0x4002	tx_open
0x4003	tx_close
0x4004	tx_rollback
0x4005	tx_info

Event type	Event name
0x4006	tx_set_commit_return
0x4007	tx_set_transaction_control
0x4008	tx_set_transaction_timeout
0x5000	trncmt
0x5001	trnrbk
0x5002	trnfgt
0x5003	trnstics

Note: Any item not acquired in a trace is displayed as an asterisk (*).

Output messages

Message ID	Description	Output destination
KFCA26750-E	Command cannot be executed.	Standard error output
KFCA26751-E	Command format is invalid.	Standard error output
KFCA26760-E	Failed to open the prf trace file.	Standard error output
KFCA26761-E	An error occurred during accessing the prf trace file.	Standard error output
KFCA26762-E	Invalid data is detected during accessing the prf trace file.	Standard error output
KFCA26770-I	Help message	Standard output

Notes

- This command can only be used if TP1/Extension 1 is installed. Operation is guaranteed only when TP1/Extension 1 is installed.
- In the example in *Output format* with no option specified, it appears that one record is output on 4 lines; however, the record is actually output on 2 lines. Therefore, the output display may differ depending on the terminal mode you are using. In this case, use a terminal mode with 80-digit columns such as vt100.
- The dump data in the example in *Output format* with the `-d` option specified is only displayed when the record includes data unique to the system or user.
- The traces listed below use the same trace numbers in a process. Consequently, if you use a command to edit these traces only, the trace numbers (*eeee*) of the applicable processes that are output in *Output format* may not be in sequential order.

- Performance verification trace
- XAR performance verification trace
- JNL performance verification trace
- LCK performance verification trace
- MCF performance verification trace
- TRN event trace
- NAM event trace
- Process service event trace
- FIL event trace

For details about trace information, see the manual *OpenTP1 Description*.

- Do not specify a trace data file larger than 2 GB in the argument of the `prfed` command. If you want to edit a trace data file exceeding 2 GB, execute the `prfed` command as follows and edit the trace file:

Entering trace data from the standard input

```
prfget [-a] > prf-trace-collection-file
cat prf-trace-collection-file | prfed [options] > trace-edition-result-file
```

Entering trace data in the prf trace file acquisition directory (`$DCDIR/spool/dcprfinf`)

This method allows you to directly enter data in a prf trace file without using the `prfget` command. Because data is directly entered in a prf trace file, execute this method after OpenTP1 has stopped. If you execute this method while OpenTP1 is running, an error may occur during entry of the prf trace data.

```
cd $DCDIR/spool/dcprfinf
prfed [options] prf-trace-file [prf-trace-file . . .]
```

prfget (Get trace information files)

Format

```
prfget [-a] [-f {_tr|_nm|_xr|_pr|_mc|_fl|_jl|_lk}]
```

Function

The `prfget` command gets the trace information stored in the prf trace file. The range to be obtained starts right after the last information was obtained, and ends with the information when the command is entered.

Options

- -a

Obtains all trace information stored in the prf trace file. Usually, with the newest ID, the command obtains only the trace information that has not so far been obtained; but, this option enables you to get the trace information with an old run ID.

- -f {_tr|_nm|_xr|_pr|_mc|_fl|_jl|_lk}

Obtains event trace information.

`_tr`: Obtains TRN event trace information.

`_nm`: Obtains NAM event trace information.

`_xr`: Obtains XAR event trace information.

`_pr`: Obtains event trace information for a process service.

`_mc`: Obtains MCF performance verification trace information.

`_fl`: Obtains FIL event trace information.

`_jl`: Obtains JNL performance verification trace information.

`_lk`: Obtains LCK performance verification trace information.

Do not specify this option unless you want to obtain event trace information. Also, do not specify for this option any argument other than `_tr`, `_nm`, `_xr`, `_pr`, `_mc`, `_fl`, `_jl`, or `_lk`. For details on how to collect event trace information, see the description of event traces in the manual *OpenTPI Description*.

Output messages

Message ID	Description	Output destination
KFCA26750-E	Command cannot be executed.	Standard error output

Message ID	Description	Output destination
KFCA26751-E	Command format is invalid.	Standard error output
KFCA26760-E	Failed to open the prf trace file.	Standard error output
KFCA26761-E	An error occurred during accessing the prf trace file.	Standard error output
KFCA26762-E	Invalid data is detected during accessing the prf trace file.	Standard error output
KFCA26771-I	Help message	Standard output

Notes

- This command can only be used if TP1/Extension 1 is installed. Operation is guaranteed only when TP1/Extension 1 is installed.
- If the information going back to the previous trace has been lost, the command starts obtaining the oldest available information that has not already been obtained, up to the information when the command is entered.
- When the `prfget` command succeeds in getting the trace information, it stores the last position by creating an inheritance file[#]. If it fails to create an inheritance file, it might obtain trace information that was already obtained.
- Two or more `prfget` commands cannot be run simultaneously.

[#]: The contents of the inheritance file are the same as the file header of the last-output prf file.

queinit (Allocate physical file for message queue)

Format

```
queinit [-r] [-s record-length] [-n records-count] physical-file-name
```

Function

The `queinit` command allocates a physical file to be used as a message queue by the message queue server.

The OpenTP1 file system must be initialized prior to entering the `queinit` command.

Options

■ `-r`

Specifies that a physical file is to be allocated even if a file with the specified name already exists. If this option is omitted and the specified file already exists, an error occurs.

■ `-s record-length`~((sector length specified in file system initialization, up to 32768 (32K) bytes, specified as a multiple of the sector length)) <<2048>> (unit:bytes)

Specifies a record length.

■ `-n records-count`~((1-262144)) <<4096>>

Specifies the number of record areas to be allocated, where each record area is of the specified record length.

Command arguments

■ `physical-file-name`~<path name>

Specifies a name for the physical file to be allocated, expressed as the full path name.

quels (Display status of queue groups)

Format

```
quels [-f] [-q message-queue-service-name] [queue-group-ID]
```

Function

The `quels` command outputs to the standard output the status of the physical files and queue files for specified queue groups. The message queue service must have already been started when the `quels` command is entered.

Options

- `-f`
Specifies that queue file status is to be displayed for each queue group. When this option is omitted, only the physical file status is displayed.
- `-q message-queue-service-name~<identifier> <<que>>`
Specifies the name of the message queue service that manages the queue group IDs displaying status.

Command arguments

- `queue-group-ID~<identifier>`
Specifies the ID of a queue group whose file status is to be displayed. When this command argument is omitted, the status of all queue groups is displayed.

Output format

Queue group ID	Record length	Number of records	Number of records in use	Number of unused records	Physical file path name
<i>aaaaaaaa</i>	<i>bbbb</i>	<i>cccc</i>	<i>dddd</i>	<i>eeee</i>	<i>ff..ff</i>
Utilization factor (%)	Number of retained messages	Number of messages waiting for retrieval	Number of messages being processed	Number of messages on READY status	Queue file name
<i>gg</i>	<i>hhhh</i>	<i>iii</i>	<i>jjj</i>	<i>kk</i>	<i>lllll ...l.</i>
:	:	:	:	:	: ...l.

1. When the `-f` option is specified, this information is displayed as many times as there are queue files.

quels (Display status of queue groups)

aaaaaaaa

Queue group ID

bbbb

Record length of the physical file

cccc

Number of records in the physical file

dddd

Number of records in use

eeee

Number of unused records

ff..ff

Physical file path name

gg

Ratio of the number of records in the queue file to the number of records in the physical file (%) (rounded down)

hhhh

Number of messages retained in the queue file

iii

Number of messages in the queue file waiting for retrieval

jjj

Number of messages in the queue file being processed

kk

Number of messages in the queue file on READY status

lllllll

Queue file name

Output messages

Message ID	Description	Output destination
KFCA01323-E	Specification of a command argument is invalid.	Standard error output
KFCA01324-E	Format of queue group ID is invalid.	Standard error output

Message ID	Description	Output destination
KFCA01325-E	Specified queue group ID was not found.	Standard error output
KFCA01326-E	Memory is inadequate.	Standard error output
KFCA01329-I	Help message	Standard output, standard error output
KFCA01339-E	Specification of the message queue service name is invalid.	Standard error output
KFCA01360-E	Error occurred in the program.	Standard error output, message log file
KFCA01361-E	No execution environment has been set up.	Standard error output

querm (Delete physical file for message queue)

Format

<code>querm <i>physical-file-name</i></code>
--

Function

The `querm` command deletes a physical file allocated to the message queue by the `queinit` command.

A physical file being used in the online mode cannot be deleted. If an attempt is made to do so, an error message is output.

Command arguments

- `physical-file-name~<path name>`
Specifies the name of the physical file to be deleted, expressed as the full path name.

rapdfgen (Automatically generate definitions for using the remote API facility)

Format

```
rapdfgen RAP-processing-listener-service-definition-file-name | -m
RAP-processing-client-manager-service-definition-file-name
```

Function

The `rapdfgen` command generates the definition necessary for executing the remote API facility.

The `rapdfgen` command executed for the definition of the user-created RAP-processing listener service outputs the user service definitions for the RAP-processing listener and RAP-processing server to a file under the `$DCCONFPATH` directory. The `rapdfgen` command executed for the definition of the user-created RAP-processing client manager service outputs the user service definitions for the RAP-processing client manager to a file under the `$DCCONFPATH` directory. For details about the definitions for the RAP-processing listener service and RAP-processing client manager service, see the manual *OpenTPI System Definition*.

The following shows the output file names:

User service definition for the RAP-processing listener:

RAP-processing listener service definition name

User service definition for the RAP-processing server:

RAP-processing listener service definition name with suffix S

User service definition for the RAP-processing client manager:

RAP-processing client manager service definition name

Options

- `-m RAP-processing-client-manager-service-definition-file-name~<path name>`

Specifies the name of the file that stores the definition for the RAP-processing client manager service.

A file name for the RAP-processing client manager service definition must be an identifier of 1 to 8 characters. The system assumes the current directory to store the definition file that is not specified with a full path name (i.e., not beginning with "/").

Command argument

- *RAP-processing-listener-service-definition-file-name*~<path name>

Specifies the name of the file that stores the definition for the RAP-processing listener service.

A file name for the RAP-processing listener service definition must be an identifier of 1 to 7 characters. The system assumes the current directory to store the definition file that is not specified with a full path name (i.e., not beginning with "/").

Output messages

Message ID	Description	Output destination
KFCA-26911-E	An error occurred in the <code>rapdfgen</code> command.	Standard error output
KFCA-26912-E	The definition has an invalid value specified.	Standard error output
KFCA-26919-I	How to use the <code>rapdfgen</code> command.	Standard output

Notes

- Do not put the definition files for the RAP-processing listener service and RAP-processing client manager service under `$DCCONFPATH`. These definition files are inputs to the `rapdfgen` command, and they overwrite the user service definitions for the RAP-processing listener and RAP-processing client manager created by the command if they are put under `$DCCONFPATH`. The operation cannot be assured if these definitions files are put under `$DCCONFPATH`.
- The system assumes that the user service definitions for RAP-processing listener, RAP-processing server, and RAP-processing client manager are under `$DCCONFPATH`. Therefore, do not move these user service definitions from under `$DCCONFPATH` to under `$DCUAPCONFPATH`. The operation cannot be assured if these user service definitions are moved to under `$DCUAPCONFPATH`.

rapls (Display the status of a RAP-processing listener or a RAP-processing server)

Format

```
rapls RAP-processing-listener-name [ [RAP-processing-listener-name] . . . ]
```

Function

The `rapls` command outputs information about the specified RAP-processing listener or RAP-processing server to standard output.

Command arguments

- *RAP-processing-listener-name*

Specify the name of the RAP-processing-listener for which you want to display the status.

Output format

Listener	Status	Ports	Number of servers	Number of clients	Number of active recovery servers standing by
<i>aa...aa</i>	<i>bbbb</i>	<i>cccc</i>	<i>ddd(eee)</i>	<i>ff..ff</i>	<i>ggg(hhh)</i>
	pid	Status	IP address	Server allocation time	Last server request accepting time
	<i>iiii</i>	<i>jkl---</i>	<i>mmm.mmm.mmm.mmm</i>	<i>yyyy/mm/dd HH:MM:SS</i>	<i>yyyy/mm/dd HH:MM:SS</i>
:	:	:	:	:	:

aa...aa

RAP-processing listener name (up to 7 characters)

bbbb

Status of the RAP-processing listener

Init: A service is being started.

Serv: A service is being executed.

Term: A service is being terminated

cccc

Port number

ddd

Number of started servers

rapls (Display the status of a RAP-processing listener or a RAP-processing server)

eee

Number of active servers that are currently executing services

ff...ff

Number of clients being connected

ggg

Number of active recovery servers standing by

hhh

Number of active standby servers that are currently executing services

iiii

Server process ID

j

Execution status of the server

I: The server is being started or restarted

F: Not scheduled

S: A service is being executed

T: The server is being terminated

k

Server transaction mode

T: Transaction mode

-: Non-transaction mode

l

RPC execution mode of the server

-: Normal RPC execution mode

C: Chained RPC execution mode

mmm . mmm . mmm . mmm

IP address of the client being served by the applicable server

yyyy/mm/dd/HH:MM:SS

Processing time

If the execution status of the server is other than S (a service is being executed),
----/--/-- --:--:-- is displayed as the server allocation time and the last

server request accepting time.

Output messages

Message ID	Description	Output destination
KFCA26911-E	An error occurred while the command was executed.	Standard error output
KFCA26922-I	How to use the <code>rapls</code> command	Standard output
KFCA27750-E	An error occurred while the command was executed.	Standard error output
KFCA27751-W	The specified RAP-processing listener is not started.	Standard error output

rapsetup (Set up execution environment for the remote API facility)

Format

```
rapsetup [-h] [-d] user-program-file-directory
```

Function

The `rapsetup` command sets up the environment for executing the remote API facility.

Options

- `-h`

Displays the help message that indicates how to use the command.

- `-d`

Deletes the execution environment for the remote API facility from the OpenTP1 execution environment.

Command argument

- *user-program-file-directory*~<path name>

Specify the directory for storing the user program file. The default is `$DCDIR/aplib`.

Output messages

Message ID	Description	Output destination
KFCA26911-E	An error occurred in the <code>rapsetup</code> command.	Standard error output
KFCA26918-I	How to use the <code>rapsetup</code> command.	Standard output

rpcdump (Output RPC trace information)

Format

```
rpcdump [{-r|-m}] [{-c|-f}] [-d text-length]
[-t begin-date-and-time [, end-date-and-time]]
[-s service-group-name [, service-name] ...]
[-b node-identifier [, node-identifier] ...]
[-v server-name [, server-name] ...]
[-p process-ID [, process-ID] ...]
[-x xid [, xid] ...]
[-n [begin-trace-number] [, end-trace-number]]
[RPC-trace-file-name]
```

Function

The `rpcdump` command edits trace information of RPC trace files and output it to the standard output.

Options

■ `-r`

Outputs the result of editing in the form of an RPC trace file.

■ `-m`

Outputs outline RPC trace information. When this option is specified, specification of the `-c`, `-f`, and `-d` options becomes invalid.

■ `-c`

Extracts and edits the relationship between the call source and the call destination, and output it in units of call/called trace information and in chronological order according to other options' output conditions. The trace information involved with a call/called relationship to the trace information extracted through other options is output even if not meeting other options' output conditions.

■ `-f`

Extracts and edits a series of RPC flow and outputs it in units of nest/nested trace information and in chronological order according to other options' output conditions. The trace information involved in the same flow as the trace information extracted through other options is output even if not meeting other options' output conditions.

When neither the `-c` option nor the `-f` option is specified, RPC trace information is displayed chronologically.

- `-d text-length~((0~4096))`

Outputs the contents of a text with the specified length. When this option is omitted, the contents of all texts are output.

- `-t [begin-date-and-time] [, end-date-and-time]`

Specifies the output range of trace information by the date and time of trace acquisition.

The times must be specified in the range from 00:00:00 on January 1, 1970 to the current date and time.

The start time and end time cannot both be omitted. When the start time is omitted, the output range is from the beginning of the RPC trace file to the specified end time. When the end time is omitted, the output range is from the specified start time to the end of the RPC trace file.

The begin and end times are expressed in the format *hhmmss* [*MMDD* [*YYYY*]] :

hh: Hour (00 ≤ hh ≤ 23); cannot be omitted

mm: Minute (00 ≤ mm ≤ 59); cannot be omitted

ss: Second (00 ≤ ss ≤ 59); cannot be omitted

MM: Month (01 ≤ MM ≤ 12); can be omitted[#]

DD: Day (01 ≤ DD ≤ 31); can be omitted[#]

YYYY: Year (4-digit year from 1970); can be omitted[#]

[#]: If the year is omitted from the begin or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, then omit the year, the month, and the day (all three).

If the `-t` option is omitted, all information in the specified file is edited and output.

- `-s service-group-name [, service-name] ~<1-31 character identifier>`

Specifies a service group name and a service name to edit and output trace information including the call destination trace information for a particular service. More than one service name can be specified for a service group, with a comma (,) placed between names. When no service name is specified, the trace information for all services in the specified service group is edited and output. A service name alone cannot be specified.

When the XATMI request/response type communication service is specified, the service group name must be a name specified in the service definition of SPP. No XATMI conversational communication service name cannot be specified.

■ **-b** *node-identifier*~<4-character identifier>

Outputs the trace information collected at an OpenTP1 node with the specified node identifier. It must be the same as the node identifier specified in the system common definition. More than one node identifier can be specified, with a comma (,) placed between identifiers.

■ **-v** *server-name*~<1-8 character identifier>

Outputs trace information collected by the specified server. More than one server name can be specified, with a comma (,) placed between names.

■ **-p** *process-ID*~((0-2147483647))

Outputs trace information collected by the specified process. More than one process IDs can be specified, with a comma (,) placed between IDs.

■ **-x** *xid*~<1-8 hexadecimal digits> ((0-ffffff))

Outputs trace information with the sequential number of the specified global transaction identifier. More than one xid can be specified, with a comma (,) placed between xids.

■ **-n** [*begin-trace-number*] [, *end-trace-number*] ((1-2147483647))

Outputs trace information ranging from the begin trace number to the end trace number. A trace number is a sequential number assigned to trace information in an RPC trace file.

When the begin trace number is omitted, the output range is from the beginning of the RPC trace file to the end trace number.

When the end trace number is omitted, the output range is from the specified begin trace number to the end of the RPC trace file.

Either the begin or end trace number must be specified.

Command arguments

■ ***RPC-trace-file-name***~<path name>

Specifies the name of the RPC trace file in which trace information is to be edited. One of the following three names can be used:

- The name of the file specified in `rpc_trace_name` of the system common definition or user service definition
- The name of the file that stores the result of executing the `rpcmrgr` command
- The name of the file that stores the result of executing the `rpcdump` command with its `-r` option specified.

When this command argument is omitted, information is dumped from `$DCDIR/spool/rpctr`.

The name of the RPC trace file (default: `rpcctr`) included in the path can be a maximum of 13 characters.

Output format

- (1)-m option specified

```

rpcdump ***** rpc trace summary data *****
File name: ff..ff
Issue date and time:
          oo...oo - nn...nn
-----
NODE  SERVER  GROUP  SERVICE
aaaa  bb...bb  cc...cc  dd...dd
pid  send      rcv  send-rply  rcv-rply  send-error  rvc-error
eee  ff..ff    gg..gg  hh...hh  ii...ii  jj..jj    kk...kk
:    :        :        :        :        :        :
...l.
...l.
...l.
...l.
...l.
...l.

```

1. Edits and output outline information for each RPC text specified by the node identifier, server name, service group name, and service name, or XATMI conversational text.

ff..ff

Name of an RPC trace file

oo...oo

Date and time of the oldest trace information that was edited

nn...nn

Date and time of the newest trace information that was edited

aaaa

Identifier of the node that collected RPC trace information (4 characters)

bb...bb

Name of the server that collected RPC trace information (1-8 characters)

cc...cc

Name of the service group at the RPC text receive destination (1-31 characters)

For an XATMI conversational text, "*****" is displayed.

dd...dd

Name of the service at the RPC text receive destination (1-31 characters)

For an XATMI conversational text, "*****" is displayed.

eee

Process ID (decimal, left-justified)

ff..ff

Number of RPC request send texts or XATMI reply texts (decimal, left-justified)

gg...gg

Number of RPC request receive texts or XATMI reply texts (decimal, left-justified)

hh...hh

Number of RPC reply send texts or XATMI response texts (decimal, left-justified)

ii...ii

Number of RPC reply receive texts or XATMI response texts (decimal, left-justified)

jj..jj

Number of RPC error reply send texts or XATMI error texts (decimal, left-justified)

kk...kk

Number of RPC error reply receive texts or XATMI error texts (decimal, left-justified)

■ (2) Neither *-r* nor *-m* option specified

```
[aa...aa]          no<bbb>  nest-level<ccc>
server : dd...dd      nid  : eeee
node   : ff.ff.ff.ff  pid  : ggggg
port   : hhhhh
date   : ii...ii
<Service information>
  group : jj..jj           ...2.
  service : kk...kk       ...2.
  status : ll..ll         xid   : mmm           ...2.
  mode   : nn...nn       priority : oo           ...1., 2.
<xx...xx>
  node   : ppp.ppp.ppp.ppp  family : q           ...3.
  port   : rrrr           tid   : s           ...3.
<XATMI information>
  service : AA...AA       ...4.
  type    : BB...BB       subtype : CC...CC   ...4.
  tpflags : DD...DD       ...4.
  tperrcode : EE...EE     ...4.
  revent  : FF...FF       ...4.
  call descriptor : GG...GG ...4.
<Data>
ttttt uu...uu           vv...vv           ...5., 6.
```

1. Displayed only when an RPC request text is sent or received.
2. May not be displayed during XATMI conversational communication
3. The send source information is displayed for receive-side trace information. The receive destination information is displayed for send-side trace information.
4. Displayed only during communication using XATMI functions. Some items may not be displayed depending on the text type or send status.
5. Displayed repeatedly until all texts are processed.
6. Is not displayed if the length of the text specified in the `-d` option is 0.

aa...aa

RPC trace type

Send

RPC request text has been sent.

Receive

RPC request text has been received.

Send-Reply

RPC reply text has been sent.

Recv-Reply

RPC reply text has been received.

Send-Error

RPC error reply text has been sent.

Recv-Error

RPC error reply text has been received.

XATMI-R/R-Send

XATMI request/response-type request text has been sent.

XATMI-R/R-Recv

XATMI request/response-type request text has been received.

XATMI-R/R-Send-Reply

XATMI request/response reply text has been sent.

XATMI R/R-Recv-Reply

XATMI request/response reply text has been received.

XATMI-R/R-Send-Error

XATMI request/response error text has been sent.

XATMI-R/R-Recv-Error

XATMI request/response error text has been received.

XATMI-CNV-Send

XATMI conversational request text has been sent.

XATMI-CNV-Recv

XATMI conversational request text has been received.

XATMI-CNV-Send-Error

XATMI conversational error text has been sent.

XATMI-CNV-Recv-Error

XATMI conversational error text has been received.

bbb

Trace serial number in an RPC trace file (decimal)

ccc

Nest display (decimal)

Route of RPC nest calls; the route *ccc* is 1, which is incremented by 1 for each additional nest. 0 is displayed when an RPC is received from a CUP, different system, or an OpenTP1 system 03-00 or an earlier version. Note that, when the transaction is started from a CUP, the RPC request from the system process accepting a request from the CUP is the route, which is 1.

dd...dd

Server name (1-8 characters)

eeee

Node identifier of an OpenTP1 node (4 characters)

fff.fff

Node address of the RPC trace collection destination (decimal, left-justified)

ggggg

Process ID of the RPC trace collection source (decimal, left-justified)

hhhhh

BSD socket port number for the RPC trace collection source (decimal,

left-justified)

ii...ii

Time at which a text was sent to an RPC trace file

Format: *xxx.xxx x hh:mm:ss xxxx yy:zzz*

Example of output

Wed Jan 1 12:34:56 2003 789:012

The final *yy:zzz* indicate the following information:

yy: milliseconds

zzz: microseconds

This output example indicates that the text was output at 12:34 and 56.789012 seconds on Wednesday, January 1st, 2003.

jj...jj

Service group name (1-31 characters)

kk...kk

Service name (1-31 characters)

ll...ll

Chained RPC display

Chained

Chained RPC is in process.

Unchained

Not a chained RPC.

mmm

Sequence number of global transaction identifier (hexadecimal, left-justified)

nn...nn

Type of RPC

Reply

Synchronous reply type RPC

Noreply

Non-response type RPC

Nowait

Asynchronous reply type RPC

Chained

Chained RPC

oo

Priority value of a service request (decimal, left-justified)

xx...xx

Indication of the sender/receiver

Send source

The send source sends an RPC text.

(The RPC trace output source receives an RPC text.)

Receive destination

The receive destination receives an RPC text. (The RPC trace output source sends an RPC text.)

PPP-PPP

Nodes address of the send source and receive destination (decimal, left-justified)

q

Network type

rrrrr

Port number of the BSD socket for the send source and receive destination (decimal, left-justified)

sss

Thread ID of the send source (decimal, left-justified)

tttt

Location of a text (hexadecimal)

uu...uu

Text body consisting of up to 16 bytes per line (hexadecimal)

ss...ss

Indication of the text body in terms of ASCII code (1-16 characters). Any character that cannot be printed as a result of conversion is displayed by a period (.).

AA...AA

XATMI service name (1-31 characters)

BB...BB

Buffer type name. Either `X_OCTET`, `X_COMMON` or `X_C_TYPE` is displayed. For the meaning of each name, see the XATMI manual.

CC...CC

Buffer sub-type name (1-16 characters). No sub-type name is displayed when the buffer type is `X_OCTET`.

DD...DD

Value of `tpflags` specified in the XATMI function (hexadecimal). Value 0 is not displayed.

EE...EE

Error code (`tperror`) of an error that occurred with the XATMI function (hexadecimal)

FF...FF

Event that occurred with the `tprecv` function. Either `TREV_SENDOONLY`, `TREV_SVCERR`, `TREV_SVCFAIL`, `TREV_SVCSUCC` or `TREV_DISCONIMM` is displayed. For the meaning of each event, see the XATMI manual.

GG...GG

call description during XATMI conversational communication (hexadecimal)

Output messages

Message ID	Description	Output destination
KFCA00308-E	RPC trace file contains no data.	Standard error output
KFCA00310-E	Specified RPC trace file was not found.	Standard error output
KFCA00312-E	Access error occurred in RPC trace file.	Standard error output
KFCA00314-I	Help message	Standard output
KFCA00352-E	RPC trace file data is incorrect.	Standard error output

Notes

- When RPC trace information is output by the `rpcdump` command, the receive information may be output before the send information. This is because the send information is written into the RPC trace file after text send is completed.

- All conditions specified by the `-t`, `-s`, `-b`, `-v`, `-p`, `-n` and `-x` options are AND conditions.
- The `rpcdump` command must be entered when no RPC trace is written. For example, by saving the RPC trace file, the write to the RPC trace file can be suppressed. If the `rpcdump` command is entered while collecting the RPC trace, an RPC trace file access error may occur depending on the timing. This error occurs if the RPC trace file is switched while RPC trace is being edited.
- If the amount of the transmitted text is greater than the value specified for the `rpc_trace_size` operand in the system common definition, the user service default definition, or the user service definition, the `rpcdump` command may not output anything or the output information may be incomplete.

rpcmgr (Merge RPC traces)

Format

```
rpcmgr RPC-trace-file-name [RPC-trace-file-name] . . .
```

Function

The `rpcmgr` command chronologically arranges trace information from the specified two or more RPC trace files, and output it to the standard output in the form of an RPC trace file. Any duplicated information is deleted from the specified RPC trace files.

Command argument

- *RPC-trace-file-name*~<path name>

Specifies the name of the RPC trace file that is to be merged. One of the following three names can be used:

- The name of the file specified in `rpc_trace_name` of the system common definition or user service definition.
- The name of the file that stores the result of executing the `rpcmgr` command
- The name of the file that stores the result of executing the `rpcdump` command with its `-r` option specified

The name of the RPC trace file (default: `rpctr`) included in the path name can be a maximum of 13 characters.

Output messages

Message ID	Description	Output destination
KFCA00308-E	No data exists in RPC trace file.	Standard error output
KFCA00310-E	Specified RPC trace file does not exist.	Standard error output
KFCA00312-E	Access error to RPC trace file occurred.	Standard error output
KFCA00350-I	Help message	Standard output
KFCA00352-E	Data in RPC trace file is invalid.	Standard error output

Notes

- If there is any deviation in the time managed by each OpenTP1, trace information is not arranged in the order it was actually collected.
- In the following situations, the `rpcmgr` command may not output any

information, or the information it outputs may be incomplete:

A message is sent or received which is larger than the value specified for the `rpc_trace_size` operand in the system common definition, the user service default definition, or the user service definition, or a message is output that spans two or more of the specified RPC trace files.

rpcstat (Display execution status of temporary closing)

Format

```
rpcstat
```

Function

The `rpcstat` command displays the execution status of temporary closing of the process under OpenTP1 management. The displayed information does not include statistics, but includes the execution status of temporary closing at the time of command entry.

Output format

```
PID   SVNAME MAXFDS H-WATER L-WATER  USING  CLOSED  CLOSING  REFUSED  PORT
aa..aa bb..bb cc..cc dd..dd ee..ee  ff..ff  gg..gg  hh..hh  ii..ii  jj..jj
```

aa...aa

Process ID

bb...bb

Server name

cc...cc

Maximum number of sockets that can be used for inter-process communications

dd....dd

Percentage of sockets which have started temporary closing

ee...ee

Percentage of sockets which are not subject to temporary closing

ff...ff

Number of sockets being used in inter-process communication

gg...gg

Number of sockets temporarily closed

hh...hh

Number of sockets requesting for temporary closing

ii...ii

Number of sockets which refused temporary closing

jj..jj

Receiving port number

Notes

- No information of the shell-started SUP is output.
- Executing this command under OpenTP1 which does not support the temporary closing status check facility displays 0 for all the values.
- Even before OpenTP1 starts up, this command can output the information if there is the shared memory that OpenTP1 uses.

rtsedit (Edit and output RTS log files)

Format

```
rtsedit { [-m] | [-e item-ID [, item-ID] ... [-q]]
          [-t [start-time] [, end-time]]
          [-u editing-unit [-s server-name] [-v service-name]]
          [-i editing-interval]
          [-o output-destination-file-name [-l number-of-lines]] }
          RTS-log-file-name [ [ Δ RTS-log-file-name] ... ]
```

Function

The `rtsedit` command gets real-time statistics from an RTS log file, edits the statistics, and outputs the result to a file or the standard output in CSV format.

Options

■ `-m`

Outputs management information about the RTS log file specified in the command argument to the standard output.

When you specify the `-m` option, you cannot specify other options.

■ `-e item-ID~<unsigned integer>` ((1000-2147483647))

Specifies the ID of the item to be edited.

For details about item IDs, see *E.5 Real-time statistics*.

■ `-q`

Outputs hyphens (-) for the number of items, maximum value, minimum value, and average in a time slot that has no real-time statistics for the item ID specified in the `-e` option.

This option is valid only when the `-e` option is specified.

If you do not specify this option, no hyphens (-) are output for the time slots that have no real-time statistics.

■ `-t [start-time] [, end-time]`

Specifies the period for which data is to be edited. For *start-time*, specify the time at which editing of the target real-time statistics is to start. For *end-time*, specify the time at which editing of the target real-time statistics is to end.

For *start-time*, you can specify a time from 00:00:00 on January 1, 1970, to the current time of the current year, month, and day.

You must specify either a start time or an end time, or both. If you omit the start time, the command outputs real-time statistics from the start of the RTS log file until the specified end time. If you omit the end time, the command outputs real-time statistics from the specified start time until the end of the RTS log file.

Specify the start time and the end time in the *hhmmss* (*[MMDD[YYYY]]*) format.

hh

Hour ($00 \leq hh \leq 23$)

Cannot be omitted.

mm

Minutes ($00 \leq mm \leq 59$)

Cannot be omitted.

ss

Seconds ($00 \leq ss \leq 59$)

Cannot be omitted.

MM

Month ($01 \leq MM \leq 12$)

Cannot be omitted.[#]

DD

Day ($01 \leq DD \leq 31$)

Cannot be omitted.[#]

YYYY

Year (specify a year after 1970 using four digits)

Cannot be omitted.[#]

#

If the year is omitted from the start or end time, the specified month, day, and time of the current year are assumed. If the year, month, and day are omitted, the current year, month, and day, and the specified time are assumed. You cannot omit only the month and day, or the month or the day. Doing so causes an option error. If you want to omit the month or the day, omit the year, the month, and the day (all three).

If you omit the *-t* option, all the information in the specified file is edited and output.

- `-u editing-unit~<all>`

Specifies the unit of editing.

`sys`

Edits the acquired real-time system-wide statistics.

`svc`

Edits the acquired real-time statistics for each server or service.

`all`

Edits all the real-time statistics (system-wide, all servers, and all services).

- `-s server-name~<string of 1-8 characters>`

Specifies the name of the server, port number, logical terminal, or first acquisition target whose real-time statistics you wish to edit.

The specification of the `-s` option is valid when `svc` is specified in the `-u` option.

- `-v service-name~<string of 1-63 characters>`

Specifies the name of the service, port number, logical terminal, or second acquisition target whose real-time statistics you wish to edit.

The specification of the `-v` option is valid when `svc` is specified in the `-u` option.

The following table shows the combinations of the `-u`, `-s`, and `-v` options, and the statistics edited as a result.

Table 13-7: Combinations of rtsedit command options and the statistics to be edited

Specification of <code>-u</code> option	Specification of <code>-s</code> option	Specification of <code>-v</code> option	Real-time statistics to be edited
<code>sys</code>	--	--	System-wide statistics
<code>service</code>	Server name	Service name	Statistics for the specified server and service
		Not specified	Statistics for the specified server
	Port number [#]	IP address [#]	Statistics for the node for which the specified service information is referenced
	Logical terminal name [#]	Not specified [#]	Statistics for the specified logical terminal
	Not specified	Service name	Statistics for the specified service
		Service group name [#]	Statistics for the specified service group

Specification of -u option	Specification of -s option	Specification of -v option	Real-time statistics to be edited
		Not specified	All non-system-wide statistics
all	--	--	All statistics

Legend:

--: Cannot be specified.

#

Indicates a specification for editing the statistics of an acquisition target that is registered by specifying `obj` for the `-u` option in the `rtspout` definition command or the `rtstats` command of the real-time statistics service definition.

■ `-i editing-interval~<unsigned integer> ((10-31536000))`

Specifies in seconds the interval for editing statistics and outputting them to a CSV file.

If you do not specify the `-i` option, the system assumes the output interval specified when real-time statistics are acquired in an RTS log file (value specified in the `rts_trcput_interval` operand in the real-time statistics service definition) as the value of the `-i` option.

If you do not specify the `-i` option and specify multiple RTS log files in the command argument, the system assumes the value of the `-i` option to be the largest interval among those for outputting real-time statistics to an RTS log file.

The following table lists example valid intervals for outputting edited real-time statistics to a CSV file.

Table 13-8: Examples of valid intervals for outputting edited real-time statistics to a CSV file

Example	Interval for outputting real-time statistics to an RTS log file (seconds)			Value specified for -i option	Valid interval for outputting edited real-time statistics to CSV file (seconds)
	RTS log file 1	RTS log file 2	RTS log file 3		
1	600	300	200	1000	1000
2	600	300	200	100	100
3	100	200	300	Not specified	300

To check the interval for outputting real-time statistics to an RTS log file, execute the

`rtsedit` command with the `-m` option specified.

- `-o output-destination-file-name~<path name of 1-65 characters>`

Specifies the name of a file in CSV format to which the edited real-time statistics are to be output.

If you do not specify this option, the command outputs the edited real-time statistics to the standard output.

- `-l number-of-lines~<unsigned integer> ((1024-65535))`

Specifies the number of lines to be output to one file (file in CSV format specified in the `-o` option).

The range for generation numbers of CSV-format files is from 1 to 999999. When all the lines specified in the `-l` option have been output, the generation number for a CSV-format file is incremented by one, and output continues. When real-time statistics are output to a file with generation number 999999, the command ends the output processing.

The specification of the `-l` option is valid only when the `-o` option is specified.

If you do not specify the `-l` option, the command does not update the generation numbers of CSV-format files and outputs all of the data to one file.

Command argument

- `RTS-log-file-name~<path name of 1-65 characters>`

Specifies the name of a RTS log file.

Output format

- (1) *When the `-m` option is specified (standard output)*

```
*** RTS logfile information ***
Logfile Name: aa...aa
Node Id: bbbb
File Update Time: cccc/cc/cc cc:cc:cc
Statistics Data Edit Interval: dd...dd
Version: ee...ee
```

} 1

1. Displayed repeatedly for each RTS log file

aa...aa

Name of the RTS log file that is read

bbbb

Node identifier

cccc/cc/cc cc:cc:cc

Time when the RTS log file was updated

dd...dd

Output interval

The interval for output of RTS log files is shown in seconds.

ee...ee

The version number of the RTS log file that was read.

"1" indicates an RTS log file acquired by version 07-00 of TP1/Server Base, and "2" indicates an RTS log acquired by version 07-01 or later of TP1/Server Base.

■ (2) When the *-m* option is not specified (CSV format)

```

rtsedit vv-rr
Logfile Name : aa...aa
Time of File Creation : bbbb/bb/bb bb:bb:bb
Range of Output Data : from cccc/cc/cc cc:cc:cc to dddd/dd/dd dd:dd:dd
Edit Interval : ee...ee

Node ID,Start Time,End Time,Server,Service,                1
ffff,gggg/gg/gg gg:gg:gg,hhhh/hh/hh hh:hh:hh,ii...ii,jj...jj,  2

Type,Event,Event ID,Total,Maximum,Minimum,Average,Units    1
kk...kk,ll...ll,mm...mm,nn...nn,oo...oo,pp...pp,qq...qq,(rr...rr)  2
    
```

1 and 2: Displayed as a single line each.

2. Displayed for each server or service for which real-time statistics are acquired for the number of items in the statistics

aa...aa

Name of the RTS log file that is read

bbbb/bb/bb bb:bb:bb

Time when the output file was created

cccc/cc/cc cc:cc:cc

Start time specified in the *-t* option

Hyphens (-) are displayed if the *-t* option is not specified.

dddd/dd/dd dd:dd:dd

End time specified in the *-t* option

Hyphens (-) are displayed if the *-t* option is not specified.

ee...ee

Editing interval

Valid interval for outputting edited real-time statistics to a CSV file. This interval is set based on the interval for outputting real-time statistics to an RTS log file and the value specified in the `-i` option.

ffff

Node identifier

gggg/gg/gg gg: gg: gg

Time at which editing of the real-time statistics starts

hhhh/hh/hh hh:hh:hh

Time at which editing of the real-time statistics ends

ii...ii

Name of the server whose real-time statistics are edited

When the server name is `_SYSTEM` and the service name is `ΔΔΔΔ`, the target statistics are system-wide statistics. If `----` appears in this field, the real-time statistics were acquired without specifying a first acquisition target.

jj...jj

Name of the service whose real-time statistics are edited

When `****` is displayed, the target statistics are the real-time statistics acquired from each applicable unit of processing other than services. When `ΔΔΔΔ` is displayed, the target statistics are the real-time statistics acquired from each server. If `----` appears in this field, the real-time statistics were acquired without specifying a second acquisition target.

kk...kk

Type of real-time statistics

ll...ll

Item name

mm...mm

Item ID

nn...nn

Number of items

If there are no real-time statistics to be edited[#], hyphens (-) are displayed. If an overflow error occurs, `****` is displayed.

oo...oo

Maximum value

If there are no data values to be edited, hyphens (-) are displayed. If an overflow error occurs, **** is displayed.

pp...pp

Minimum value

If there are no data values to be edited, hyphens (-) are displayed. If an overflow error occurs, **** is displayed.

qq...qq

Average

If there are no data values to be edited, hyphens (-) are displayed. If an overflow error occurs, **** is displayed.

rr...rr

Unit of real-time statistics

vv-rr

Version number - revision number

#

If you specify the `-e` and `-q` options when, for the following reasons, there are no real-time statistics to be edited, the system displays hyphens (-) for the number of items, maximum value, minimum value, and average.

- The value specified in the `-i` option (editing interval) is smaller than the interval for outputting real-time statistics to an RTS log file.
- Although the value specified in the `-i` option (editing interval) is equal to the interval for outputting real-time statistics to an RTS log file, there is no data to be edited within the time slot of the editing interval due to a delay in acquiring real-time statistics.
- There is a time slot in which no real-time statistics have been acquired because, for example, the output of real-time statistics is temporarily stopped or some items to be acquired have been deleted.
- Multiple RTS log files are specified in the command argument and the acquisition times for those RTS log files are not sequential.

Output messages

Message ID	Description	Output destination
KFCA32714-E	Processing cannot continue due to insufficient process memory.	Standard error output
KFCA32719-E	The command was used incorrectly.	Standard error output
KFCA32720-E	The value specified in a command option is invalid.	Standard error output
KFCA32721-E	The value specified in a command argument is invalid.	Standard error output
KFCA32725-E	An error occurred during file operations.	Standard error output
KFCA32726-E	The specified file already exists.	Standard error output
KFCA32727-E	The specified file is not an RTS log file.	Standard error output
KFCA32728-E	The combination of version and revision of the RTS log file is incorrect.	Standard error output
KFCA32729-E	The command terminated because all the generations of output files were used.	Standard error output
KFCA32730-E	Invalid data was detected while an RTS log file was being accessed.	Standard error output
KFCA32731-E	The specified file does not exist.	Standard error output
KFCA32735-I	Help message	Standard output
KFCA32744-E	The user is not authorized to access the file.	Standard error output
KFCA32745-E	The specified file name is incorrect.	Standard error output
KFCA32759-E	Editing is not possible because the contents of the RTS log file already exist.	Standard error output

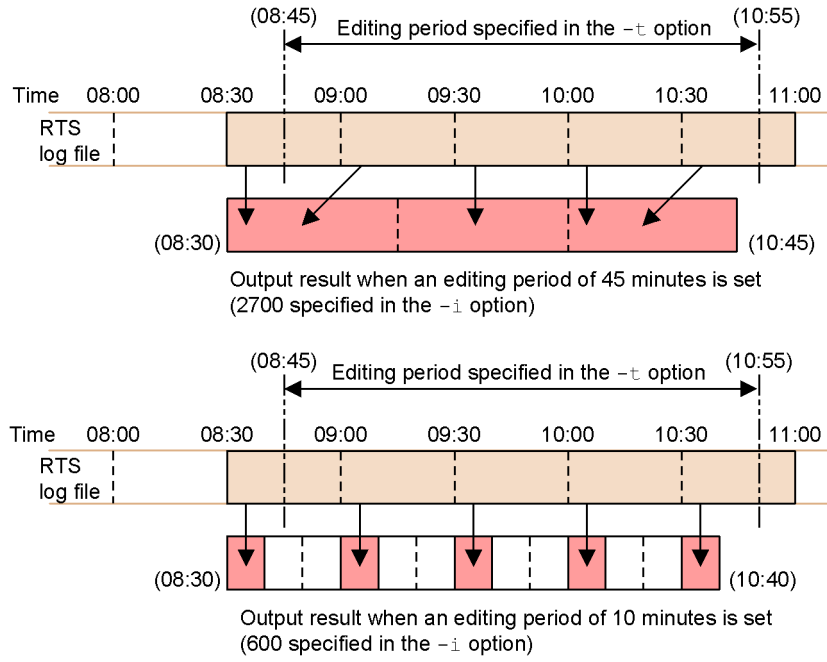
Notes

- If you specify an RTS log file acquired using version 07-00 of TP1/Server Base for editing, zeroes (0) are displayed instead of hyphens (-) as the maximum value, minimum value, and average value for the `write` (item ID: 1104) and `read` (item ID: 1107) items in the journal information.
- When editing RTS log files acquired using version 07-00 of TP1/Server Base together with RTS log files acquired using later versions, if you specify an editing interval that is longer than the output interval of the RTS log files, zeroes (0) are displayed as the minimum value of the `write` (item ID: 1104) and `read` (item ID: 1107) items in the journal information. Also, the average value displayed for these items may be less than the actual average.
- If you specify an RTS log file that is in use in a command argument, the command

may not be able to output correct information.

- If a KFCA32729-E message is output, either increase the value specified in the `-i` option or decrease the number of RTS log files specified in the command argument, and then re-execute the `rtsedit` command.
- If you specify multiple RTS log files that were acquired on different OpenTP1 systems, make sure that they do not have the same node ID. If you specify RTS log files that have the same node ID, statistics may be compiled on the assumption that the data refers to the same node, or command processing may terminate with the message KFCA32759-E.
- Figure 13-1 shows the relationship between RTS log files and the `-t` and `-i` options in the following situation:
 - Interval for outputting real-time statistics to an RTS log file: 30 minutes
 - Value specified in the `-t` option: 084500, 105500

Figure 13-1: Relationship between RTS log files and specified options



Legend:

- : Real-time statistics to be edited.
- : Time slot in which the real-time statistics are output.
- : Time slot in which hyphens (-) are output instead of real-time statistics when the -q option is specified, and nothing is output when the -q option is omitted.

rtsls (Output real-time statistics to standard output)

Format

```
rtsls [-c] [-n generation-number] [-m] [-l]
      [-e item-ID [, item-ID] ...]
      | -u output-unit [-s server-name] [-v service-name]
```

Function

The `rtsls` command outputs, to the standard output, management information and real-time statistics of the real-time statistics service. This data was collected in the shared memory used for the real-time statistics service.

For details about the real-time statistics that are output, see *E.5 Real-time statistics*.

Options

- `-c`

Outputs real-time statistics in CSV format to the standard output.

- `-n generation-number`~<unsigned integer> ((1-3))

Specifies the generation number of the shared memory for the RTS service that contains the real-time statistics to be output.

If you do not specify the `-n` option, the command outputs the real-time statistics to the shared memory with a generation number that indicates that statistics have already been acquired in the shared memory.

- `-m`

Outputs the management information for the real-time statistics service to the standard output.

- `-l`

Outputs a list of the names of services from which real-time statistics are acquired and the relevant item IDs.

- `-e item-ID`~<unsigned integer> ((1000-2147483647))

Specifies an item ID for the real-time statistics to be output.

For details about item IDs, see *E.5 Real-time statistics*.

- `-u output-unit`~<all>

Specifies the output unit.

`sys`

Outputs the acquired real-time system-wide statistics.

`svc`

Outputs the acquired real-time statistics for each server or service.

`all`

Outputs all the real-time statistics (system-wide, all servers, and all services).

- `-s server-name~<string of 1-8 characters>`

Specifies the name of the server, port number, logical terminal, or first acquisition target whose real-time statistics you wish to output.

The specification of the `-s` option is valid only when `svc` is specified in the `-u` option.

- `-v service-name~<string of 1-63 characters>`

Specifies the name of the service, port number, logical terminal, or second acquisition target whose real-time statistics you wish to output.

The specification of the `-v` option is valid only when `svc` is specified in the `-u` option.

The following table shows the combinations of the `-u`, `-s`, and `-v` options, and the output targets.

Table 13-9: Combinations of rtsls command options and the statistics to be output

Specification of <code>-u</code> option	Specification of <code>-s</code> option	Specification of <code>-v</code> option	Real-time statistics to be output
<code>sys</code>	-	-	System-wide statistics
<code>service</code>	Server name	Service name	Statistics for the specified server and service
		Not specified	Statistics for the specified server
	Port number [#]	IP address [#]	Statistics for the node for which the specified service information is referenced
	Logical terminal name [#]	Not specified [#]	Statistics for the specified logical terminal
	Not specified	Service name	Statistics for the specified service
		Service group name [#]	Statistics for the specified service group
	Not specified	All non-system-wide statistics	
<code>all</code>	--	--	All statistics

Legend:

--: Cannot be specified.

#

Indicates a specification for editing the statistics of an acquisition target that is registered by specifying `obj` for the `-u` option in the `rtspout` definition command or the `rtstats` command of the real-time statistics service definition.

If you specify neither the `-e` option or the `-u` option, the system assumes that `all` is specified for the `-u` option.

Output format

- (1) When the `-m` option is specified

The output format is as follows:

```
***** Real time statistical information Node id:aaaa *****
Shm id:bbbbbbb,Number of shm:cccccccc,RTS status:dddd
Max service:eeee,Used service:ffff,Unused service:gggg
Max item:hhh,Trcpnt interval:iiiii[sec] Swap message:jjj
***** Real time statistical storage area information *****
Number of generations:kk,Used number:ll
Start time[1]:mmmm/mm/mm mm:mm:mm End time[1]:nnnn/nn/nn nn:nn:nn
Start time[2]:mmmm/mm/mm mm:mm:mm End time[2]:nnnn/nn/nn nn:nn:nn
Start time[3]:mmmm/mm/mm mm:mm:mm End time[3]:nnnn/nn/nn nn:nn:nn
***** Real time statistical log file information *****
File put:ooo,File size:ppppppp[KB],Number of generations:qq,Used number:rr
File name:ss....ss
```

The output format is as follows when the `-m` and `-c` options are specified (CSV format):

```
***** Real time statistical information Node id:aaaa *****
Shm id:bbbbbbb,Number of shm:cccccccc,RTS status:dddd
Max service:eeee,Used service:ffff,Unused service:gggg
Max item:hhh,Trcpnt interval:iiiii[sec],Swap message:jjj
***** Real time statistical storage area information *****
Number of generations:kk,Used number:ll
Start time[1]:mmmm/mm/mm mm:mm:mm,End time[1]:nnnn/nn/nn nn:nn:nn
Start time[2]:mmmm/mm/mm mm:mm:mm,End time[2]:nnnn/nn/nn nn:nn:nn
Start time[3]:mmmm/mm/mm mm:mm:mm,End time[3]:nnnn/nn/nn nn:nn:nn
***** Real time statistical log file information *****
File put:ooo,File size:ppppppp[KB],Number of generations:qq,Used number:rr
File name:ss....ss
```

aaaa

OpenTP1 node identifier

bbbbbbbb

Identifier of the shared memory for the real-time statistics service (decimal)

cccccccc

Number of the shared memory for the real-time statistics service (decimal)

dddd

Status of the real-time statistics service

START

The service is running.

eeee

Maximum number of services for which statistics can be acquired (decimal)

The maximum number of services for which statistics can be acquired is displayed. This number is the value specified in the `rts_service_max` operand in the real-time statistics service definition.

ffff

Number of services from which statistics are currently being acquired (decimal)

gggg

Number of unused services (decimal)

The value obtained by subtracting the number of services from which statistics are currently being acquired from the maximum number of services for which statistics can be acquired (specified in the `rts_service_max` operand in the real-time statistics service definition) is displayed.

hhhh

Maximum number of items to be acquired (decimal)

The maximum number of items to be acquired is displayed. This number is the value specified in the `rts_item_max` operand in the real-time statistics service definition.

iiiiii

Acquisition interval (decimal)

jjj

Whether swap messages are output

YES

Swap messages are output.

NO

Swap messages are not output.

kk

Generation number of the area in the shared memory for the real-time statistics service from which the statistics are acquired (decimal)

ll

Number of generations of the area in the shared memory for the real-time statistics service from which statistics are currently being acquired (decimal)

mmmm/mm/mm mm:mm:mm

Time at which the acquisition of statistics started

Hyphens (-) are displayed when the acquisition of statistics has not started.

nnnn/nn/n nn:nn:nn

Time at which the acquisition of statistics ended

Hyphens (-) are displayed when the acquisition of statistics has not ended.

ooo

Whether the real-time statistics are output to an RTS log file

YES

The real-time statistics are output to an RTS log file.

NO

The real-time statistics are not output to an RTS log file.

ppppppp

Size of the RTS log file (decimal)

qq

Number of RTS log file generations

rr

Generation number of the RTS log file to which statistics are currently being output (decimal)

ss....ss

RTS log file name

■ (2) When the *-l* option is specified

The output format is as follows:

```

***** Real time statistical information Node id:aaaa *****
Server      Service
Used item   Unused item  Event id
bb....bb   cc....cc
ddd        eee          fffffff fffffff ....
    
```

The output format is as follows when the *-l* and *-c* options are specified (CSV format):

```

***** Real time statistical information Node id:aaaa *****
Server,Service,Used item,Unused item,Event id
bb....bb,cc....cc,ddd,eee,ffffff fffffff ....
    
```

1. Title of the data
2. Displayed repeatedly for the number of servers or services from which real-time statistics are acquired

aaaa

OpenTP1 node identifier

bb....bb

Server name

When the server name is `_SYSTEM` and the service name is `ΔΔΔΔ`, the target statistics are system-wide statistics. If `----` appears in this field, the real-time statistics were acquired without specifying a first acquisition target.

cc....cc

Service name

When `****` is displayed, the target statistics are the real-time statistics acquired from each applicable unit of processing other than services. When `ΔΔΔΔ` is displayed, the target statistics are the real-time statistics acquired from each server. If `----` appears in this field, the real-time statistics were acquired without specifying a second acquisition target.

ddd

Number of items in the real-time statistics that are currently being acquired (decimal)

eeee

Number of unused items (decimal)

The value obtained by subtracting the number of items in the real-time statistics that are currently being acquired from the maximum number of items to be acquired (specified in the `rts_item_max` operand in the real-time statistics service definition) is displayed.

ffffffff

Item ID in the real-time statistics that are currently being acquired (decimal)

■ (3) When the `-e` option is specified

The output format is as follows:

```

***** Real time statistical information Node id:aaaa *****
Start time[n]:bbb/bb/bb bb:bb:bb End time[n]:ccc/cc/cc cc:cc:cc
Serverz Service
Type Event id Total Maximum Minimum Average Units
dd....dd ee....ee
fff hhhhhhhhhh iiiiiiiii jiiiiijjj kkkkkkkkkkk llllllll mm...mm
    
```

The output format is as follows when the `-e` and `-c` options are specified (CSV format):

```

***** Real time statistical information Node id:aaaa *****
Start time[n]:bbb/bb/bb bb:bb:bb,End time[n]:ccc/cc/cc cc:cc:cc
Server,Service,Type,Event,Event id,Total,Maximum,Minimum,Average,Units
dd....dd,ee....ee,fff,gg....gg, hhhhhhhhhh, iiiiiiiii, jiiiiijjj, kkkkkkkkkkk, llllllll, mm...mm
    
```

1. Title of the data
2. Displayed repeatedly for the number of servers or services from which real-time statistics are acquired for each item ID

aaaa

OpenTP1 node identifier

bbb/bb/bb bb:bb:bb

Time at which the acquisition of statistics started

ccc/cc/cc cc:cc:cc

Time at which the acquisition of statistics ended

dd....dd

Server name

When the server name is `_SYSTEM` and the service name is `ΔΔΔΔ`, the target statistics are system-wide statistics. If `----` appears in this field, the real-time statistics were acquired without specifying a first acquisition target.

ee...ee

Service name

When `****` is displayed, the target statistics are the real-time statistics acquired from each applicable unit of processing other than services. When `ΔΔΔΔ` is displayed, the target statistics are the real-time statistics acquired from each server. If `----` appears in this field, the real-time statistics were acquired without specifying a second acquisition target.

fff

Type of real-time statistics

gg...gg

Item name

This is output only for CSV format.

hhhhhhhhhh

Item ID (decimal)

iiiiiiiiii

Number of statistics items that are generated (decimal)

`****` is displayed if an overflow error occurs.

jjjjjjjjjj

Maximum value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

kkkkkkkkkk

Minimum value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

llllllllll

Average value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

mm...mm

Unit of the real-time statistics

n

Generation number of the area in the shared memory for the RTS service from which statistics are acquired

■ (4) When the *-u* option is specified

The output format is as follows:

```

***** Real time statistical information Node id:aaaa *****
Start time[n]:bbb/bb/bb bb:bb:bb End time[n]:ccc/cc/cc cc:cc:cc
Server Service
Type Event id Total Maximum Minimum Average Units
dd...dd ee...ee
fff hhhhhhhhhh iiiiiiiii jjjjjjjjj kkkkkkkkkkk lllllllll mm...mm
    
```

The output format is as follows when the *-u* and *-c* options are specified (CSV format):

```

***** Real time statistical information Node id:aaaa *****
Start time[n]:bbb/bb/bb bb:bb:bb,End time[n]:ccc/cc/cc cc:cc:cc
Server,Service,Type,Event,Event id,Total,Maximum,Minimum,Average,Units
dd...dd,ee...ee,fff,gg...gg, hhhhhhhhhh, iiiiiiiii, jjjjjjjjj, kkkkkkkkkkk, lllllllll, mm...mm
    
```

1. Title of the data
2. Displayed repeatedly for the number of servers or services from which real-time statistics are acquired
3. Displayed repeatedly for the number of real-time statistics items that are acquired from servers or services

aaaa

OpenTP1 node identifier

bbb/bb/bb bb:bb:bb

Time at which the acquisition of statistics starts

ccc/cc/cc cc:cc:cc

Time at which the acquisition of statistics ends

dd...dd

Server name

When the server name is `_SYSTEM` and the service name is `ΔΔΔΔ`, the target statistics are system-wide statistics. If `----` appears in this field, the real-time statistics were acquired without specifying a first acquisition target.

ee...ee

Service name

When `****` is displayed, the target statistics are the real-time statistics acquired from each applicable unit of processing other than services. When `ΔΔΔΔ` is displayed, the target statistics are the real-time statistics that are acquired from each server. If `----` appears in this field, the real-time statistics were acquired without specifying a second acquisition target.

fff

Type of real-time statistics

gg...gg

Item name

This is output only for CSV format.

hhhhhhhhhh

Item ID (decimal)

iiiiiiii

Number of statistics items that are generated (decimal)

`****` is displayed if an overflow error occurs.

jjjjjjjj

Maximum value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

kkkkkkkkkk

Minimum value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

llllllll

Average value in the statistics (decimal)

Hyphens (-) are displayed if there are no data values. `****` is displayed if an overflow error occurs.

mm...mm

Unit of the real-time statistics

n

Generation number of the area in the shared memory for the RTS service from which statistics are acquired

Output messages

Message ID	Description	Output destination
KFCA32712-E	The version and revision combination of the real-time statistics service is incorrect.	Standard error output
KFCA32714-E	Processing cannot continue due to insufficient process memory.	Standard error output
KFCA32719-E	The command was used incorrectly.	Standard error output
KFCA32720-E	The value specified in a command option is invalid.	Standard error output
KFCA32722-E	Processing cannot be executed because shared memory is not available.	Standard error output
KFCA32723-E	Processing cannot be executed because the real-time statistics service has not started.	Standard error output
KFCA32736-I	Help message	Standard output

Notes

- Real-time statistics are not output if the acquisition of real-time statistics from the shared memory with the generation number specified in the `-n` option has not started.
- If the acquisition of real-time statistics from the shared memory with the generation number specified in the `-n` option has not ended, the command outputs the real-time statistics collected between the specified start time for acquiring statistics and the execution of the `rtsls` command.

rtssetup (Set execution environment for real-time statistics service)

Format

```
rtssetup [-d] [storage-directory-for-servers]
```

Function

The `rtssetup` command sets the environment required to execute the real-time statistics service.

Option

- `-d`

Deletes the executable modules (`rtssup` and `rtsspp`) and user service definition files (`RTSSUP` and `RTSSPP`) of the real-time statistics service from the OpenTP1 execution environment.

Command argument

- *storage-directory-for-servers*~<path name>

Specifies the directory that stores the management servers (`RTSSUP` and `RTSSPP`) for the real-time statistics service.

If you do not specify *storage-directory-for-servers*, the system assumes `$DCDIR/aplib`.

Output messages

Message ID	Description	Output destination
KFCA32719-E	The command was used incorrectly.	Standard error output
KFCA32738-I	Help message	Standard output
KFCA32739-E	An error occurred while the command was being executed.	Standard error output

Notes

- The `rtssetup` command does not set the environment in the following cases:
 - If *storage-directory-for-servers* contains files or directories that have the same names as the execution modules (`rtssup` and `rtsspp`) to be created by the `rtssetup` command
 - If the directory specified in the `$DCCONFPATH` environment variable contains files or directories that have the same names as the user service definition files (`RTSSUP` and `RTSSPP`) created by the `rtssetup` command

- When you specify a user other than an OpenTP1 administrator in the `uid` operand in the user service default definition, in the `rts_log_file_name` operand in the real-time statistics service definition, specify a directory that this user can write to.

The real-time statistics service always operates correctly as long as the user has the write authority for the directory that will contain the RTS log file, regardless of the user ID with which the service was started. If the user does not have the write authority, the service outputs the KFC A32734-W message and stops outputting log information to the RTS log file. Note that the `uid` and `groups` operands are not specified in the definition files created using the `rtsssetup` command (`RTSSUP` and `RTSSPP`). For this reason, if these operands have been specified in the user service default definition, the real-time statistics service is started as the specified user and group.

- The `dcsetup` command cannot delete the executable modules (`rtssup` and `rtsspp`) and user service definition files (`RTSSUP` and `RTSSPP`) created by the `rtsssetup` command. Use the `rtsssetup` command to delete the executable modules and user service definition files before you execute the `dcsetup` command to remove OpenTP1 from your operating system.

If you leave these files on the system when you remove OpenTP1, use the `rtsssetup` command to delete and re-create them when you next install OpenTP1.

rtsstats (Change targets and items for real-time statistics acquisition)

Format

```
rtsstats{-a [-r] | -d }
      {-u sys |
        -u srv -s server-name |
        -u svc -s server-name -v service-name |
        -u obj [-o acquisition-target-name-1] [-b acquisition-target-name-2] }
      [-e item-ID [, item-ID] ... ]
      [-f name-of-definition-file-for-items-to-be-acquired-in-real-time]
```

Function

The `rtsstats` command changes the targets for the acquisition of real-time statistics and the items to be acquired.

Options

- `-a`
Adds the targets of real-time statistics acquisition and the items to be acquired.
- `-r`
When you add an item to be acquired for the specified target for real-time statistics acquisition, this option deletes all existing items and adds the new item.
If you do not specify the `-r` option, existing items remain and the new item is added.
- `-d`
Deletes the targets of real-time statistics acquisition and the acquired items.
- `-u {sys | srv | svc | obj}`
Specifies the type of the new target of real-time statistics acquisition.
 - `sys`
Specifies the entire system as the target of real-time statistics acquisition.
When `sys` is specified, the value of the `rts_service_max` operand in the real-time statistics service definition is not affected.
 - `srv`
Specifies a server as the target of real-time statistics acquisition.
The following statistics are acquired when you add a user server name as the target in the `-s` option:

- Statistics for the specified server
- Statistics for all the services specified in the `service` operand of the user service definition for the specified user server
- Statistics for processing that is not related to services

This causes the number of targets for which statistics are to be acquired to be the number of services specified in the `service` operand plus 2.

If you specify the name of a server that has no services, such as the system server or an SUP, the command acquires statistics for only the specified server.

`svc`

Specifies a service of a user server as the target of real-time statistics acquisition.

`obj`

When you specify this option, also use the `-o` and `-b` options to specify an object as the target about which statistics are acquired.

For details on the combination of the target objects and options, see the explanation of the `-o` and `-b` options.

- `-s server-name~<Alphanumeric string of 1-8 characters beginning with an alphabetic character>`

Specifies the name of the target server.

When `srv` or `svc` is specified in the `-u` option, always specify the `-s` option. When `sys` or `obj` is specified in the `-u` option, the `-s` option cannot be specified.

- `-v service-name~<identifier of 1-31 characters>`

Specifies the name of the target service.

When `svc` is specified in the `-u` option, always specify the `-v` option. When `sys`, `srv`, or `obj` is specified in the `-u` option, the `-v` option cannot be specified.

- `-o acquisition-target-name-1~<1-8 characters>`

- `-b acquisition-target-name-2~<1-63 characters>`

Specifies the name of the target object when you specify `obj` in the `-u` option.

The following table shows the objects you can specify in the `-o` and `-b` options as targets. When `obj` is specified in the `-u` option, specify arguments in the `-o` and `-b` options according to the table.

Table 13-10: Arguments of the -o and -b options in the rtstats command, and the object determined from the specified arguments

Argument of the -o option	Argument of the -b option	Object determined from the arguments
Port number [#]	IP address [#]	Node for which the specified service information is referenced
Logical terminal name	Not specified	Specified logical terminal
Not specified	Service group name	Specified service group name

#

Specify the port number and IP address of a node specified in the `all_node` operand of the system common definition or in the domain definition file.

When `sys`, `srv`, or `svc` is specified in the `-u` option, the `-o` or `-b` option cannot be specified.

- `-e item-ID~<decimal number from 1000 to 9999>`

Specifies the item ID of the item to be added or deleted.

You can specify a maximum of 64 item IDs. For details about item IDs, see *E.5 Real-time statistics*.

- `-f name-of-definition-file-for-items-to-be-acquired-in-real-time~<identifier of 1-8 characters>`

Specifies the name of the definition file for the items to be acquired in real time that contains the item to be added or deleted.

If you want to add items, this option adds the items for which you have specified `Y` in the specified definition file for the items to be acquired in real time.

If you want to delete items, this option deletes all items except for the items for which you have specified `Y` in the specified definition file for the items to be acquired in real time.

Output messages

Message ID	Description	Output destination
KFCA32712-E	The combination of version and revision of the real-time statistics service is incorrect.	Standard error output
KFCA32714-E	Processing cannot continue due to insufficient process memory.	Standard error output
KFCA32719-E	The command was used incorrectly.	Standard error output

Message ID	Description	Output destination
KFCA32720-E	The value specified in a command option is invalid.	Standard error output
KFCA32722-E	Processing cannot be executed because shared memory is not available.	Standard error output
KFCA32723-E	Processing cannot be executed because the real-time statistics service has not started.	Standard error output
KFCA32724-E	The specified target for the acquisition of real-time statistics is not registered.	Standard error output
KFCA32737-I	Help message	Standard output
KFCA32742-E	No more targets can be added because the number of registered targets for real-time statistics acquisition is already at the maximum.	Standard error output
KFCA32743-E	No more items can be added because the number of registered items to be acquired as real-time statistics is already at the maximum.	Standard error output
KFCA32752-E	Processing cannot be executed because the extended feature of the real-time statistics service has not started.	Standard error output
KFCA32754-E	An error occurred when the RPC was executed.	Standard error output
KFCA32755-E	A timeout error occurred during the wait for a response from the extended feature of the real-time statistics service.	Standard error output
KFCA32756-E	Processing cannot be executed because the area for storing service requests for the extended feature of the real-time statistics service is insufficient.	Standard error output
KFCA32758-E	An error occurred during the analysis of the specified definition file.	Standard error output
KFCA32760-E	The number of arguments specified in the command option is invalid.	Standard error output

Notes

- The `rtsstats` command can be used only if the extended feature of the real-time statistics service (`RTSSPP`) has been started.
- The maximum number of targets for acquiring real-time statistics that you can add using the `rtsstats` command is the value specified in the `rts_service_max` operand in the real-time statistics service definition. You cannot change the value specified in the `rts_service_max` operand while the service is online.
- The maximum number of items that can be acquired for one target is the value specified in the `rts_item_max` operand in the real-time statistics service

definition. You cannot change the value specified in the `rts_item_max` operand while the service is online.

- If you omit the `-e` and `-f` options when the `-a` option is specified, only the area for storing real-time statistics is secured. The items to be acquired are not set. In this case, the number of targets for real-time statistics acquisition increases by one.
- When you specify the `-e` or `-f` option when the `-d` option is specified, the area for storing real-time statistics is not released even if all the items to be acquired have been deleted.

If you want to release the area for storing real-time statistics and decrease the number of targets, execute the `rtstats` command with the `-d` option specified but without the `-e` and `-f` options specified.

- In the `rts_item_max` operand in the real-time statistics service definition, specify a value that also covers the number of items to be acquired by the `dc_rts_utrace_put` function. You need to include the number of items to be acquired by the `dc_rts_utrace_put` function because you cannot use the `rtstats` command to add or delete the items for this function.
- When you use the `rtstats` command to change targets or the items to be acquired, the change is applied to all the generations in the shared memory for the real-time statistics service, including the generations that already completed the acquisition of real-time statistics.
- When you execute the `rtstats` command to change multiple targets (services) with `srv` specified in the `-u` option, the number of targets (services) or items to be acquired may reach the maximum value for the specified server. Although a KFCA32742-E message or a KFCA32743-E message is output, in this case, processing continues for other targets (services) that can be successfully changed.

When the number of targets (services) or the number of items to be acquired reaches the maximum value and no more targets (services) or items to be acquired can be registered for some servers, the error message only displays the first server where an error occurred. For details about the targets (services) or the acquisition items that cannot be registered, use the `rtsls` command to check the configuration of shared memory.

- When you specify `svc` for the `-u` option, the command does not check whether the server name specified in the `-s` option or the service name specified in the `-v` option actually exists. When `obj` is specified in the `-u` option, whether the values specified in the `-o` and `-b` options are valid is not checked. If unnecessary acquisition targets have been registered, execute the `rtsls` command to check their configuration, and delete them by executing the `rtstats` command with the `-d` option specified.
- If you add or remove services from the `service` operand in the user service

definition after the `rtsstats` command is executed, their status as targets for statistics acquisition does not change even if the `srv` argument is specified for the `-u` option. To add or remove such services as targets for statistics acquisition, execute the `rtsstats` command again.

Services added or deleted using the facility for dynamic loading of service functions after the `rtsstats` command is executed are also not automatically added or removed as targets of the real-time statistics service. To add or remove such services as targets for statistics acquisition, execute the `rtsstats` command again.

sdcchprc (Change the number of processes)

Format

```
sdcchprc {-a|-s server-name} [-p number-of-resident-processes
[, maximum-number-of-processes]]
```

Function

Changes the number of resident processes running on a server or the maximum number of processes that can run on a server while the server is operating. This command can be used for user servers and some system servers. As a result of the change, new processes are generated or excess processes are terminated. The number of processes changed by this command is effective until the server is terminated (including when the server is forcibly terminated). The system does not inherit the change after full system recovery. You can also only display the values currently set.

Options

- -a

The number of processes of all the queue-receiving user servers, excluding the servers that work with debuggers, and of the MHPs is to be changed.

- -s *server-name*~<identifier, 1-8 characters>

The number of processes of the specified server is to be changed.

To specify a system service, use this option.

You can specify the following server names for system services:

Transactional RPC execution process

Server name: `_clttrn`

CPU execution process

Server name: `_cltcon`

- -p *number-of-resident-processes* [, *maximum-number-of-processes*] ~<unsigned integer> ((0-1024))

Specifies the number of resident processes or the maximum number of processes that has been changed in the format of the `parallel_count` operand of the user service definition.

If you specify only the number of the resident processes, the maximum number of processes is also changed to the same value. Other specification rules are the same as for the `parallel_count` operand of the user service definition. For the

`parallel_count` operand of the user service definition, see the manual *OpenTPI System Definition*.

If you omit this option, the value currently set is displayed in the following format.

Display format

Number of servers	<i>aaaa</i>		<i>...1.</i>
Server name	<i>bb...bb</i>	When started	After being changed
		<i>cccc, dddd</i>	<i>eeee, ffff</i> <i>...2.</i>

1. Displayed only when you specify the `-a` option.
2. When you specify the `-a` option, this line is displayed for each server.

aaaa

Number of servers to be displayed (Up to four decimal digits)

bb...bb

Server name (up to eight alphanumerics)

cccc

Number of resident processes when the server is started (up to four decimal digits)

dddd

Maximum number of processes when the server is started (up to four decimal digits)

eeee

Number of resident processes after being changed by the `scdchprc` command (up to four decimal digits)

For the server whose number of processes has not been changed after startup, '****' is displayed.

ffff

Maximum number of processes after being changed by the `scdchprc` command (up to four decimal digits)

For the server whose number of processes has not been changed after startup, '****' is displayed.

Output messages

Message ID	Description	Output destination
KFCA00843-I	The command was normally processed.	Message log file

Message ID	Description	Output destination
KFCA00844-E	The command processing failed.	Message log file, standard error output
KFCA00883-I	Help message	Standard output, standard error output

Notes

- You must install TP1/Extension 1 to use this command. The operation cannot be guaranteed if TP1/Extension 1 is not installed.
- This command does not change the value specified in the `parallel_count` operand of the user service definition.
- Even when this command seems to be completed, the command may be still executing. Therefore, the number of processes may not be changed immediately. When reducing the number of processes, processes are normally terminated, therefore, it may take some time to reduce the number of processes depending on the execution status of the processes.
- To reduce the maximum number of processes to less than the number of processes being started, reduce the number of processes being started regardless of the number of service requests remained for the server. (Reduce the number of processes with the multi-server load balancing facility being ignored.)
- Increasing the number of resident processes exceeding the maximum number of processes for OpenTP1 or the system causes a command error.
- If you change the resident server at startup (number of resident processes = maximum number of processes) to the non-resident server (number of resident processes < maximum number of processes), the change is not reflected to the following definition operands related to the non-resident process.

User service definition

```

    termed_after_service
    service_wait_time

```

User service default definition

```

    termed_after_service
    service_wait_time

```

- You cannot apply the `scdchprc` command to the following servers. If you specify any of these servers in the `-s` option, the command outputs the message KFCA00844-E and returns an error.

- An SUP
- A server that receives requests from a socket
- A server that works with debuggers
- System services (including the following servers)
 - A RAP-processing server
 - An RTSSPP (server for extended feature of real-time statistics service)
- When the -a option is specified, the load on the system may increase due to the large number of user server processes being started or changed in a short space of time.

scdhold (Shut down scheduling)

Format

```
scdhold {-a|-s server-name [-c server-name]|-ap
        |-s server-name -p}
```

Function

The `scdhold` command shuts down the scheduling of a server or service working under the scheduling service. However, the scheduling of a service can be shut down only with `service_hold=Y` (shut down on a service basis) specified in the user service definition.

This command cannot be used to shut down scheduling for the system service or for MHP (a server for which `type=MHP` is specified in the user service definition or a server for which `type=MHP` is specified in the user service default definition and for which the type operand is omitted from the user service definition). To shut down scheduling for MHP, the `mcfstdctsv` command must be used.

Options

- `-a`
Specifies that scheduling for all servers operating under the scheduling service, except for MHP and the system service, is to be shut down.
- `-s server-name~<identifier, 1-8 characters>`
Specifies the name of a server whose scheduling is to be shut down.
- `-c service-name~<1-31 character identifier>`
Shuts down the scheduling of the server specified in the `-s` option or the specified service. For a server for which `service_hold=Y` (shut down on a service basis) has not been specified in the user service definition, an error message (KFCA00890-E) is displayed. When this option is omitted, the scheduling of the server specified in the `-s` option is shut down.
- `-ap`
Shuts down the scheduling of all servers (except MHP and the system service) working under the scheduling service. Service requests from clients are accepted.
- `-s server-name -p~<1-8 character identifier>`
Shuts down the scheduling of the server with the specified server name. Service requests to applicable servers are accepted.

Output messages

Message ID	Description	Output destination
KFCA00881-I	Help message	Standard output, standard error output
KFCA00890-E	Error occurred during <code>scdhold</code> command processing.	Standard error output
KFCA00891-E	Usage of the <code>scdhold</code> command is invalid.	Standard error output
KFCA00892-E	<code>scdhold</code> command execution environment has not been set up.	Standard error output
KFCA00893-E	Specified server name is invalid.	Standard error output
KFCA00894-E	<code>scdhold</code> command cannot be executed due to inconsistent versions.	Standard error output
KFCA00895-E	Batch shutdown processing failed.	Standard error output
KFCA00897-E	Specified server has already been shut down.	Standard error output

Notes

You cannot apply the `scdhold` command to the following servers. If you specify any of these servers in the `-s` option, the command outputs the message KFCA00893-E and returns an error.

- An SUP
- A server that receives requests from a socket
- A server that works with debuggers
- System services (including the following servers)
 - A RAP-processing server
 - An RTSSPP (server for extended feature of real-time statistics service)

scdls (Display scheduling status)

Format

```
scdls [{-a|-s server-name|-ac|-s server-name -c|-ae|-s server-name -e|
      -b[scheduling-buffer-group-name]|-p} [-t]]
```

Function

The `scdls` command outputs the scheduled status of a server working under the scheduling service, or service status to the standard output.

This command outputs the status of a buffer pool for storing messages to the standard output for each scheduling buffer group.

This command also outputs the load level of the scheduler daemon receiving the service request to the standard output.

Options

- `-a`

Specifies that information about all servers, except the system service, that are active at the time the `scdls` command is entered is to be displayed. When this option is omitted, `-a` is assumed.

- `-s server-name~<identifier, 1-8 characters>`

Specifies the name of a server whose scheduling status is to be displayed.

- `-ac`

Displays the status of all servers working under the scheduling service, and service status. For a server for which `service_hold=N` (do not shut down on a service basis) has been specified in the user service definition, only its status is displayed.

- `-s server-name -c~<1-8 character identifier>`

Displays the service status of the specified server. For a server for which `service_hold=N` (do not shut down on a service basis) is specified in the user service definition, only its status is displayed.

- `-ae`

Displays the statuses of all services on all servers working under the scheduling service.

Statuses are displayed only for servers for which `service_hold=Y` (shut down on a service basis) or the `scdsvcdef` definition command (which specifies operations on a service basis) has been specified in the user service definition.

- `-s server-name -e<identifier of 1 to 8 characters>`

Displays the service status of the specified server.

Statuses are displayed only for servers for which `service_hold=Y` (shut down on a service basis) or the `scdsvcdef` definition command (which specifies operations on a service basis) has been specified in the user service definition.

- `-b scheduling-buffer-group-name`

Displays the status of a buffer pool for storing messages used by the scheduling buffer group, and the scheduling status of the user server in that scheduling buffer group.

Specify the scheduling buffer group name specified in the schedule service definition.

When a scheduling buffer group name is not specified, information on all the scheduling buffer group names is displayed.

- `-p`

Displays the load level of the scheduler daemon receiving the service request.

- `-t`

Displays the time when a service request was last retrieved from the schedule queue. Do not specify this option at the same time as the `-b` or `-p` option.

Output format

- (1) *Scheduling status displayed*

```

total server count aaaa
Server ST QUECNT MAXCNT USED_POL FREE_POL MAX_UPOL MAX_FPOL
bb...bb c dddddd eeeee fffffff gggggggg hhhhhhhh iiiiii
Service group SCD_TIME
jj...jj kk...kk
4

```

1. This information is displayed only when `scdls -a` is specified.

3. If `scdls -a` is specified, this information is displayed as many times as there are servers operating under the scheduler.

The 2. lines and the 3. lines are usually each displayed on a single line; on some display devices, they may be broken into multiple lines.

4. If the `-t` option is specified, this information is displayed only as many times as there are servers.

aaaa

Number of servers operating under the scheduler (decimal)

bb...bb

Server name (up to 8 characters)

c

Status of the corresponding server:

S: Under server preparation

A: Can be scheduled

E: Under server termination processing

H: Server shutdown status

P: Service shutdown status being ready to accept service requests

dddddd

Number of service requests remaining in the scheduling queue (decimal, 6 digits)

eeeeee

Maximum number of service requests having remained in the scheduling queue (decimal, 6 digits)

fffffff

Size of the message storage pool currently in use (decimal)

gggggggg

Size of the message storage pool currently not in use (decimal)

hhhhhhhh

Maximum size of the message storage pool that can be used (decimal)

iiiiiii

Current maximum size of unused contiguous area in the message storage pool (decimal)

jj...jj

Service group name (up to 31 characters)

kk...kk

Time when a service request was last retrieved from the schedule queue

The time is displayed according to the setting in the TZ environment variable, based on the number of seconds from 00:00:00 on January 1, 1970. If no service requests have ever been retrieved from the schedule queue, the time is displayed as follows:

Example:

Thu Jan 1 09:00:00 1970 (when JST-9 is set for the TZ environment variable)

Notes

- If the status of the corresponding server is E, 0 may be displayed for ddddd through iiii.ii.
- The information indicating the schedule status of MHP is the server name, server status and service group name. Ignore the other information, which is the control information for the schedule service.

■ (2) *Service status displayed*

```

total server count  aaaa                               1

Server  ST  Service_group      ST  Service
bb...bb  c  dd...dd             e  ff...ff

┌──────────────────┬──────────┐
                              2          3

ST  Service              QUECNT  MAXCNT  USED_POL  MAX_UPOL  4
c  ff...ff              gggggg  hhhhhh  iiiiii   iiiiii   5
└──────────────────┘

                                                                 6

PAR_C  MAX_P  SCD_TIME  4
kkkk  IIII  mm...mm  5
┌────────┬────────┐
          6          7

```

1. This information is displayed only when `scdls -ac` is specified.
2. If `scdls -ac` is specified, this information is displayed as many times as there are servers working under the scheduling service.
3. For a server for which `service_hold=Y` (shut down on a service basis) is specified in the user service definition, this information is displayed as many times as there are services. If the `-e` option is specified, the information is displayed in the formatting indicated by the number 6.
4. and 5. This information is usually displayed on a single line, but may be broken into multiple lines on some display devices.
6. When `service_hold=Y` (shut down on a service basis) has been specified in the user service definition, or the `scdsvcddef` definition command is specified to control queues on a service basis, this information is displayed as many times as there are services. This information is displayed following the information indicated by the number 2 without a line break. If the `-c` option is specified, the information is displayed in the formatting indicated by the number 3.

7. If the `-t` option is specified, this information is displayed as many times as there are services. This information is displayed following the information indicated by the number 3 or 6, without a line break.

aaaa

Number of servers operating under the scheduling service (decimal, 1-4 digits)

bb...bb

Server name (1-8 characters)

c

Status of the corresponding server

S: Under server preparation

A: Can be scheduled

E: Under server termination processing

H: Server shutdown status

P: Shutdown status being ready to accept service requests

dd...dd

Service group name (1-31 characters)

e

Status of the service

S: Under server preparation

A: Can be scheduled

E: Under server termination processing

H: Service shutdown status

P: Shutdown status being ready to accept service requests

ff...ff

Service name (1-31 characters)

gg...gg

The number of service requests currently in the schedule queue for the service (up to 6 decimal digits)[#]

hh...hh

The maximum number of service requests that have been queued in the schedule queue for the service (up to 6 decimal digits)[#]

ii..ii

The current size of the buffer pool for storing messages used by the service (up to 8 decimal digits. Unit: bytes)[#]

jj..jj

The maximum size of the buffer pool for storing messages used by the service (up to 8 decimal digits. Unit: bytes)[#]

kk...kk

The number of instances of the service being executed concurrently (up to 4 decimal digits)[#]

ll...ll

The maximum number of instances of the service that have been executed concurrently (up to 4 decimal digits)[#]

mm...mm

Time when a request for the service was last retrieved from the schedule queue[#]

The time is displayed according to the TZ environment variable, based on the number of seconds from 00:00:00 on January 1, 1970. If no requests for the service have ever been retrieved from the schedule queue, the time is displayed as follows:

Example:

Thu Jan 1 09:00:00 1970 (when JST-9 is set for the TZ environment variable)

#

If the `scdsvcdef` definition command for controlling queues on a service basis is not specified in the user service definition, `** . . *` is displayed for these items.

■ (3) *Status of buffer pool for storing messages displayed*

```
scheduling buffer group count   aaaa                               ...1.
group  pool length used   max. use                               ...2.
bb..bb  cc..cc   dd..dd  ee..ee                               ...3.
server status quecount max. value used   max. use                               ...4.
ff..ff   g      hh..hh      ii..ii   jj..jj   kk..kk                               ...5.
```

The lines 1. to 5. are each displayed on a single line.

1. This information is displayed only when a scheduling buffer group name is not specified.
 5. This information is displayed for each of the servers sharing the scheduling buffer.
- If a scheduling buffer group name is not specified, lines of 2. to 5. are displayed for each of the buffer groups.

aaaa

Number of scheduling buffer groups (1-4 digits decimal)

bb...bb

Scheduling buffer group name (up to 8 characters identifier)

cc...cc

Length of the buffer pool for storing messages in the scheduling buffer group (1-8 digits decimal, bytes)

dd...dd

Size of the buffer pool for storing messages currently in use (1-8 digits decimal, bytes)

ee...ee

Current maximum size of the available buffer pool for storing messages

If the buffer pool for storing message is insufficient for storing a message, ***** is displayed.

ff...ff

Server name (up to 8 characters identifier)

g

Server status (one character)

S: Preparing

A: Available for scheduling

E: In process of server termination processing

H: Shutdown

F: Shutdown but available for accepting service

hh...hh

Number of service requests remaining in the scheduling queue of the server (1-6 digits decimal)

ii...ii

Maximum number of service requests having remained in the scheduling queue of the server (1-6 digits decimal)

jj..jj

Size of the buffer pool for storing messages used by the server (1-8 digits decimal, bytes)

kk...kk

Maximum size of the buffer pool for storing messages having used by the server (1-8 digits decimal, bytes)

■ (4) Load level of scheduler daemon displayed

SCDNAME	TYPE	THREAD(NOW PEAK PEAKTIME)			
<i>aaaaaaaa</i>	<i>bbbbbbb:bbbb</i>	<i>cccc</i>	<i>dddd</i>	<i>ee:ee:ee</i>	1.
					2.
SERVERBUSY(TOTAL PEAK PEAKTIME)	SCHEDULE(NOW PEAK PEAKTIME)				
<i>ffffff</i>	<i>ggggggggg</i>	<i>hh:hh:hh</i>	<i>iiii.iiii</i>	<i>jjjj-jjjj</i>	<i>kk:kk:kk</i>
					2.
RECEIVE(NOW PEAK PEAKTIME)					
<i>llllllll</i>	<i>mmmmmmmmmm</i>	<i>nn:nn:nn</i>			
					1.
					2.

1 and 2:

This information is usually displayed on a single line. On some display devices, however, it may be broken into multiple lines.

2.

The information is displayed for each of the scheduler daemons if the multi-scheduler facility starts multiple scheduler daemons.

aa...aa

Server name of scheduler daemon (up to 8 characters)

bb...bb

Identifier of scheduler daemon (up to 14 characters), consisting of the multi-scheduler group name and port number delimited by ": "

For a master scheduler daemon, the multi-scheduler group name is indicated as *********, and the port number is indicated as ********* when the `scd_port` operand of the schedule service definition is not specified.

cc...cc

Number of active processing threads (1-5 digits decimal)

dd....dd

Maximum number of active processing threads (1-5 digits decimal)

ee...ee

Time when the number of active processing threads rises to a maximum (8 characters)

When there is no active processing thread, "***.***.***" is displayed.

ff...ff

Number of times a shortage of processing threads occurs (1-10 digits decimal)

gg...gg

Maximum number of times a shortage of processing threads occurs successively (1-10 digits decimal)

hh...hh

Time when the number of times a shortage of processing threads occurs rises to a maximum (8 characters)

When no shortage of processing threads has occurred, "***.***.***" is displayed.

ii...ii

Period of time between when a processing thread receives a service request and when scheduling is completed (11 digits including 6 decimal places)

If the period of time for scheduling becomes less than 0 because the time of the OS is reset or because of any other reasons, "*****.*****" is displayed.

jj...jj

Maximum period of time between when a processing thread receives a service request and when scheduling is completed (11 digits including 6 decimal places)

kk...kk

Time when the period of time between when a processing thread receives a service request and when scheduling is completed rises to a maximum (8 characters)

When no service requests are received, "***.***.***" is displayed.

ll...ll

Number of service requests received in unit time (1-10 digits decimal)

This indicates the number of service requests received during the interval processing time that the scheduler daemon performs every 10 seconds. If the interval processing time takes much time, the unit time may become longer than 10 seconds.

mm...mm

Maximum number of service requests received in unit time (1-10 digits decimal)

nn...nn

Time when the number of service requests received in unit time rises to a maximum (8 characters)

When no service requests are received, "***.**.*" is displayed.

Notes

The information of scheduler daemon's load level is updated or referenced without locking to minimize the influence on the processing performance. Therefore, the displayed information may be invalid depending on the timing. In this case, re-execute the command. The information displayed by this option only includes service requests provided via the scheduler daemon. If a service request is scheduled to SPP of the local node, the information of the service request may not be updated because it bypasses the scheduler daemon.

Output messages

Message ID	Description	Output destination
KFCA00880-I	Help message	Standard output, standard error output
KFCA00890-E	Error occurred during <code>scdls</code> command processing.	Standard error output
KFCA00891-E	Usage of the <code>scdls</code> command is invalid.	Standard error output
KFCA00892-E	<code>scdls</code> command execution environment has not been set up.	Standard error output
KFCA00893-E	Specified server name is invalid.	Standard error output
KFCA00894-E	<code>scdls</code> command cannot be executed due to inconsistent versions.	Standard error output

scdrles (Release scheduling shutdown)

Format

```
scdrles {-a|-s server-name [-c service-name] | -ap
        |-s server-name -p}
```

Function

The `scdrles` command restarts the scheduling of servers operating under the scheduling service, or the specified service. Scheduling of a service can be restarted only when `service_hold=Y` (shut down on a service basis) is specified in the user service definition.

This command cannot be used to release scheduling shutdown for the system service or MHP (a server for which `type=MHP` is specified in the user service definition or a server for which `type=MHP` is specified in the user service default definition and for which the type operand is omitted from the user service definition). To release scheduling shutdown for MHP, the `mcftactsv` command must be used.

Options

- `-a`
Specifies that scheduling shutdown is to be released for all servers operating under the scheduling service, except for MHP and the system service.
- `-s server-name~<identifier, 1-8 characters>`
Specifies the name of a server whose scheduling shutdown is to be released.
- `-c service-name~<1-31 character identifier>`
Restarts the scheduling of the server specified in the `-s` option or the specified service. For a server for which `service_hold=Y` (shut down on a service basis) has not been specified in the user service definition, an error message (KFCA00890-E) is displayed. When this option is omitted, the scheduling of the server specified in the `-s` option is restarted.
- `-ap`
Restarts the scheduling of servers (except MHP and the system service) which are working under the scheduling service and which have been shut down by the `scdhold` command with its `-p` option specified.
- `-s server-name -p~<1-8 character identifier>`
Restarts the scheduling of the server with the specified server name. A command error occurs if the scheduling has not been shut down with the `scdhold` command with its

-p option specified.

Output messages

Message ID	Description	Output destination
KFCA00882-I	Help message	Standard output, standard error output
KFCA00890-E	Error occurred during <code>scdrls</code> command processing.	Standard error output
KFCA00891-E	Usage of the <code>scdrls</code> command is invalid.	Standard error output
KFCA00892-E	<code>scdrls</code> command execution environment has not been set up.	Standard error output
KFCA00893-E	Specified server name is invalid.	Standard error output
KFCA00894-E	<code>scdrls</code> command cannot be executed due to inconsistent versions.	Standard error output
KFCA00896-E	Batch shutdown release processing failed.	Standard error output
KFCA00897-E	Specified server has not been shut down.	Standard error output

Notes

If you specify multiple service groups with the `scdrls` command, or execute the `scdrls` command multiple times consecutively, the large amount of activity relating to process activation increases the system's CPU usage. You can plan the system operation and leave an interval between executions of the `scdrls` command so as to avoid the undesirable effects that accompany high CPU usage.

You cannot apply the `scdrls` command to the following servers. If you specify any of these servers in the `-s` option, the command outputs the message KFCA00893-E and returns an error.

- An SUP
- A server that receives requests from a socket
- A server that works with debuggers
- System services (including the following servers)
 - A RAP-processing server
 - An RTSSPP (server for extended feature of real-time statistics service)

scdrsprc (Stop and restart processes)

Format

```
scdrsprc {-s server-name|-a}
```

Function

The `scdrsprc` command stops and restarts user server processes running under the scheduling service.

Options

- `-s server-name`

Specifies the name of a user server whose processes are to be stopped and restarted.

- `-a`

Stops and restarts all the queue-receiving user servers and MHPs operating under the scheduling service.

Output messages

Message ID	Description	Output destination
KFCA00884-I	Help message	Standard error output
KFCA00890-E	Error occurred during <code>scdrsprc</code> command processing.	Standard error output
KFCA00891-E	Usage of the <code>scdrsprc</code> command is invalid.	Standard error output
KFCA00892-E	<code>scdrsprc</code> command cannot be executed because the scheduler is not running.	Standard error output
KFCA00893-E	Specified server name, service group name, or service name is invalid.	Standard error output
KFCA00894-E	<code>scdrsprc</code> command cannot be executed due to inconsistent versions.	Standard error output

Notes

- To execute the `scdrsprc` command, TP1/Extension 1 must be installed. The operation of this command is not guaranteed if TP1/Extension 1 is not installed.
- The `scdrsprc` command finishes the moment it successfully receives a request. Therefore, process swapping will not have completed fully at the target user server when the command finishes.

- Executing the `scdrsprc` command immediately terminates all processes that are waiting for service requests. However, processes that are currently executing server requests terminate only when the request is completed. New processes are activated at preset intervals (10 seconds) by the OpenTP1 scheduler daemon. It may therefore take some time to swap every process.
- You cannot apply the `scdrsprc` command to the following servers. If you specify any of these servers in the `-s` option, the command outputs the message KFCA00893-E and returns an error:
 - An SUP
 - A server that receives requests from a socket
 - A server for which either of the following operands is specified in the user service definition or user service default definition:
 - `termed_after_service = Y`
 - `service_wait_time = length of time to wait for service requests from non-resident processes of the user server`
 - A server that works with debuggers
 - System services (including the following servers)
 - A RAP-processing server
 - An RTSSPP (server for extended feature of real-time statistics service)
- When the `-a` option is specified, the load on the system may increase due to the large number of user server processes being stopped and restarted in a short space of time.

stsclose (Close a status file)

Format

```
stsclose {-n logical-file-name|-f physical-file-name}
```

Function

The `stsclose` command closes a currently open status file.

A status file on current status cannot be closed.

This command can close only a status file specified in the status service definition, and the command is effective only while the status service is running.

Options

- `-n logical-file-name`~<identifier, 1-8 characters>

Specifies the name of the logical file to be closed. The files in both the A and B systems constituting the specified logical file are closed simultaneously.

This command argument can be specified in the following cases:

- Either the A-system file or the B-system file is on BLOCKADE or STANDBY status
- Both the A-system file and the B-system file are on BLOCKADE or STANDBY status

When a logical file name is specified, the user need not know the path name for the status file entity.

- `-f physical-file-name`~<path name>

Specifies the name of the physical file to be closed, expressed as the full path name.

If an error has occurred in a status file, this command argument is specified to close the file in the system on STANDBY status, not in the system on shutdown status.

Entering both `stsclose -f A-system-physical-file-name` and `stsclose -f B-system-physical-file-name` is the same as entering `stsclose -n logical-file-name`.

Output messages

Message ID	Description	Output destination
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01050-E	Error occurred during process-to-process communication.	Standard error output

Message ID	Description	Output destination
KFCA01070-E	Error occurred during <code>stsclose</code> command processing.	Standard error output
KFCA01084-I	Help message	Standard output, standard error output
KFCA01099-E	Error occurred during <code>OpenTP1</code> function call.	Standard error output

stsfills (Display contents of a status file)

Format

```
stsfills -f physical-file-name [-cx]
```

Function

The `stsfills` command outputs the contents of a status file to the standard output in the offline mode.

Options

- `-f physical-file-name~<path name>`

Specifies the name of a physical file, expressed as the full path name.

- `-c`

Specifies that the status file is to be checked. The following types of information are checked:

- Record (beginning or ending) integrity check number for record damage;
- File management information during `stsininit` command execution for the record length and number of records
- Status management information for completion of file update processing
- Status file management information for the number of records and the record type

If the check reveals an inconsistency, the contents of the specified status file are displayed, and an error message is output.

When this option is omitted, the status file is not checked.

- `-x`

Specifies that the status file is to be opened in the lock mode. If the `stsfills` command is lock at the time the status service is started, OpenTP1 may terminate abnormally with a lock error.

When this option is omitted, the status file is not opened in the lock mode. If the specified status file is being used by OpenTP1, the correct status may not be displayed.

Output format

```

Path name: aa...aa
  Initial- Current   R    R    Utiliza- Available   Number of
  ization- file selec- length count  tion    consecutive management
  time     tion time  length count  factor  records    records
bb...bb   cc...cc      dddd ee...ee  fff%   gg...gg    hh...hh
    
```

aa...aa

Name of the physical file (up to 63 characters)

bb...bb

Initialization date and time (in the format year/month/day Δ hour: minutes: seconds). The last two digits of the year are displayed for the year.

cc...cc

Current file selection time (in the format year/month/day Δ hour: minutes: seconds). The last two digits of the year are displayed for the year.

dddd

Record length (decimal)

ee...ee

Number of records (decimal)

fff

Record utilization factor for the file

gg...gg

Number of consecutive records available in the file (decimal)

hh...hh

Number of management records in the file (decimal)

Note

If an error occurs in the file or the file has been damaged, the displayed information may not be correct.

Output messages

Message ID	Description	Output destination
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01070-E	Error occurred during <code>stsfills</code> command processing.	Standard error output

stsfills (Display contents of a status file)

Message ID	Description	Output destination
KFCA01088-I	Help message	Standard output, standard error output
KFCA01091-E	Memory is inadequate.	Standard error output
KFCA01099-E	Error occurred during OpenTP1 function call.	Standard error output

stsinit (Create and initialize a status file)

Format

```
stsinit -f physical-file-name [-s record-length] [-c records-count]
```

Function

The `stsinit` command initializes an OpenTP1 file system to the format that can be used by the status service.

This command cannot initialize a status file being used (on current or standby status) by the status service in the online mode.

Options

- `-f physical-file-name~<path name>`
Specifies the name of the physical file to be initialized, expressed as the full path name.
- `-s record-length~((512-32768)) <<4608>> (bytes)`
Specifies the record length of the status file.
Specifies the multiple of sector length specified in OpenTP1 file system creation (`filmkfs` command).
- `-c records-count~((32-4194304)) <<256>>`
Specifies the number of records in the status file.

Notes

- The current status file is managed only while the status service is active. The user must ensure that the current status file used in the online mode will not be initialized.
- The physical files in the A and B systems constituting a single logical file must be initialized so that they both have the same record length and the same number of records.
- The number of records may be changed in units of logical files, but the record length cannot be changed.
- The default value should be used for the record length. If the record length is less than 4608 bytes, the number of status file I/O operations increases. If it is greater than 4608 bytes, status file utilization efficiency decreases.
- The number of records in a status file depends on the OpenTP1 system configuration. The status file record utilization factor can be checked with the

stsinit (Create and initialize a status file)

`sts1s` command in the online mode or the `stsfills` command in the offline mode. The number of records should be changed on the basis of the record utilization factor. If the status file record utilization factor becomes high during online operation, a status file containing more records than the current file can be initialized as a standby file.

stsls (Display status of status files)

Format

```
stsls [{-n logical-file-name|-f physical-file-name|-a|-l|-p}]
```

Function

The `stsls` command outputs the status of status files to the standard output in the online mode.

The command is effective only when the status service is running.

Options

- `-n logical-file-name~<identifier, 1-8 characters>`

Specifies the name of the logical file whose status is to be displayed.

- `-f physical-file-name~<path name>`

Specifies the name of the physical file whose status is to be displayed, expressed as the full path name.

- `-a`

Specifies that the status of all status files in use by the status service is to be displayed in summary format.

- `-l`

Specifies that the status of all logical files being used by the status service is to be displayed.

- `-p`

Specifies that the status of all physical files being used by the status service is to be displayed.

When all options are omitted, the status of all status files being used by the status service is displayed.

Output format

The following shows the output format when all options are omitted:

stsls (Display status of status files)

Logical file name	Logical status	Utilization factor	Available consecutive records	Number of management records	...
<i>aaaaaaa</i>	<i>bbbbbbb</i>	<i>ccc%</i>	<i>ddddddd</i>	<i>eeeeeee</i>	...2.
System	Physical status	Record length	Number of records	Path name	...
<i>f</i>	<i>ggggggg</i>	<i>hhhhhhh</i>	<i>iiiiiii</i>	<i>jj..jj</i>	...1., 2.
<i>f</i>	<i>ggggggg</i>	<i>hhhhhhh</i>	<i>iiiiiii</i>	<i>jj..jj</i>	...1., 2.

1.Physical file status display.

2.Logical file status display.

aaaaaaa

Logical file name (up to 8 characters)

bbbbbbb

Logical file status

ACTIVE: Current file

BLOCKADE: Error shutdown file

CLOSE: Invalid file

NONE: Status in which there is no file entity

STANDBY: Spare file

ccc

Record utilization factor for the file (percent, 3 digits)

ddddddd

Number of consecutive records available in the file (decimal)

eeeeeee

Number of management records in the file (decimal)

f

Whether physical file is for system A or B:

A: A system

B: B system

ggggggg

Physical file status:

a: Current file

- b: Error shutdown file
- c: Closed file
- i: Initialized status
- l: Logical error
- n: Status in which there is no file entity
- o: Open file
- p: Physical error
- r: File status recovery by error information
- s: Spare file
- u: Used file

The display position of the letter indicating the status depends on the physical file's status.

hhhhhhh

Record length (decimal)

iiiiiii

Number of records (decimal)

jj..jj

Physical file name (up to 63 characters)

Output messages

Message ID	Description	Output destination
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01050-E	Error occurred during process-to-process communication.	Standard output
KFCA01070-E	Error occurred during <code>stsls</code> command processing.	Standard error output
KFCA01085-I	Help message	Standard output, standard error output
KFCA01091-E	Memory is inadequate.	Standard error output
KFCA01099-E	Error occurred during OpenTP1 function call.	Standard error output

Notes

The status of a status file is managed only while the status service is running. If the `stssinit` or `stssrm` command is executed for any of the following closed files during

stsls (Display status of status files)

status service operation, the file status to be displayed by the `stsls` command will not change:

- CLOSE
- NONE
- BLOCKADE

If the `stssinit` or `stssrm` command, the `stssopen` command, and the `stsls` command are executed in this order, the current file status is displayed.

stsopen (Open a status file)

Format

```
stsopen {-n logical-file-name|-f physical-file-name}
```

Function

The `stsopen` command opens a status file initialized by the `stsinit` command or a status file closed by the `stsclose` command.

This command can open only a status file specified in the status service definition, and the command is effective only when the status service is running.

If the current status file is a one-system operation file, executing the `stsinit` command to initialize the physical file in the shutdown-state system and then executing the `stsopen` command make it possible to recover the file as a current status file.

Options

- `-n logical-file-name~<identifier, 1-8 characters>`

Specifies the name of a logical file to be opened. The files in both the A and the B systems constituting the specified logical file are opened simultaneously.

This command argument can be specified in the following cases:

- Either the A-system file or the B-system file is on CLOSE, NONE, or BLOCKADE status
- Both the A-system file and the B-system file are on CLOSE, NONE, or BLOCKADE status

When a logical file name is specified, the user need not know the path name for the status file entity.

- `-f physical-file-name~<path name>`

Specifies the name of a physical file to be opened, expressed as the full path name.

This command argument is specified to open a status file that was placed on shutdown status due to a status file error or to open a physical file that has been initialized by the `stsinit` command in order to increase its size.

Entering both `stsopen -f A-system-physical-file-name` and `stsopen -f B-system-physical-file-name` is the same as entering `stsopen -n logical-file-name`.

stlopen (Open a status file)

Output messages

Message ID	Description	Output destination
KFCA01008-E	Error occurred in status file.	Standard error output
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01050-E	Error occurred during process-to-process communication.	Standard error output
KFCA01070-E	Error occurred during <code>stlopen</code> command processing.	Standard error output
KFCA01083-I	Help message	Standard output, standard error output
KFCA01091-E	Memory is inadequate.	Standard error output
KFCA01099-E	Error occurred during OpenTP1 function call.	Standard error output

stsrn (Delete a status file)

Format

```
stsrn -f physical-file-name
```

Function

The `stsrn` command deletes a status file.

This command cannot delete a status file being used (on current or standby status) by the status service in the online mode.

Options

- `-f physical-file-name~<path name>`

Specifies the name of a physical file to be deleted.

Output messages

Message ID	Description	Output destination
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01070-E	Error occurred during <code>stsrn</code> command processing.	Standard error output
KFCA01082-I	Help message	Standard output, standard error output
KFCA01091-E	Memory is inadequate.	Standard error output
KFCA01099-E	Error occurred during OpenTP1 function call.	Standard error output

Notes

Because a current status file is managed only while the status service is active, the following must be noted about executing the `stsrn` command:

- A current status file being used in the online mode must not be deleted.
- Only a status file placed on shutdown status due to an error should be deleted.

stsswap (Swap status files)

Format

stsswap

Function

The `stsswap` command switches the status of a status file from current to spare. As a result, the current status file is used as a standby and the standby as a current one.

Output messages

Message ID	Description	Output destination
KFCA01040-E	Error occurred in physical file.	Standard error output
KFCA01050-E	Error occurred during process-to-process communication.	Standard error output
KFCA01062-I	<code>stsswap</code> command processing has started.	Standard output
KFCA01063-I	<code>stsswap</code> command processing has been completed.	Standard output
KFCA01064-E	Error occurred during swap processing.	Standard error output
KFCA01070-E	Error occurred during <code>stsswap</code> command processing.	Standard error output
KFCA01086-I	Help message	Standard output, standard error output
KFCA01091-E	Memory is inadequate.	Standard error output
KFCA01099-E	Error occurred during <code>OpenTP1</code> function call.	Standard error output

tamadd (Add a TAM table)

Format

```
tamadd [-r loading-timing] [-a access-mode] [-i] [-j]
      TAM-table-name TAM-file-name
```

Function

The `tamadd` command adds a specified TAM table and catalogs it into the online system.

The added and cataloged TAM table is placed on logical shutdown status.

Options

- `-r loading-timing`~<alphabetic> <<start>>

Specifies the TAM table loading timing:

`start`

Load the TAM table during execution of the `tamadd` command

`cmd`

Load the TAM table during execution of the `tamload` command

`lib`

Load the TAM table when `dc_tam_open` function is issued

- `-a access-mode`~<alphabetic> <<read>>

Specifies the TAM table's access mode:

`read`

Reference type

`rewrite`

Update type that cannot be added or deleted

`write`

Update type that can be added or deleted

`reclck`

Update type that does not secure table locks. `reclck` allows addition and deletion.

- -i

Assumes the specified TAM table as an I/O failure continuation table. This promises an advantage: if an I/O error occurs when a TAM file is updated, the appropriate TAM table is not placed on error shutdown status in the same online system. In other words, the TAM file can be accessed even after the I/O error has occurred. One thing to consider is to prevent mismatch in TAM file status when online operation restarts. To this end, the affected TAM table (I/O failure continuation table) is removed from the online system when it restarts. The user must recover the TAM file, and add and catalog to the online system again.

If the -i option is omitted, the TAM file is placed on error shutdown status when an I/O error occurs during updating of the TAM file.

- -j

When a TAM record is updated, collects the differential journal. The differential journal is collected by comparing the record before updating and that after updating. The records are compared from the beginning in 32 bytes. Where a difference is encountered first from the beginning becomes the journal collection starting position. Then, the records are compared from the end in 32 bytes. Where a difference is encountered first from the end becomes the journal collection ending position.

When there is more than one difference in one record, a larger journal is collected than the amount of difference.

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of the TAM table to be added and cataloged.

- *TAM-file-name*~<path name>

Specifies the name of the TAM file corresponding to the TAM table, expressed as the full path name.

Output messages

Message ID	Description	Output destination
KFCA01700-E	Specified file is not a TAM file.	Standard error output
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01706-E	Length of TAM file name exceeds 63 characters.	Standard error output
KFCA01709-E	User does not have access right to TAM file.	Standard error output

Message ID	Description	Output destination
KFCA01710-E	User does not have access right to a special file.	Standard error output
KFCA01712-E	Number of arguments specified in command is invalid.	Standard error output
KFCA01713-E	Specified file was not found.	Standard error output
KFCA01715-E	Specified TAM file name is not a special file name.	Standard error output
KFCA01716-E	Specified file is being used by another process.	Standard error output
KFCA01749-I	Cataloging of TAM table has been completed.	Message log file
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01758-E	Specified TAM table name has already been cataloged.	Standard error output
KFCA01760-E	Specified TAM file name has already been cataloged.	Standard error output
KFCA01761-E	I/O error occurred.	Standard error output
KFCA01762-E	Record damage occurred.	Standard error output
KFCA01764-E	V/R of TAM table is invalid.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01772-E	Specified special file name has not been initialized for the OpenTP1 file system.	Standard error output
KFCA01773-E	System version used for file system creation does not match system version used to execute operation command.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02856-I	Help message	Standard output
KFCA02857-E	There is no free space for cataloging the specified TAM table.	Standard error output
KFCA02858-E	There are too many open OpenTP1 files.	Standard error output
KFCA02866-E	Record length of the specified TAM file exceeds the maximum record length.	Standard error output
KFCA02868-E	Loading timing argument is invalid.	Standard error output
KFCA02870-E	Access mode argument is invalid.	Standard error output

Message ID	Description	Output destination
KFCA02886-E	The specified TAM table cannot be cataloged since the value of a table attribute is different from its previous value.	Standard error output

Notes

- When start is specified for the loading timing, the corresponding TAM table will be loaded the next time the TAM service is started.
- When the user is adding and cataloging a TAM table once removed from the online system, OpenTP1 checks for the following items. If different file attributes are found, a `tamadd` command error occurs.
 - Record length
 - Key length
 - Key startup location
 - Maximum number of records
 - Index type
 - Number of hash entries
 - Presence/absence of the key area in records
- The user may have removed a TAM table from the online system with the table opened by a UAP and then added the table to the system again. In this case, the status of whether the `-u` option was specified or not when the table was opened is inherited until it is re-opened with the appropriate UAP. If the user again adds the table and opens it, the status for specification of the `-u` option at this time becomes valid. Thus, it is recommended that the user remove the TAM table for which the `-u` option was specified when added to the online system after the UAP that opened the table, or specify the same option as used when the table was added previously when the table is added online again.
- When you add and catalog a TAM table again that was once removed from the online system, you cannot change the option of the TAM table from `-a reclck` (if it is specified) to a different option or from an option other than `-a reclck` to `-a reclck`. If you change the option, the `tamadd` command will output the KFCA02886-E message.

tambkup (Back up a TAM file)

Format

```
scedhold [-d] [-o] {TAM-file-name file-name | -s TAM-file-name}
```

Function

The `tambkup` command backs up the contents of a specified TAM file into a specified file or standard output.

The contents backed up from the TAM file can be restored with the `tamrstr` command.

The `tambkup` command also outputs the specified TAM file record to the specified file in the form of a TAM data file.

Options

- -d

Creates a TAM data file from effective records in a TAM file. The TAM data file is to store data sorted in ascending order on the basis of a key value.

The `tambkup` command with its `-d` option specified must be executed when online operation terminates normally or after executing the `tamhold` and `tamrm` commands.

An error occurs if there is no effective record in the TAM file.

An error also occurs if broken records in the TAM file are detected.

- -o

Backs up a TAM file in online mode.

This option is valid only when the TAM file is in the online mode.

If the option is not specified, the TAM file is to be backed up in the offline mode. The offline mode backup procedure is as follows:

1. Execute the `tamhold` command to logically shut down the TAM table.
2. Execute the `tamrm` command to remove the logically shutdown TAM table from the online system.
3. Execute the `tambkup` command with its `-o` option not specified to backup the TAM file.

- -s

This option must be specified to output the backup to the standard output.

The `-d` and `-o` options are mutually exclusive.

Command arguments

- *TAM-file-name*~<path name>

Specifies the name of the backup source TAM file, expressed as the full path name.

- *file-name*~<path name>

Specifies the name of the backup destination file.

Specifies the name of the TAM data file when the `-d` option is specified.

This command argument cannot be specified with the `-s` option.

Notes

- When the `tambkup` command with its `-d` option specified is executed to exchange the contents of a TAM file, the user must ensure matching in journal generation number. If an attempt is made to recover the TAM file (execute the `tamfrc` command) or restart the online system with generation mismatching, system operation is not guaranteed.
- The information stored in a TAM file before exchange is not reflected in the file at online restart if:

The user executes the `tambkup` command with its `-d` option specified for a TAM file that was removed from the online system with the `tamhold` and `tamrm` commands, thereby exchanging the contents of the TAM file and then executes the `tamadd` command to add and catalog the file to the online system.

- The `tambkup` command with its `-o` option not specified cannot be used for a TAM file being used online.
- Once online backup is completed, a message (KFCA01738-I) is output. This message is usually output to the standard output, but output to the standard error output when the `-s` option is specified. The message contains the generation number and block number of journal files to be recovered. Recovery of a TAM file requires the unloaded-journals file starting with the generation number and block number.
- The memory requirement for the `tambkup` command depends on the size of the TAM file to be backed up. Use the following formula to estimate the memory requirement (in bytes):

$$400000 + A + B$$

Legend:

A: *record-count* x *record-length*. Add the value of *A* only when the `-d` option is specified.

B: Size of the target TAM file. For the formula for estimating TAM file size, see *H.7 Estimating the size of a TAM file*.

tamcre (Initialize a TAM file)

Format

```
tamcre -r record-length -l key-area-length -k key-begin-position
      -m max-records-count [-t] [-u hash-entry-utilization-factor
      [-x] [-y]] [-s] [-d TAM-data-file-name] TAM-file-name
```

Function

The `tamcre` command creates a new TAM file based on specified options.

Options

- `-r record-length~<unsigned integer> ((1-1000000000))`
Specifies the record length of the TAM file.
- `-l key-area-length~<unsigned integer> ((1-1000000000))`
Specifies the key length.
- `-k key-begin-position~<unsigned integer> ((1-1000000000))`
Specifies the length from the beginning of a record to the key begin position.
If the `-s` option is specified, 0 must be specified for this option.
- `-m max-records-count~<unsigned integer> ((1-1000000000))`
Specifies the maximum number of records in the TAM table.
- `-t`
Specifies that the TAM table is to be created in tree format.
If this option is omitted, the TAM table will be created in hash format. When this option is omitted, the `-u` option must be specified.
- `-u hash-entry-utilization-factor~<unsigned integer> ((1-100))`
Specifies the utilization factor for the index to be used as the hash area.
When this option is omitted, the `-t` option must be specified.
- `-x`
When there is any initial data to create the TAM file in the hash format, the synonym area is usually optimized to reduce the size of the TAM file and amount of shared memory.
The `-x` option specifies not to optimize the synonym area.

Specify this option when using the TAM file as:

- The hash format TAM table to be added or deleted online by a command.
- The hash format TAM table whose records are to be added or deleted by a UAP.

■ -y

Changes the hash function to be used for a hash-format TAM file. Specify this option in the following cases:

- When you use a key whose key value does not change much even though the key length is rather long
- When the amount of synonym information to be acquired using the `tamhsls` command after the creation of a TAM file is considerably large

■ -s

Specifies that the key area is to be deleted from the records.

When this option is specified, the record length of the TAM file becomes "specified record length - key area length".

■ -d *TAM-data-file-name*~<path name>

Specifies a name for the file in which the TAM table data is to be stored.

When this option is omitted, a physical file only is allocated.

Command arguments

■ *TAM-file-name*~<path name>

Specifies the name of the OpenTP1 file system in which the TAM table is to be stored.

If an existing TAM file name is specified, an error message (KFCA02836-E) is displayed. A new TAM file name must be specified.

Notes

- If the `-t` and `-u` options are both specified, an error occurs.
- To initialize a hash type TAM file with the `tamcre` command, the index area as large as the maximum number of records specified in the `-m` option must be allocated, and the area specified in the `-u` option (hash entry utilization percentage) as the hash area. In addition, how many records are to be contained in the synonym area must be calculated from the data to be read from a data file, and the result allocated as the synonym area. Thus, the resulting total of records exceeds the maximum number of records specified in the `-m` option.
- When you specify the `-y` option, the hash function for retrieving records from key values changes and you can expect increased accuracy in the distribution of hash values. However, the performance of the hash function itself may decrease when

compared to a hash function that is used for a TAM file without the `-y` option.

- The memory requirement for the `tamcre` command depends on the size of the TAM file to be created. Use the following formula to estimate the memory requirement (in bytes):

$$400000 + A + B + C$$

Legend:

A: *record-count* x *record-length*. Add the value of *A* only when the `-d` option is specified.

B: Size of the target TAM file. For the formula for estimating the TAM file size, see *H.7 Estimating the size of a TAM file*.

C: *record-count* x 8 for a 32-bit OS; *record-count* x 16 for a 16-bit OS

tamdel (Delete a TAM file)

Format

```
tamdel TAM-file-name
```

Function

The `tamdel` command deletes a specified TAM file.

Command arguments

- *TAM-file-name*~<path name>

Specifies the name of the TAM file to be deleted, expressed as the full path name.

tamfrc (Recover a TAM file)

Format

```
tamfrc [-s] [-e] [-g] [-k key] [-m] [-j]
      {-f recovery-target-definition-file-name |
       TAM-table-name TAM-file-name}
      journal-file-name [ [  $\Delta$  journal-file-name ] . . . ]
```

Function

The `tamfrc` command recovers a TAM file from specified journal files.

Options

- `-s`
Specifies that the previous TAM FRC is not to be inherited.
When this option is omitted, the previous TAM FRC is inherited.
- `-e`
Specifies that the inheritance file is to be deleted when TAM FRC terminates.
When TAM FRC is executed with this option specified, the `-s` option must be specified the next time TAM FRC is executed.
When this option is omitted, the inheritance file is not deleted.
- `-g`
Specifies that TAM FRC is to be executed even if the `-s` option is specified and unload journal files with journal generation number 1 are not specified.
When this option is omitted, processing is cancelled.
When the `-s` option is omitted, this option is ignored, if specified.
- `-k key~((001-999)) <<001>>`
Specifies the key of the applicable FRC when multiple TAM FRCs are to be executed.
In addition, specify different keys when the `damfrc` command and the `mqafrc` command are executed simultaneously with the `tamfrc` command or when the `damfrc` command, `mqafrc` command, and `tamfrc` commands are executed in succession.
To inherit the previous TAM FRC, its key must be specified.

- -m

Integrates the journal records needed for file recovery on a file. When this option is omitted, a buffer area is allocated on memory for integration of journal records.

- -j

Specifies that TAM FRC is to be executed using journal files integrated by the `jnlcolc` command.

When this option is specified, the `-s`, `-e` and `-m` options must not be specified.

When this option is omitted, TAM FRC is executed using unload journal files created by the `jnlunlfg` command.

- -f *recovery-target-definition-file-name*~<path name>

Specifies the name of the file that defines the TAM file to be recovered.

The recovery target definition file is created in the following format, using a text editor:

```
[Δ] TAM-table-name Δ TAM-file-name (linefeed)
[[Δ] TAM-table-name Δ TAM-file-name (linefeed)]
:      :
```

TAM-table-name~<1-to-32-character identifier>

Name of the TAM table to be recovered

TAM-file-name~<1-to-63-character path name>

Name of the recovery destination TAM file (full path name)

Specify a file restored from the backup file. When specifying multiple files, either the files restored from the online backup files or the files restored from the offline backup files can be specified. These two kinds of files cannot be both specified.

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of the TAM table to be recovered.

- *TAM-file-name*~<path name>

Specifies the name of the TAM file to be recovered, expressed as the full path name.

- *journal-file-name*~<path name>

Specifies the name of a journal file to be used during TAM FRC execution.

If the `-j` option is specified, the name of an integrating journal file must be specified; if the `-j` option is omitted, the name of an unload journal file must be specified.

If multiple journal generations are to be processed, all the corresponding journal file names must be specified. No more than one integrating journal file can be specified.

Recovering online backed-up files does not require all unloaded-journals files to be specified. The user has only to specify the unloaded-journals files beginning with the generation number and block number of the journal files to be recovered that are contained in the message (KFCA01738-I) that is output at completion of online backup.

Recovery of offline backed-up files requires specification of all unloaded-journals files.

Notes

- If the `tamfrc` command is incorrectly specified, a message with no message ID may be output to suggest the correct use of the `tamfrc` command.
- The file attributes (table format, record length, maximum number of records, key value, etc.) of a TAM file specified in `tam-file-name` must match those of a TAM file that corresponds to the TAM table accessed in the online mode.
- The `jnlcolc` command is used inside the `tamfrc` command; an error message for the `jnlcolc` command may be displayed. Take proper action according to the instructions contained in the manual *OpenTPI Messages*.
- When the `tambkup` command with its `-d` option specified is executed to exchange the contents of a TAM file, the user must ensure matching in journal generation number. If an attempt is made to recover the TAM file (execute the `tamfrc` command) with generation mismatching, system operation is not guaranteed.
- When the recovery target definition file is specified by the `-f` option, do not specify a TAM table name or a TAM file name in the command argument.
- Set the write authority in the current directory to allow the `tamfrc` command to create a temporary file in the current directory under the following conditions. The disk capacity of the temporary file is as follows:
 1. When the `tamfrc` command without the `-j` option attempts to recover the TAM file restored from the offline backup file^{#1}:
Maximum $4096 + a$ (bytes)
 2. When the `tamfrc` command without the `-j` option attempts to recover the TAM file restored from the online backup file^{#2}:
Maximum $90 + 4096 + a$ (bytes)
 3. Other than above (the `tamfrc` command has the `-j` option specified):
No temporary file is created.

Legend:

a: Total disk capacity for the unload journal file^{#3} specified at execution of the `tamfrc` command

#1: TAM file backed up by executing the `tambkup` command offline (without the `-o` option).

#2: TAM file backed up by executing the `tambkup` command with the `-o` option specified.

#3: Use the `ls` command of UNIX to see the total disk capacity. If two or more unload journal files are specified, the sum of their disk capacity is applicable.

- Do not specify an unload journal file that is acquired by specifying the `-t` option in the `jnlunlfg` command.

tamhold (Shut down a TAM table logically)

Format

<code>tamhold TAM-table-name</code>

Function

The `tamhold` command shuts down a specified TAM table logically.

Command arguments

- *TAM-table-name*~<character, identifier, 1-32 characters>

Specifies the name of the TAM table to be placed on logical shutdown status.

Output messages

Message ID	Description	Output destination
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01750-I	Logical shutdown of TAM table has been completed.	Message log file
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02852-I	Help message	Standard output
KFCA02860-E	Specified TAM table is already on error shutdown status.	Standard error output
KFCA02861-E	Specified TAM table is already on logical shutdown status.	Standard error output
KFCA02874-W	Logical shutdown processing will be retried, because a transaction is executing.	Standard error output

tamhsls (Display the synonym information about a hash-format TAM file and TAM table)

Format

```
tamhsls { -m TAM-table-name | TAM-file-name }
```

Function

The `tamhsls` command displays the information about the synonyms in the specified hash-format TAM table and TAM file.

Options

- `-m`

Displays the information about the synonyms in the loaded TAM table while OpenTP1 is online. When you specify this option, specify in the command argument the name of the TAM table that is already loaded.

Command arguments

- *TAM-table-name*~<identifier of 1 to 32 characters>

Specify the name of a hash-format TAM table that is already loaded for which you want to display the information about synonyms. When you specify a tree-format TAM table, the information will not be displayed.

- *TAM-file-name*~<path name>

Specify the name of a hash-format TAM file for which you want to display the information about synonyms using a full path name. When you specify a tree-format TAM table, the information will not be displayed.

Output format

```
HASH FUNCTION NO = a
RECNO  EFFECT  SYN_COUNT  KEY
bb..bb  cc...cc  dd...dd   ee...ee
```

a

Hash function number used

0: Old hash number

1: New hash number

bb...bb

Record number in the TAM table

cc...cc

Validity

0: Not used.

1: Being used.

dd...dd

Number of records in the synonym area corresponding to the applicable record (hash area). When a UAP accesses a record in a TAM table, the UAP uses the hash function to convert a key value into a hash value and uses it as a record number. Next, the UAP compares key value *ee...ee* indicating the record number and the key value the UAP specifies. If *ee...ee* does not match the key value specified by the UAP, the UAP performs a sequential search for *dd...dd* records in the synonym area.

ee...ee

Key value (hexadecimal) in the applicable record (hash area)

Output messages

Message ID	Description	Output destination
KFCA01700-E	The specified file is not a TAM file.	Standard error output
KFCA01704-E	The TAM service is not started.	Standard error output
KFCA01707-E	The TAM table name is incorrect.	Standard error output
KFCA01709-E	You do not have the access permission for the TAM file.	Standard error output
KFCA01710-E	You do not have the access permission for the special file.	Standard error output
KFCA01711-E	The disk partition is not initialized as an OpenTP1 file system.	Standard error output
KFCA01715-E	The specified TAM file is not a special file.	Standard error output
KFCA01716-E	The specified file is being used by another process.	Standard error output
KFCA01773-E	The version is invalid.	Standard error output
KFCA02811-E	The memory for the process-specific area became insufficient while the command was executed.	Standard error output
KFCA02812-E	An I/O error occurred.	Standard error output
KFCA02818-E	Input from the TAM file is impossible.	Standard error output
KFCA02835-E	The record is damaged.	Standard error output

Message ID	Description	Output destination
KFCA26205-E	The TAM command is used incorrectly.	Standard error output
KFCA26206-E	Help message	Standard output
KFCA26207-E	An error occurred when you attempted to acquire TAM synonym information.	Standard error output

tamlockls (Convert TAM locked resource names)

Format

```
tamlockls resource-name [resource-name] . . .
```

Function

The `tamlockls` command converts the specified resource names as follows:

- Table resource: TAM table name
- Record resource: TAM table name and record key value

Command argument

- *resource-name*~<character string>

Specifies the resource name to be converted in the following format.

Table resource

Six-byte character string beginning with T.

Record resource

Sixteen-byte character string beginning with R.

Both resources are the output information of the `lockls` command or deadlock information file.

Output format

```
Resource Name = [aa...aa]
                --> TAM Table name = [bb...bb]
                --> TAM Record Key (Length=[cc])
                [dd...dd] : ee...ee           ...l.
```

1. Up to 16 bytes are output on this line. Data of more than 16 bytes are output on multiple lines. The line consisting of less than 16 bytes is padded with "00".

aa...aa

Resource name specified in the argument

bb...bb

TAM table name for the resource name

cc

Key value length of the record for the record resource

dd...dd

Key value of the record for the record resource (16-byte hexadecimal)

ee...ee

Key value of the record for the record resource. A printable character is displayed as it is. An unprintable character is displayed as ".".

Output message

Message ID	Description	Output destination
KFCA20203-E	An error occurred during collection of lock information.	Standard error output
KFCA20204-I	Help message	Standard output
KFCA20205-E	Command argument has error.	Standard error output

tamload (Load a TAM table)

Format

```
tamload TAM-table-name
```

Function

The `tamload` command loads a specified TAM table into memory.

This command can load only TAM tables for which `cmd` was specified in the TAM service definition as the loading timing. If the loading timing is not `cmd`, an error will occur.

When the `tamload` command is entered with the TAM table on error shutdown status, an error occurs. Also, with the TAM table on status other than error shutdown, an error occurs if loading a TAM table whose shutdown status has been released with the `tamrles` command is attempted using the `tamload` command.

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of a TAM table.

Output messages

Message ID	Description	Output destination
KFCA01700-E	Specified file is not a TAM file.	Standard error output
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01761-E	I/O error occurred.	Standard error output
KFCA01762-E	Record is damaged.	Standard error output
KFCA01764-E	V/R of TAM table is invalid.	Standard error output

Message ID	Description	Output destination
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02851-I	Help message	Standard output
KFCA02867-E	Loading timing is not an operation command entry.	Standard error output
KFCA02869-E	TAM table has been loaded.	Standard error output
KFCA02871-I	Loading of TAM table has been completed.	Message log file
KFCA02882-E	TAM table cannot be loaded because it is on error shutdown status.	Standard error output
KFCA02897-E	Processing cannot be continued because TAM table is on error recovery waiting status.	Standard error output

tamls (Display status of TAM tables)

Format

```
tamls [-a] [TAM-table-name]
```

Function

The `tamls` command outputs the status of TAM tables to the standard output.

Options

■ `-a`

Specifies to output the following information:

- TAM file name
- TAM file status
- Access type
- Loading timing
- TAM record length
- Key area in records
- Key length
- Security
- I/O failure processing mode
- Key start location

Command arguments

■ `TAM-table-name`~<character string, identifier, 1-32 characters>

Specifies the name of a TAM table.

When this command argument is omitted, the status of all TAM tables is displayed.

Output format

■ (1) The *-a* option specified

```
TAM table name      = xx...xx      ...l.
TAM file name      = xx...xx      ...l.
TAM file status    = aa...aa      ...l.
TAM table status   = bb...bb      TAM table number      = xx...xx      ...l.
Index type         = cc...cc      Number of records in use = xx...xx      ...l.
Access type        = dd...dd      Maximum number of records = xx...xx      ...l.
Loading timing     = ee...ee      TAM record length       = xx...xx      ...l.
Key area in records = xx...xx      Key length               = xx...xx      ...l.
Security           = f              I/O failure processing mode = gg...gg      ...l.
Journal collection mode = hh...hh  Key start location      = xx...xx      ...l.
```

1. If no TAM table name is specified, the information is displayed repeatedly as many times as the number of TAM tables used.

aa...aa

TAM file status

normal: Non-shutdown status

failure shutdown: Error shutdown status

bb...bb

TAM table status (one of the following):

normal: Non-shutdown status

logical shutdown: Logical shutdown status

failure shutdown: Error shutdown status

failure recovery: Error recovery wait status

cc...cc

Index type (one of the following):

tree: Tree format:

hash: Hash format

dd...dd

Access type

read: Reference type

rewrite: Update type that cannot be added or deleted

write: Update type that can be added or deleted

`reclck`: Update type that does not secure table locks. `reclck` allows addition and deletion.

ee...ee

Loading timing

`start`: When the `tamadd` command is executed

`cmd`: When the `tamload` command is executed

`lib`: When the `dc_tam_open` function is issued

f

Access authority check

Y: Checked

N: Not checked

gg...gg

Failure processing mode of TAM file on I/O error

`continue`: Continues processing

`stop`: Assumes error shutdown status and Terminates processing

hh...hh

Journal collection mode

`condense`: Differential journal collection mode

`no condense`: Entire-record journal collection mode

■ (2) *The -a option not specified*

```
TAM table name   = xx...xx          ...l.  
TAM table status = aa...aa TAM table number   = xx...xx          ...l.  
Index type      = bb...bb  Number of records in use = xx...xx          ...l.  
Maximum number of records = xx...xx          ...l.
```

1. If no TAM table name is specified, the information is displayed repeatedly as many times as the number of TAM tables used.

aa...aa

TAM table status

`normal`: Non-shutdown status

`logical shutdown`: Logical shutdown status

`failure shutdown`: Error shutdown status

failure recovery: Error recovery wait status

bb...bb

Index type

tree: Tree type

hash: Hash type

Output messages

Message ID	Description	Output destination
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01761-E	I/O error occurred.	Standard error output
KFCA01764-E	V/R of TAM table is invalid.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02854-I	Help message	Standard output
KFCA02878-E	Specified table is not a TAM table.	Standard error output

tamrles (Release TAM table from shutdown)

Format

```
tamrles [-o] [-f] TAM-table-name
```

Function

The `tamrles` command releases a specified TAM table from shutdown status.

If removed from the online system with the `tamrm` command or unloaded with the `tamunload` command, the TAM table released from shutdown is on non-shutdown status.

An error occurs if the `tamhold` command and the `tamrles` command with no option specified are executed in this order for a TAM table that has been released from shutdown with the `tamrles` command with the `-f` option specified.

Options

■ `-o`

Specifies that the TAM table is to be released from error shutdown status. Later, the TAM table can be accessed in the online mode.

If the TAM table is not on error shutdown status and this option is specified, an error occurs.

When the TAM service is started, the TAM table that was shut down due to an error during the time from opening to loading of a file cannot be released from shutdown. The `-o` option cannot coexist with the `-f` option.

■ `-f`

Releases a TAM table from shutdown due to an error. An error occurs if an attempt is made to access the released TAM table in the online mode. In order for the TAM table to be accessed, it must be removed from the line, recovered, and added again.

An error occurs if the `-f` option is specified for a TAM table whose status is other than error shutdown status.

When the TAM service is started, the TAM table that was shut down due to an error during the time from opening to loading of a file cannot be released from shutdown. The `-o` option cannot coexist with the `-f` option.

Following are the operation commands to be executed, the functions to be issued, and the processing to be performed, after execution of the `tamrles` command with its `-f` option specified.

Operation commands to be executed, functions to be issued	Action
tamrles	displays 'failure recovery' (error recovery waiting status)
tamrm, tamrles, tamunload, tamhold	treats as shutdown status
tamload	error
tamrles after executing tamhold	displays 'failure recovery' (error recovery waiting status)
tamrles after executing tamhold	error
tambkup with -o option specified	error
dc_tam_get_inf function	returns the return value, DCTAM_STS_OHLD (error shutdown status)
dc_tam_get_inf function after executing tamhold	returns the return value, DCTAM_STS_LHLD (logical shutdown status).

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of a TAM table to be released from shutdown status.

Output messages

Message ID	Description	Output destination
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01751-I	TAM table shutdown release processing has been completed.	Message log file
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output

tamrles (Release TAM table from shutdown)

Message ID	Description	Output destination
KFCA02853-I	Help message	Standard output
KFCA02859-E	Specified TAM table is not on shutdown status.	Standard error output
KFCA02863-E	Specify the -o option, because the specified TAM table is on error shutdown status.	Standard error output
KFCA02864-E	There is no need to specify the -o option, because the specified TAM table is on logical shutdown status.	Standard error output
KFCA02881-E	TAM table cannot be released from shutdown status, because it has not been loaded. Delete the TAM table.	Standard error output
KFCA02896-E	More than one option cannot be specified.	Standard error output
KFCA02897-E	Processing cannot be continued because TAM table is on error recovery waiting status.	Standard error output

tamrm (Remove a TAM table)

Format

tamrm <i>TAM-table-name</i>

Function

The `tamrm` command removes a specified TAM table from the online system.

Before the `tamrm` command is executed, the TAM table must be placed on logical shutdown status by the `tamhold` command.

A TAM table on error shutdown status can also be removed from the online system.

If the maximum number of TAM files specified in `tam_max_tblnum` in the TAM service definition have been used, no new TAM file can be added or cataloged by the `tamadd` command even after removing a TAM table from the online system with the `tamrm` command. TAM tables with the same file attributes as the one removed from the line can be added and cataloged.

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of a TAM table to be removed.

Output messages

Message ID	Description	Output destination
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01748-I	TAM table removal has been completed.	Message log file
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01761-E	I/O error occurred.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output

tamm (Remove a TAM table)

Message ID	Description	Output destination
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02855-I	Help message	Standard output
KFCA02859-E	Specified TAM table is not on shutdown status.	Standard error output
KFCA02875-W	Removal processing will be retried, because a transaction is executing.	Standard error output
KFCA02893-E	Retry processing will be suspended, because transaction has not terminated.	Standard error output

Note

When the online system is restarted due to a failure of the online system while a TAM table is being removed from the online system, the file attributes of the TAM table (record length and number of records) are not inherited.

tamrstr (Restore a TAM file)

Format

```
tamrstr {file-name TAM-file-name} [-s TAM-file-name]
```

Function

The command restores to a TAM file the contents of a file backed up by the `tambkup` command from the file or standard input.

Options

- `-s`

This option must be specified when the restoration source is a standard input.

Command arguments

- `file-name~<path name>`

Specifies the name of the source file for restoration.

This command argument cannot be specified with the `-s` option.

- `TAM-file-name~<path name>`

Specifies the name of the restoration destination TAM file, expressed as the full path name.

Note

The memory requirement for the `tamrstr` command depends on the size of the TAM file to be restored. Use the following formula to estimate the memory requirement (in bytes):

$$400000 + A \times 2$$

Legend:

Size of the target TAM file. For the formula for estimating the TAM file size, see *H.7 Estimating the size of a TAM file*.

tamunload (Unload a TAM table)

Format

```
tamunload TAM-table-name
```

Function

The `tamunload` command unloads a specified TAM table. It can also unload a TAM table that is on error shutdown status.

Before the `tamunload` command is executed, the TAM table must be placed on logical shutdown status by the `tamhold` command.

Command arguments

- *TAM-table-name*~<character string, identifier, 1-32 characters>

Specifies the name of a TAM table.

Output messages

Message ID	Description	Output destination
KFCA01701-E	Memory is inadequate.	Standard error output
KFCA01703-E	Address information of TAM service cannot be retrieved.	Standard error output
KFCA01704-E	TAM service has not been started.	Standard error output
KFCA01707-E	TAM table name is invalid.	Standard error output
KFCA01712-E	Number of arguments specified in the command is invalid.	Standard error output
KFCA01753-E	Overtime occurred.	Standard error output
KFCA01757-E	TAM table name exceeds 32 characters.	Standard error output
KFCA01765-E	V/R of TAM server is invalid.	Standard error output
KFCA01767-E	Network error occurred in RPC.	Standard error output
KFCA01770-E	TAM service is terminating.	Standard error output
KFCA01771-W	Unload processing will be retried, because a transaction is executing.	Standard error output
KFCA01775-E	Error occurred during definition analysis.	Standard error output
KFCA02850-I	Help message	Standard output
KFCA02859-E	Specified TAM table is not on shutdown status.	Standard error output

Message ID	Description	Output destination
KFCA02867-E	Loading timing is not an operation command entry.	Standard error output
KFCA02872-I	TAM table unloading has been completed.	Message log file
KFCA02873-E	TAM table has already been unloaded.	Standard error output
KFCA02893-E	Retry processing will be suspended, because transaction has not terminated.	Standard error output

tptrnls (Display undecided transaction information for OSI TP communication)

Format

```
tptrnls
```

Function

The `tptrnls` command displays information, including ID and status, about undecided transactions that perform the OSI TP communication using XATMI. This command also uses the OSI TP format to display information about transaction branches across more than one OSI TP communication.

Output format

```
TRNGTRID = aa...aa TRNBID = bb...bb
STATUS = cc...cc PID = dd...dd SERVER = ee...ee SERVICE = ff...ff
[ ATOMIC-ACTION-ID = gg...gg-hh...hh-ii...ii
  BRANCH-ID = jj...jj-kk...kk-ll...ll
  PARTNER-AE = mm...mm ]
```

The above format covers one transaction branch. The command outputs this format for each of the undecided transactions. The information enclosed in the brackets [] is output for each of the times there is more than one OSI TP communication that a transaction branch is performing.

1. *aa...aa*
Global transaction ID (character string)
2. *bb...bb*
Transaction branch ID (character string)
3. *cc...cc*
Transaction branch status (character string)
4. *dd...dd*
Process ID of the process in which the transaction is operating (decimal)
5. *ee...ee*
Name of the server that started the transaction branch (character string)
6. *ff...ff*

Name of the service that started the transaction branch (character string)

7. *gg...gg*

Master AP name of the atomic action identifier (hexadecimal)

8. *hh...hh*

Master AE qualifier of the atomic action identifier (decimal)

9. *ii...ii*

Suffix of the atomic action identifier (atomic action number) (decimal)

10. *jj...jj*

Superior AP name of the branch identifier (hexadecimal)

11. *kk...kk*

Superior AE qualifier of the branch identifier (decimal)

12. *ll...ll*

Suffix of the branch identifier (branch number) (decimal)

13. *mm...mm*

Remote AE name (hexadecimal)

1. and 2. are the same as TRNGID and TRNBID, respectively, output by the `trnls` command.

3. is the same as the transaction's first status output by the `trnls` command.

4., 5. and 6. are the same as PID, server, and service, respectively, output by the `trnls` command.

7., 8., 10., 11. and 13. are the same as the items defined for TP1/NET/OSI-TP-Extended.

Output messages

Message ID	Description	Output destination
KFCA03770-E	Execution of the XATMI command failed.	Standard error output
KFCA03771-I	There is no undecided transaction concerning XATMI.	Standard output

Notes

- This command is available only when the OpenTP1 system is running.
- An error during execution of the command terminates the command after the output of the KFCA03770-E message.

tptrnls (Display undecided transaction information for OSI TP communication)

- If there is no information about undecided transaction branches concerning XATMI, the command terminates after the output of the KFCA03771-I message.
- Since this command does not take any options or arguments, they will be ignored if specified for this command.

trncmt (Commit transactions)

Format

```
Trncmt {-t [-af] | -T transaction-global-identifier [-af]}
```

Function

When the information displayed by the `trnls` command indicates `READY(p,n)` status, the `trncmt` command commits forcibly a transaction branch, notifies the other transaction branches, and then terminates the transaction. This command should be entered when a root transaction branch is to be committed.

The `trncmt` command is entered when no transaction branch constituting a global transaction is able to determine the transaction for some reason (communication error, etc.). All the transaction branches in the global transaction must be committed in order to avoid inconsistency with other transactions. If a communication error has occurred, the transaction cannot be terminated until communication among its transaction branches has been completed. When the `-f` option is specified, the transactions are terminated forcibly. If the communication error is temporary, it is better to execute the `trncmt` command without specifying the `-f` option.

Options

- `-t`

Specifies that all transactions on `READY(p,n)` status that are managed by the transaction manager on the corresponding computer are to be committed. Information about the transactions to be committed will be output to the standard output.

- `-T transaction-global-identifier~<16 alphanumeric characters>`

Specifies the transaction global identifier that is to be committed, but only if it is on `READY(p,n)` status. Information about the transaction to be committed will be output to the standard output.

The transaction global identifier can be obtained by the `trnls -t` command.

- `-a`

Specifies that all information about the transactions is to be output to the standard output.

When this option is omitted, only the OpenTP1 system node ID through the service name will be output to the standard output.

- `-f`

Specifies that the transactions are to be terminated forcibly.

trncmt (Commit transactions)

When this option is omitted, the transactions are not terminated forcibly.

Output format

When the `-a` option is specified, the output format is the same as output format (1) for the `trnls` command; otherwise, it is the same as output format (5) for the `trnls` command.

Output messages

Message ID	Description	Output destination
KFCA00970-E	Processing error occurred during <code>trncmt</code> command execution.	Standard error output
KFCA00976-I	Help message	Standard output
KFCA00978-E	Usage of <code>trncmt</code> command is invalid.	Standard error output
KFCA00979-E	<code>trncmt</code> command execution environment has not been set up.	Standard error output

trndlinf (Delete undecided-transaction information files)

Format

```
trndlinf -d day-count
```

Function

The `trndlinf` command deletes undecided-transaction information files.

Options

- `-d day-count`~((1 to 24855))

Specifies the number of days for which the files are not deleted. Deleted are the undecided-transaction information files created "24 x *day-count*" or more hours before the execution time of the `trndlinf` command.

Output messages

Message ID	Description	Output destination
KFCA00970-E	Processing error occurred during <code>trndlinf</code> command execution.	Standard error output
KFCA00972-I	Help message	Standard output
KFCA00978-E	Usage of <code>trndlinf</code> command is invalid.	Standard error output
KFCA00979-E	<code>trndlinf</code> command execution environment has not been set up.	Standard error output

trnfgt (Terminate transactions forcibly)

Format

```
Trnfgt {-t [-af] |-T transaction-global-identifier [-af]}
```

Function

The `trnfgt` command terminates transactions forcibly.

Options

- `-t`

Specifies that all transactions on `HEURISTIC_FORGETTING(p,n)` status that are managed by the transaction manager for the corresponding computer are to be terminated. Information about the transactions to be terminated will be output to the standard output.

- `-T transaction-global-identifier<16 alphanumeric characters>`

Specifies the transaction global identifier that is to be terminated, but only if it is on `HEURISTIC_FORGETTING(p,n)` status. Information about the transaction to be terminated will be output to the standard output.

The transaction global identifier can be obtained by the `trnls -t` command.

- `-a`

Specifies that all information about the transactions is to be output to the standard output.

When this option is omitted, only the OpenTP1 system node ID through the service name will be output to the standard output.

Output format

When the `-a` option is specified, the output format is the same as output format (1) for the `trnls` command; otherwise, it is the same as output format (5) for the `trnls` command.

Output messages

Message ID	Description	Output destination
KFCA00970-E	Processing error occurred during <code>trnfgt</code> command execution.	Standard error output
KFCA00974-I	Help message	Standard output
KFCA00978-E	Usage of <code>trnfgt</code> command is invalid.	Standard error output

Message ID	Description	Output destination
KFCA00979-E	trnfgt command execution environment has not been set up.	Standard error output

trnlInkrm (Catalog and delete resource managers)

Format

```
trnlInkrm { [-A OpenTP1-RM-name-to-be-added
             [, OpenTP1-RM-name-to-be-added] ... ]
           [-D OpenTP1-RM-name-to-be-deleted
             [, OpenTP1-RM-name-to-be-deleted] ... ]
           [-a non-OpenTP1-RM-name-to-be-added
             [, non-OpenTP1-RM-name-to-be-added] ... ]
           -s RM-switch-name [, RM-switch-name] ...
           -o 'RM-related-object-name [RM-related-object-name] ... '
             [, 'RM-related-object-name
               [RM-related-object-name] ... ' ] ... ]
           [-d non-OpenTP1-RM-name-to-be-deleted
             [, non-OpenTP1-RM-name-to-be-deleted] ... ]
           | [-n] }
           [-C 'compile-option-name [compile-option-name] ... ' ]
           [-B 'linkage-option-name [linkage-option-name] ... ' ] [-l] [-f] [-P]
```

Function

The `trnlInkrm` command adds or deletes resource managers used under OpenTP1, and recreates a transaction service control execution-form program and a client service execution-form program and a standard transaction control object file (`dc_trn_allrm.o`).

When an OpenTP1 program product is installed, XA interface object files provided by resource managers have not been linked with the transaction service control execution-form program and a client service execution-form program. When executed, the `dcsetup` command determines the installed OpenTP1 program product and automatically links XA interface object files for the OpenTP1 resource managers (DAM, TAM, MCF, ISAM, and MQA) with OpenTP1 resource manager. In order for a transaction executing under OpenTP1 to access non-OpenTP1 resource managers, the `trnlInkrm` command must be executed to catalog the non-OpenTP1 resource managers after execution of the `dcsetup` command and before startup of OpenTP1, thereby recreating a transaction service control execution-form program and a client service execution-form program. For using non-Hitachi databases, see the manual *OpenTP1 Programming User's Guide*.

Up to 32 resource managers can be cataloged under OpenTP1.

XA interface object files for all resource managers to be accessed by UAPs operating under OpenTP1 must be linked with the transaction service control execution-form program and a client service execution-form program. Otherwise, an error (`dc_rpc_open_error`) occurs at startup of a UAP that is accessing a resource manager not linked with OpenTP1.

The `trnlncrm` command cannot be executed when OpenTP1 is in the online mode.

If OpenTP1 is on restart status, the `-f` option must be specified in the `trnlncrm` command. Once the `trnlncrm` command with its `-f` option specified has been normally terminated, OpenTP1 is no longer be restarted.

If the resource manager registration status is changed with the `trnlncrm` command, the UAP to which a standard transaction control object file has been linked requires linkage again.

The `trnlncrm` command searches for the target C compiler in the following sequence:

1. `/bin/cc` and `/lib/ccom`

Both `/bin/cc` and `/lib/ccom` are required.

2. `/usr/bin/cc`

3. `/usr/vac/bin/cc`

If the target C compiler is not found in the above search sequence, OpenTP1 assumes the setting specified in the `PATH` environment variable of the process that executes the `trnlncrm` command. The `-P` option must be specified when the user wants to use the value specified in the environment variable (`PATH`).

Options

- `-A OpenTP1-RM-name-to-be-added~<1-31 alphanumeric characters>`

Specifies the name of a resource manager provided by OpenTP1 that is to be added. The OpenTP1 resource managers are:

OpenTP1_DAM

TP1/FS/Direct Access resource manager

OpenTP1_TAM

TP1/FS/Table Access resource manager

OpenTP1_MCF

TP1/Message Control resource manager

OpenTP1_MQA

TP1/Message Queue resource manager

OpenTP1_ISM

ISAM/B resource manager

The RM switch name and RM-related object name need not be specified for resource managers specified by the `-A` option. More than one RM name can be specified, with

a comma (,) placed between names.

- *-D OpenTP1-RM-name-to-be-deleted*~<1-31 alphanumeric characters>

Specifies the name of a resource manager provided by OpenTP1 that is to be deleted. The OpenTP1 resource managers are:

OpenTP1_DAM

TP1/FS/Direct Access resource manager

OpenTP1_TAM

TP1/FS/Table Access resource manager

OpenTP1_MCF_A

TP1/Message Control resource manager

OpenTP1_MQA

TP1/Message Queue resource manager

OpenTP1_ISM

ISAM/B resource manager

The RM switch name and RM-related object name need not be specified for resource managers specified by the *-D* option. More than one RM name can be specified, with a comma (,) placed between names.

- *-a non-OpenTP1-RM-name-to-be-added*~<1-31 alphanumeric characters>

Specifies the name of an additional resource manager not provided by OpenTP1. Do not specify a resource manager provided by OpenTP1.

The RM switch name and RM-related object name must be specified for resource managers specified by the *-a* option. More than one RM name can be specified, with a comma (,) placed between names.

- *-s RM-switch-name*~<1-32 alphanumeric characters beginning with an alphabetic character or an underscore>

Specifies the name of a resource manager switch to be added. For the switch names, see the applicable resource manager specifications. More than one RM switch name can be specified, with a comma (,) placed between names. The RM switch names correspond to the RM names in the order they are specified.

- *-o RM-related-object-name*~<alphanumeric characters>

Specifies the name of an object file (for the XA interface) related to the non-OpenTP1 resource manager to be added.

More than one RM-related object name can be specified for one resource manager. For the RM-related object names, see the applicable resource manager specifications.

If commas (,) need be specified for RM-related object files, '\' must be placed before the comma. More than one RM-related object file name can be specified, with a space placed between names.

When RM-related object file names are specified for multiple resource managers, the set of the RM-related object names for one resource manager must be enclosed in single quotation marks (') and a comma (,) placed between sets.

The RM-related object names correspond to the RM names in the order they are specified.

- `-d non-OpenTP1-RM-name-to-be-deleted`~<1-31 alphanumeric characters>

Specifies the name of a non-OpenTP1 resource manager to be deleted. In no case must OpenTP1 resource managers be specified in this option.

The RM switch name and RM-related object name need not be specified for any resource manager specified in this option. More than one RM name can be specified, with a comma (,) placed between names.
- `-n`

Recreates a transaction service control execution-form program and client service execution-form program without changing the resource managers cataloged into OpenTP1.
- `-C compile-option-name`~<string of 1-512 characters>

Specifies the compile option to be used when compilation is executed.

Enclose the compile option name in single quotation marks ('). If you need to include a comma (,) in the compile option name, insert a backslash (\) before the comma.

When you specify multiple compile option names, use spaces to delimit them.

Under normal circumstances, there is no need to specify this option.
- `-B linkage-option-name`~<string of 1-512 characters>

Specifies the linkage option to be used when library linkage is performed.

Enclose the linkage option name in single quotation marks ('). If you need to include a comma (,) in a linkage option name, insert a backslash (\) before the comma.

When you specify multiple linkage option names, use spaces to delimit them. Because the `trnlkrm` command uses the `cc` command to perform linkage, the linkage options you specify in the `trnlkrm` command must be options that can be specified in the `cc` command.

Under normal circumstances, there is no need to specify this option.
- `-l`

Outputs the progress of executing the `trnlkrm` command to the standard output.

■ -f

Forcibly executes the `trnlnkrm` command independently of the OpenTP1 status. However, the command cannot be executed while OpenTP1 is online mode because of recreating an OpenTP1 transaction service execution-form program and a client service execution-form program.

The `trnlnkrm` command can be specified only when OpenTP1 is to be normally started, with the resource managers changed, after non-normal termination (that is, planned termination A, planned termination B, forced termination, or abnormal termination).

Once the command with this option specified has normally terminated, OpenTP1 can no longer be restarted.

■ -P

A C compiler is selected according to the value specified in the `PATH` environment variable for the `trnlnkrm` command execution process. The `trnlnkrm` command looks for "cc" in the order specified in `PATH` and use the first "cc" encountered.

When none of the options is specified, the method for specifying the `trnlnkrm` command is output to the standard output.

Notes

- Once the `trnlnkrm` command has normally terminated, OpenTP1 can no longer be restarted; normal startup is only the method to start OpenTP1.
- When options are specified over more than one line, place the continuation mark '\ ' between options and perform a line feed. If a line feed is performed within the specified value of an option (for example, between RM-related object names specified in the `-o` option), the command may operate incorrectly.

trnls (Display status of transactions)

Format

```
trnls {-t [-{a|c}]|-T transaction-global-identifier [-{a|c}]
      |-bc|-B system-node-ID [-{a|c}]|-rc
      |-R RM-name + RM-extension [-{a|c}]}
```

Function

The `trnls` command displays information about transactions managed by the transaction manager or displays the number of transaction branches.

The following lists the display information for transactions:

1. Transaction global identifier
An identifier combining a system node ID and a global transaction number (a unique number assigned by the system to manage a global transaction)
2. Transaction branch identifier
An identifier combining a system node ID and a transaction branch number (a unique number assigned by the system to manage a transaction branch)
3. Transaction's first status
Transaction branch processing status
4. Transaction's second status
Status related to the transaction branch process
5. Transaction's third status
Transaction branch communication status
6. Process ID
ID of the process in which the transaction branch is operating
7. Server name
Name of the server activating the transaction branch
8. Service name
Name of the service activating the transaction branch
9. Transaction descriptor
An index number used to identify each transaction branch with the same transaction global identifier

10. Branch descriptor

An index number used to identify a transaction branch that branches out from another transaction branch

11. Parent transaction descriptor

The transaction descriptor of a transaction that has generated the corresponding transaction branch

The following lists the display information for the number of transaction branches:

1. Transaction global identifier

An identifier combining a system node ID and a global transaction number (a unique number assigned by the system to manage a global transaction)

2. System node ID

OpenTP1 system node ID

3. Resource manager name

Name of the resource manager in use under the transaction branch

4. Number of transaction branches

Number of transaction branches under processing

Options

■ -t

Specifies that information about all transactions is to be displayed.

■ -T *transaction-global-identifier*~<16 alphanumeric characters>

Specifies the transaction global identifier whose information is to be displayed.

The transaction global identifier can be obtained by the `trnls -t` command.

■ -b

Specifies that information about branched transactions is to be displayed.

■ -B *system-node-ID*~<8 alphanumeric characters>

Specifies the ID of a system when information about transactions that have branched out to the computer with the specified system node ID is to be displayed.

■ -r

Specifies that information about transactions connected to resource managers is to be displayed.

- `-R RM-name + RM-extension~<1-33 alphanumeric characters>`

Displays the information about a transaction connected to the resource manager with the specified name.

The resource manager names provided by OpenTP1 are:

OpenTP1_DAM

TP1/FS/Direct Access resource manager name

OpenTP1_TAM

TP1/FS/Table Access resource manager name

OpenTP1_MCF_A

TP1/Message Control resource manager name

To display the information about a transaction connected to a certain MCF, the MCF RM name + MCF RM extension (the manager process identifier for the MCF to be displayed) must be specified.

OpenTP1_MQA

TP1/Message Queue resource manager name

OpenTP1_ISM

ISAM/B resource manager name

The "RM-extension" must be appended when the `-i` option is specified in the definition command `trnstring` of the transaction service definition.

- `-a`

Specifies that all the transaction information is to be displayed.

- `-c`

Specifies that the number of transaction branches is to be displayed.

If the `-a` and `-c` options are omitted, the information in (1) through (8) above is displayed.

Output format

- (1) *"trnls -ta" specified*

TRNGID	TRNBID	Status	PID	...1.
aaaaaaaaabbbbbbb	aaaaaaaaaccccccc	dd...dd (e, f)	gg...gg	...2.
Server	Service	ENTRYID	BRANCHID	PENTRYID
hh..hh	ii...ii	jjjjjjjj	kkkkkkkkkk	llllllll
				...2.

trnls (Display status of transactions)

The 1. lines and the 2. lines are each displayed on a single line.

2. This information is displayed as many times as there are transactions.

aaaaaaaa

OpenTP1 system node ID (8 characters)

bbbbbbbb

Global transaction number (8 digits)

cccccccc

Transaction branch number (8 digits)

dd...dd

Transaction's first status (up to 20 characters):

BEGINNING

Transaction branch start processing is underway

ACTIVE

Executing

SUSPENDED

Suspended

IDLE

Changing to synchronous point processing

PREPARE

Under commit (phase 1) processing

READY

Waiting for commit (phase 2) processing

HEURISTIC_COMMIT

Heuristic decision commit processing is underway

HEURISTIC_ROLLBACK

Heuristic decision rollback processing is underway

COMMIT

Commit processing is underway

ROLLBACK_ACTIVE

Waiting for rollback processing

ROLLBACK

Rollback processing is underway

HEURISTIC_FORGETTING

Transaction branch termination processing after heuristic decision is underway

FORGETTING

Transaction branch termination processing is underway

e

Transaction's second status (1 character):

u

User server executing in a user server process

r

Transaction branch recovery processing in a transaction recovery process

p

Waiting for completion of other transaction branch recovery processing in a transaction recovery process

If the first status is READY and the root transaction branch is not on the same computer, the transaction will be placed on user instruction wait status. If you are using the XA resource service, the transaction will be placed in wait state for instructions from the XA resource service.

f

Transaction's third status (1 character):

s

Under sending

r

Under receiving

n

Not under sending/receiving

gg...gg

Process ID (decimal)

hh...hh

trnls (Display status of transactions)

Server name (up to 8 characters)

ii...ii

Service name (up to 32 characters) (for SUP, this item is blank)

jjjjjjjjj

Transaction descriptor (decimal)

kkkkkkkkkk

Branch descriptor (decimal) (for the root transaction branch, "*****" is displayed)

lllllllll

Parent transaction descriptor (decimal) (for the root transaction branch, "*****" is displayed)

Note

The term "under sending" or "under receiving" refers to synchronization between transaction branches.

- (2) *"trnls -tc" or "trnls -T transaction-global-identifier -c" specified*

TRNGID	Number of transaction branches	...
<i>aaaaaaaaabbbbbbb</i>	<i>cccc</i>	<i>...1.</i>

1. This information is displayed as many times as there are global transactions.

aaaaaaaa

OpenTP1 system node ID (8 characters)

bbbbbbbb

Global transaction number (8 digits)

cccc

Number of transaction branches in the global transaction (decimal)

- (3) *"trnls -bc" or "trnls -B system-node-ID -c" specified*

System node ID	Number of transaction branches	...
<i>aaaaaaaa</i>	<i>bbbb</i>	<i>...1.</i>

1. This information is displayed as many times as there are computers executing transactions.

aaaaaaaa

OpenTP1 system node ID (8 characters)

bbbbbb

Number of transaction branches currently being executed on the corresponding computer (decimal)

- (4) *"trnls -rc" or "trnls -R RM-name+RM-extension-c" specified*

Resource manager	Number of transaction branches	...
<i>aa...aa</i>	<i>bbbbbb</i>	<i>...1.</i>

1. This information is displayed as many times as there are resource managers executing transactions.

aa...aa

Resource manager name + resource manager extension (up to 33 characters)

bbbbbb

Number of transaction branches currently being executed on the corresponding resource manager (decimal)

- (5) *-a and -c options both omitted*

TRNGID	TRNBID	Status	PID	Server	Service	...
<i>aaaaaaaaabbbbbbb</i>	<i>aaaaaaaaaccccccc</i>	<i>dd...dd (e,f)</i>	<i>gg...gg</i>	<i>hh...hh</i>	<i>ii...ii</i>	<i>...1.</i>

1. This information is displayed as many times as there are transactions.

aaaaaaaa

OpenTP1 system node ID (8 characters)

bbbbbbbbb

Global transaction number (8 digits)

ccccccc

Transaction branch number (8 digits)

dd...dd

Transaction's first status (up to 20 characters) (for details, see (1) above)

e

Transaction's second status (1 character) (for details, see (1) above)

trnls (Display status of transactions)

f

Transaction's third status (1 character) (for details, see (1) above)

gg...gg

Process ID (decimal)

hh...hh

Server name (up to 8 characters)

ii...ii

Service name (up to 32 characters) (for SUP, this item is blank)

Output messages

Message ID	Description	Output destination
KFCA00970-E	Processing error occurred during <code>trnls</code> command execution.	Standard error output
KFCA00975-I	Help message	Standard output
KFCA00978-E	Usage of the <code>trnls</code> command is invalid.	Standard error output
KFCA00979-E	<code>trnls</code> command execution environment has not been set up.	Standard error output

Notes

As many transactions started from a CUP (Client User Program) as the maximum number of processes specified in `parallel_count` of the client service definition can be executed at the same time. The status of the transactions executing on these processes can be checked by the `trnls` command.

Executing this command displays a server name as `_clttrn`.

trnlsrm (Display information about resource manager)

Format

```
trnlsrm [-o file-name [, file name]...] [-s]
```

Function

The `trnlsrm` command displays the information about resource managers cataloged in OpenTP1, UAPs or transaction control objects.

Options

- `-o file-name~<full path name>`

Outputs the information about the resource manager cataloged in the specified file to the standard output. Only files storing UAP execution-form programs or transaction control object files can be specified. More than one file name can be specified, with a comma (,) placed between names.

- `-s`

Outputs the information about resource managers cataloged in OpenTP1 to the standard output.

When none of the options is specified, the `-s` option is assumed.

Output format

File name	Resource manager name	Attribute	Switch name	Object name	
<i>aa...aa</i>	<i>bb...bb</i>	<i>cd</i>	<i>ee...ee</i>	<i>ff...ff</i>	<i>...1.</i>
:	:	:	:	<i>ff...ff</i>	<i>...2.</i>
:	:	:	:	:	:

The information on lines 1. and 2. is displayed as many times as the number of the resource managers cataloged. For display of information about the same file, its name appears on the leading line alone.

2. When there is more than one RM-related object, information is displayed as many times as the number of the objects.

aa...aa

system or the specified file name (excluding the path name)

bb...bb

trnlsrm (Display information about resource manager)

Resource manager name (1-31 characters)

c

The first attribute of a resource manager

B: OpenTP1 resource manager

N: Non-OpenTP1 resource manager

d

The second attribute of a resource manager

O: One-phase commit control enabled

N: One-phase commit control disabled

ee...ee

RM switch name (1-32 characters)

***** is displayed for OpenTP1 resource managers.

ff...ff

RM-related object name

***** is displayed for OpenTP1 resource managers.

trnmkobj (Create a transaction control object file)

Format

```
trnmkobj -o transaction-control-object-name
          [-R OpenTP1-RM-name [, OpenTP1-RM-name] ...]
          [-r non-OpenTP1-RM-name [, non-OpenTP1-RM-name] ...]
          [-C 'compile-option-name [  $\Delta$ compile-option-name] ... ']
          [-l] [-P]
```

Function

The `trnmkobj` command creates a transaction control object file whose name is *transaction-control-object-name.o* under `$DCDIR/spool/trnrncmd/userobj` according to the resource manager accessed by a UAP. The user must link the created transaction control object file and the object file provided by the resource manager (XA interface object file) with the UAP. By doing so the transaction that accesses the resource manager under OpenTP1 can be executed.

When the `cc` command is executed to link a transaction control object file with a UAP, the "transaction control object name .o" must be specified before the `-L` option.

For a UAP executing a transaction that does not access resource managers under OpenTP1 or a UAP that executes no transaction under OpenTP1, the `trnmkobj` command need not be executed.

Linking the same resource manager with all UAPs that configure a global transaction optimizes commit processing (suppresses process-to-process communication), thus enhancing transaction performance.

The `trnmkobj` command need not be executed when using the standard transaction control object file (`$DCDIR/spool/trnrncmd/userobj/dc_trn_allrm.o`) consisting of all resource managers registered to OpenTP1.

When neither the `-R` option nor the `-r` option is specified, the same resource manager as cataloged in OpenTP1 is used to create a transaction control object file.

If the resource manager registration status is changed using the `trnlncrm` command, the UAP to which either of the following two files has been linked requires linkage again:

- Standard transaction control object file
- Transaction control object file created with the `trnmkobj` command with neither the `-R` nor `-r` option omitted

A transaction control object file can be used between UAPs that use the same resource manager. This means no need of executing the `trnmkobj` command for each UAP.

Only resource managers that are cataloged in OpenTP1 can be specified.

The `trnmkobj` command searches for the target C compiler in the following sequence:

1. `/bin/cc` and `/lib/ccom`

Both `/bin/cc` and `/lib/ccom` are required.

2. `/usr/bin/cc`

3. `/usr/vac/bin/cc`

If the target C compiler is not found in the above search sequence, OpenTP1 assumes the setting specified in the `PATH` environment variable of the process that executes the `trnmkobj` command. The `-P` option must be specified when the user wants to use the value specified in the environment variable (`PATH`).

Options

- `-o transaction-control-object-name~<1-12 alphanumeric characters>`

Specifies the name of a transaction control object file.

- `-R OpenTP1-RM-name~<1-31 alphanumeric characters>`

Specifies the name of an OpenTP1 resource manager to be accessed by UAPs. No non-OpenTP1 resource managers can be specified. The resource manager names provided by OpenTP1 are:

OpenTP1_DAM

TP1/FS/Direct Access resource manager name

OpenTP1_TAM

TP1/FS/Table Access resource manager name

OpenTP1_MCF_A

TP1/Message Control resource manager name

OpenTP1_MQA

TP1/Message Queue resource manager name

OpenTP1_ISM

ISAM/B resource manager name

More than one RM name can be specified, with a comma (,) placed between names.

- `-r non-OpenTP1-RM-name~<1-31 alphanumeric characters>`

Specifies the name of a non-OpenTP1 resource manager to be accessed by UAPs.

Only resource managers that are cataloged in OpenTP1 can be specified. More than

one RM name can be specified, with a comma (,) placed between names.

- `-C compile-option-name~<string of 1-512 characters>`

Specifies the compile option to be used when compilation is executed.

Enclose the compile option name in single quotation marks ('). If you need to include a comma (,) in a compile option name, insert a backslash (\) before the comma. When you specify multiple compile option names, use spaces to delimit them.

Under normal circumstances, there is no need to specify this option.

- `-l`

Outputs the progress of executing the `trnmkobj` command to the standard output.

- `-P`

A C compiler is selected according to the value specified in the `PATH` environment variable for the `trnmkobj` command execution process. The `trnmkobj` command looks for `cc` in the order specified in `PATH` and uses the first `cc` encountered.

When none of the options is specified, the method of specifying the `trnmkobj` command is output to the standard output.

trnrbk (Roll back transactions)

Format

```
Trnrbk { -t [-af] | -T transaction-global-identifier [-af] }
```

Function

When the information displayed by the `trnls` command indicates `READY(p,n)` status, the `trnrbk` command rolls back a transaction branch forcibly, notifies the other transaction branches, and then terminates the transactions. This command should be entered when a root transaction branch is to be rolled back.

The `trnrbk` command is entered when no transaction branch constituting a global transaction is able to determine the transaction for some reason (communication error, etc.). All the transaction branches in the global transaction must be rolled back in order to avoid inconsistency with other transactions. If a communication error has occurred, the transaction cannot be terminated until communication among its transaction branches has been completed. When the `-f` option is specified, the transaction is terminated forcibly. If the communication error is temporary, it is better to execute the `trnrbk` command without specifying the `-f` option.

Options

- `-t`

Specifies that all transactions on `READY(p,n)` status that are managed by the transaction manager on the corresponding computer are to be rolled back. Information about the transactions to be rolled back will be output to the standard output.

- `-T transaction-global-identifier`~<16 alphanumeric characters>

Specifies the transaction global identifier that is to be rolled back, but only if it is on `READY(p,n)` status. Information about the transaction to be rolled back will be output to the standard output.

The global transaction identifier can be obtained by the `trnls -t` command.

- `-a`

Specifies that information about the transactions is to be output to the standard output.

When this option is omitted, only the OpenTP1 system node ID through the service name will be output to the standard output.

- `-f`

Specifies that the transactions are to be terminated forcibly.

When this option is omitted, the transactions are not terminated forcibly.

Output format

When the `-a` option is specified, the output format is the same as output format (1) for the `trnls` command; otherwise, it is the same as output format (5) for the `trnls` command.

Output messages

Message ID	Description	Output destination
KFCA00970-E	Processing error occurred during <code>trnrbk</code> command execution.	Standard error output
KFCA00977-I	Help message	Standard output
KFCA00978-E	Usage of <code>trnrbk</code> command is invalid.	Standard error output
KFCA00979-E	<code>trnrbk</code> command execution environment has not been set up.	Standard error output

trnstics (Start and terminate collecting transaction statistics)

Format

```
Trnstics {-s|-e}
```

Function

The `trnstics` command starts or terminates collecting transaction statistics onto a journal file.

When the `-s` option is specified, statistics is collected from a transaction that is to be started after the `trnstics` command is normally terminated. No statistics can be collected from a transaction that has started before normal termination of the `trnstics` command.

Only the transaction from which statistics can be collected is a transaction that has been executed by a user service for which 1-4 is specified in `trn_statistics_type` of the user service definition.

When the `-e` option is specified, no statistics is collected from a transaction that is to start after normal termination of the `trnstics` command.

At restart of OpenTP1, the specification of the `trnstics` command cannot be inherited. In this case, processing follows the specification of `trn_tran_statistics` (specify whether statistics for each transaction branch is to be collected) in the transaction service definition before restart of OpenTP1.

Options

- `-s`

Starts collecting transaction statistics onto a journal file.

- `-e`

Terminates collecting transaction statistics onto a journal file.

Output messages

Message ID	Description	Output destination
KFCA00970-E	Error occurred in <code>trnstics</code> command processing.	Standard error output
KFCA00973-I	Help message	Standard output
KFCA00978-E	Use of <code>trnstics</code> command is invalid.	Standard error output
KFCA01099-E	<code>trnstics</code> command execution environment has not been set.	Standard error output

Notes

Lower transaction performance will result with more different types of transaction statistics to be collected.

usmdump (Output memory dump)

Format

```
usmdump [-a] [-i user-identifier] [file-name]
```

Function

The `usmdump` command dumps the contents of shared memory during OpenTP1 operation to a UNIX file in the dump directory (`$DCDIR/spool/save`). The dump file is then compressed by the `compress` command, or by the `gzip` command if you are using Linux.

Options

■ `-a`

Dumps all of the information in the shared memory allocated to OpenTP1. If you omit the `-a` option, the command dumps information from only those areas in shared memory that are actually in use.

■ `-i user-identifier`

Specifies the user identifier for the shared memory to be dumped. The dumped information depends on the user identifier you specify. Specify one of the following user identifiers:

- `all`: Dumps all of the information in shared memory that is being managed by OpenTP1.
- `BTN`: Dumps the information in dynamic shared memory and static shared memory.
- Information set in `OWNER` at execution of the `dcshmls -r` command: Dumps the information in the shared memory pool for the service specified in `OWNER`. For example, specify `dam` to dump the information in the shared memory pool allocated to the DAM service.

When the `dcshmls -r` command is executed, `btn` is shown as the `OWNER` of the information in dynamic shared memory and static shared memory. To output this information, either omit the `-i` option, or specify `BTN` as the argument for the `-i` option.

If you omit this option, OpenTP1 operation is the same as if `BTN` were specified.

Command arguments

- `file-name~<character string, 1-31 bytes> <<shmdump>>`

Specifies the output destination file to use when outputting the information in shared memory. The information in shared memory is dumped with the specified file name to `$DCDIR/spool/save`. If you specify the file name in the form `$DCDIR/spool/save/file-name`, make sure that the full path does not exceed 255 bytes. If you omit this command argument, the command outputs the information in shared memory to the file `shmdump` in `$DCDIR/spool`.

Microseconds (decimal)

cc...cc

Application server name (character string, 1-16 characters)

dddd

Display number (decimal)

ee...ee

Process ID (decimal)

ff..ff

Transaction management table number (decimal)

gg...gg

Internal maintenance information (decimal, or character string 1-20 characters)

hh...hh

Name of request code (1-32 characters)

xar_start: Transaction branch start processing

xar_call: Remote procedure call from within a transaction branch

xar_end: Transaction branch termination processing

xar_prepare: Transaction branch preparation for commit processing (phase 1)

xar_commit: Transaction branch commit processing (phase 2)

xar_rollback: Transaction branch rollback processing

xar_recover: Report transaction branches with the status Prepared or Heuristically Completed.

xar_forget: Discard transaction branches with the status Heuristically Completed.

xar_tbl_reserve: Internal maintenance information (detect entry release when the table is full)

xar_trnrcvr: Internal maintenance information (detect RAP disconnection or failure of RAP-processing server)

ii

Position of information acquisition (2 characters)

IN: Function entry

OK: Function exit (normal termination)

ER: Function exit (abnormal termination)

jj...jj

XA return code character string (character string, 1-20 characters) or return code (decimal)

kk...kk

Application server XID information within first 64 bytes of GTRID (hexadecimal)

ll...ll

Application server XID information within first 64 bytes of BQUAL (hexadecimal)

mm...mm

Application server XID detailed information: Format ID (hexadecimal)

nn...nn

Application server XID detailed information: GTRID size (hexadecimal)

oo...oo

Application server XID detailed information: BQUAL size (hexadecimal)

pp...pp

Application server XID: GTRID detailed information (hexadecimal or characters)

qq...qq

Application server XID: BQUAL detailed information (hexadecimal or characters)

rr...rr

OpenTP1 XID information: TRNGID character string (1-16 characters)

ss...ss

OpenTP1 XID information: TRNBID character string (1-16 characters)

tt...tt

OpenTP1 XID detailed information (hexadecimal)

Output messages

Message ID	Description	Output destination
KFCA32029-E	Processing could not continue due to insufficient memory in the process-specific area.	Standard error output

Message ID	Description	Output destination
KFCA32100-I	Help message	Standard output
KFCA32101-E	XAR event trace information file was not found.	Standard error output
KFCA32102-E	An error occurred during file manipulation.	Standard error output
KFCA32103-E	<code>xarevtr</code> command usage is invalid.	Standard error output
KFCA32104-E	<code>xarevtr</code> command cannot be executed.	Standard error output
KFCA32105-E	An error occurred while the <code>xarevtr</code> command was executing.	Standard error output

xarfills (Display XAR file status)

Format

```
xarfills XAR-file-name
```

Function

The `xarfills` command outputs the status of an XAR file to standard output.

Command arguments

- *XAR-file-name*~<path name>

Specifies the name of the XAR file.

Output format

Record length[B]	: <i>aa...aa</i>
Number of records	: <i>bb...bb</i>
Number of transaction branches	: <i>cc...cc</i>
RI length that can be stored	: <i>dd...dd</i>
XAR file status	: <i>ee...ee</i>
File creation time	: <i>ff...ff</i>
Online start time	: <i>gg...gg</i>
Last close time	: <i>hh...hh</i>
XAR file name	: <i>ii...ii</i>

aa...aa

XAR file record length (bytes)

bb...bb

Number of records in the XAR file

cc...cc

Number of transaction branches

dd...dd

Maximum size of RI data that can be stored in the XAR file

ee...ee

XAR file status

RUNNING: Normal

HOLD: Shut down

INIT: Initializing

ff..ff

Time at which the XAR file was created (in the format: *day-of-week* Δ *month* Δ *day* Δ *hour:minute:second* Δ *year*)

gg...gg

Time at which the file was placed in online mode (in the format: *day-of-week* Δ *month* Δ *day* Δ *hour:minute:second* Δ *year*)

hh...hh

Time at which the XAR file was last closed (in the format: *day-of-week* Δ *month* Δ *day* Δ *hour:minute:second* Δ *year*)

ii...ii

XAR file name

Output messages

Message ID	Description	Output destination
KFCA32017-E	Processing could not continue due to insufficient memory in the process-specific area.	Standard error output
KFCA32025-E	The specified XAR file path is invalid.	Standard error output
KFCA32026-E	User does not have access rights to the special file.	Standard error output
KFCA32027-E	The attempt to open the XAR file exceeded the maximum number of open files allowed by the system.	Standard error output
KFCA32028-E	An I/O error occurred while accessing the XAR file.	Standard error output
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32030-E	The system version used for file system creation does not match the system version used to execute the command.	Standard error output
KFCA32031-E	The special file to which the XAR file is allocated is not initialized in the OpenTP1 file system.	Standard error output
KFCA32036-I	The specified XAR file does not exist.	Standard error output
KFCA32037-E	The specified XAR file is being used by another process.	Standard error output
KFCA32038-E	The user does not have access rights to the XAR file.	Standard error output
KFCA32103-E	Usage of the <code>xarfills</code> command is invalid.	Standard error output
KFCA32111-E	The specified file is not an XAR file.	Standard error output

xarfills (Display XAR file status)

Message ID	Description	Output destination
KFCA32112-I	Help message	Standard output

xarforce (Change the status of an XAR transaction)

Format

```
xarforce {-c|-r|-f}
          {-t OpenTP1-transaction-ID |
           -u client-transaction-ID |
           -n entry-number}
```

Function

The `xarforce` command changes the status of the transaction corresponding to a specific OpenTP1 transaction ID, client transaction ID, or entry number.

Options

- `-c`
Changes the status of a transaction managed by the XA resource service to Heuristic Commit. With this option, you can only change the status of a transaction in prepared state, as displayed by the `xarls` command.
- `-r`
Changes the status of a transaction managed by the XA resource service to Heuristic Rollback. With this option, you can only change the status of a transaction in prepared state, as displayed by the `xarls` command.
- `-f`
Invalidates a transaction managed by the XA resource service. With this option, you can only invalidate a transaction whose status as displayed by the `xarls` command is heuristic hazard, heuristic commit, heuristic rollback, or heuristic mix.
- `-t OpenTP1-transaction-ID<80 character hexadecimal>`
Specifies the ID of the OpenTP1 transaction whose status is to be changed. Specify the OpenTP1 transaction ID as displayed by the `xarls` command with the `-a` or `-p` option specified.
- `-u client-transaction-ID<280 character hexadecimal>`
Specifies the client transaction ID of the transaction whose status is to be changed. Specify the ID as displayed by the `xarls` command with the `-a` or `-p` option specified.
- `-n entry-number<decimal between 1 and 8192>`
Specifies the entry number of the transaction whose status is to be changed. Specify the entry number as displayed by the `xarls` command with the `-a` or `-p` option specified.

Output messages

Message ID	Description	Output destination
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarforce</code> command.	Standard error output
KFCA32123-E	Processing could not execute because shared memory was unavailable.	Standard error output
KFCA32124-E	V/R of the XA resource service is invalid.	Standard error output
KFCA32126-E	XA resource service is shut down.	Standard error output
KFCA32127-E	Timeout occurred during an RPC.	Standard error output
KFCA32128-E	Network error occurred during an RPC.	Standard error output
KFCA32130-I	Help message	Standard output
KFCA32133-E	There is no transaction with the specified transaction ID.	Standard error output
KFCA32134-E	There is no transaction with the specified entry number.	Standard error output
KFCA32135-E	The status of the transaction specified by the transaction ID could not be changed.	Standard error output
KFCA32136-E	The status of the transaction specified by the entry number could not be changed.	Standard error output
KFCA32137-I	The transaction corresponding to the transaction ID was committed.	Standard output
KFCA32138-I	The transaction corresponding to the transaction ID was rolled back.	Standard output
KFCA32139-I	The transaction corresponding to the transaction ID was invalidated.	Standard output
KFCA32140-I	The transaction corresponding to the entry number was committed.	Standard output
KFCA32141-I	The transaction corresponding to the entry number was rolled back.	Standard output
KFCA32142-I	The transaction corresponding to the entry number was invalidated.	Standard output
KFCA32143-E	The XA resource server is inactive.	Standard error output
KFCA32144-E	The OpenTP1 system is inactive.	Standard error output
KFCA32156-E	An error was detected during execution of the <code>xarforce</code> command.	Standard error output

Notes

- To execute the command using an entry number, stop access to the XA resource service before you execute the command.

The following table shows the relationships between the command options and transaction status.

Option	Transaction Status				
	A	I	P	R	H
-c	N	N	Y	N	N
-r	N	N	Y	N	N
-f	N	N	N	N	Y

Legend:

A: Active

I: Idle

P: Prepared

R: Rollback only

H: Heuristically completed

Y: Can be specified

N: Cannot be specified

xarhold (Shut down the XA resource service)

Format

xarhold

Function

The `xarhold` command shuts down the XA resource service.

Output messages

Message ID	Description	Output destination
KFCA32007-I	The XA resource service has been shut down.	Standard output
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarhold</code> command.	Standard error output
KFCA32121-I	Help message	Standard output
KFCA32124-E	V/R of the XA resource service is invalid.	Standard error output
KFCA32126-E	The XA resource service is already shut down.	Standard error output
KFCA32127-E	Timeout occurred during an RPC.	Standard error output
KFCA32128-E	Network error occurred during an RPC.	Standard error output
KFCA32143-E	The XA resource service is inactive.	Standard error output

Notes

- If an error occurs during the closure of an XAR file that is incorporated into the XA resource service, the XA resource service is shut down forcibly.

xarinit (Create an XAR file)

Format

```
xarinit -f physical-file-name -n number-of-records [-s record-length]
```

Function

The `xarinit` command creates an XAR file within the OpenTP1 file system, and initializes the file to enable use in online mode.

Options

- `-f physical-file-name~<path name>`

Specifies the full path name of the physical file to be created. If you specify the name of a file that already exists, an error results.

- `-n number-of-records~<unsigned integer> ((1-8192))`

Specifies the number of records of the XAR file to be created.

- `-s record-length~<unsigned integer> (bytes)`

Specifies the record length of the XAR file to be created. Specify a multiple of the sector length specified at creation of the OpenTP1 file system by the `filmkfs` command (an ordinary UNIX file typically has a record length of 512 bytes). If this option is omitted, the sector length of the OpenTP1 file system is used as the record length of the XAR file.

Output messages

Message ID	Description	Output destination
KFCA32017-E	Processing could not continue due to insufficient memory in the process-specific area.	Standard error output
KFCA32025-E	The specified XAR file path is invalid.	Standard error output
KFCA32026-E	The user does not have access rights to the special file.	Standard error output
KFCA32027-E	The attempt to open the XAR file exceeded the maximum number of open files allowed by the system.	Standard error output
KFCA32028-E	An I/O error occurred while accessing the XAR file.	Standard error output
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32030-E	The system version used for file system creation does not match the system version used to execute the command.	Standard error output

Message ID	Description	Output destination
KFCA32031-E	The special file to which the XAR file is allocated is not initialized in the OpenTP1 file system.	Standard error output
KFCA32032-E	The specified XAR file name is invalid.	Standard error output
KFCA32033-E	The specified XAR file already exists.	Standard error output
KFCA32034-E	The OpenTP1 file system has insufficient file capacity.	Standard error output
KFCA32035-E	The number of files allowed by the OpenTP1 file system has been exceeded.	Standard error output
KFCA32037-E	The specified XAR file is being used by another process.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarinit</code> command.	Standard error output
KFCA32106-I	Help message	Standard output
KFCA32107-E	Invalid number of records specified.	Standard error output
KFCA32108-E	The specified XAR file name is longer than 63 characters.	Standard error output
KFCA32162-E	Invalid record length specified.	Standard error output

Notes

- When using the MSDTC linkage facility (by specifying `xar_msdtc_use=Y` in the XA resource service definition), we recommend that you specify a value of 1024 bytes or larger for the `-s` option.
- When using the MSDTC linkage facility, the record length of the XAR file may be too short for some client environments, causing transaction determination processing to fail with the message KFCA32045-E. If this occurs, re-create the XAR file as follows:
 1. Shut down OpenTP1.
 2. Recreate the XAR files by executing the `xarinit` command and specifying for the `-s` option the value output as the required length of XAR files in the message KFCA32045-E.
 3. Start OpenTP1 in the normal manner.
- The structure of XAR files has been changed from version 07-01 of TP1/Server Base. For this reason, operation cannot be guaranteed if you use XAR files created by version 07-01 or later of TP1/Server Base with TP1/Server Base 07-00 or earlier.

xarls (Display XAR transaction information)

Format

```
xarls [-c] [-a|-p process-ID] [-r]
```

Function

The `xarls` command displays on the console information about either a specific transaction or about all transactions managed by the XA resource service.

Options

- `-c`

Displays the management information of the XA resource service.

- `-a`

Displays the management information of the XA resource service, and the status of all the transactions being managed by the XA resource service.

- `-p process-ID~<decimal>`

Specifies the process ID of the transaction to be displayed. The transactions associated with the specified process ID are displayed.

- `-r`

Displays the status of all the transactions being managed by the XA resource service, recovery information for the transactions, and the transaction information contained in the XID.

Output formats

- (1) When displaying transaction information

```
Entry number: AA...AA
Transaction start time: BB...BB
Process ID: CC...CC
Port number: XX...XX
Transaction status: DD...DD
Rollback-only mark: EE...EE
Rollback-only time: FF...FF
OpenTPl transaction ID: GG...GG
TRNGID: HH...HH TRNBID: II...II
Client transaction ID: JJ...JJ
DID: KK...KK NodeID: LL...LL
RI: MM...MM
```

AA...AA

Entry number (decimal)

BB...BB

Transaction start time (ctime format)

CC...CC

Process ID (decimal)

XX...XX

Port number of the RAP-processing listener (decimal)

DD...DD

Transaction status

A: Active

I: Idle

P: Prepared

HH: Heuristic Hazard

HC: Heuristic Commit

HR: Heuristic Rollback

HM: Heuristic Mix

***: Initial status

EE...EE

Rollback-only mark

INIT: Initial status

RBONLY1: Marked active transaction

RBONLY2: No transaction

RBONLY3: Unmarked active transaction

FF...FF

Time when rollback-only status occurred (ctime format)

GG...GG

OpenTP1 transaction ID (80 character hexadecimal)

HH...HH

Transaction global identifier (character string, 16 characters)

II...II

Transaction branch identifier (character string, 16 characters)

JJ...JJ

Client transaction ID (280 character hexadecimal)

KK...KK

DID information (32 character hexadecimal)

A DID is a transaction identifier managed by MSDTC. **** is displayed for transaction branches that have no DID information.

LL...LL

Node ID information (32 character hexadecimal)

**** is displayed for transaction branches that have no node ID information.

MM...MM

Recovery information used for MSDTC linkage (hexadecimal (length varies))

**** is displayed for transaction branches that have no recovery information.

■ (2) When displaying management information

```
Version: aa...aa
XAR block status: bb...bb
Number of transaction branches: cc...cc
Session time: dd...dd
Trace record number: ee...ee
Event level: ff...ff
Maximum number of records: gg...gg
Writable file number: hh...hh
RI length that can be stored: ii...ii
----- XAR file information -----
XAR file status: jj...jj
Physical file name: kk...kk
Number of records: ll...ll
Record length: mm...mm
File creation time: nn...nn
Online start time: oo...oo
Last close time: pp...pp
----- XAR file information -----
XAR file status: jj...jj
Physical file name: kk...kk
Number of records: ll...ll
Record length: mm...mm
File creation time: nn...nn
Online start time: oo...oo
Last close time: pp...pp
```

Online XAR file information

Backup XAR file information

aa...aa

XAR version (decimal)

bb...bb

XAR shutdown status (character string)

RUNNING: Normal

HOLD: Shut down

cc...cc

Number of transaction branches (decimal)

dd...dd

Length of time to wait for idle transaction branches (decimal)

ee...ee

Trace record number (decimal)

ff..ff

Event level (decimal)

gg...gg

Maximum number of records (decimal)

hh...hh

File number of writable file (decimal)

1: Online XAR file

2: Backup XAR file

3: No writable files

ii...ii

Maximum size of RI data that can be stored in the XAR file (decimal)

jj..jj

XAR file status (character string)

RUNNING: Normal

HOLD: Shut down

INIT: Initializing

kk...kk

Physical file name (character string)

ll...ll

Number of records in the XAR file (decimal)

mm...mm

Length of records in the XAR file (decimal)

nn...nn

Time of file creation (ctime format)

oo...oo

Time at which the XAR file entered online mode (ctime format)

xarls (Display XAR transaction information)

pp...pp

Time at which the XAR file was last closed (ctime format)

Output messages

Message ID	Description	Output destination
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarls</code> command.	Standard error output
KFCA32123-E	Processing could not execute because shared memory was unavailable.	Standard error output
KFCA32124-E	V/R of the XA resource service is invalid.	Standard error output
KFCA32129-I	Help message	Standard output
KFCA32131-E	No transactions were found.	Standard error output
KFCA32132-E	There is no transaction with the specified process ID.	Standard error output
KFCA32143-E	The XA resource service is inactive.	Standard error output

Notes

To minimize its impact on processing performance, the `xarls` command references information without locking. In this case, `***` is displayed as the status of transactions managed by the XA resource service that were in the process of being released at execution of the `xarls` command. Zeroes (0) are displayed for the decimal and hexadecimal format data of these transactions.

xarrles (Release XA resource service from shutdown status)

Format

xarrles

Function

The `xarrles` command incorporates the XAR files specified in the definition file into the XA resource service, and releases the XA resource service from shutdown status.

Output messages

Message ID	Description	Output destination
KFCA32008-I	Released XA resource service from shutdown status.	Standard output
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarrles</code> command.	Standard error output
KFCA32120-I	Help message	Standard output
KFCA32122-E	Failed to release the XA resource service from shutdown status.	Standard error output
KFCA32124-E	V/R of the XA resource service is invalid.	Standard error output
KFCA32125-E	The XA resource service is not shut down.	Standard error output
KFCA32127-E	Timeout occurred during an RPC.	Standard error output
KFCA32128-E	Network error occurred during an RPC.	Standard error output
KFCA32144-E	The OpenTP1 system is inactive.	Standard error output

Notes

- The following limitations apply to the XAR files you can specify in the definition file:
 - The file must have the same sector length as the XAR file that was being used before the XA resource service entered shutdown status.
 - The file must have the same number of records as the XAR file that was being used before the XA resource service entered shutdown status.

xarrm (Delete an XAR file)

Format

```
xarrm -f physical-file-name
```

Function

The `xarrm` command deletes an XAR file from the OpenTP1 file system. The specified file cannot be deleted if it is in use.

Options

- `-f physical-file-name~<path name>`

Specifies the full path name of the physical file to delete.

Output messages

Message ID	Description	Output destination
KFCA32017-E	Processing could not continue due to insufficient memory in the process-specific area.	Standard error output
KFCA32025-E	The specified XAR file path is invalid.	Standard error output
KFCA32026-E	The user does not have access rights to the special file.	Standard error output
KFCA32027-E	The attempt to open the XAR file exceeded the maximum number of open files allowed by the system.	Standard error output
KFCA32028-E	An I/O error occurred while accessing the XAR file.	Standard error output
KFCA32029-E	Processing could not continue due to insufficient memory.	Standard error output
KFCA32030-E	The system version used for file system creation does not match the system version used to execute the command.	Standard error output
KFCA32031-E	The special file to which the XAR file is allocated is not initialized in the OpenTP1 file system.	Standard error output
KFCA32032-E	The specified XAR file name is invalid.	Standard error output
KFCA32036-E	The specified XAR file does not exist.	Standard error output
KFCA32037-E	The specified XAR file is being used by another process.	Standard error output
KFCA32038-E	The user does not have access rights to the XAR file.	Standard error output
KFCA32103-E	Invalid usage of the <code>xarrm</code> command.	Standard error output
KFCA32108-E	The specified XAR file name is longer than 63 characters.	Standard error output

Message ID	Description	Output destination
KFCA32109-I	Help message	Standard output
KFCA32111-E	The specified file is not an XAR file.	Standard error output

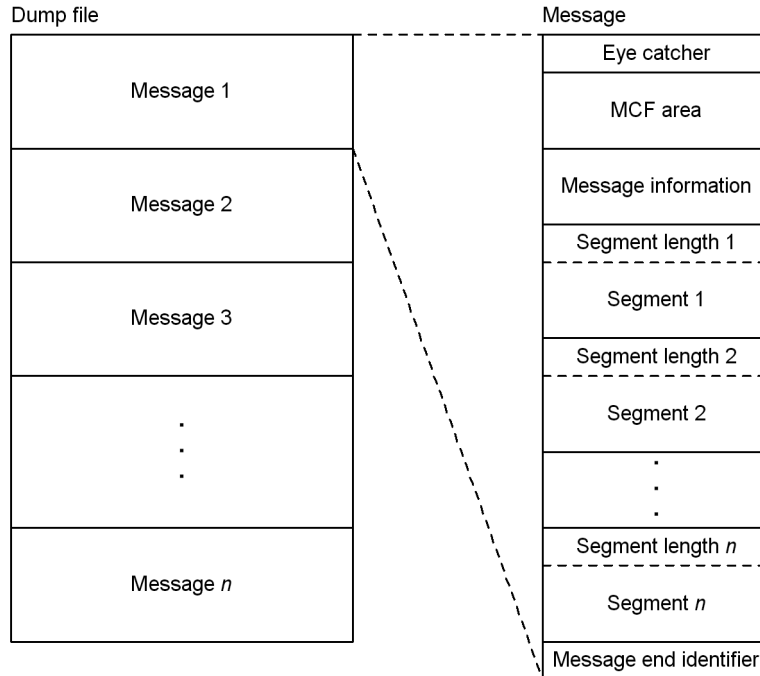
Appendixes

- A. Dump File Format of Input/Output Queues
- B. OpenTP1 Events
- C. Information Output for Audited Events
- D. Journal Information Collected by the Message Control Facility
- E. Details of Statistics
- F. List of Files Output by OpenTP1
- G. Estimating the Size of the Physical File for a Message Queue
- H. Estimating the Size of OpenTP1 Files
- I. Estimating the Number of Record Locks
- J. Estimating the Resources Used by UNIX Message Exchange Functions
- K. Estimating the Size of an OpenTP1 File System
- L. Acquiring Performance Verification Traces
- M. Details About Scenario Templates

A. Dump File Format of Input/Output Queues

Executing the `mcftdmpqu` command causes messages to be successively output onto a dump file. The following figure shows the format of a dump file.

Figure A-1: Dump file format



The dump file format shown in Figure A-1 is explained:

- Eye catcher (4 bytes)
Indicates which is the output message, a message for input queues or for output queues.
ITQ Δ : Message for input queues
OTQ Δ : Message for output queues
- MCF area (960 bytes)
Indicates the area used by MCF.
- Message information (512 bytes)
Indicates the area that stores information about output messages. It can be

referenced by the structure `dc_mcf_dump_info`. Figure A-2 shows the format of the structure.

- Segment length (4 bytes), segment
Indicates the area giving the length of a segment, and the segment having the indicated segment length. A segment length and a segment makes up one set, which is repeated as many time as the number of segment used.
- Message end identifier (4 bytes)
Indicates the area showing the end of a message, which stores NULL.

Figure A-2: Format of structure dc_mcf_dump_info

```

/*****
/*                               Input/Output Queue Dump Message Information */
/*****
struct dc_mcf_dump_info {
  char  le_name[16];      /* Input source logical terminal name#1      */
  char  cn_name[16];      /* Input source connection ID                */
  char  mcf_sid[3];       /* MCF identifier                            */
  char  ap_name[9];       /* Application name#2                        */
  char  ap_kind[5];       /* Application type#2                        */
                               /* value<"mcf " =system application          */
                               /*      "user"=user application>            */
  char  sg_name[32];      /* Service group name                        */
  char  sv_name[32];      /* Service name                              */
  char  map_name[9];      /* Map name                                  */
  char  yobil[2];        /* Reserved 1                                */
  struct {                /* Message write time                        */
    unsigned int tvsec;    /* Total seconds from January 1, 1970        */
    int tvsec;            /* Less than one second                      */
  } write_time;
  struct {                /* Receive serial number structure           */
    unsigned int time;    /* Receive serial number collection time      */
                               /* (Total seconds from January 1, 1970)      */
    int no_in_time;      /* Sub-number at the same time (initial value=1) */
    char yobil[8];       /* Reserved 1                                */
  } rcv_no;
  struct {                /* Send serial number structure#1           */
    int no;              /* Send serial number                        */
    char type[2];        /* Send serial number identification          */
                               /* value<"n"=general                         */
                               /*      "p"=Priority                          */
                               /*      "o"=reply                            */
    char yobil[2];       /* Reserved 1                                */
  } send_no;
  char  exec_ap_name[9];  /* Startup destination application name#1,3  */
  char  exec_ap_kind[5]; /* Startup destination application type#1,3  */
                               /* value<"mcf " =system application          */
                               /*      "user"=user application>            */
  char  send_le_name[16]; /* Output destination logical terminal name  */
  char  yobi2[326];      /* Reserved 2                                */
};

```

#1

Set for output queue dump only.

#2

For a message sent from an SPP, ***** set in the application name and **** in the application type.

#3

If no application is started, ***** is appended to the startup destination application name and *** to the startup destination application type.

B. OpenTP1 Events

OpenTP1 allows the user to catalog events such as startup and termination in the JP1 Event Service. You can use the cataloged events with the JP1 job management services. For information about the JP1 event service facility, see the manual *Job Management Partner 1/Base User's Guide*. For information about the JP1 job management services, see the manual *Job Management Partner 1/Automatic Job Management System 2 Description* and the manual *Job Management Partner 1/Automatic Job Management System 2 Planning and Administration Guide*.

B.1 Cataloging events

To catalog OpenTP1 events in the JP1 event service facility, specify `jp1_use=Y` in the system common definition.

B.2 OpenTP1 events that can be cataloged

The following table lists the OpenTP1 events that can be cataloged in the JP1 event service facility.

Table B-1: OpenTP1 events that can be cataloged in the JP1 event service facility

Event	Event ID (basic code only)	Message	Detailed information [#]	Cataloging timing
Startup	0x00010000	NULL	start	After startup of user server scheduling
Normal termination	0x00010001	NULL	end_normal"	After system service end
Planned termination A	0x00010001	NULL	end_planA	After system service end
Planned termination B	0x00010001	NULL	end_planB	After system service end
Forced termination	0x00010001	NULL	end_force	After system service end
Abnormal termination	0x00010002	NULL	end_error	Immediately after determination of abnormal termination, or before startup of termination process
Message log output	0x00010003	OpenTP1 message	NULL	Immediately after message log output

#

Detailed information is given in text format, as follows:

ss Δ *nnnn* Δ *mm...mm* Δ \0

ss

System identifier (0-2 bytes)

nnnn

Node identifier (4 bytes)

mm...mm

Character string of above detailed information lines (5-10 bytes)

C. Information Output for Audited Events

The following table lists the information output for audited events.

Table C-1: Information output for audited events

Audited event	Information output for event	
	Item name	Contents
OpenTP1 startup	Message ID (<i>msgid</i>)	KFCA33400-I
	Component (<i>compid</i>)	adm
	Event type (<i>ctgry</i>)	StartStop
	Event result (<i>result</i>)	Success
	Subject ID information (<i>subj</i> : (<i>euid</i> , <i>pid</i>))	User name or ID of the user who executed the <code>dcstart</code> command.
	Object (<i>obj</i>)	Node identifier
	Action information (<i>op</i>)	Start
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$290 + 2 * a$
OpenTP1 standby	Message ID (<i>msgid</i>)	KFCA33401-I
	Component (<i>compid</i>)	adm
	Event type (<i>ctgry</i>)	StartStop
	Event result (<i>result</i>)	Success
	Subject ID information (<i>subj</i> : (<i>euid</i> , <i>pid</i>))	User name or ID of the superuser.
	Object (<i>obj</i>)	Node identifier
	Action information (<i>op</i>)	Start

Audited event	Information output for event	
	Item name	Contents
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$326 + 2*a$
Normal termination of OpenTP1	Message ID (<i>msgid</i>)	KFCA33402-I
	Component (<i>compid</i>)	adm
	Event type (<i>ctgry</i>)	StartStop
	Event result (<i>result</i>)	Success
	Subject ID information (<i>subj: (euid, pid)</i>)	User name or ID of the user who executed the <i>dcstop</i> command
	Object (<i>obj</i>)	Node identifier
	Action information (<i>op</i>)	Stop
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$289 + 2*a$
	Abnormal termination of OpenTP1	Message ID (<i>msgid</i>)
Component (<i>compid</i>)		adm

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Event type (<i>ctgry</i>)	Failure
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj</i> : (<i>eu</i> id, <i>pi</i> d))	The PID associated with the process that led to the OpenTP1 system going down (0 is output if the process is a process service.)
	Object (<i>obj</i>)	Node identifier (**** is output if the node identifier cannot be acquired.)
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from</i> : <i>ip</i> v4)	--
	Request source port number (<i>from</i> : <i>port</i>)	--
	Request destination host (<i>to</i> : <i>ip</i> v4)	--
	Request destination port number (<i>to</i> : <i>port</i>)	--
	Log message size (Units: bytes)	321
Critical error in process service	Message ID (<i>msgid</i>)	KFCA33404-E
	Component (<i>compid</i>)	prc
	Event type (<i>ctgry</i>)	Failure
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj</i> : (<i>eu</i> id, <i>pi</i> d))	PID of process service
	Object (<i>obj</i>)	Node identifier (**** is output if the node identifier cannot be acquired.)
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	--
Request source host (<i>from</i> : <i>ip</i> v4)	--	

Audited event	Information output for event	
	Item name	Contents
	Request source port number (from:port)	--
	Request destination host (to:ipv4)	--
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	331
User server startup	Message ID (msgid)	KFCA33405-I
	Component (compid)	adm
	Event type (ctgry)	StartStop
	Event result (result)	Success
	Subject ID information (subj: (euid, pid))	User name or ID of the user who executed the dcsvstart command
	Object (obj)	User server name
	Action information (op)	Start
	Object location information (objloc)	Node identifier
	Request source host (from:ipv4)	--
	Request source port number (from:port)	--
	Request destination host (to:ipv4)	--
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	$285 + 2*(a + b)$
	Remarks	These items are also output at startup and termination of the following servers: RAP-processing server, RAP-processing listener, RAP-processing client manager, TP1/EE, RTSSPP, RTSSUP, MQC gateway server, mqrsp, mqrsp
Normal termination of user server	Message ID (msgid)	KFCA33406-I
	Component (compid)	adm

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Event type (<i>ctgry</i>)	StartStop
	Event result (<i>result</i>)	Success
	Subject ID information (<i>subj: (euid, pid)</i>)	User name or ID of the user who executed the <code>dcsvstop</code> command
	Object (<i>obj</i>)	User server name
	Action information (<i>op</i>)	Stop
	Object location information (<i>objloc</i>)	Node identifier
	Request source host (<i>from: ipv4</i>)	--
	Request source port number (<i>from: port</i>)	--
	Request destination host (<i>to: ipv4</i>)	--
	Request destination port number (<i>to: port</i>)	--
	Log message size (Units: bytes)	$284 + 2*(a + b)$
	Remarks	These items are also output at startup and termination of the following servers: RAP-processing server, RAP-processing listener, RAP-processing client manager, TP1/EE, RTSSPP, RTSSUP, MQC gateway server, mqrsp, mqrsp
Abnormal termination of user server	Message ID (<i>msgid</i>)	KFCA33407-E
	Component (<i>compid</i>)	adm
	Event type (<i>ctgry</i>)	Failure
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of the user server process that went down
	Object (<i>obj</i>)	User server name
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	Node identifier

Audited event	Information output for event	
	Item name	Contents
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	313 + 2*b
	Remarks	These items are also output at startup and termination of the following servers: RAP-processing server, RAP-processing listener, RAP-processing client manager, TP1/EE, RTSSPP, RTSSUP, MQC gateway server, mqrsp, mqrsp
User server shutdown	Message ID (<i>msgid</i>)	KFCA33408-I
	Component (<i>compid</i>)	scd
	Event type (<i>ctgry</i>)	Failure
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj:(euid, pid)</i>)	PID of scheduler service
	Object (<i>obj</i>)	User server name
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	Node identifier
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	348 + 2*b

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
Service shutdown on user server	Message ID (<i>msgid</i>)	KFCA33409-I
	Component (<i>compid</i>)	scd
	Event type (<i>ctgry</i>)	Failure
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of scheduler service
	Object (<i>obj</i>)	User server name, service name
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	Node identifier
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$365 + 2 * (b(\text{user-server-name}) + b(\text{service-name}))$
Successful client user authentication	Message ID (<i>msgid</i>)	KFCA33410-I
	Component (<i>compid</i>)	nam
	Event type (<i>ctgry</i>)	Authentication
	Event result (<i>result</i>)	Success
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of name service
	Object (<i>obj</i>)	Received login name
	Action information (<i>op</i>)	Login
	Object location information (<i>objloc</i>)	Node identifier
	Request source host (<i>from:ipv4</i>)	Y

Audited event	Information output for event	
	Item name	Contents
	Request source port number (from:port)	Y
	Request destination host (to:ipv4)	--
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	364 + 2*b
	Remarks	Output when Y is specified for the client_uid_check operand in the system common definition.
Unsuccessful client user authentication	Message ID (msgid)	KFCA33411-W
	Component (compid)	nam
	Event type (ctgry)	Authentication
	Event result (result)	Failure
	Subject ID information (subj: (eid, pid))	PID of name service
	Object (obj)	Received login name
	Action information (op)	Login
	Object location information (objloc)	Node identifier
	Request source host (from:ipv4)	Y
	Request source port number (from:port)	Y
	Request destination host (to:ipv4)	--
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	360 + 2*b
	Remarks	Output when Y is specified for the client_uid_check operand in the system common definition.
Service function started execution	Message ID (msgid)	KFCA33412-I

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Component (<i>compid</i>)	rpc
	Event type (<i>ctgry</i>)	AccessControl
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of the user server that received the service function request
	Object (<i>obj</i>)	Name of requested service
	Action information (<i>op</i>)	Enforce
	Object location information (<i>objloc</i>)	Name of the requested service group
	Request source host (<i>from:ipv4</i>)	Y
	Request source port number (<i>from:port</i>)	Y (Outputs the reception port number of the request source.)
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$552 + 2*(b + c) + FSV + FSVG$
	Remarks	<ul style="list-style-type: none"> Not acquired for SPP of XATMI, and SPP.NET. May also be output for the following servers, depending on the product versions used: Client extended service, RAP-processing server, RAP-processing listener, PAR-processing client manager, TP1/EE, MQC gateway server, mqrsp, mqrspup
Service function completed execution	Message ID (<i>msgid</i>)	KFCA33413-I
	Component (<i>compid</i>)	rpc
	Event type (<i>ctgry</i>)	AccessControl
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of the user server that received the service function request

Audited event	Information output for event	
	Item name	Contents
	Object (<i>obj</i>)	Name of requested service
	Action information (<i>op</i>)	Enforce
	Object location information (<i>objloc</i>)	Name of the requested service group
	Request source host (<i>from:ipv4</i>)	Y
	Request source port number (<i>from:port</i>)	Y (Outputs the reception port number of the request source.)
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$553 + 2*(b + c) + FSV + FSVG$
	Remarks	<ul style="list-style-type: none"> Not acquired for SPP of XATMI, and SPPNET. May also be output for the following servers, depending on the product versions used: Client extended service, RAP-processing server, RAP-processing listener, PAR-processing client manager, TP1/EE, MQC gateway server, mqrsp, mqrsp
Invalid message discarded	Message ID (<i>msgid</i>)	KFCA33414-W
	Component (<i>compid</i>)	rpc
	Event type (<i>ctgry</i>)	AnomalyEvent
	Event result (<i>result</i>)	Occurrence
	Subject ID information (<i>subj:(eid, pid)</i>)	PID of the process that detected the invalid message
	Object (<i>obj</i>)	Reception port number
	Action information (<i>op</i>)	Occur
	Object location information (<i>objloc</i>)	Node identifier (*** is output if the node identifier cannot be acquired.)

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Request source host (<i>from:ipv4</i>)	Y
	Request source port number (<i>from:port</i>)	Y (0 is output for UNIX domain communication.)
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	542
RPC call completed	Message ID (<i>msgid</i>)	KFCA33415-I
	Component (<i>compid</i>)	rpc
	Event type (<i>ctgry</i>)	AccessControl
	Event result (<i>result</i>)	Success/Failure
	Subject ID information (<i>subj: (euid, pid)</i>)	PID of the user server that issued the request
	Object (<i>obj</i>)	Destination service of RPC request
	Action information (<i>op</i>)	Enforce
	Object location information (<i>objloc</i>)	Name of the service group comprising the requested service
	Request source host (<i>from:ipv4</i>)	Y
	Request source port number (<i>from:port</i>)	Y (Outputs the reception port number of the call source.)
	Request destination host (<i>to:ipv4</i>)	Y (Not output if an error is detected before the destination of the RPC request can be established.)
	Request destination port number (<i>to:port</i>)	Y (Not output if an error is detected before the destination of the RPC request can be established.)
	Log message size (Units: bytes)	$709 + 2*(b + c) + FSV + FSVG$

Audited event	Information output for event	
	Item name	Contents
	Remarks	<ul style="list-style-type: none"> An entry is not output to the audit log for this event if the return value of <code>dc_rpc_call</code> is <code>DCRPCER_PROTO</code> or <code>DCRPCER_INVALID_ARGS</code>. May also be output for the following servers, depending on the product versions used: Client extended service, RAP-processing server, RAP-processing listener, PAR-processing client manager, TP1/EE, MQC gateway server, mqrsp, mqrsp
RPC response received (when using the <code>dc_rpc_poll_any_replies</code> function)	Message ID (<code>msgid</code>)	KFCA33416-I
	Component (<code>compid</code>)	rpc
	Event type (<code>ctgry</code>)	AccessControl
	Event result (<code>result</code>)	Success/Failure
	Subject ID information (<code>subj: (euid, pid)</code>)	PID of the user server that issued <code>dc_rpc_poll_any_replies()</code>
	Object (<code>obj</code>)	Destination service of RPC request (***** is output if an error is detected when using asynchronous-response RPC, before a response is received.)
	Action information (<code>op</code>)	Enforce
	Object location information (<code>objloc</code>)	Name of the service group comprising the requested service (***** is output if an error is detected when using asynchronous-response RPC, before a response is received.)
	Request source host (<code>from:ipv4</code>)	--
	Request source port number (<code>from:port</code>)	--
	Request destination host (<code>to:ipv4</code>)	--

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	$538 + 2*(b + c) + FSV + FSVG$
	Remarks	May also be output for the following servers, depending on the product versions used: Client extended service, RAP-processing server, RAP-processing listener, PAR-processing client manager, TP1/EE, MQC gateway server, mqrsp, mqrsp
Invalid RAP message discarded	Message ID (msgid)	KFCA33417-W
	Component (compid)	scs
	Event type (ctgry)	AnomalyEvent
	Event result (result)	Occurrence
	Subject ID information (subj: (euid, pid))	PID of the process that detected the invalid message
	Object (obj)	Reception port number
	Action information (op)	Occur
	Object location information (objloc)	Node identifier
	Request source host (from:ipv4)	Y
	Request source port number (from:port)	Y
	Request destination host (to:ipv4)	Y
	Request destination port number (to:port)	Y
	Log message size (Units: bytes)	535
Error accessing the OpenTP1 file system	Message ID (msgid)	KFCA33418-W
	Component (compid)	fil
	Event type (ctgry)	ContentAccess
	Event result (result)	Failure

Audited event	Information output for event	
	Item name	Contents
	Subject ID information (<i>subj</i> : (<i>eid</i> , <i>pid</i>))	User name or ID of the process that requested access to the file
	Object (<i>obj</i>)	Name of the OpenTP1 file (If the name of the OpenTP1 file system cannot be acquired, this item is not output.)
	Action information (<i>op</i>)	Refer/Add/Update/Delete
	Object location information (<i>objloc</i>)	Name of the OpenTP1 file system (If the name of the OpenTP1 file system cannot be acquired, this item is not output.)
	Request source host (<i>from</i> : <i>ipv4</i>)	--
	Request source port number (<i>from</i> : <i>port</i>)	--
	Request destination host (<i>to</i> : <i>ipv4</i>)	--
	Request destination port number (<i>to</i> : <i>port</i>)	--
	Log message size (Units: bytes)	$345 + 2*a + b + c$
Command execution	Message ID (<i>msgid</i>)	KFCA33419-I
	Component (<i>compid</i>)	cmd
	Event type (<i>ctgry</i>)	Maintenance
	Event result (<i>result</i>)	Success/Failure/Occurrence
	Subject ID information (<i>subj</i> : (<i>eid</i> , <i>pid</i>))	User name or ID of the user who executed the command.
	Object (<i>obj</i>)	Command name
	Action information (<i>op</i>)	Maintain
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from</i> : <i>ipv4</i>)	--
	Request source port number (<i>from</i> : <i>port</i>)	--
	Request destination host (<i>to</i> : <i>ipv4</i>)	--

C. Information Output for Audited Events

Audited event	Information output for event	
	Item name	Contents
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	$306 + 2*(a + b) + CPARM$
Startup of OpenTP1 service	Message ID (msgid)	KFCA33420-I
	Component (compid)	nts
	Event type (ctgry)	StartStop
	Event result (result)	Success
	Subject ID information (subj: (euid, pid))	Service logon account
	Object (obj)	Service name (Windows service name)
	Action information (op)	Start
	Object location information (objloc)	--
	Request source host (from:ipv4)	--
	Request source port number (from:port)	--
	Request destination host (to:ipv4)	--
	Request destination port number (to:port)	--
	Log message size (Units: bytes)	$268 + 2*(a + b)$
	Remarks	Output only in the Windows version
	Termination of OpenTP1 service	Message ID (msgid)
Component (compid)		nts
Event type (ctgry)		StartStop
Event result (result)		Success
Subject ID information (subj: (euid, pid))		Service logon account
Object (obj)		Service name (Windows service name)

Audited event	Information output for event	
	Item name	Contents
	Action information (<i>op</i>)	Stop
	Object location information (<i>objloc</i>)	--
	Request source host (<i>from:ipv4</i>)	--
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$259 + 2*(a + b)$
	Log message size (Units: bytes)	Output only in the Windows version
User-specific information acquired from a UAP	Message ID (<i>msgid</i>)	KFCA34000- <i>x</i> to KFCA34999- <i>x</i>
	Component (<i>compid</i>)	User-specified value. (Output in the format * <i>AA</i> , where <i>AA</i> is the value specified by the audit log output API.)
	Event type (<i>ctgry</i>)	User-specified value
	Event result (<i>result</i>)	Success/Failure/Occurrence
	Subject ID information (<i>subj:(euid, pid)</i>)	User name or ID of the user server that called the audit log output API
	Object (<i>obj</i>)	Service name
	Action information (<i>op</i>)	User-specified value
	Object location information (<i>objloc</i>)	User server name
	Request source host (<i>from:ipv4</i>)	Y
	Request source port number (<i>from:port</i>)	--
	Request destination host (<i>to:ipv4</i>)	--
	Request destination port number (<i>to:port</i>)	--
	Log message size (Units: bytes)	$290 + a + b + c + msg$

Legend:

Y: This item is output.

--: N/A

Explanation of variables used in equations for calculating log message size

To calculate the log message size, replace the variables in the equation with the values for those items. The variables *FSV*, *FSVG*, *CPARM*, and *msg* refer to data output in comment (msg) format. The following table gives the meaning of each variable:

Variable	Description
<i>a</i>	The number of characters output as subject ID information
<i>b</i>	The number of characters output as object information
<i>c</i>	The number of characters output as object location information
<i>FSV</i>	The number of characters in the name of the requesting service
<i>FSVG</i>	The number of characters in the name of the service group that issued the request
<i>CPARM</i>	The number of characters in the command parameters
<i>msg</i>	The number of characters in the comment

Calculating the log message size

Calculate the log message size by using the equation particular to the event. To the result of the equation, add the number of characters in the value of the `DCDIR` environment variable. The result is the log message size for that event.

Example:

The following is an example of calculating the log message size for an OpenTP1 startup event (associated with message ID KFCA33400-I). The name of the user who executed the `dcstart` command is `tp1user`. The value of environment variable `DCDIR` is `/usr/OpenTP1`, adding 12 to the result.

Equation	= 290+2*a
	= 290+2*7
	= 304
Chars in DCDIR	= 12
Log size	= 304 + 12
	= 316

Note on log message sizes

The equations for calculating log message size are meant to provide a certain amount of leeway. For this reason, the actual size of the log data may be smaller than the estimate.

D. Journal Information Collected by the Message Control Facility

The following shows journal information (AJ, CJ, GJ, IJ, MJ, and OJ) collected by the message control facility on the unloaded-journals file. Also provided are the conditions for collecting journals by the message control facility and the calculation formulae of the necessary amount of journal.

The following table shows the symbols indicating the attribute used in the MCF journal record format.

Table D-1: Attribute symbols

Attribute symbols	Meaning
char	One-byte area with character attribute
uchar	One-byte area with unsigned attribute
char[*]	Character attribute area. Nothing specified for * indicates a variable length. Decimal specified for * indicates a number of bytes. To reference this area, use the character string operation with the number of characters specified.
ulong	Four-byte area with unsigned numeric attribute

D.1 AJ record format

Item	Position (byte)	Attribute	Description
Reserved	0	char[24]	Reserved area
Output logical terminal name	24	char[16]	Name of the output logical terminal that sent the output message.
Application name	40	char[10]	Name of the application that sent the output message to the logical terminal. This information is not collected for an SPP.
Reserved	50	char[18]	Reserved area
Journal collection time	68	ulong	Time the journal was written to the MCF journal buffer. (Seconds counted from 0: 00: 00 at January 1, 1970)
Journal collection time	72	ulong	Time the journal was written to the MCF journal buffer. (Fewer than a second, counted from 0: 00: 00 at January 1, 1970)

Item	Position (byte)	Attribute	Description
Reserved	76	char[4]	Reserved area
Output message type	80	char	Output message type indicated by the character: n: Ordinary unidirectional send message o: Query message p: Priority unidirectional send message
Assignment of output serial number	81	uchar	Whether or not the message has output serial number is indicated as follows: 0x00: Message without output serial number 0x01: Message with output serial number
Reserved	82	char[2]	Reserved area
Message output serial number	84	ulong	Output serial number when the output message was sent. This information is valid when the assignment of output serial number is 0x01.
Reserved	88	char[24]	Reserved area

D.2 GJ record format

Item	Position (byte)	Attribute	Description
Reserved	0	char[8]	Reserved area
Input logical terminal name	8	char[18]	Name of the input logical terminal that received the input message.
Reserved	24	char[16]	Reserved area
Application name	40	char[10]	Name of the application that processes the input message.
Reserved	50	char[2]	Reserved area
Message input serial number	52	char[12]	The input serial number of the message received from the input logical terminal. This information is not a character string, and must be handled as 12-byte area.
Reserved	64	char[4]	Reserved area
Journal collection time	68	ulong	Time the journal was written to the MCF journal buffer. (Seconds counted from 0: 00: 00 at January 1, 1970)

Item	Position (byte)	Attribute	Description
Journal collection time	72	ulong	Time the journal was written to the MCF journal buffer. (Fewer than a second, counted from 0: 00: 00 at January 1, 1970)
Reserved	76	char[4]	Reserved area
Map name	80	char[9]	Map name of the NEXT map, if any.
Reserved	89	char[15]	Reserved area
Input message size	104	ulong	Size of the message in bytes received from the input logical terminal.
Input message	108	char[]	The message received from the input logical terminal is set. The size of this variable-length information is as large as indicated by the input message size.

D.3 IJ record format

Item	Position (byte)	Attribute	Description
Reserved	0	char[8]	Reserved area
Input logical terminal name	8	char[16]	Name of the input logical terminal that received the input message.
Reserved	24	char[16]	Reserved area
Application name	40	char[10]	Name of the application that processes the input message.
Reserved	50	char[2]	Reserved area
Message input serial number	52	char[12]	The input serial number of the received message. This information is not a character string, and must be handled as 12-byte area.
Reserved	64	char[4]	Reserved area
Journal collection time	68	ulong	Time the journal was written to the MCF journal buffer. (Seconds counted from 0: 00: 00 at January 1, 1970)

Item	Position (byte)	Attribute	Description
Journal collection time	72	ulong	Time the journal was written to the MCF journal buffer. (Fewer than a second, counted from 0: 00: 00 at January 1, 1970)
Reserved	76	char[4]	Reserved area
Map name	80	char[9]	Map name of the NEXT map, if any.
Input message type	89	char	Input message type indicated by the character: i: Query message n: Unidirectional receive message
Order identifier	90	char	The identifier indicates the order of the message when the input message is divided. s: Input message is not divided. f: First message after which message (m, l) is received. m: Middle message after which message (m, l) is received. l: Last message, followed by no message.
Reserved	91	char[13]	Reserved area
Input message size	104	ulong	Size of the message in bytes received by the input logical terminal.
Input message	108	char[]	The message received by the input logical terminal is set. The size of this variable-length information is as large as indicated by the input message size.

D.4 MJ record format

Item	Position (byte)	Attribute	Description
Reserved	0	char[8]	Reserved area
Input logical terminal name	8	char[16]	For input, indicates the name of the input logical terminal that received the message.
Output logical terminal name	24	char[16]	For output, indicates the name of the output logical terminal to which the message was sent.
Reserved	40	char[28]	Reserved area
Journal collection time	68	ulong	Time the journal was written to the MCF journal buffer. (Seconds counted from 0: 00: 00 at January 1, 1970)

D. Journal Information Collected by the Message Control Facility

Item	Position (byte)	Attribute	Description
Journal collection time	72	ulong	Time the journal was written to the MCF journal buffer. (Fewer than a second, counted from 0: 00: 00 at January 1, 1970)
Reserved	76	char[4]	Reserved area
Connection name	80	char[16]	Name of the connection used for receiving or sending the message.
Message journal type	96	char	Message journal type indicated by the character: i: Message journal before input editing o: Message journal before output editing
Order identifier	97	char	The identifier indicates the order of the message when the input or output message is divided. s: Message is not divided. f: First message after which message (m, l) is received. m: Middle message after which message (m, l) is received. l: Last message, followed by no message.
Reserved	98	char[14]	Reserved area
Input or output message size	112	ulong	Size of the input or output message in bytes.
Input or output message	116	char[]	The message received by the input or output logical terminal is set. The size of this variable-length information is as large as indicated by the input or output message size.

D.5 OJ record format

Item	Position (byte)	Attribute	Description
Reserved	0	char[24]	Reserved area
Output logical terminal name or started application name	24	char[16]	Name of the output logical terminal to which the output message was sent, or the started application name
Application name	40	char[10]	Name of the application that sent the output message. This information is not collected for an SPP.
Reserved	50	char[18]	Reserved area

Item	Position (byte)	Attribute	Description
Journal collection time	68	ulong	Time the journal was written to the MCF journal buffer. (Seconds counted from 0: 00: 00 at January 1, 1970)
Journal collection time	72	ulong	Time the journal was written to the MCF journal buffer. (Fewer than a second, counted from 0: 00: 00 at January 1, 1970)
Reserved	76	char[4]	Reserved area
Map name	80	char[9]	Map name specified when the output message was sent.
Output message type	89	char	Output message type indicated by the character: n: Ordinary unidirectional send message o: Query message p: Priority unidirectional send message Δ: Application startup message
Reserved	90	char	Reserved area
Assignment of output serial number	91	char	Whether or not the message has output serial number is indicated as follows: 0x00: Message without output serial number 0x01: Message with output serial number
Message output serial number	92	ulong	Output serial number when the output message was sent. This information is valid when the assignment of output serial number is 0x01.
Reserved	96	char[8]	Reserved area
Send message size	104	ulong	Size of the message in bytes sent to the logical terminal or started application.
Send message	108	char[]	The message sent to the logical terminal or started application is set. The size of this variable-length information is as large as indicated by the message size.

D.6 Conditions for collecting journals by the message control facility

Type	Condition	Collecting process	When to collect
AJ	Specify <code>mcfalcle -o aj=yes</code> in the MCF communication configuration definition.	Communication process	When outputting a message (sending completed)
GJ	Specify <code>mfaalcap -j gj=yes</code> in the MCF application definition.	UAP process	When receiving a message (collected in the <code>dc_mcf_receive()</code> function).
IJ	Specify <code>mfaalcap -j ij=yes</code> in the MCF application definition.	Communication process	When inputting a message (before entering to the input queue)
MJ	Enter the <code>mcfactmj</code> command.	Communication process	When inputting a message (before entering to the input queue) When outputting a message (after reading from the output queue)
OJ	MHP: MCF application definition When <code>mfaalcap -j oj=yes</code> is specified SPP: User service definition (or user service default definition) When <code>mcf_spp_oj=Y</code> is specified	UAP process	When sending a message (collected in the <code>dc_mcf_send()</code> function) or when the startup of an application is requested (collected in the <code>dc_mcf_execap()</code> function)
CJ (serial number)	Send a message with the serial number specified.	UAP process	When sending a message (collected in the <code>dc_mcf_send()</code> function)
CJ (last serial number)	Send a message with the serial number specified.	Communication process	When outputting a message (sending completed)
CJ (QUE#)	Specify <code>mfaalcap -g quekind=disk</code> in the MCF application definition.	Communication process	When inputting a message (after entering to the input queue)
CJ (QUE#)	Specify <code>mfaalcap -g quekind=disk</code> in the MCF application definition, or specify <code>mcfalcle -k quekind=disk</code> in the MCF communication configuration definition.	UAP process	At the synchronization point
CJ (QUE#)	Specify <code>mcfalcle -k quekind=disk</code> in the MCF communication configuration definition.	Communication process	When outputting a message (output queue is deleted)

#

Recovery journal when a message queue server collects journals.

D.7 Calculation formula of the necessary amount of journal collected by the message control facility

Type	Calculation formula (unit: bytes)
AJ	176
GJ	$\uparrow (204+\text{seg})/4 \uparrow \times 4$
IJ	$\uparrow (172+\text{seg})/4 \uparrow \times 4$
MJ	$\uparrow (180+\text{seg})/4 \uparrow \times 4$
OJ	$\uparrow (204+\text{seg})/4 \uparrow \times 4$
CJ (serial number)	Outside a transaction: 128 (MHP with mcfalcap -n trnmode=nontrn specified in the application definition) For transaction (for other than above): 160
CJ (last serial number)	144
CJ (QUE [#])	$\uparrow \{88+(24 \times (\uparrow \text{msg}/\text{ql} \uparrow + \uparrow 960/\text{ql} \uparrow)) + \text{msg}^*\}/4 \uparrow \times 4$ (collected in the communication process during input of a message)
CJ (QUE [#])	$\uparrow \{260 + \sum \{(24 \times (\uparrow \text{msg}/\text{ql} \uparrow + \uparrow 960/\text{ql} \uparrow))\} + \sum \text{msg}^*\}/4 \uparrow \times 4$ (collected in the UAP process at the synchronization point) Σ : Sum of the estimations $(24 \times (\uparrow \text{msg}/\text{ql} \uparrow + \uparrow 960/\text{ql} \uparrow))$ for messages received or sent in the transaction.
CJ (QUE [#])	$\uparrow \{88+(24 \times (\uparrow \text{msg}/\text{ql} \uparrow + \uparrow 960/\text{ql} \uparrow)) + \text{msg}^*\}/4 \uparrow \times 4$ (collected in the communication process during output of a message)

#

Recovery journal when a message queue server collects journals.

$\uparrow \uparrow$: Rounded up to the nearest integer.

seg: Segment length (A message can be divided into segments. One message may be one segment or may consist of segments. IJ, GJ, MJ and OJ are collected in units of segments.)

aps: Number of applications started by the dc_mcf_execap() function

les: Number of logical terminals that send the response, branch or resend messages.

ap: 32 when using a disk for the input queue

D. Journal Information Collected by the Message Control Facility

0 when using memory for the input queue

msg: Sent or received message length

q1: Queue file physical record length (value specified with the -s option of the queinit command)

msg*: Add this value when the value of `que_io_maxrecsize` in the message queue service definition is larger than the message.

E. Details of Statistics

E.1 System statistics

The following table shows the details of the system's statistics.

Table E-1: Details of the system's statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
RPC information	RPC call (response time)	<p>Length of time between sending a request from the client to the server in <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code>, and receiving a response from the server.</p> <p>This value is obtained when <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are successful.</p> <p>Synchronous-response RPC</p> <p>Length of time between starting a <code>dc_rpc_call</code> or <code>dc_rpc_call_to</code> request, and receiving a response. The time includes the length of time for retries if the transmission of a request to the server in the <code>dc_rpc_call</code> or <code>dc_rpc_call_to</code> function fails and transmission is retried.</p> <p>Chained RPC</p> <p>Length of time between starting a <code>dc_rpc_call</code> or <code>dc_rpc_call_to</code> request, and receiving a response.</p> <p><Number of occurrences></p> <p>Number of times <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are issued</p>	1
	User service execution (execution time)	<p>Length of time from the start of execution of a service function requested by <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code>, until the service function issues a return value.</p> <p>This value is obtained in <code>dc_rpc_mainloop</code>.</p> <p><Number of occurrences></p> <p>Number of times <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are issued</p>	2
	RPC timeout	<p>Timeout errors during RPC response wait processing.</p> <p><Number of occurrences></p> <p>Timeout error count</p>	3
	RPC failures	<p>Internal failures during RPC processing.</p> <p><Number of occurrences></p> <p>Number of above failures</p>	4

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
Schedule information	Schedule waits (queue length)	<p>The number of remaining service requests in the schedule queue of the user server (SPP).</p> <p>When the queue length is lengthy, increase the number of resident processes specified by the <code>parallel_count</code> clause in the user service definition. If there is any nonresident process, decrease the specification of the <code>balance_count</code> clause to make it easier to start the nonresident process.</p> <p>When the queue length is small, decrease the number of resident processes specified by the <code>parallel_count</code> clause in the user service definition, if necessary. If there is any nonresident process, increase the specification of the <code>balance_count</code> clause to make it harder to start the nonresident process. When there is difference between the average and the minimum value of the queue length, adjust the ratio of resident processes to nonresident processes specified in the <code>balance_count</code> clause, as well as the maximum number of processes.</p> <p><Number of occurrences> Number of times the client made service requests (<code>dc_rpc_call()</code>) to the user server</p>	11
	Schedule (message size)	<p>The length of the service request message received by the user server (SPP).</p> <p>The message length includes 512 bytes of control information added by the system.</p> <p>It is preferable to specify a maximum message size in the <code>message_bufLen</code> clause in the user service definition. The following shows the range of preferable specifications for the <code>message_store_bufLen</code> clause in the user service definition:</p> <p>Message size (average) x queue length (max.) ≤ <code>message_store_bufLen</code> ≤ message size (max.) x queue length (max.)</p> <p><Number of occurrences> Number of times the client made service requests (<code>dc_rpc_call</code>) to the user server.</p>	12
	Size of the currently used buffer pool for storing messages	<p>The length of the buffer pool currently used by the user server (SPP) for storing messages, or shared by the scheduling buffer group, i.e., a group of user servers, to which the SPP belongs.</p> <p><Number of occurrences> Number of times the client made service requests (<code>dc_rpc_call</code>) to the user server</p>	13

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Message size that could not be scheduled due to an insufficient buffer pool for storing messages	Of the service requests to the user server (SPP), the length of the service request messages that could not be scheduled due to an insufficient buffer pool for storing messages. <Number of occurrences> Number of times the user server failed to schedule service requests (<code>dc_rpc_call</code>) from the client due to an insufficient buffer pool for storing messages.	14
Lock information	Lock acquisition (wait time)	The time from the beginning of a lock wait status until the wait status is released. The lock here means an internal lock by <code>dc_lck_get</code> , or an internal lock of the lock service issued by DAM, TAM, and MQA. <Number of occurrences> Number of times the lock wait occurred	21
	Lock wait (queue length)	Queue length when lock wait occurred. <Number of occurrences> Number of times the lock wait occurred	22
	Deadlock	Number of deadlocks that occurred. <Number of occurrences> Same as above	23
DAM information	Read (input length)	Input length of the DAM file data in the unit used for issuing <code>dc_dam_read</code> . <Number of occurrences> Number of times the <code>dc_dam_read</code> was issued	31
	Read errors	Errors that occurred in the <code>dc_dam_read</code> interface part with the OS. Errors such as an invalid API argument are not counted. <Number of time occurrences> Count of above error	32
	Write (output length)	Output length of the DAM file data in the unit used for issuing <code>dc_dam_write</code> or <code>dc_dam_rewrite</code> . <Number of occurrences> Number of times the <code>dc_dam_write</code> or <code>dc_dam_rewrite</code> was issued	33
	Write errors	Output errors during actual input/output operations with a disk. Errors such as an invalid API argument are not counted. <Number of occurrences> Count of above error	34

E. Details of Statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Use of update buffer (update size)	Amount of recovery journal (FJ) Tune the size and number of journal files by calculating the amount of a journal for using DAM. <Number of occurrences> Number of times the recovery journal (FJ) was obtained.	35
	The first DAM API issued in a transaction branch (total number of transaction branches)	The number of transactions using DAM. The maximum value of this information is used as the peak number of transactions using DAM. To obtain an optimum memory (amount of dynamic shared memory for TP1/Server Base), specify the maximum value of this information for <code>dam_tran_process_count</code> in the DAM service definition. <Number of occurrences> Not collected.	36
	Request for allocating a shared memory cache block	Number of times the DAM file data block that is allocated in the resource manager's shared memory pool was allocated. <Number of occurrences> Number of events	38
	Shared memory allocation (shared memory pool usage ratio)	The periodically-obtained usage ratio of the resource manager shared memory pool. We recommend that you specify the size of the shared memory for the resource manager (<code>dam_cache_size</code> in the DAM service definition) based on this value. If the UAP frequently drops because the maximum value approaches 100%, increase the value of <code>dam_cache_size</code> . <Number of occurrences> Number of transactions that updated the DAM file specified with deferred update divided by 10	39
Process information	UAP abnormal termination	Abnormal terminations of UAP processes. <Number of occurrences> Sum of the UAP processes which terminated abnormally	41
	Abnormal termination of system server	Abnormal terminations of system service processes. <Number of occurrences> Sum of the system servers which terminated abnormally	42
	Process generation (total number of processes)	Sum of the number of system service processes and UAP processes started in the OpenTP1 system at specified intervals. <Number of occurrences> Not collected	43

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
TAM information	TAM file real updating (number of written bytes)	<p>Number of data bytes (total value) actually updated in the TAM file during online operation.</p> <p>To calculate the average number of bytes updated for one occurrence, use the following formula:</p> <ul style="list-style-type: none"> Average number of bytes updated for one occurrence = $(\text{current-maximum-value} - \text{previous-maximum-value}) / \text{number-of-occurrences}$ <p>If there is no previous data, use the following formula:</p> <ul style="list-style-type: none"> Average number of bytes updated for one occurrence = $(\text{maximum-value} - \text{minimum-value}) / (\text{number-of-occurrences} - 1)$ <p><Number of occurrences></p> <p>Total number of data updates for the TAM file.</p> <p>Timer-start, cpd, or tamhold triggers the real update of the TAM file.</p>	51
	Commits, rollbacks (record reference count)	<p>The number of records referenced by one transaction. Since the TAM server outputs this value as statistics during commit processing, the number of occurrences is counted up no matter whether or not the <code>dc_tam_read</code> is issued. Therefore, a system that does not issue the <code>dc_tam_read</code> counts up the number of occurrences while the maximum value, minimum value and average remain zero.</p> <p><Number of occurrences></p> <p>Number of commits or rollbacks for TAM.</p>	55
	Commits, rollbacks (record update count)	<p>The number of records updated by one transaction. Since the TAM server outputs this value as statistics during commit processing, the number of occurrences is counted up no matter whether or not the <code>dc_tam_write</code> is issued. Therefore, a system that does not issue the <code>dc_tam_write</code> counts up the number of occurrences while the maximum value, minimum value and average remain zero.</p> <p><Number of occurrences></p> <p>Number of commits or rollbacks for TAM.</p>	56
Transaction information	Commits	<p>Transactions determined to be committed.</p> <p><Number of occurrences></p> <p>The number of transactions determined to be committed</p>	61
	Rollbacks	<p>Transactions determined to be rolled back.</p> <p><Number of occurrences></p> <p>The number of transactions determined to be rolled back</p>	62

E. Details of Statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
Name information	Cache hits	<p>The number of times the service information set in the service information cache area is referenced at the node. The service information cache area means the area for setting service information of the server started at a different node as for the name service. Service information means the address information needed for RPC with the server. This information is set when a client requests reference to service information while the name service of the same node as for the client does not have the service information.</p> <p><Number of occurrences> Number of times above information was referenced</p>	71
	Local hits	<p>The number of times the service information set in the service information local area is referenced in the node. The service information local area means the area for setting service information of the server started at the same node as for the name service. Starting the server sets the service information.</p> <p><Number of occurrences> Number of times above information was referenced</p>	72
	Lookup	<p>The number of times the reference to service information is requested in the node. This data is counted up even if neither the service information local area nor the service information cache area stores service information.</p> <p><Number of occurrences> Number of times the reference to service information is requested</p>	73
Checkpoint dump information	Checkpoint dump collection (collection interval)	<p>Checkpoint trigger interval. This means the time interval between a previous checkpoint and the current checkpoint. Since a checkpoint occurs depending on output count of a regular journal block, the frequency is evaluated by time.</p> <p><Number of occurrences> Number of checkpoint dumps triggered</p>	81
	Checkpoint dump validation (collection time)	<p>The time between triggering and validating the checkpoint dump collection. This means the time that elapsed before completing validation of checkpoint dump after an event triggers checkpoint dump and each system server starts dump processing.</p> <p><Number of occurrences> Number of checkpoint dumps triggered</p>	82

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
Message queue information	Read message (input length)	The message length read from the queue file. <Number of occurrences> Read message length	91
	Write message (output length)	The message length written to the queue file. <Number of occurrences> Written message length	92
	Read errors	FIL server errors during input from the physical file (abnormal cases). <Number of occurrences> Count of above error	93
	Write errors	FIL server errors during output to the physical file (abnormal cases). <Number of occurrences> Count of above error	94
	Waits for an available buffer	The number of times the necessary input/output buffers became insufficient due to too many requests concentrated on the queue file. Use this value to check the number of input/output buffers in the queue service definition. <Number of occurrences> Number of events	95
	Real reads	Input count from the physical file when there is no message in the input/output buffer. To reduce this value, increase the number of input/output buffers in the queue service definition. <Number of occurrences> Count of above input	96
	Real writes	Output count to the physical file. Compare this value with the number of write messages shown above to see how many write requests were input and/or output in batches. To reduce the input/output count, increase the number of input/output buffers in the queue service definition. The output count to the physical file should be zero when the deferred write facility is in use, because any output to the physical file is counted as the deferred writing (real write count). <Number of occurrences> Count of above output	97

E. Details of Statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Deferred writing (real write count)	Output count to the physical file by deferred writing when the deferred write facility is being used. <Number of occurrences> Count of above output	99
	Deferred writing for each physical file (number of records written in deferred mode)	The value obtained for tuning the number of input/output buffers (<code>quegrp -n</code> option) in the message queue service definition. <Number of occurrences> Number of records written in deferred mode	151
	Deferred writing for each physical file (messages written in deferred mode)	The value obtained for tuning the number of input/output buffers (<code>quegrp -n</code> option) in the message queue service definition. <Number of occurrences> Percentage of messages written in deferred mode	152
Journal information	Buffer full	The number of journal buffers for a system is twice of the number of parallel disk accesses, plus 1. One of these buffers is the current buffer. Buffer full means the condition that an attempt to buffer the journal record to the current buffer failed due to insufficient buffer area available. <Number of occurrences> Number of events	102
	Waits for available buffer	The condition of journal records waiting to be buffered because all journal buffers are full or in the process of being output. <Number of occurrences> Number of events	103
	Journal output (block length)	Data length of a journal block Use this value to obtain the amount of the journal that occurred in the specified unit of time. <Number of occurrences> Journal block output count	104
	Journal output (data length of the non-bus section)	OpenTPI internal information. <Number of occurrences> Number of events	105

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Input/output waits (number of waiting buffers)	The number of buffers waiting for output upon completion of journal output. The actual value times 100 is obtained to prevent the average value from being rounded. <Number of occurrences> Completed journal output count	107
	Write	System's internal output count that includes both the journal data and file control information during a journal swap. <Number of occurrences> Count of above output	108
	Write errors	Errors when data is written to a journal file. <Number of occurrences> Count of above error	109
	Swap (swap time)	The overhead time when journal files are swapped. <Number of occurrences> Journal file was swapping count	110
	Journal input (data length)	Length of data input from a journal file. <Number of occurrences> Count of data input from the journal file	111
	Read	The count for system's internal input that includes both the journal data and file control information during journal swap. <Number of occurrences> Above input count	113
	Read errors	Errors when data is read from a journal file. <Number of occurrences> Count of above error	114
MCF information	Message input to the memory queue	Writing of messages to the memory queue on the OpenTP1 node. <Number of occurrences> Message write count shown above	121
	Message input to the disk queue	Writing of messages to the disk queue on the OpenTP1 node. <Number of occurrences> Message write count shown above	122

E. Details of Statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Message output from the memory queue	Retrieving messages from the memory queue in the OpenTP1 node. <Number of occurrences> Message retrieval count shown above	123
	Message output from the disk queue	Retrieving messages from the disk queue in the OpenTP1 node. <Number of occurrences> Message retrieval count shown above	124
	Disconnects due to a connection error	Connection errors in the OpenTP1 node. <Number of occurrences> Count of above error	125
Shared memory control information	Size of static shared memory used	Total size of static shared memory blocks being used after static shared memory blocks have been allocated or released. <Number of occurrences> Number of statistics journals collected during the editing period + number of times the function for allocating or releasing static shared memory was issued	131
	Maximum size of the required amount of static shared memory	Maximum size of the static shared memory blocks in use. This information is obtained after the allocation or release of blocks if the previous pool size has been updated. The pool size is the total size of the shared memory in use, and includes empty areas in the pool created by releasing allocated shared memory. Accordingly, specify a value larger than this size in <code>static_shmpool_size</code> of the system environment definition. <Number of occurrences> Same as the size of static shared memory in use (if the previous pool size has been updated)	132
	Size of dynamic shared memory used	Total size of the dynamic shared memory blocks being used after dynamic shared memory blocks have been allocated or released. <Number of occurrences> Number of statistics journals collected during the editing period + number of times the function for allocating or releasing dynamic shared memory was issued	133

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Maximum size of the required amount of dynamic shared memory	Maximum size of the dynamic shared memory blocks in use. This information is obtained after allocation or release of blocks if the previous pool size has been updated. Specify a value larger than this size in <code>dynamic_shmpool_size</code> of the system environment definition. <Number of occurrences> Same as the size of dynamic shared memory in use (if the previous pool size has been updated)	134
MQA service information	Get message (input length)	The length of the message that was read from the queue by the application and MQT server. <Number of occurrences> Number of times the AP and MQT server retrieved messages from the queue	161
	Put message (output length)	The length of the message that was entered into the queue by the application and MQT server. <Number of occurrences> Number of times the AP and MQT server entered messages to the queue	162
	Read errors	I/O errors during input of messages from a file. <Number of occurrences> Count of above error	163
	Write errors	I/O errors during output of messages to a file. <Number of occurrences> Count of above error	164
	Waits for an available buffer	The number of times the buffers became insufficient. Increase the number of input/output buffers if this value is counted. <Number of occurrences> Number of events	165
	Real reads	I/O for reading messages from the file. If this value is counted, increase the number of input/output buffers to reduce I/O for receiving messages. <Number of occurrences> Number of events	166
	Real writes	I/Os for writing messages to a file. <Number of occurrences> Number of events	167

E. Details of Statistics

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
	Write request time	Processing time for writing messages to a file (I/O time + time for processing of request to the MQAI/O process). <Number of occurrences> Number of requests for writing messages to the file	169
	Number of messages activated at one trigger	Number of messages left in the queue when a trigger was generated. <Number of occurrences> Trigger message generation count	170
	Wait time for starting message transfer	The time between issuing MQPUT and startup of message transfer. <Number of occurrences> Number of messages transferred	171
	Wait time for message arrival	The time between issuing MQPUT and completion of message transfer. <Number of occurrences> Number of messages transferred	172
IST service information	Input length	Information about the input length of the IST table data in the unit used for issuing dc_ist_read. <Number of occurrences> Number of times the dc_ist_read was issued	191
	Output length	Information about the output length of the IST table data in the unit used for issuing dc_ist_write. <Number of occurrences> Number of times the dc_ist_write was issued	192
	Number of updates from other nodes	Number of update messages received prompting the local node to apply the IST table update operations performed at other nodes. <Number of occurrences> Number of events	193
	Number of updates to other nodes	Number of times the local node updated the IST table and notified other nodes of the updating. <Number of occurrences> Number of events	194

Types of statistics	Event (obtained value)	Details	ID for edition by dcreport#
XATMI service information	XATMI call (response time)	<p>The time between sending a service request from the client and receiving a response during OSI TP communication using the XATMI interface.</p> <p>Synchronous-response The time between sending a service request with <code>tpcall</code> and receiving a response.</p> <p>Asynchronous-response The time between sending a service request with <code>tpacall</code> and receiving a response with <code>tpgetrply</code>.</p> <p>Non-response Invalid value.</p> <p><Number of occurrences> Number of times the <code>tpcall</code> or <code>tpacall</code> was issued. Those that returned by error before sending a service request are not counted.</p>	211
	XATMI user service execution time	<p>The time from the beginning of execution of a service function at the server during OSI TP communication using the XATMI interface until the service function issues <code>return</code>.</p> <p><Number of occurrences> Number of times the user service was executed.</p>	212
	XATMI service failure count	<p>Number of failures in OSI TP communication using the XATMI interface</p> <p>Client side Failures that occurred within a transaction after the service request was sent but before the transaction is determined, and failures that occurred outside a transaction after the service request was sent but before the client receives a response from the service. These failures include communication failures, server UAP failures, and error responses from the service.</p> <p>Server side Failures that occurred within a transaction after the service request was received but before the transaction is determined, and failures that occurred outside a transaction after the service request was received but before the service function terminates. These failures include communication failures, client UAP failures, and rollback directions from the client.</p> <p><Number of occurrences> Number of above failures.</p>	213

#

Argument to be specified when the `dcreport` command is executed.

The following table lists the items to be edited in the system statistics.

The *Unit* in the table is the number of times an event occurred or the unit of the obtained value if there is one. If a particular event did not occur (the number of occurrences is 0), the edited values are meaningless.

Table E-2: Items to be edited in system statistics

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
RPC information	RPC calls (response times)	Y	Y	Y	Y	Micro-seconds	Y	1
	Execution of user service (execution times)	Y	Y	Y	Y	Micro-seconds	Y	2
	RPC overtimes	Y	N	N	N	Number of events	Y	3
	RPC errors	Y	N	N	N	Number of events	Y	4
Scheduling information	Schedule wait (wait queue length)	Y	Y	Y	Y	Requests	Y	11
	Schedule (message size)	Y	Y	Y	Y	Bytes	Y	12
	Size of the currently used buffer pool for storing messages	Y	Y	Y	Y	Bytes	Y	13

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Message size that could not be scheduled due to insufficient buffer pool for storing messages	Y	Y	Y	Y	Bytes	Y	14
Lock information	Lock acquisition (wait time)	Y	Y	Y	Y	Milli-seconds	N	21
	Lock wait (wait queue length)	Y	Y	Y	Y	Requests	N	22
	Deadlocks	Y	N	N	N	Number of events	N	23
DAM information	read (input length)	Y	Y	Y	Y	Bytes	Y	31
	read errors	Y	N	N	N	Number of events	Y	32
	write (output length)	Y	Y	Y	Y	Bytes	Y	33
	write errors	Y	N	N	N	Number of events	Y	34
	Use of update buffer (update size)	Y	Y	Y	Y	Bytes	Y	35
	First DAM API issued in a transaction branch (total number of transaction branches)	#2	Y	Y	Y	Transaction branches	Y	36

E. Details of Statistics

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Shared memory cache block allocation request	Y	N	N	N	Number of events	Y	38
	Shared memory allocation (shared memory utilization rate) ^{#3}	Y	Y	Y	Y	%	N	39
Process information	UAP abnormal terminations	Y	N	N	N	Number of events	N	41
	System server abnormal terminations	Y	N	N	N	Number of events	N	42
	Process generations (total number of processes)	#2	Y	Y	Y	Processes	N	43
TAM information	TAM file real updating (number of written bytes)	Y	Y	Y	Y	Bytes	N	51
	Commits, rollbacks (number of record referencing operations)	Y	Y	Y	Y	Number of events	Y	55

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Commits, rollbacks (number of record updating operations)	Y	Y	Y	Y	Number of events	Y	56
Transaction information	Commits	Y	N	N	N	Number of events	Y	61
	Rollbacks	Y	N	N	N	Number of events	Y	62
Name information	Cache hits	Y	N	N	N	Number of events	Y	71
	Local hits	Y	N	N	N	Number of events	Y	72
	lookup	Y	N	N	N	Number of events	Y	73
Checkpoint dump information	Collection of checkpoint dump (collection interval)	Y	Y	Y	Y	Milli-seconds	N	81
	Validation of checkpoint dump (collection time)	Y	Y	Y	Y	Milli-seconds	N	82
Message queue information	Read message (input length)	Y	Y	Y	Y	Bytes	N	91
	Write message (output length)	Y	Y	Y	Y	Bytes	N	92
	Read errors	Y	N	N	N	Number of events	N	93

E. Details of Statistics

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Write errors	Y	N	N	N	Number of events	N	94
	Waits for an available buffer	Y	N	N	N	Number of events	N	95
	Real reads	Y	N	N	N	Number of events	N	96
	Real writes	Y	N	N	N	Number of events	N	97
	Deferred writing (real write count)	Y	Y	Y	Y	Times	N	99
	Deferred writing for each physical file (number of records that were written in deferred mode)	Y	Y	Y	Y	Number of records	N	151
	Deferred writing for each physical file (percentage of messages that were written in deferred mode)	Y	Y	Y	Y	%	N	152
Journal information	Buffer full occurrences	Y	N	N	N	Number of events	Y	102
	Waits for an available buffer	Y	N	N	N	Number of events	Y	103

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Journal output (block length)	Y	Y	Y	Y	Bytes	N	104
	Journal output (data length of non-bus section) ^{#4}	Y	Y	Y	Y	Bytes	N	105
	Input/output waits (Number of waiting buffer faces)	Y	Y	Y	Y	Number of events	N	107
	write	Y	N	N	N	Number of events	N	108
	Write errors	Y	N	N	N	Number of events	N	109
	Swap (swap time)	Y	Y	Y	Y	Micro-seconds	N	110
	Journal input (data length)	Y	Y	Y	Y	Bytes	N	111
	read	Y	N	N	N	Number of events	N	113
	Read errors	Y	N	N	N	Number of events	N	114
MCF information	Message inputs to memory queue	Y	N	N	N	Number of events	N	121
	Message inputs to disk queue	Y	N	N	N	Number of events	N	122

E. Details of Statistics

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Message outputs from memory queue	Y	N	N	N	Number of events	N	123
	Message outputs from disk queue	Y	N	N	N	Number of events	N	124
	Disconnects due to a connection error	Y	N	N	N	Number of events	N	125
Shared memory control information	Size of static shared memory used	Y	Y	Y	Y	Bytes	Y	131
	Maximum static shared memory size requirement	Y	Y	Y	Y	Bytes	Y	132
	Size of dynamic shared memory used	Y	Y	Y	Y	Bytes	Y	133
	Maximum dynamic shared memory size requirement	Y	Y	Y	Y	Bytes	Y	134
MQA service information	Get message (input length)	Y	Y	Y	Y	Bytes	Y	161
	Put message (output length)	Y	Y	Y	Y	Bytes	Y	162
	Read errors	Y	--	--	--	Number of events	Y	163

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	Write errors	Y	--	--	--	Number of events	Y	164
	Waits for an available buffer	Y	--	--	--	Number of events	Y	165
	Real reads	Y	--	--	--	Number of events	Y	166
	Real writes	Y	--	--	--	Number of events	Y	167
	Write request time	Y	Y	Y	Y	Micro-seconds	Y	169
	Number of messages activated at one trigger	Y	Y	Y	Y	Number of events	Y	170
	Time of wait for message transfer start	Y	Y	Y	Y	Milli-seconds	Y	171
	Time of wait for message arrival	Y	Y	Y	Y	Milli-seconds	Y	172
IST service information	Input length	Y	Y	Y	Y	Bytes	Y	191
	Output length	Y	Y	Y	Y	Bytes	Y	192
	Number of updates from other nodes	Y	N	N	N	Number of events	N	193
	Number of updates to other nodes	Y	N	N	N	Number of events	N	194
XATMI service information	XATMI call response time	Y	Y	Y	Y	Micro-seconds	Y	211

Statistics type	Edited contents					Unit	Editing in units of user servers	ID for edition by dcreport# 1
	Event (obtained value)	Number	Edited values					
			Avg	Max	Min			
	XATMI user service execution time	Y	Y	Y	Y	Micro-seconds	Y	212
	XATMI service error count	Y	N	N	N	Number of events	Y	213

Legend:

Number: Number of occurrences

Avg: Average

Max: Maximum

Min: Minimum

Y: Can be output to a system journal file or can be edited in units of user servers.

N: Cannot be output to a system journal file or cannot be edited in units of user servers.

--: Neither edited nor output.

#1

Argument to be specified when the `dcreport` command is executed.

#2

Output when the `dcreport` command is executed.

#3

These statistics are not output if you do not update the DAM file with the deferred update specification. Note that these statistics are output once for every 10 deferred updates.

#4

This is OpenTP1 internal information.

E.2 Statistics about transaction

The following table shows the details of statistics about transaction.

Table E-3: Details of statistics about transaction

Item	Details
Transaction branch execution time	The real time between starting the service as a transaction and completing synchronization point processing. <Number of occurrences> Not collected.
Transaction branch execution time for synchronization point processing	The actual length of time spent for synchronization point processing after termination of the service executed as a transaction (included in the branch execution time above) <Number of occurrences> Not collected.
Transaction branch determination	The method of determining transaction branches which the service executed <Number of occurrences> The number of transaction branches determined to be committed, and the number of those determined to be rolled back
Transaction branch and child-branch determinations	The method of determining transaction branches including those divided from the service (child-branches) <Number of occurrences> The numbers of transaction branches determined to be committed, those determined to be rolled back, and those determined by the command.
Process type for transaction branch determinations	The type of the process that determined transactions <Number of occurrences> The number of transactions determined by the user server process and the number of those determined by the transaction recovery process

E.3 Statistics about response

(1) Details about response statistics

The following table shows the details of statistics about response.

Table E-4: Details of statistics about response

Item	Details
RPC type	Number of RPCs issued to the applicable service from the node that outputs the journal, for each type

Item	Details
Response time ^{#1}	Length of time between the point when <code>dc_rpc_call</code> is issued and the point when the response text is received. However, the length of time differs depending on the flag as shown below: DCNOFLAGS (synchronous-response RPC) Length of time between the point when <code>dc_rpc_call</code> is issued and the point when the response text is received DCRPC_CHAINED (chained RPC) Length of time between the point when <code>dc_rpc_call</code> is issued and the point when the response text is received DCRPC_NOWAIT (asynchronous-response RPC) Length of time between the point when <code>dc_rpc_call</code> is issued and the point when the response is written in the out area DCRPC_NOREPLY (non-response RPC) No response is acquired.
Service execution time ^{#2}	Length of time required to execute a service function. This information is not acquired for non-response RPCs.
Service wait time ^{#2}	Length of time between the point at which a request is collected in the schedule queue and the point at which the request is fetched from the schedule queue. This information is not acquired for non-response RPCs.

#1

Response time refers to the following:

$$\text{Response time} = \text{service-execution-time} + \text{service-wait-time} + \text{communication-time}$$

Communication time is not included in *service-execution-time* and *service-wait-time*.

#2

The service execution time and the service wait time of the MHP are not acquired.

In the response statistics, only the number of successful RPCs is counted. The number of failed RPCs due to timeouts or other reasons is not included.

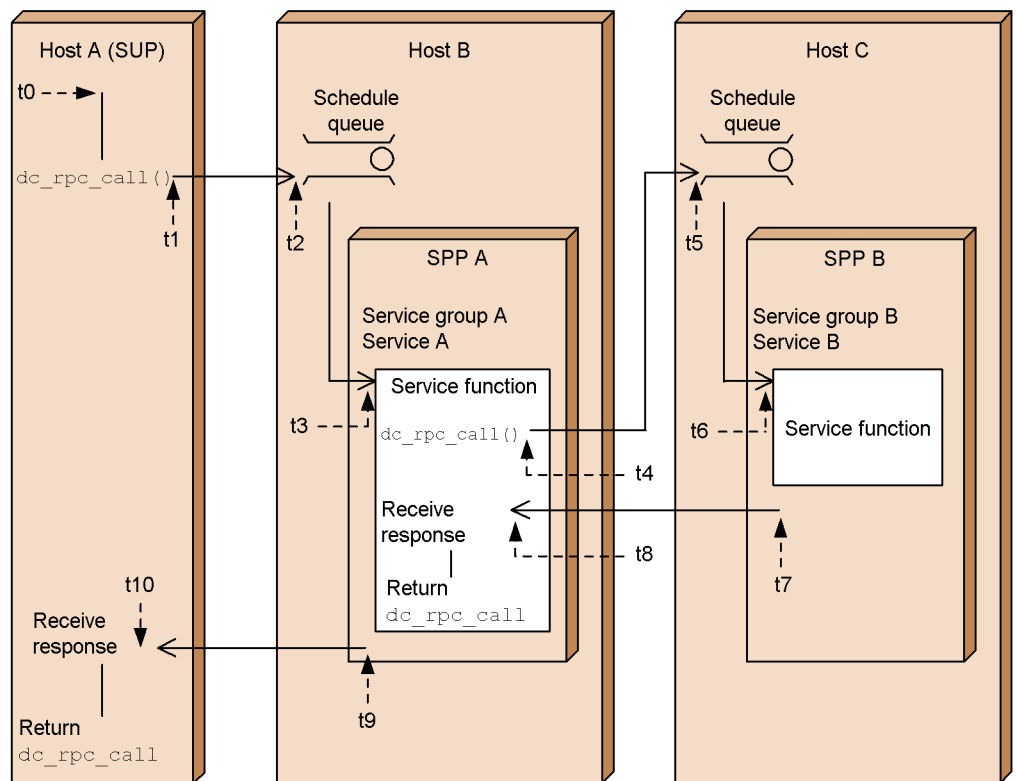
(2) Output example of response statistics

Response statistics are output differently depending on the execution type of each service. The following shows an example output of statistics for each type of execution.

(a) When the target service exists in a node different from the node that requests the service

The following figure shows a case when the target service exists in a node different from the node that requests the service.

Figure E-1: When the target service exists in a node different from the node that requests the service



Legend:

t_0 : Service start point

t_1 to t_{10} : Amounts of time elapsed from t_0 to the points indicated by the applicable arrows (units: microseconds)

When you edit the journals acquired in the hosts in Figure D-1, the response statistics are output as follows for each host:

- Edited journal of host A

```

jnlstts vv-rr          ***** Response Time Statistics *****          Page: 1
File name              : aaa
File creation date/time : XX-XX-XX XX:XX:XX
Output specification date/time : **_*_* **_*_*_* ~ **_*_*_* **_*_*_*

```

```

-----
Node identifier       = XXXX
Service group name   = A

```

```

***** Editing period: XX-XX-XX XX:XX to XX-XX-XX XX:XX *****

```

```

<Service name A>

```

```

RPC type
Synchronous-response      : 1 (events)
Non-response              : 0 (events)
Asynchronous-response     : 0 (events)
Chained                   : 0 (events)
Total                    : 1 (events)

```

```

-----
Event          Event count  Maximum  Minimum  Average  (unit)
(acquired value)
Response time   1          #1       #1       #1 (microseconds)
Service execution time 1          #2       #2       #2 (microseconds)
Service wait time 1          #3       #3       #3 (microseconds)
-----

```

#1

Values of t10 to t1 in Figure D-1

#2

Values of t9 to t3 in Figure D-1

#3

Values of t3 and t2 in Figure D-1

The details about each output item are as follows:

Number of events for each RPC type

Host A issued a synchronous-response RPC to service A in service group A once.

Response time

Length of time between the point when the SUP on host A issues `dc_rpc_call` and the point when the SUP receives a response

Execution of user service (service execution time)

Length of time between the point when service A in service group A starts processing and the point when service A sends a response. However, the communication time for sending the response is not included.

Service wait (service wait time)

Length of time between the point when the SUP on host A enters a request in the schedule queue and the point when the request is fetched from the schedule queue

- Edited journal of host B

```

jn1stts vv-rr          ***** Response Time Statistics *****          Page:  1
File name              : aaa
File creation date/time : XX-XX-XX XX:XX:XX
Output specification date/time : **_*_* **.*.* to **_*_* **.*.*
-----
Node identifier        = XXXX
Service group name    = A
-----
***** Editing period: XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
<Service name A>
RPC type
  Synchronous-response      :      0 (events)
  Non-response              :      0 (events)
  Asynchronous-response     :      0 (events)
  Chained                   :      0 (events)
  Total                     :      0 (events)
-----
Event                    Event count    Maximum   Minimum   Average   (unit)
(acquired value)
Response time            0             #1       #1       #1 (microseconds)
Service execution time   1             #2       #2       #2 (microseconds)
Service wait time        1             #3       #3       #3 (microseconds)
-----
Node identifier        = XXXX
Service group name    = B
-----
***** Editing period: XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
<Service name: B>
RPC type
  Synchronous-response      :      1 (events)
  Non-response              :      0 (events)
  Asynchronous-response     :      0 (events)
  Chained                   :      0 (events)
  Total                     :      1 (events)
-----
Event                    Event count    Maximum   Minimum   Average   (unit)
(acquired value)
Response time            1             #4       #4       #4 (microseconds)
Service execution time   1             #5       #5       #5 (microseconds)
Service wait time        1             #6       #6       #6 (microseconds)
-----

```

#1

0

#2

Values of t9 to t3 in Figure D-1

#3

Values of t3 and t2 in Figure D-1

#4

Values of t8 to t4 in Figure D-1

#5

Values of t7 and t6 in Figure D-1

#6

Values of t6 and t5 in Figure D-1

The details about each output item are as follows:

Service A in service group A

Number of events for each RPC type

Host B has never called service A in service group A.

Response time

Length of time between the point when the SUP on host A issues `dc_rpc_call` and the point when the SUP receives a response. This value cannot be acquired since the journal of host B is edited. Therefore, 0 is written.

Execution of user service (service execution time)

Length of time between the point when service A in service group A starts processing and the point when service A sends a response. However, the communication time for sending the response is not included.

Service wait (service wait time)

Length of time between the point when the SUP on host A enters a request in the schedule queue and the point when the request is fetched from the schedule queue

Service B in service group B

Number of events for each RPC type

Host B issued a synchronous-response RPC to service B in service group B once.

Response time

Length of time between the point when SPP A on host B issues `dc_rpc_call` and the point when SPP A receives a response

Execution of user service (service execution time)

Length of time between the point when service B in service group B starts processing and the point when service B sends a response. However, the communication time for sending the response is not included.

Service wait (service wait time)

Length of time between the point when SPP A on host B enters a request in the schedule queue and the point when the request is fetched from the schedule queue

- Edited journal of host C

```

jn1stts vv-rr          ***** Response Time Statistics *****          Page: 1
File name      : aaa
File creation date/time : XX-XX-XX XX:XX:XX
Output specification date/time : **_**_** **.*.*.* to **_**_** **.*.*.*
-----
Node identifier      = XXXX
Service group name  = B
-----
***** Editing period: XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
<Service name B>
  RPC type
  Synchronous-response      : 0 (events)
  Non-response              : 0 (events)
  Asynchronous-response     : 0 (events)
  Chained                   : 0 (events)
  Total                     : 0 (events)
-----
Event          Event count  Maximum  Minimum  Average  (unit)
(acquired value)      value      value      value
Response time          0         #1         #1         #1 (microseconds)
Service execution time 1         #2         #2         #2 (microseconds)
Service wait time      1         #3         #3         #3 (microseconds)
-----

```

#1

0

#2

Values of t9 to t3 in Figure D-1

#3

Values of t3 and t2 in Figure D-1

The details about each output item are as follows:

Number of events for each RPC type

Host C has never called service B in service group B.

Response time

Length of time between the point when SPP A on host B issues `dc_rpc_call` and the point when SPP A receives a response. However, this value cannot be acquired since the journal of host C is edited. Therefore, 0 is written.

Execution of user service (service execution time)

Length of time between the point when service B in service group B starts processing and the point when service B sends a response. However, the communication time for sending the response is not included.

Service wait (service wait time)

Length of time between the point when SPP A on host B enters a request in the schedule queue and the point when the request is fetched from the schedule queue

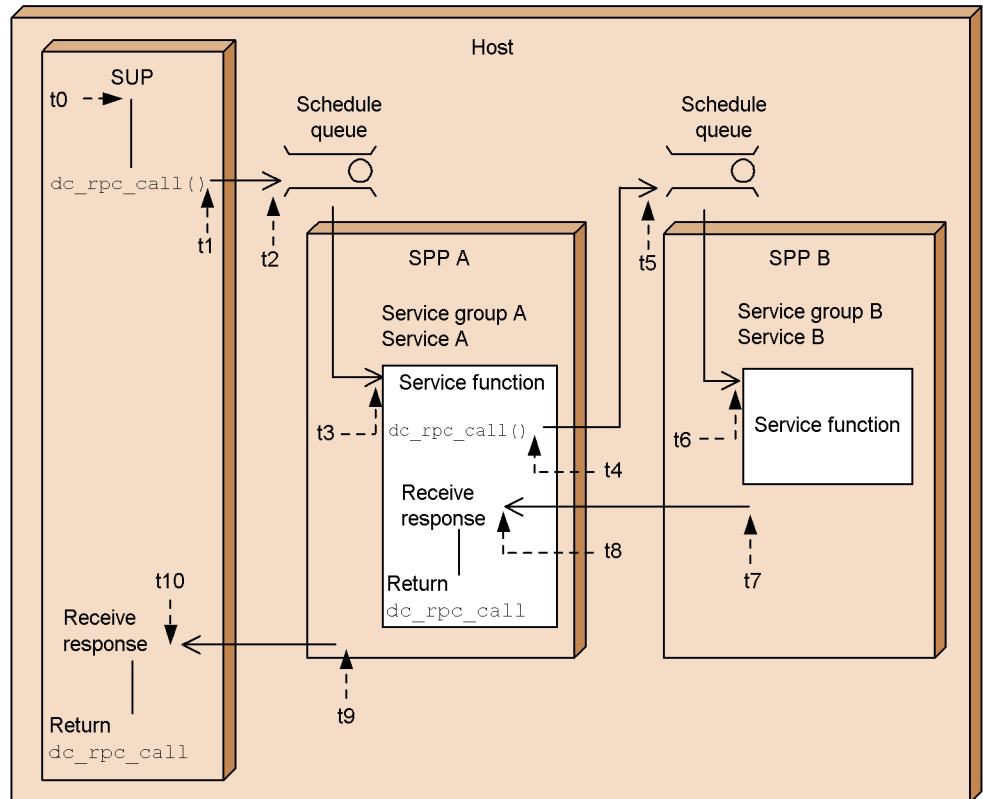
The relationship between the journals of the hosts and the statistics acquired from the journals is as follows:

- When you edit the journal of host A, you can acquire the statistics on SPP A operating in host B.
- When you edit the journal of host B, you can acquire the statistics on SPP B operating in host C.
- When you edit the journal of host B, you cannot acquire the statistics on SPP A.
- When you edit the journal of host C, you cannot acquire the statistics on SPP B.

(b) When a service is executed in the local node

The following figure shows a case of executing a service in the local node.

Figure E-2: When a service is executed in the local node



Legend:

t0: Service start point

t1 to t10: Amounts of time elapsed from t0 to the points indicated by the applicable arrows (units: microseconds)

When you edit the journal that is acquired in the host in Figure D-2, the response statistics are output as follows:

E. Details of Statistics

```

jnlstts vv-rr      ***** Response Time Statistical Information ***** Page: 1

File name: aaa
File creation date and time: **--** **:**
Output specification date and time from XX-XX-XX XX:XX:XX to XX-XX-XX XX:XX:XX
-----
Node identifier    = XXXX
Service group name = A
-----
      ***** Editing object time from XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
< service name : A >
RPC type
  Reply           :           1 (events)
  No-reply        :           0 (events)
  No-wait         :           0 (events)
  Chained         :           0 (events)
  Total          :           1 (events)
-----
Event (collected value)  Number of  Maximum  Minimum  Average
                          events          value    value    value
                          (times)
Response time             1            #1       #1       #1
                          (microsec.)
Execution of user        2            #2       #2       #2
service (execution time) (microsec.)
Service wait             2            #3       #3       #3
                          (microsec.)
-----
Node identifier    = XXXX
Service group name = B
-----
      ***** Editing object time from XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
< service name : B >
RPC type
  Reply           :           1 (events)
  No-reply        :           0 (events)
  No-wait         :           0 (events)
  Chained         :           0 (events)
  Total          :           1 (events)
-----
Event (collected value)  Number of  Maximum  Minimum  Average
                          events          value    value    value
                          (times)
Response time             1            #4       #4       #4
                          (microsec.)
Execution of user        2            #5       #5       #5
service (execution time) (microsec.)
Service wait             2            #6       #6       #6
                          (microsec.)
-----

```

#1

Values of t10 to t1 in Figure D-2

#2

Values of t9 to t3 in Figure D-2

#3

Values of t3 and t2 in Figure D-2

#4

Values of t8 to t4 in Figure D-2

#5

Values of t7 and t6 in Figure D-2

#6

Values of t6 and t5 in Figure D-2

The details about each output item are as follows:

Service A in service group A

Number of events for each RPC type

The local node issued a synchronous-response RPC to service A in service group A once.

Response time

Length of time between the point when the SUP issues `dc_rpc_call` and the point when the SUP receives a response

Execution of user service (service execution time)

Length of time between the point when service A in service group A starts processing and the point when service A sends a response. However, the communication time for sending the response is not included.

The number of events is 2 since `rpc_response_statistics=Y` is specified for both the SUP and SPP A.

The average value is determined by dividing (maximum value + minimum value) by 2.

Service wait (service wait time)

Length of time between the point when the SUP enters a request in the schedule queue and the point when the request is fetched from the schedule queue

The number of events is 2 since `rpc_response_statistics=Y` is specified for both the SUP and SPP A.

The average value is determined by dividing (maximum value + minimum value) by 2.

Service B in service group B

Number of events for each RPC event

The local node issued a synchronous-response RPC to service B in service group B once.

Response time

Length of time between the point when SPP A issues `dc_rpc_call` and the point when SPP A receives a response

Execution of user service (service execution time)

Length of time between the point when service B in service group B starts processing and the point when service B sends a response. However, the communication time for sending the response is not included.

The number of events is 2 since `rpc_response_statistics=Y` is specified for both SPP A and SPP B.

The average value is determined by dividing (maximum value + minimum value) by 2.

Service wait (service wait time)

Length of time between the point when SPP A enters a request in the schedule queue and the point when the request is fetched from the schedule queue

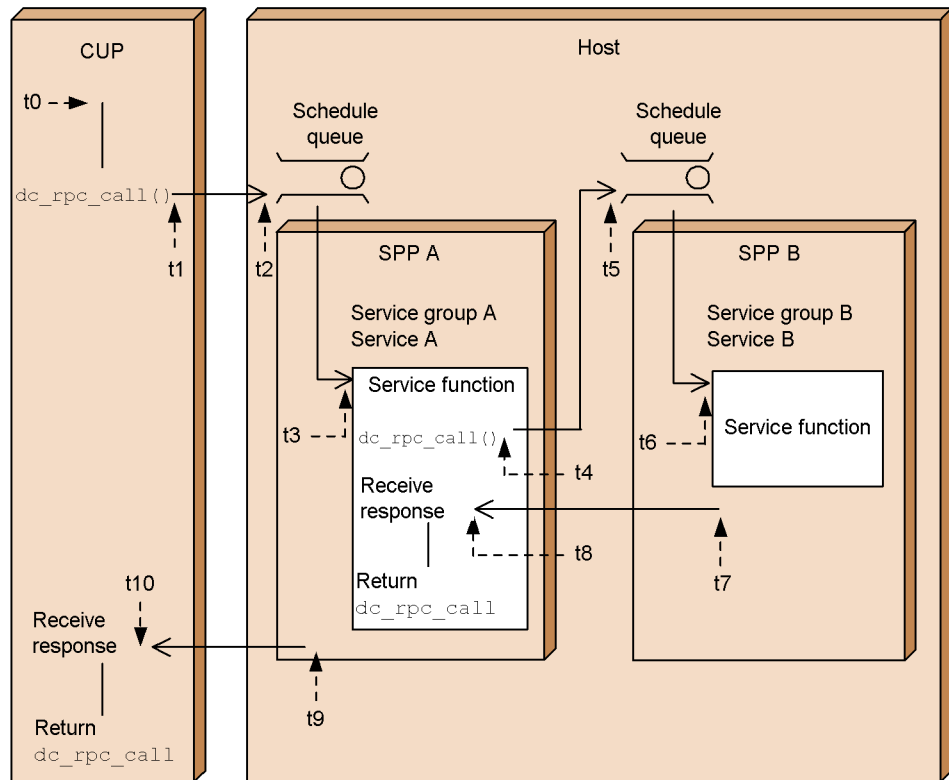
The number of events is 2 since `rpc_response_statistics=Y` is specified for both SPP A and SPP B.

The average value is determined by dividing (maximum value + minimum value) by 2.

(c) When a service is executed from a client

The following figure shows a case when a service is executed from a client.

Figure E-3: When a service is executed from a client



Legend:

t_0 : Service start point

t_1 to t_{10} : Amounts of time elapsed from t_0 to the points indicated by the applicable arrows (units: microseconds)

When you edit the journal acquired in the host in Figure D-3, the response statistics are output as follows:

E. Details of Statistics

```

jnlstts vv-rr      ***** Response Time Statistical Information ***** Page: 1

File name: aaa
File creation date and time: **--** **:**
Output specification date and time from XX-XX-XX XX:XX:XX to XX-XX-XX XX:XX:XX
-----
Node identifier    = XXXX
Service group name = A
-----
***** Editing object time from XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
< service name : A >
RPC type
  Reply           :           0 (events)
  No-reply        :           0 (events)
  No-wait         :           0 (events)
  Chained         :           0 (events)
  Total          :           0 (events)
-----
Event (collected value)   Number of   Maximum   Minimum   Average
                           events         value     value     value
                           (times)
Response time              1           #1        #1        #1
                           (microsec.)
Execution of user         2           #2        #2        #2
service (execution time) (microsec.)
Service wait              2           #3        #3        #3
                           (microsec.)
-----
Node identifier    = XXXX
Service group name = B
-----
***** Editing object time from XX-XX-XX XX:XX to XX-XX-XX XX:XX *****
< service name : B >
RPC type
  Reply           :           1 (events)
  No-reply        :           0 (events)
  No-wait         :           0 (events)
  Chained         :           0 (events)
  Total          :           1 (events)
-----
Event (collected value)   Number of   Maximum   Minimum   Average
                           events         value     value     value
                           (times)
Response time              1           #4        #4        #4
                           (microsec.)
Execution of user         2           #5        #5        #5
service (execution time) (microsec.)
Service wait              2           #6        #6        #6
                           (microsec.)
-----

```

#1

0

#2

Values of t9 to t3 in Figure D-3

#3

Values of t3 and t2 in Figure D-3

#4

Values of t8 to t4 in Figure D-3

#5

Values of t7 and t6 in Figure D-3

#6

Values of t6 and t5 in Figure D-3

The details about each output item are as follows:

Service A in service group A

Number of events for each RPC type

The local node has never called service A in service group A. Even if the client is in the same node, calls from the client cannot be included in the count since clients cannot acquire response statistics.

Response time

Length of time between the point when the CUP issues `dc_rpc_call` and the point when the CUP receives a response. However, this value cannot be acquired since clients cannot acquire response statistics. Therefore, 0 is written.

Execution of user service (service execution time)

Length of time between the point when service A in service group A starts processing and the point when service A sends a response. However, the communication time for sending the response is not included.

Service wait (service wait time)

Length of time between the point when the CUP enters a request in the schedule queue and the point when the request is fetched from the schedule queue

Service B in service group B

Number of events for each RPC type

The local node issued a synchronous-response RPC to service B in service group B once.

Response time

Length of time between the point when SPP A issues `dc_rpc_call` and the

point when SPP A receives a response

Execution of user service (service execution time)

Length of time between the point when service B in service group B starts processing and the point when service B sends a response. However, the communication time for sending the response is not included.

The number of events is 2 since `rpc_response_statistics=Y` is specified for both SPP A and SPP B.

The average value is determined by dividing (maximum value + minimum value) by 2.

Service wait (service wait time)

Length of time between the point when SPP A enters a request in the schedule queue and the point when the request is fetched from the schedule queue

The number of events is 2 since `rpc_response_statistics=Y` is specified for both SPP A and SPP B.

The average value is determined by dividing (maximum value + minimum value) by 2.

E.4 Statistics about communication delay

The following table shows the details of statistics about communication delay.

Table E-5: Details of statistics about communication delay

Item	Details
Communication delay	The value of the response time minus server processing time. This means the time required for communication (sending a request + sending a response). <Number of occurrences> Number of times the synchronous RPC (including chained RPC) was issued

E.5 Real-time statistics

(1) Details about real-time statistics

The following table shows the details about real-time statistics.

Table E-6: Details about real-time statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
CPD (checkpoint dump information)	Collection of cpd	Number of items	Number of events that caused the collection of checkpoint dumps
		Length of time	Interval between the previous checkpoint and the current checkpoint (interval of events that cause checkpoint collection). Since a checkpoint is collected after the output of a preset number of journal blocks, the frequency of collecting checkpoint dumps is expressed as a time value here.
	Validation of cpd	Number of items	Number of events that caused the collection of checkpoint dumps
		Length of time	Length of time from an event that collects a checkpoint dump and the startup of collection processing in each system server to the completion of validation of the checkpoint dump
JNL (journal information)	Buffer full occurrences	Number of items	Buffering is disabled because the free area in the buffer was too small when an attempt was made to store a journal record in the current buffer. There are three journal buffers in the system, of which one is the current buffer.
	Waits for an available buffer	Number of items	The system is waiting for a free buffer. Journals cannot be buffered because all three journal buffers are full or journals are being output.
	Journal output	Number of times	Number of times journal blocks are output
		Data length	Data length in a journal block. You can make an assumption about the probable number of journals that are generated during a unit of time based on this value.
	I/O waits	Number of times	Number of times journal output is completed

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Number of buffers x 100	Number of buffers waiting for output when journal output is completed. The number of buffers is multiplied by 100 to prevent truncation of the decimal part of the average value.
	Write	Number of times	Number of outputs, including journal outputs and outputs of file management information when journal swapping occurs.
		Length of time	Length of time required for writing journal files.
	Swap	Number of times	Number of times journal files are swapped
		Length of time	Length of time required for swapping journal files
	Journal input	Number of times	Number of times data is input from a journal file
		Data length	Length of data input from a journal file
	Read	Number of times	Number of inputs, including journal inputs and inputs of file management information when journal swapping occurs.
		Length of time	Length of time required for reading journal files.
	LCK (lock information)	Lock acquisition	Number of items
Length of time			Length of time from the occurrence of a lock wait and entry into the wait status to release of the wait status. The lock described here indicates the internal locks set by the lock service issued by the <code>dc_lck_get</code> function, DAM, TAM, and MQA.
Lock wait		Number of items	Number of waits for a lock
		Queue length	Length of the wait queue when lock waits are generated

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Deadlock	Number of items	Number of deadlocks that occurred. The deadlocks detected by the system server are not acquired in the real-time statistics of each service.
NAM (name information)	Global Cache hits	Number of times	Number of times the information about the service group stored in the global cache area is referenced during a search for information about the service group. Note that the number of references may not match the number of communications. ^{#1}
	Local Cache hits	Number of times	Number of times the information about the service group stored in the local cache area is referenced during a search for information about the service group. Note that the number of references may not match the number of communications. ^{#1}
	Lookup	Number of times	Number of times the information about the service group stored in the global cache area or the local cache area is referenced when a UAP executes an RPC. When the information about the applicable service group is not stored in the global cache area or the local cache area, global cache hits or local cache hits are not counted. However, the number of searches for service information is counted. As a result, the number of global cache hits and the number of local cache hits may not match the number of searches for service information.
OSL (shared memory management information)	Common static memory (acquired)	Number of times	Number of times the function for allocating or releasing static shared memory is issued plus 1
		Memory block size	Total size of static shared memory blocks used after static shared memory blocks are allocated or released
	Common static memory (pool)	Number of times	Number of times the pool size is updated plus 1

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Memory block size	<p>Maximum size of static shared memory blocks in use (size up to the highest memory block currently in use). This information is acquired if the previous pool size is updated after the allocation or release of blocks.</p> <p>The pool size is the total size of the shared memory in use, including empty areas in the pool created by releasing the allocated shared memory.</p> <p>Accordingly, specify a value larger than this size in the <code>static_shmpool_size</code> operand in the system environment definition.</p> <p>This information is the total size from the beginning of acquisition, not the size acquired during the preset length of time for acquisition specified in the <code>rts_trcput_interval</code> operand in the real-time statistics service definition.</p>
	Common dynamic memory (acquired)	Number of times	Number of times the function for allocating or releasing the dynamic shared memory is issued plus 1
		Memory block size	Total size of dynamic shared memory blocks in use after allocating or releasing the dynamic shared memory blocks
	Common dynamic memory (pool)	Number of times	Number of times the pool size is updated plus 1

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Memory block size	<p>Maximum size of dynamic shared memory blocks in use (size up to the lowest memory block currently in use). This information is acquired if the previous pool size is updated after allocation or release of blocks. Specify a value larger than this size in the <code>dynamic_shmpool_size</code> operand in the system environment definition.</p> <p>This information is the total size from the beginning of acquisition, not the size acquired during the preset length of time for acquisition specified in the <code>rts_trcput_interval</code> operand in the real-time statistics service definition.</p>
PRC (process information)	Process generations	Number of times	Number of processes started by the OpenTP1 system. This number includes the number of processes of non-resident servers. Because only the number of processes started by the process service is counted, the process service itself and the processes that are not started by the process service are not included.
	UAP abnormal terminations	Number of times	Number of times UAP processes terminate abnormally for each user server. This number includes cases when a user server goes down but does not include processes that terminate abnormally after <code>dc_rpc_close</code> . For details about the relationship between process termination and item counts, see Table E-7.

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	System server abnormal terminations	Number of times	<p>Number of times the system server terminates abnormally. This information may not be acquired if the system server terminates abnormally and the OpenTP1 system goes down.</p> <p>Note</p> <p>If OpenTP1-provided servers that operate as SPPs (such as TP1/EE, the RAP-processing server, and RTSSUP) go down, those terminations are included in UAP abnormal terminations, not System server abnormal terminations.</p> <p>For details about the relationship between process termination and item counts, see Table E-7.</p>
	Process terminations	Number of times	<p>Number of times processes terminate, including normally terminated processes, processes that are stopped by the <code>dcsvstop</code> command with the <code>-f</code> option or by the <code>prckill</code> command, and by processes that are forcibly stopped due to an actual time-monitoring timeout error. This information does not include the counts of UAP abnormal terminations and System server abnormal terminations. For details about the relationship between process termination and item counts, see Table E-7.</p>
	Number of processes	Number of times	<p>Number of observations (monitoring) during a preset length of time. Monitoring is usually performed every 60 seconds. Monitoring is performed every 2 seconds when the <code>dcstart</code>, <code>dcstop</code>, <code>dcsvstart</code>, <code>dcsvstop</code>, <code>eesvstart</code>, or <code>eesvstop</code> command is executed.</p>

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Number of processes	Number of system services and UAP processes started in the OpenTP1 system. Because this information includes only the number of processes started by the process service, the process service itself and the processes that are not started by the process service are not included in the count. This information is the total number from the beginning of acquisition, not the number acquired during the preset length of time for acquisition specified in the <code>rts_trcput_interval</code> operand in the real-time statistics service definition.
QUE (message queue information)	Read message	Number of times	Number of times messages are read from the queue file
		Message length	Length of a message read from the queue file
	Write message	Number of times	Number of times messages are written into the queue file
		Message length	Length of a message written into the queue file
	Read errors	Number of items	Number of cases where errors occurred during input from a physical file (abnormal-condition cases)
	Write errors	Number of items	Number of cases where errors occurred during output to a physical file (abnormal-condition cases)
	Waits for an available buffer	Number of items	Number of cases where required I/O buffers became insufficient due to congestion of requests for the target queue file. Use this value to check the number of I/O buffers in the message queue service definition.
	Real reads	Number of times	Number of inputs generated from a physical file when the I/O buffer contains no messages. To decrease this value, increase the number of I/O buffers in the message queue service definition.

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Real writes	Number of times	Number of outputs generated for a physical file. When you compare this value and the value for <code>Write message</code> , you can determine how many write requests were input and output at one time. To decrease the number of inputs and outputs, increase the number of I/O buffers in the message queue service definition. When you use the deferred write facility, the number of outputs to a physical file is included in the count of <code>Number of delay writings</code> . Therefore, Hitachi recommends that you set 0 for <code>Real writes</code> .
	Number of delay writings	Number of times	Number of deferred write requests that are generated when the deferred write facility is used
		Number of times	Number of writes (<code>Real writes</code>) to a physical file per deferred write request when the deferred write facility is used
	Number of delay writing records	Number of times	Number of times deferred write requests are generated for each physical file when the deferred write facility is used
		Number of records	Number of records whose writing is deferred. Although this information is acquired for each physical file, the information is displayed for all the physical files and not for each physical file. Use this value to tune the number of I/O buffers in the message queue service definition.
	Delay writing record rate	Number of times	Number of times deferred write requests are generated for each physical file when the deferred write facility is used

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Percentage of messages	Percentage of all messages whose writing is deferred. Although this information is acquired for each physical file, the information is displayed for all the physical files and not for each physical file. Use this value to tune the number of I/O buffers in the message queue service definition.
RPC (RPC information)	RPC calls	Number of times	Number of times synchronous-response <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are called. This information is acquired when synchronous-response <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are successful.
		Length of time	Length of time between sending a request from the client to the server in synchronous-response <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> , and receiving a response from the server. This value is acquired when synchronous-response <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are successful. The time includes the length of time for retries if the transmission of a request to the server in synchronous-response <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> fails and the transmission of the request to the server is retried.
	RPC calls(chained)	Number of times	Number of times chained <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are called between the start and end of a chained RPC. This information is acquired when chained <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are successful.
		Length of time	Length of time between sending a request from the client to the server in a chained <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> , and receiving a response from the server between the start and end of a chained RPC. This value is acquired when a chained <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> is successful.

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Execution of user service	Number of times	Number of times service functions requested by <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> are executed. When the service retry facility is used to retry service functions, the number of times the service functions are retried is not included.
		Length of time	Length of time between the start of the execution of service functions requested by <code>dc_rpc_call</code> and <code>dc_rpc_call_to</code> and the point at which the service functions issue <code>return</code> values. This value is obtained in <code>dc_rpc_mainloop</code> . When the service retry facility is used to retry service functions, the length of time required to retry the service functions is included.
	RPC overtimes	Number of items	Number of timeout errors that are generated during the response wait processing of the RPC
SCD (schedule information)	Schedule Waits	Number of times	Number of times service requests (<code>dc_rpc_call</code>) are issued by the client to the applicable user server

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Queue length	<p>Number of service requests remaining in the schedule queue of the applicable user server (SPP).</p> <p>When this value is too large, increase the value specified in the <code>parallel_count</code> operand in the user service definition.</p> <p>When non-resident processes exist, decrease the value specified in the <code>balance_count</code> operand so that the non-resident processes can be easily started.</p> <p>When this value is too small, if necessary, decrease the number of resident processes specified in the <code>parallel_count</code> operand in the user service definition.</p> <p>When non-resident processes exist, increase the value specified for the <code>balance_count</code> operand as necessary so that the non-resident processes are not easily started.</p> <p>When the difference between the average and the maximum value is too great, adjust the ratio of the resident process count and the non-resident process count specified in the <code>parallel_count</code> operand and the maximum number of processes.</p>
	Schedule	Number of times	Number of times service requests (<code>dc_rpc_call</code>) are issued by the client to the applicable user server
		Message length	Length of the service request message received by the applicable user server (SPP). The message length includes the control information (512 bytes) added by the system. ^{#2}
	Size of using buffer	Number of times	Number of times service requests (<code>dc_rpc_call</code>) are issued by the client to the applicable user server
		Buffer pool size	Size of the buffer pool being used to store messages for the applicable user server (SPP). However, if a shared buffer pool is used, the size of the shared buffer pool is acquired.

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Message size in case of lack of buffer	Number of times	Of the service requests (<code>dc_rpc_call</code>) that are issued by the client to the applicable user server, the number of service requests that cannot be scheduled due to the insufficient size of the buffer pool for storing messages
		Message length	Of the service requests to the applicable user server (SPP), the length of the service request message that cannot be scheduled due to the insufficient size of the buffer pool for storing messages
	Schedule Stay	Number of times	Number of times service requests are fetched from the schedule queue
		Length of time	Length of time between the storage in the schedule queue of service requests (<code>dc_rpc_call</code>) that are sent to the applicable user server and fetching of those service requests
	Schedule Waits of service	Number of times	Number of times clients have issued service requests (<code>dc_rpc_call</code>) to the applicable user server. This information is acquired only if the <code>scdsvcdef</code> definition command is specified in the user service definition.
		Queue length	Number of service requests that have been in the schedule queue of the applicable user server (SPP). This information is acquired only if the <code>scdsvcdef</code> definition command is specified in the user service definition.
	Size of using buffer of service	Number of times	Number of times clients have issued service requests (<code>dc_rpc_call</code>) to the applicable user server. This information is acquired only if the <code>scdsvcdef</code> definition command is specified in the user service definition.

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Buffer pool size	Size of the buffer pool being used to store messages for the applicable user server (SPP). However, if a shared buffer pool is used, the size of the shared buffer pool is acquired. This information is acquired only if the <code>scdsvcddef</code> definition command is specified in the user service definition.
	Parallel Service	Number of times	Number of times service requests are fetched from the schedule queue. This information is acquired only if the <code>scdsvcddef</code> definition command is specified in the user service definition.
		Number of services	The number of service requests that the applicable user server (SPP) is executing concurrently. This information is acquired only if the <code>scdsvcddef</code> definition command is specified in the user service definition.
TRN (transaction information)	Commits	Number of items	Number of transactions that are committed
	Rollbacks	Number of items	Number of transactions that are rolled back
	Commit settlement by command	Number of items	Number of transactions that are committed by commands
	Rollback settlement by command	Number of items	Number of transactions that are rolled back by commands
	Hazard settlement by command	Number of items	Number of transactions that are heuristically settled as hazards by commands
	Mixed settlement by command	Number of items	Number of transactions that are heuristically settled as mixed by commands
	Branch execution time	Number of times	Number of times transaction branches are executed

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Length of time	Actual length of time between the start of services as transactions and the completion of synchronous point processing
	Branch synchronous point processing time	Number of times	Number of times synchronous point processing is executed
		Length of time	Actual length of time required for synchronous point processing that is executed after the termination of services executed as transactions (this time is included in Branch execution time shown above)
RTS (real-time statistics)	Arbitrary section	Number of times	Number of times <code>dc_rts_utrace_put</code> is called
		Length of time	Length of time between calling of <code>dc_rts_utrace_put</code> with the <code>DCRTS_START</code> flag and calling of <code>dc_rts_utrace_put</code> with the <code>DCRTS_END</code> flag. Statistics for Arbitrary section are not acquired as the statistics for the entire system.
DAM (DAM information)	Read	Number of times	Number of times <code>dc_dam_read</code> is called. This information is acquired when <code>dc_dam_read</code> is successful.
		Data length	Buffer length specified in the argument of <code>dc_dam_read</code> .
	Read errors	Number of items	Number of times errors are detected in DAM cache blocks and the DAM blocks read from a disk and the number of read errors on a disk. Errors such as invalid API arguments are not included in this count.
	Write	Number of times	Number of times <code>dc_dam_write</code> and <code>dc_dam_rewrite</code> are called. This information is acquired when <code>dc_dam_write</code> and <code>dc_dam_rewrite</code> are successful.
		Data length	Buffer length specified in the arguments of <code>dc_dam_write</code> and <code>dc_dam_rewrite</code> .

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Write errors	Number of items	Number of errors that occurred while the data in DAM cache blocks was written to a disk. Depending on the time when a transaction is committed, this information may not be acquired for each service. Errors such as invalid API arguments are not included in this count.
	FJ output frequency	Number of times	Number of times recovery journals (FJ) are output. Depending on the time when a transaction is committed, this information may not be acquired for each service.
		Journal size	Number of recovery journals (FJ). Depending on the time when a transaction is committed, this information may not be acquired for each service.
	Number of DAM transaction branches	Number of times	Number of transaction branches related to the DAM
		Number of transactions	Number of transaction branches related to the DAM that are executed by the first DAM service function executed in the transaction. If the number of transaction branches related to the DAM is less than the value specified in the <code>dam_tran_process_count</code> operand in the DAM service definition, the system outputs the value specified in the <code>dam_tran_process_count</code> operand. The maximum value for this item is the maximum number of transaction branches related to the DAM that are simultaneously executed. When you specify this value in the <code>dam_tran_process_count</code> operand in the DAM service definition or use it in the formula for the <code>dam_cache_size</code> operand or the <code>dam_cache_size_fix</code> operand, the shared memory for the DAM cache is optimized.
	DAM cache-block securing frequency	Number of times	Number of times DAM cache blocks are allocated

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Common memory for DAM use rate	Number of times	Number of transactions that updated the DAM file specified for deferred update divided by 10
		Usage rate of shared memory pool	Usage rate of the shared memory for the DAM cache. Note that this value is updated only within the transactions that update the DAM files specified for deferred update.
TAM (TAM information)	Table file real renewal	Number of times	Number of times data is updated in a TAM file. Timer-start, cpd, and the tamhold command are available as events for updating a TAM file. When multiple TAM files are updated for one update event, the count increases only by one.
		Number of data bytes	Number of bytes in the data that is actually updated in a TAM file during online operation. This value includes all the data written for one update event. When multiple TAM files are updated, this value is the total number of bytes updated in those TAM files.
	Table file real renewal time	Number of times	Number of times data is updated in a TAM file. Timer-start, cpd, and the tamhold command are available as events for updating a TAM file. When multiple TAM files are updated for one update event, the count is increased only by one.
		Length of time	Processing time required for a TAM file update. This information indicates the length of time, in microseconds, required for one update event.
	Number of tam record references	Number of times	Number of commits and rollbacks related to the TAM

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Number of records	Number of records that are referenced in one transaction. The number of records is counted, regardless of whether <code>dc_tam_read</code> is called, because this value is determined and output by the commit processing of the TAM server. Therefore, in a system that does not call <code>dc_tam_read</code> , only the number of records is counted, and the maximum value, minimum value, and average are all 0.
	Number of tam record renewal	Number of times	Number of commits and rollbacks related to the TAM
		Number of records	Number of records updated in one transaction. The number of records is counted, regardless of whether <code>dc_tam_write</code> is called, because this value is determined and output by the commit processing of the TAM server. Therefore, in a system that does not call <code>dc_tam_write</code> , only the number of records is counted, and the maximum value, minimum value, and average are all 0.
	Read	Number of times	Number of times <code>dc_tam_read</code> is called. This information is acquired when <code>dc_tam_read</code> is successful. ^{#3}
		Data length	Length of data input to a TAM file each time <code>dc_tam_read</code> is read. ^{#3}
	Read errors	Number of items	Number of errors that occurred in <code>dc_tam_read</code> in the interface with the OS. Errors such as invalid API arguments are not included in this count. ^{#3}
	Write	Number of times	Number of times <code>dc_tam_write</code> and <code>dc_tam_rewrite</code> are called. This information is acquired when <code>dc_tam_write</code> and <code>dc_tam_rewrite</code> are successful. ^{#4}

E. Details of Statistics

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Data length	Length of data output from a TAM file each time <code>dc_tam_write</code> or <code>dc_tam_rewrite</code> is called ^{#4}
	Write errors	Number of items	Number of errors that occurred in <code>dc_tam_write</code> and <code>dc_tam_rewrite</code> in the interface with the OS. Errors such as invalid API arguments are not included in this count. ^{#4}
XAR (XA resource service information)	Start() request	Number of times	Number of times transaction start requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction start requests sent from an application server
	Start() request error	Number of items	Number of errors that occurred in the transaction start requests sent from an application server to the RAP-processing server
	Call() request	Number of times	Number of times service requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the service requests sent from an application server
	Call() request error	Number of items	Number of errors that occurred in the service requests sent from an application server to the RAP-processing server
	End() request	Number of times	Number of times transaction end requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction end requests sent from an application server

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	End () request error	Number of items	Number of errors that occurred in the transaction end requests sent from an application server to the RAP-processing server
	Prepare () request	Number of times	Number of times transaction preparation processing requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction preparation processing requests sent from an application server
	Prepare () request error	Number of items	Number of errors that occurred in the transaction preparation processing requests sent from an application server to the RAP-processing server
	Commit () request	Number of times	Number of times transaction commit processing requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of the execution by the RAP-processing server of the transaction commit processing requests sent from an application server
	Commit () request error	Number of items	Number of errors that occurred in the transaction commit processing requests sent from an application server to the RAP-processing server
	Rollback () request	Number of times	Number of times transaction rollback processing requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction rollback processing requests sent from an application server

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Rollback() request error	Number of items	Number of errors that occurred in the transaction rollback processing requests sent from an application server to the RAP-processing server
	Recover() request	Number of times	Number of times transaction notification requests that have been prepared or heuristically determined are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction notification requests that have been prepared or heuristically determined and sent from an application server
	Recover() request error	Number of items	Number of errors that occurred in the transaction notification requests that have been prepared or heuristically determined and sent from an application server to the RAP-processing server
	Forget() request	Number of times	Number of times transaction abort requests are sent from an application server to the RAP-processing server
		Length of time	Length of time from the start to the end of execution by the RAP-processing server of the transaction abort requests sent from an application server
	Forget() request error	Number of items	Number of errors that occurred in the transaction abort requests sent from an application server to the RAP-processing server
MCF (MCF information)	Schedule Stay	Number of times	Number of times the MHP retrieved messages from the input queue
		Length of time	Wait time until the MHP retrieves a message after it is placed in the input queue by the MCF communication process
	Execution of user service	Number of times	Number of times MHP executed services

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
		Length of time	Time from the start of service function execution to issuance of return by the service function

#1

The name service stores information in the global cache area about the service group running on the nodes specified in the `all_node` operand in the system common definition, and stores information in the local cache area about the service group running on the local node. When a UAP or a command executes an RPC, the required information about the service group is searched for in the global cache area and the local cache area.

#2

For the `message_buflen` operand in the user service definition, Hitachi recommends that you set the value of the message size (maximum). For the `message_store_buflen` operand in the user service definition, Hitachi recommends that you specify a value within the following range:

$$\text{Message length (average)} \times \text{schedule wait queue length (maximum)} \leq \text{message_store_buflen} \leq \text{message length (maximum)} \times \text{schedule wait queue length (maximum)}$$

#3

Includes `dc_dam_read` that accesses TAM tables.

#4

Includes `dc_dam_write` and `dc_dam_rewrite` that access TAM tables.

The following table shows the relationship between process termination and the applicable item counts in the real-time statistics.

Table E-7: Process termination and applicable item counts

Process and termination	Applicable item count
When a UAP process terminates with a CPU time monitoring timeout error	UAP abnormal terminations
When OpenTP1-provided servers operating as SPPs (TP1/EE, RAP-processing server, RTSSUP, RTSSPP) go down	
When a UAP process goes down	

Process and termination	Applicable item count
When the system server goes down but the system does not go down (trnrvd (transaction recovery service) and cltrnd (client-executed process))	System server abnormal terminations
When a UAP process terminates normally	Process terminations
When a UAP process is forcibly terminated by using the dcsvstop command with the -f option specified or the prckill command	
When a UAP process terminates due to an actual time-monitoring timeout error	

The following table shows the real-time statistics information that is edited.

Table E-8: Edited real-time statistics information

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
CPD (checkpoint dump information)	Collection of cpd	events (number of items)	Yes	No	1000	rts_cpd_collet_cpd	N	N	Y
		microsec (length of time in microseconds)	No	Yes					
	Validation of cpd	events (number of items)	Yes	No	1001	rts_cpd_validt_cpd	N	N	Y
		microsec (length of time in microseconds)	No	Yes					
JNL (journal information)	Buffer full occurrences	events (number of items)	Yes	No	1100	rts_jnl_buf_full	N	N	Y

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
	Waits for an available buffer	events (number of items)	Yes	No	1101	rts_jnl_wait_buf	N	N	Y
	Journal output	times (number of times)	Yes	No	1102	rts_jnl_jnl_output	N	N	Y
		bytes (data length in bytes)	No	Yes					
	I/O waits	times (number of times)	Yes	No	1103	rts_jnl_io_wait	N	N	Y
		surfaces (number of buffers x 100)	No	Yes					
	Write	times (number of times)	Yes	No	1104	rts_jnl_write	N	N	Y
		microsec (length of time in microseconds)	No	Yes					
	Swap	times (number of times)	Yes	No	1105	rts_jnl_swap	N	N	Y
		microsec (length of time in microseconds)	No	Yes					
	Journal input	times (number of times)	Yes	No	1106	rts_jnl_jnl_input	N	N	Y
		bytes (data length in bytes)	No	Yes					
	Read	times (number of times)	Yes	No	1107	rts_jnl_read	N	N	Y

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		microsec (length of time in microseconds)	No	Yes			N	N	Y
LCK (lock information)	Lock acquisition	events (number of items)	Yes	No	1200	rts_lck_lock_acqst	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Lock wait	events (number of items)	Yes	No	1201	rts_lck_lock_wait	Y	Y	Y
		events (queue length)	No	Yes					
	Deadlock	events (number of items)	Yes	No	1202	rts_lck_deadlock	Y	Y	Y
NAM (name information)	Global Cache hits	times (number of times)	Yes	No	1300	rts_nam_global_cache_hit	Y	Y	Y
	Local Cache hits	times (number of times)	Yes	No	1301	rts_nam_local_cache_hit	Y	Y	Y
	Lookup	events (number of times)	Yes	No	1302	rts_nam_lookup	Y	Y	Y
OSL (shared memory management information)	Common static memory (acquired)	times (number of times)	Yes	No	1400	rts_osl_stamem_acq	N	N	Y

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		bytes (memory block size in bytes)	No	Yes					
	Common static memory (pool)	times (number of times)	Yes	No	1401	rts_osl_stamem_pol	N	N	Y
		bytes (memory block size in bytes)	No	Yes					
	Common dynamic memory (acquired)	times (number of times)	Yes	No	1402	rts_osl_dynmem_acq	N	N	Y
		bytes (memory block size in bytes)	No	Yes					
	Common dynamic memory (pool)	times (number of times)	Yes	No	1403	rts_osl_dynmem_pol	N	N	Y
		bytes (memory block size in bytes)	No	Yes					
PRC (process information)	Process generation	times (number of times)	Yes	No	1500	rts_prc_prc_generation	N	Y	Y
	UAP abnormal terminations	times (number of times)	Yes	No	1501	rts_prc_uap_abnml	N	Y	Y

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
	System server abnormal terminations	times (number of times)	Yes	No	1502	rts_prc_sys_abnml	N	N	Y
	Process terminations	times (number of times)	Yes	No	1503	rts_prc_prc_term	N	Y	Y
	Number of processes	times (number of times)	Yes	No	1504	rts_prc_prc_num	N	N	Y
		Process (number of processes)	No	Yes					
QUE (message queue information)	Read message	times (number of times)	Yes	No	1601	rts_que_read	N	Y	Y
		bytes (message length in bytes)	No	Yes					
	Write message	times (number of times)	Yes	No	1602	rts_que_write	N	Y	Y
		bytes (message length in bytes)	No	Yes					
	Read errors	events (number of items)	Yes	No	1603	rts_que_read_err	N	Y	Y
	Write errors	events (number of items)	Yes	No	1604	rts_que_write_err	N	Y	Y

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target			
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S	
			Ttl	Max, Min, Avg						
	Waits for an available buffer	events (number of items)	Yes	No	1605	rts_que_wait_buf	N	Y	Y	
	Real reads	times(number of times)	Yes	No	1606	rts_que_real_read	N	Y	Y	
	Real writes	times(number of times)	Yes	No	1607	rts_que_real_write	N	Y	Y	
	Number of delay writings	times(number of times)	Yes	No	1650	rts_que_delay_wrt	N	N	Y	
		times(number of times)	No	Yes						
	Number of delay writing records	times(number of times)	Yes	No	1651	rts_que_delay_rec	N	N	Y	
		records (number of records)	No	Yes						
	Delay writing record rate	times(number of times)	Yes	No	1652	rts_que_delay_msg	N	N	Y	
		% (percentage of messages)	No	Yes						
	RPC (RPC information)	RPC calls	times(number of times)	Yes	No	1700	rts_rpc_rpc_call	Y	Y	Y
			microsec (length of time in microseconds)	No	Yes					

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
	RPC calls (chained)	times (number of times)	Yes	No	1701	rts_rpc_rpc_called	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Execution of user service	times (number of times)	Yes	No	1730	rts_rpc_usr_svc	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	RPC overtimes	events (number of items)	Yes	No	1731	rts_rpc_rpc_ovrtim	Y	Y	Y
	SCD (schedule information)	Schedule Waits	times (number of times)	Yes	No	1800	rts_scd_scd_wait	Y #	Y
events (queue length)			No	Yes					
Schedule		times (number of times)	Yes	No	1801	rts_scd_schedule	Y	Y	Y
		bytes (message length in bytes)	No	Yes					
Size of using buffer		times (number of times)	Yes	No	1802	rts_scd_using_buf	Y #	Y	Y

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				SVC	SRV	SYS
			Ttl	Max, Min, Avg					
		bytes (buffer pool size in bytes)	No	Yes					
	Message size in case of lack of buffer	times(number of times)	Yes	No	1803	rts_scd_lack_buf	Y	Y	Y
		bytes (message length in bytes)	No	Yes					
	Schedule Stay	times(number of times)	Yes	No	1804	rts_scd_scd_stay	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Schedule Waits of service	times(number of times)	Yes	No	1805	rts_scd_svc_scd_wait	Y	Y	Y
		events (queue length)	No	Yes					
	Size of using buffer of service	times(number of times)	Yes	No	1806	rts_scd_svc_using_buf	Y	Y	Y
		bytes (buffer pool size in bytes)	No	Yes					
	Parallel Service	times(number of times)	Yes	No	1807	rts_scd_parallel	Y	Y	Y

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		counts (number of services)	No	Yes					
TRN (transaction information)	Commits	events (number of items)	Yes	No	1900	rts_trn_commit	Y	Y	Y
	Rollbacks	events (number of items)	Yes	No	1901	rts_trn_rollback	Y	Y	Y
	Commit settlement by command	events (number of items)	Yes	No	1902	rts_trn_cmt_cmd	N	N	Y
	Rollback settlement by command	events (number of items)	Yes	No	1903	rts_trn_rbk_cmd	N	N	Y
	Hazard settlement by command	events (number of items)	Yes	No	1904	rts_trn_haz_cmd	N	N	Y
	Mixed settlement by command	events (number of items)	Yes	No	1905	rts_trn_mix_cmd	N	N	Y
	Branch execution time	times (number of times)	Yes	No	1906	rts_trn_branch	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
Branch synchronous point processing time	times (number of times)	Yes	No	1907	rts_trn_sync_point	Y	Y	Y	

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		microsec (length of time in microseconds)	No	Yes					
RTS (real-time statistics)	Arbitrary section	times (number of times)	Yes	No	100000 0-21474 83647	-	Y	Y	N
		microsec (length of time in microseconds)	No	Yes					
DAM (DAM information)	Read	times (number of times)	Yes	No	2000	rts_dam_read	Y	Y	Y
		bytes (data length in bytes)	No	Yes					
	Read errors	events (number of items)	Yes	No	2001	rts_dam_read_err	Y	Y	Y
	Write	times (number of times)	Yes	No	2002	rts_dam_write	Y	Y	Y
		bytes (data length in bytes)	No	Yes					
	Write errors	events (number of items)	Yes	No	2003	rts_dam_write_err	Y	Y	Y
	FJ output frequency	times (number of times)	Yes	No	2004	rts_dam_fj	Y	Y	Y
		bytes (journal size in bytes)	No	Yes					
Number of DAM transaction branches	times (number of times)	Yes	No	2005	rts_dam_trn_branch	Y	Y	Y	

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		transaction (number of transactions)	No	Yes					
	DAM cache-block securing frequency	times (number of times)	Yes	No	2006	rts_dam_cache_block	Y	Y	Y
	Common memory for DAM use rate	times (number of times)	Yes	No	2007	rts_dam_shm_pool	N	N	Y
		% (percentage of shared memory pool used)	No	Yes					
TAM (TAM information)	Table file real renewal	times (number of times)	Yes	No	2100	rts_tam_real_renew	N	N	Y
		bytes (number of data bytes)	No	Yes					
	Table file real renewal time	times (number of times)	Yes	No	2101	rts_tam_real_renew_time	N	N	Y
		microsec (length of time in microseconds)	No	Yes					
	Number of tam record references	times (number of times)	Yes	No	2102	rts_tam_rec_refer	N	Y	Y
		records (number of records)	No	Yes					

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
	Number of tam record renewal	times (number of times)	Yes	No	2103	rts_tam_rec_renew	N	Y	Y
		records (number of records)	No	Yes					
	Read	times (number of times)	Yes	No	2104	rts_tam_read	Y	Y	Y
		bytes (data length in bytes)	No	Yes					
	Read errors	events (number of items)	Yes	No	2105	rts_tam_read_err	Y	Y	Y
	Write	times (number of times)	Yes	No	2106	rts_tam_write	Y	Y	Y
bytes (data length in bytes)		No	Yes						
Write errors	events (number of items)	Yes	No	2107	rts_tam_write_err	Y	Y	Y	
XAR (XA resource service information)	Start () request	times (number of times)	Yes	No	2500	rts_xar_start	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Start () request error	events (number of items)	Yes	No	2501	rts_xar_start_err	N	Y	Y
	Call () request	times (number of times)	Yes	No	2502	rts_xar_call	N	Y	Y

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
		microsec (length of time in microseconds)	No	Yes					
	Call () request error	events (number of items)	Yes	No	2503	rts_xar_call_err	N	Y	Y
	End () request	times (number of times)	Yes	No	2504	rts_xar_end	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	End () request error	events (number of items)	Yes	No	2505	rts_xar_end_err	N	Y	Y
	Prepare () request	times (number of times)	Yes	No	2506	rts_xar_prepare	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Prepare () request error	events (number of items)	Yes	No	2507	rts_xar_prepare_err	N	Y	Y
	Commit () request	times (number of times)	Yes	No	2508	rts_xar_commit	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
	Commit () request error	events (number of items)	Yes	No	2509	rts_xar_commit_err	N	Y	Y
	Rollback () request	times (number of times)	Yes	No	2510	rts_xar_rollback	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Rollback () request error	events (number of items)	Yes	No	2511	rts_xar_rollback_err	N	Y	Y
	Recover () request	times (number of times)	Yes	No	2512	rts_xar_recover	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Recover () request error	events (number of items)	Yes	No	2513	rts_xar_recover_err	N	Y	Y
	Forget () request	times (number of times)	Yes	No	2514	rts_xar_forget	N	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Forget () request error	events (number of items)	Yes	No	2515	rts_xar_forget_err	N	Y	Y

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition target		
	Event (real-time statistics item name)	Unit (acquired information)	Value to be output				SVC	SRV	SYS
			Ttl	Max, Min, Avg					
MCF (MCF information)	Schedule Stay	times (number of times)	Yes	No	2300	rts_mcf_ap_scd_stay	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					
	Execution of user service	times (number of times)	Yes	No	2301	rts_mcf_ap_usr_srvc	Y	Y	Y
		microsec (length of time in microseconds)	No	Yes					

Legend:

Ttl: Total

Max: Maximum

Min: Minimum

Avg: Average

SVC: Acquired for each service.

SRV: Acquired for each server.

SYS: Acquired for the entire system.

Yes or Y: Applicable.

No or N: Not applicable.

#

For information acquired on a service basis, only times (number of times) is valid.

(2) Details about real-time statistics (when a target is specified)

The following table shows the details about real-time statistics when a target is specified.

Table E-9: Details about real-time statistics (when a target is specified)

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
NAM (name information)	Lookup to specified node	Number of items	Number of times search requests for service information are sent from each node to the specified reference node. When a UAP or a command executes an RPC, they may send search requests for service information to the node specified in the <code>all_node</code> operand in the system common definition to find the required information about the service group.
	Response of Lookup from specified node	Number of items	Number of times each node receives the responses to the search requests for service information from the specified reference node. When a UAP or a command executes an RPC, they may send search requests for service information to the node specified in the <code>all_node</code> operand in the system common definition to find the required information about the service group.
MCF (MCF information)	Residence time on input message ^{#1}	Number of times	Number of times the MHP retrieved messages from the input queue
		Time	Wait time until the MHP retrieves a message after it is placed in the input queue by the MCF communication process.
	Residence time on output message of synchronous ^{#1}	Number of times	Number of times the MCF communication process processed synchronous send messages.
		Time	Length of time until processing is performed by the MCF communication process after the UAP requests transmission of a synchronous send message.

Type (type of real-time statistics)	Edited information		Details
	Event (real-time statistics item name)	Acquired information	
	Residence time on output message of response ^{#1}	Number of times	Number of times the MCF communication process processed inquiry response send messages.
		Time	Length of time until processing is performed by the MCF communication process after a transaction is determined to have been committed following a UAP request for transmission of an inquiry response send message.
	Residence time on output message of priority ^{#1}	Number of times	Number of times the MCF communication process processed priority branch send messages.
		Time	Length of time until processing is performed by the MCF communication process after a transaction is determined to have been committed following a UAP request for transmission of a priority branch send message.
	Residence time on output message of normal ^{#1}	Number of times	Number of times the MCF communication process processed normal branch send messages.
		Time	Length of time until processing is performed by the MCF communication process after a transaction is determined to have been committed following a UAP request for transmission of a priority branch send message.
	Number on input message queue ^{#2}	Number of times	Number of times the MCF communication process placed messages in the input queue, and the number of times MHP retrieved messages from the input queue
		Number of items	Number of messages held in the input queue. If this value is large, increase the parallel_count operand value in the user service definition.

#1

Information acquired for each logical terminal.

#2

Information acquired for each service group.

The following table shows the format for specifying a target for acquiring real-time statistics.

Table E-10: Format for specifying a target for acquiring real-time statistics

Type (type of real-time statistics)	Event (real-time statistics item name)	Format for specifying target	
		Acquisition target name 1 ^{#1}	Acquisition target name 2 ^{#2}
NAM (name information)	Lookup to specified node	Port number ^{#3}	IP address ^{#3}
	Response of Lookup from specified node		
MCF (MCF information)	Residence time on input message	Logical terminal	None
	Residence time on output message of synchronous		
	Residence time on output message of response		
	Residence time on output message of priority		
	Residence time on output message of normal		
	Number on input message queue	None	Service group name

#1

When acquiring statistics:

Specify this in the `-o` option in the `rtspstat` definition command with `obj` specified for the `-u` option or the `rtstats` command.

When displaying or editing statistics:

Specify this in the `-s` option in the `rtsls` command with `svc` specified for the `-u` option or the `rtsedit` command.

#2

When acquiring statistics:

Specify this in the `-b` option in the `rtspout` definition command with `obj` specified for the `-u` option or the `rtstats` command.

When displaying or editing statistics:

Specify this in the `-v` option in the `rtsls` command with `svc` specified for the `-u` option or the `rtsedit` command.

#3

Specify the IP address and port number of the node specified in the `all_node` operand in the system common definition or specified in the domain definition file.

The following table shows the information produced from real-time statistics when a target is specified.

Table E-11: Edited real-time statistics information (when a target is specified)

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition unit#		
	Event (real-time statistics item name)	Units (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
NAM (name information)	Lookup to specified node	events (number of items)	Yes	No	1303	rts_nam_node_lookup	N	N	N
	Response of Lookup from specified node	events (number of items)	Yes	No	1304	rts_nam_node_lookup_responce	N	N	N

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition unit#		
	Event (real-time statistics item name)	Units (acquired information)	Value to be output				S V C	S R V	S Y S
			Ttl	Max, Min, Avg					
MCF (MCF information)	Residence time on input message	times (number of times)	Yes	No	2350	rts_mcf_in_msg_scd_wait	N	N	N
		microsec (length of time in microseconds)	No	Yes					
	Residence time on output message of synchronous	times (number of times)	Yes	No	2351	rts_mcf_out_msg_sync_scd_wait	N	N	N
		microsec (length of time in microseconds)	No	Yes					
	Residence time on output message of response	times (number of times)	Yes	No	2352	rts_mcf_out_msg_resp_scd_wait	N	N	N
		microsec (length of time in microseconds)	No	Yes					

E. Details of Statistics

Type (type of real-time statistics)	Edited information				Item ID (Event ID)	Operand name	Acquisition unit#		
	Event (real-time statistics item name)	Units (acquired information)	Value to be output				SVC	SRV	SYS
			Ttl	Max, Min, Avg					
	Residence time on output message of priority	times (number of times)	Yes	No	2353	rts_mcf_out_ms_g_prio_scd_wait	N	N	N
		microsec (length of time in microseconds)	No	Yes					
	Residence time on output message of normal	times (number of times)	Yes	No	2354	rts_mcf_out_ms_g_norm_scd_wait	N	N	N
		microsec (length of time in microseconds)	No	Yes					
Number on input message queue	times (number of times)	Yes	No	2380	rts_mcf_que_scd_wait_num	N	N	N	
	num (count)	No	Yes						

Legend:

Ttl: Total

Max: Maximum

Min: Minimum

Avg: Average

SVC: Acquired for each service.

SRV: Acquired for each server.

SYS: Acquired for the entire system.

Yes or Y: Applicable.

No or N: Not applicable.

#

When you specify a desired object as the target, real-time statistics are acquired from the object specified in acquisition target names 1 and 2.

F. List of Files Output by OpenTP1

This appendix lists the files output by OpenTPI organized by product.

(1) TP1/Server Base (common to both UNIX and Windows versions)

The table below lists the files that are output by TP1/Server Base (common to both UNIX and Windows versions). For details about each file, see Table F-2 *Descriptions of the files output by TP1/Server Base (common to both UNIX and Windows versions)*.

Table F-1: List of files output by TP1/Server Base (common to both UNIX and Windows versions)

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	Audit log file	<i>log_audit_path-operand-value</i> /audit.log Default: \$DCDIR/auditlog/audit.log	07-02	D, F, H	Text	When an audited event occurs	Y
2	Backup file for audit log file	<i>log_audit_path-operand-value</i> /auditNNN.log (NNN: 001-255) Default: \$DCDIR/auditlog/auditNNN.log (NNN: 001-255)	07-02	K	Text	When the size of the audit log file exceeds the <i>log_audit_size</i> operand value	Y
3	OpenTP1 automatic start checking file	\$DCDIR/spool/.automode	Initial	J	Binary	When OpenTP1 starts	N
4	Current work path control file 1	UNIX version: \$DCDIR/spool/.dcurrent_path Windows version: \$DCDIR\spool\dcurrent_path	UNIX version: 06-02 Windows version: 07-00	J	Binary	When <i>prcd</i> starts (not when the <i>dcstart</i> command is executed)	N
5	<i>prcd</i> port information storage file	\$DCDIR/spool/.portid	Initial	J	Binary	When <i>prcd</i> starts (not when the <i>dcstart</i> command is executed)	N

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
6	Command log	\$DCDIR/spool/cmdlog/cmdlog <i>N</i> (<i>N</i> : 1 or 2)	UNIX version: 07-00 Windows version: 06-50	C, E, H	Text	When the target command is executed	NE
7	<ul style="list-style-type: none"> • Trace information dump file • Lock file (internal file) 	\$DCDIR/spool/dcdfmtr <i>N</i> , \$DCDIR/spool/dcdfmtr1 (<i>N</i> : 1 or 2)	07-00	J	Binary	When the ddefchk command is executed	N
8	FIL event trace information file	\$DCDIR/spool/dcfilinf/_fl_ANN (ANN: 001-003)	07-03	D, F, H	Binary	When <i>Y</i> is specified in the <code>prf_trace</code> operand, 1 is specified in the <code>fil_prf_trace_option</code> operand, and the time required for file access processing is equal to or greater than the <code>fil_prf_trace_delay_time</code> operand value	NE
9	Core file for <code>jnlchgfg</code> command	\$DCDIR/spool/dcjnlinf/errinf/chgfg_ <i>N</i> (<i>N</i> : PID of the command that shut down)	07-03	A	Binary	When the <code>jnlchgfg</code> command terminates abnormally	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
10	Core file for jnlcuef process	\$DCDIR/spool/dcjnlinf/errinf/cuef_N (N: PID of the process that shut down)	07-03	A	Binary	When the internal command's process terminates abnormally	NE
11	Invalid journal information file for journal file	\$DCDIR/spool/dcjnlinf/errinf/cuef_N (N: PID of the command process that detected the invalid journal)	07-03	A	Binary	When the jnlunlfg, jnlchgfg, or jnlls command detects an invalid journal	NE
12	Core file for jnlls command	\$DCDIR/spool/dcjnlinf/errinf/ls_N (N: PID of the command that shut down)	07-03	A	Binary	When the jnlls command terminates abnormally	NE
13	Invalid journal information file for archive-journal source node	\$DCDIR/spool/dcjnlinf/errinf/r_N (N: Generation number of the journal that contains invalid data)	Initial	J	Text	When OpenTP1 is re-run and an error is detected in a loaded block	N
14	Core file for jnlunlfg command	\$DCDIR/spool/dcjnlinf/errinf/unlfg_N (N: PID of the command that shut down)	07-03	A	Binary	When the jnlunlfg command terminates abnormally	NE
15	Journal message queue management file	\$DCDIR/spool/dcjnlinf/jnlqid	Initial	J	Binary	When the journal service creates a message queue	N

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
16	JNL performance verification trace information file	\$DCDIR/spool/ dcjnlinf/prfinf/ _jn_NNN (NNN: 001-256)	07-03	D, F, H	Binary	When γ is specified in the prf_trace operand and 00000001 or 00000002 is specified in the jnl_prf_event_trace_level operand	NE
17	Deadlock and timeout information files	\$DCDIR/spool/ dclckinf	Initial	B	Text	When TAM, DAM, or the dc_lck_xxx function is used and a deadlock or lock timeout occurs	NE
18	LCK performance verification trace information file	\$DCDIR/spool/ dclckinf/_lk_NNN (NNN: 001-256)	07-03	D, F, H	Binary	When transaction processing or lock processing is underway	NE
19	Socket file for the message log notification function	\$DCDIR/spool/ dcloginf/.logntf	03-03	J	Binary	When the message log notification function is used	N
20	Message log file	\$DCDIR/spool/ dclogN (N: 1 or 2)	Initial	C, E, H	Text	When OpenTP1 is run	N

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
21	NAM event trace information file	\$DCDIR/spool/ dcnaminf/ _nm_NNN (NNN: 001-003)	07-02	D, F, H	Binary	When Y is specified in the prf_trace operand and a value other than 00000000 is specified in the nam_prf_trace_level operand	NE
22	Process service event trace information file	\$DCDIR/spool/ dcprcinf/_pr_NNN (NNN: 001-003)	07-02	D, F, H	Binary	When Y is specified in the prf_trace operand while Y is specified in the prc_prf_trace operand	NE
23	prfget command inheritance file	Earlier than 06-01 (earlier than 06-50 for Windows version): \$DCDIR/spool/ dcprfinf/getinf 06-01 to 07-02: \$DCDIR/spool/ dcprfinf/getinf, _trinf 07-02 or later: \$DCDIR/spool/ dcprfinf/getinf, _trinf, _nminf, _prinf, _xrinf	Initial	J	Binary	When the prfget command is executed without the -a option specified	N

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
24	Performance verification trace information file	Earlier than 06-01 (earlier than 06-50 for Windows version): \$DCDIR/spool/ dcrprinf/prf00N (N: 1-3) 06-01 or later: \$DCDIR/spool/ dcrprinf/prf_NNN (NNN: 001-256)	Initial	D, F, H	Binary	When OpenTP1 is run	NE
25	Lock file for RAP-processing	\$DCDIR/spool/ dcrapinf/ RAP-processing-listener-name	Initial	J	Binary	When the remote API service starts	N
26	Invalid message file for RAP-processing	03-05-/T, 03-06-/R, 05-03-/J or later: \$DCDIR/spool/ dcrapinf/ RAP-processing-listener-name.msg 05-04-/G, 06-00 or later: \$DCDIR/spool/ dcrapinf/ RAP-processing-listener-name.msg, RAP-processing-client-manager-name.msg Earlier than 06-00 that is not included in the above versions: \$DCDIR/spool/ dcrapinf/ server-name_process-ID	Initial	J, E	Text	When an invalid message is received by the remote API service	NE
27	RTS log file	rts_log_file_name-operand-valueN (N: 1-10) Default: \$DCDIR/spool/ dcrtsinf/rtslogN (N: 1-10)	07-00	D, F, H	Binary	When RTSSUP is started with Y specified in the rts_log_file operand	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
28	Backup file for RTS log file	<i>rts_log_file_name-operation-and-valueN</i> .bk (<i>N</i> : 1-10) Default: \$DCDIR/spool/ dcrtsinf/ rtslog <i>N</i> .bk (<i>N</i> : 1-10)	07-02	K	Binary	When RTSSUP is started	NE
29	Server recovery journal	\$DCDIR/spool/ dcsj1	Initial	J	Binary	Anytime during online operation	N
30	Transaction recovery journal	\$DCDIR/spool/ dctjlinf	Initial	J	Binary	Anytime during online operation	N
31	Undetermined-transaction information file	\$DCDIR/spool/ dctrninf/ <i>file-name</i>	Initial	B	Text	When OpenTP1 is started for re-run	NE
32	TRN event trace information file (PRF trace)	\$DCDIR/spool/ dctrninf/trace/ prf/_tr_ <i>NNN</i> (<i>NNN</i> : 001-256)	UNIX version: 06-01 Windows version: 07-00	D, F, H	Binary	When transaction processing is underway	NE
33	TRN event trace file	\$DCDIR/spool/ dctrninf/trace/ trnevtr <i>N</i> (<i>N</i> : 1 or 2)	Initial	D, F, H	Binary	When an error has occurred in an XA function	N

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
34	Multiple-file management file (internal file)	\$DCDIR/spool/ dcusmctl	Initial	J	Binary	When the OpenTP1 system is shut down and a shared memory dump file is output due to intentional collection of data and forced termination specified by the <code>-f</code> and <code>-d</code> options in the <code>dcstop</code> command	N
35	XAR performance verification trace information file	\$DCDIR/spool/ dcxarinf/_xr_NNN (NNN: 001-256)	07-02	D, F, H	Binary	When the XA resource service is used by specifying <code>y</code> in the <code>prf_trace</code> operand and a value other than <code>00000000</code> in the <code>xar_prf_trace_level</code> operand	NE
36	XAR event trace information file	\$DCDIR/spool/ dcxarinf/trace/ xarevtrN (N: 1 or 2)	06-00	D, F, H	Binary	When the XA resource service is used	NE
37	Backup file for XAR event trace information file	\$DCDIR/spool/ dcxarinf/trace/ xarevtrM.bkN (M: 1 or 2, N: 1-3)	06-00	K	Binary	When the XA resource service is used	NE
38	Error log information	\$DCDIR/spool/ errlog/errlogN (N: 1 or 2)	Initial	C, E, H	Text	When OpenTP1 is run	N

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
39	Directory for system internal lock control	\$DCDIR/spool/olkfifs	Initial	J	Text	When the dcmakeup command is executed or when OpenTP1 starts	N
40	Directory for system internal synchronization control	\$DCDIR/spool/olkrsfs	Initial	J	Text	When the dcmakeup command is executed or when OpenTP1 starts	N
41	Control file for resources for internal control (FIFO file)	\$DCDIR/spool/oslcnt1	Initial	J	Binary	When the dcmakeup command is executed or when OpenTP1 starts	N
42	RPC log file	\$DCDIR/spool/rpclogN (N: 1 or 2)	Initial	C, E, H	Text	When a communication error (including internal communications) occurs	N
43	RPC trace file	<i>rpc_trace_name-oper and-value</i> Default: \$DCDIR/spool/rpctrN (N: 1 or 2)	Initial	C, E, H	Binary	When the dc_rpc_cal1 or dc_rpc_cal1_to function is issued	Y
44	Backup file for FIL event trace information file	\$DCDIR/spool/save/_f1_MMM.bkN (MMM: 001-003, N: 1 or 2)	07-03	K	Binary	When OpenTP1 is shut down	NE

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
45	Backup file for JNL performance verification trace information file	\$DCDIR/spool/save/_jl_MMM.bkN (MMM: 001-256, N: 1 or 2)	07-03	K	Binary	When OpenTP1 is shut down	NE
46	Backup file for LCK performance verification trace information file	\$DCDIR/spool/save/_lk_MMM.bkN (MMM: 001-256, N: 1 or 2)	07-03	K	Binary	When OpenTP1 is shut down	NE
47	Backup file for NAM event trace information file	\$DCDIR/spool/save/_nm_MMM.bkN (MMM: 001-003, N: 1 or 2)	07-02	K	Binary	When OpenTP1 is shut down	NE
48	Backup file for process service event trace information file	\$DCDIR/spool/save/_pr_MMM.bkN (MMM: 001-003, N: 1 or 2)	07-02	K	Binary	When OpenTP1 is shut down	NE
49	Backup file for TRN event trace information file (PRF trace)	\$DCDIR/spool/save/_tr_MMM.bkN (MMM: 001-256, N: 1 or 2)	UNIX version: 06-01 Windows version: 07-00	K	Binary	When OpenTP1 is shut down	NE
50	Backup file for XAR performance verification trace information file	\$DCDIR/spool/save/_xr_MMM.bkN (MMM: 001-256, N: 1 or 2)	07-02	K	Binary	When OpenTP1 is shut down	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
51	Core file	<i>prc_coresave_path-op erand-value/ server-nameN</i> (<i>N</i> : 1-3) Default: \$DCDIR/spool/ save/ <i>server-nameN</i> (<i>N</i> : 1-3)	Initial	C, H	Binary	When the process is shut down [#]	NE
52	OpenTP1 debug information file	<i>prc_coresave_path-op erand-value/ server-nameN.deb</i> (<i>N</i> : 1-3) Default: \$DCDIR/spool/ save/ <i>server-nameN.deb</i> (<i>N</i> : 1-3)	Initial	C, H	Text	When the process is shut down [#]	NE
53	<ul style="list-style-type: none"> • Trace information dump file • Lock file (internal file) 	\$DCDIR/spool/ save/dcmtrdp <i>N</i> , dcmtrdpl (<i>N</i> : 1 or 2)	Initial	C, E, H	Binary	When OpenTP1 is run	N
54	Backup file for performance verification trace information file	Earlier than 06-01: \$DCDIR/spool/ save/ prf_MMM.bk <i>N</i> (<i>MMM</i> : 001-003, <i>N</i> : 1 or 2) 06-01 or later: \$DCDIR/spool/ save/ prf_MMM.bk <i>N</i> (<i>MMM</i> : 001-256, <i>N</i> : 1 or 2)	07-00	K	Binary	When OpenTP1 is shut down	NE

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
55	Schedule queue information file	\$DCDIR/spool/ scdqidN (N: 1 or 2)	Initial	C, E, H	Binary	When the queue-receiving server is started, terminated, shut down, or released from shutdown	N
56	Shared memory dump file	\$DCDIR/spool/ <i>output-file-name-specified-for-usmdump-command-execution</i> Default: \$DCDIR/spool/ shmdump When DAM, TAM, IST, RTS, or TP1/Message Queue is used and the -i option is specified, the file names are as follows: shmdump.dam.Z, shmdump.tam.Z, shmdump.ist.Z, shmdump.rts.Z, shmdump.mXX.Z (XX: 00-ff)	Initial	L	Binary	When a file name is omitted in the argument during execution of the usmdump command	Y
57	Shared memory dump file	\$DCDIR/spool/ shmdumpN (N: 1-3) When DAM, TAM, IST, RTS, or TP1/Message Queue is used and the -i option is specified, the file names are as follows: shmdump.damN.Z, shmdump.tamN.Z, shmdump.istN.Z, shmdump.rtsN.Z, shmdump.mXXN.Z (N: 1-3, XX: 00-ff)	Initial	E	Binary	When the system is shut down and when the dctestop command is executed with the -f and -d options specified	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
58	Transaction control object	\$DCDIR/spool/trnrmcmd/userobj	Initial	J	Binary	When the dcsetup, trnlnkrm, or trnmkobj command is executed	NY
59	Shared memory information file	\$DCDIR/tmp/dcommenv	Initial	J	Binary	When OpenTP1 starts	N
60	Resources for internal control (FIFO file)	\$DCDIR/tmp/olkmsg	Initial	J	Text	When namedpipe is specified in the schedule_method operand in the user service default definition or user service definition	N
61	jnlcolc command inheritance file	<i>current-path-where-jnlcolc-command-executed/jnlcolcNNN (NNN: 001-999)</i>	Initial	J	Binary	When the jnlcolc command is executed without the -l option specified	Y
62	Backup file for jnlcolc command inheritance file	<i>current-path-where-jnlcolc-command-executed/jnlcolcNNN.bak (NNN: 001-999)</i>	Initial	J	Binary	When the jnlcolc command is executed without the -l option specified and the specified inheritance file already exists	Y

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
63	jnlrput command inheritance file	<i>current-path-where-jnlrput-command-executed/jnlrputNNN</i> (NNN: 001-999)	Initial	J	Binary	When the jnlrput command is executed with the -e option specified and the -l option omitted	Y
64	Backup file for jnlrput command inheritance file	<i>current-path-where-jnlrput-command-executed/jnlrputNNN.bak</i> (NNN: 001-999)	Initial	J	Binary	When the following conditions are met: <ul style="list-style-type: none"> The jnlrput command is executed with the -e operand specified and the -l option omitted The specified inheritance file already exists 	Y
65	Current work path control file 2	UNIX version: <i>current-work-path/.dcdir_path</i> Windows version: <i>current-work-path/dcdir_pat</i>	UNIX version: 06-02 Windows version: 07-00	J	Binary	When <code>prcd</code> starts (not when the <code>dstart</code> command is executed)	N
66	Work file for <code>dcrasget</code> command	<i>target-directory/work</i>	UNIX version: 06-01 Windows version: 06-50	J	Binary	When the <code>dccspool</code> command is executed	AD (<code>dcrasget</code>)

Legend:

A: Monotonic increase (with no deletion function)

B: Monotonic increase (with a deletion function)

C: Round-robin (with no backup acquisition function)

D: Round-robin (with a backup acquisition function)

E: Round-robin (data is output immediately after a specified amount is reached and then a new file is used)

F: Round-robin (data is output immediately before a specified amount is reached and then a new file is used)

H: Round-robin (existing data is deleted from the target file and then new data is written from the beginning)

J: Control file or temporary file

K: Backup file

L: File whose maximum size is predetermined and to which data is output each time by a method such as command execution

Y: Can be deleted at the user's discretion

NE: Must not be deleted, unless error checking is not required.

NY: Must not be deleted, with the exception of user-created files.

AD: Deleted when the command is terminated (parentheses enclose the name of the relevant command)

N: Must not be deleted

#

Includes process shutdown when the command is executed under the following conditions:

- The `-f` and `-d` options are specified in the `dcstop` command.
- The `-f` and `-d` options are specified in the `dcsvstop` command.

The table below describes the files that are output by TP1/Server Base (common to both UNIX and Windows versions).

Table F-2: Descriptions of the files output by TP1/Server Base (common to both UNIX and Windows versions)

No.	Name	Related definition	Size	Max files	Description
1	Audit log file	<Log service definition> <ul style="list-style-type: none"> • log_audit_out operand Whether to use the audit log function • log_audit_path operand Output directory for the audit log file • log_audit_size operand Maximum size of audit log file • log_audit_count operand Maximum number of audit log files • log_audit_message operand Message ID for an item for which an audit log is collected 	log_audit_size operand value	1 generation	This is an audit log file that is obtained when <i>y</i> is specified in the log_audit_out operand. For details about the events that are subject to audit log collection, see the description of the definition for acquiring audited events in the manual <i>OpenTP1 System Definition</i> .

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
		<p><User service definition> <User service default definition></p> <ul style="list-style-type: none"> • log_audit_out_suppress operand Suppress output of audit logs by this user server • log_audit_message operand Message ID for an item for which an audit log is collected <p><RAP-processing listener service definition></p> <ul style="list-style-type: none"> • log_audit_out_suppress operand Suppress output of audit logs by RAP-processing listener and RAP-processing server • log_audit_message operand Message ID for an item for which an audit log is collected <p><RAP-processing client manager service definition></p> <ul style="list-style-type: none"> • log_audit_out_suppress operand Suppress output of audit logs by the RAP-processing client manager. • log_audit_message operand Message ID for an item for which audit log is collected 			
2	Backup file for audit log file	Same as for audit log file (No. 1)	log_audit_size operand value	The maximum value is the log_audit_count operand value - 1 generation	This is a backup file for the audit log file.

No.	Name	Related definition	Size	Max files	Description
3	OpenTP1 automatic start checking file	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.
4	Current work path control file 1	None	Several bytes (variable)	1 file	This file is used to control OpenTP1 operation.
5	pred port information storage file	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.
6	Command log	None	1024 kilobytes	2 generations	This file is used to output command logs that are collected during command execution. For details about the operation commands for which command logs are output, see <i>12.1.4 List of operation commands</i> . Note that the default is that command logs are not collected for MCF commands. To collect command logs, specify Y in the DCMCF_CMDLOG environment variable.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
7	<ul style="list-style-type: none"> • Trace information dump file • Lock file (internal file) 	None	64 kilobytes (dump file), 272 bytes (lock file)	3 files (dump file: 2 generations, lock file: 1 file)	These files are used to control OpenTP1 operation. They are the module trace storage file and lock file used during execution of the <code>dedefchk</code> command.
8	FIL event trace information file	<p><System common definition></p> <ul style="list-style-type: none"> • <code>prf_trace</code> operand Whether performance verification trace information is to be acquired • <code>fil_prf_trace_option</code> operand Whether FIL event traces are to be acquired • <code>fil_prf_trace_delay_time</code> operand Threshold value for file access processing time that is used as the condition for acquiring FIL event traces <p><Performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_trace_backup</code> operand Acquires a backup of the trace file 	Maximum 1 megabyte	3 generations	This is a FIL event trace information file. It is used to acquire FIL event trace information when <code>Y</code> is specified in the <code>prf_trace</code> operand and <code>1</code> in the <code>fil_prf_trace_option</code> operand. If <code>N</code> is specified in the <code>prf_trace_backup</code> operand, the backup file is not acquired.
9	Core file for <code>jnlchgfg</code> command	None	Size of memory used for process ^{#1}	Monotonic increase	This is a core file of the <code>jnlchgfg</code> command. It is acquired when the <code>jnlchgfg</code> command terminates abnormally.

No.	Name	Related definition	Size	Max files	Description
10	Core file for jnlcuef process	None	Size of memory used for process ^{#1}	Monotonic increase	This is a core file of the jnlcuef process. It is acquired when the jnlcuef process terminates abnormally.
11	Invalid journal information file for journal files	<System journal service definition> <ul style="list-style-type: none"> jnl_max_datasize operand Maximum length of record data 	jnl_max_datasize operand value	Monotonic increase	This is an invalid journal information file for journal files. It is acquired when the jnl1s command detects an invalid journal.
12	Core file for jnl1s command	None	Size of memory used for process ^{#1}	Monotonic increase	This is a core file of the jnl1s command. It is acquired when the jnl1s command terminates abnormally.
13	Invalid journal information file for archive-journal source node	<System journal service definition> <ul style="list-style-type: none"> jnl_max_datasize operand Maximum length of record data jnl_dual operand Whether dual journal files are to be used 	jnl_max_datasize operand value	jnl_dual=N: 1 file jnl_dual=Y: 2 files	This is an invalid journal information file for the archive-journal source node. It is acquired when there is an error in a block that was read when OpenTP1 started the re-run (abnormal). ^{#2}
14	Core file for jnlunlfg command	None	Size of memory used for process ^{#1}	Monotonic increase	This is a core file of the jnlunlfg command. It is acquired when the jnlunlfg command terminates abnormally.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
15	Journal message queue management file	<p><System journal service definition></p> <ul style="list-style-type: none"> • jnl_dual operand Whether dual journal files are to be used • jnl_max_file_dispersion operand Maximum number of dispersions for enabling parallel access <p><Archive journal service definition></p> <ul style="list-style-type: none"> • jnl_dual operand Whether dual journal files are to be used • jnl_max_file_dispersion operand Maximum number of dispersions for enabling parallel access 	<ul style="list-style-type: none"> • For archive-journal source node 16 bytes x (jnl_max_file_dispersion operand value x n + 1) (n: 2 for jnl_dual=Y, 1 for jnl_dual=N) • For archive journal node 16 bytes x (jnl_max_file_dispersion operand value x n x r + 1) (n: 2 for jnl_dual=Y, 1 for jnl_dual=N) (r: Number of resource groups specified in -a in jnldfs definition command in global archive journal service definition) 	1 file	This file is used to control OpenTP1 operation.

No.	Name	Related definition	Size	Max files	Description
16	JNL performance verification trace information file	<p><System common definition></p> <ul style="list-style-type: none"> • prf_trace operand Whether performance verification trace information is to be acquired • jnl_prf_event_trace_level operand Whether JNL performance verification traces are to be acquired <p><JNL performance verification trace definition></p> <ul style="list-style-type: none"> • prf_file_size operand Size of the trace file • prf_file_count operand Number of trace file generations • prf_trace_backup operand Acquires a backup of trace file 	prf_file_size operand value	prf_file_count operand value	<p>This is a JNL performance verification trace information file. The trace information is acquired when Y is specified in the prf_trace operand and 00000001 or 00000002 is specified in the jnl_prf_event_trace_level operand.</p> <p>In version 07-03 or later, a backup file is not acquired when N is specified in the prf_trace_backup operand.</p>

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
17	Deadlock and timeout information files	<p data-bbox="544 215 799 237"><Lock service definition></p> <ul data-bbox="544 241 847 734" style="list-style-type: none"> <li data-bbox="544 241 847 376">• lck_deadlock_info operand Whether deadlock and timeout information is to be output <li data-bbox="544 380 847 566">• lck_deadlock_info_remove operand Whether the deadlock information file and timeout information file are to be deleted at system startup <li data-bbox="544 571 847 734">• lck_deadlock_info_remove_level operand Deletion level of the deadlock information file and timeout information file 	Several kilobytes (variable)	Variable	<p data-bbox="1222 215 1390 958">These are the deadlock and timeout information files. Deadlock information and lock wait timeout information are acquired when Y is specified in the lck_deadlock_info operand. The files are deleted during OpenTP1 startup if the lck_deadlock_info_remove and lck_deadlock_info_remove_level operands are specified. The default is that the files are not deleted.</p>

No.	Name	Related definition	Size	Max files	Description
18	LCK performance verification trace information file	<p><System common definition></p> <ul style="list-style-type: none"> • <code>prf_trace</code> operand Whether performance verification trace information is to be acquired <p><Lock service definition></p> <ul style="list-style-type: none"> • <code>lck_prf_trace_level</code> operand Acquisition level for LCK performance verification trace information <p><Performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_trace_backup</code> operand Acquires a backup of the trace file <p><LCK performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_file_size</code> operand Size of LCK performance verification trace information file • <code>prf_information_level</code> operand Display level for messages related to LCK performance verification traces • <code>prf_file_count</code> operand Number of generations of LCK performance verification trace information files 	<code>prf_file_size</code> operand value	<code>prf_file_count</code> operand value	<p>This is an LCK performance verification trace information file. Trace information is acquired for various types of lock control associated with transaction processing when <code>Y</code> is specified in the <code>prf_trace</code> operand. The default is that this information is acquired.</p> <p>In version 07-03 or later, a backup file is not acquired when <code>N</code> is specified in the <code>prf_trace_backup</code> operand.</p>

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
19	Socket file for message log notification function	<p><Log service definition></p> <ul style="list-style-type: none"> log_notify_out operand Whether the message log notification function is to be used log_notify_xxx operand Whether information corresponding to log_notify_xxx is to be added 	0	1 file	This is a socket file used by the message log notification function.
20	Message log file	<p><Log service definition></p> <ul style="list-style-type: none"> log_filesize operand Maximum size of message log file 	log_filesize operand value	2 generations	This is an OpenTP1 message log file.
21	NAM event trace information file	<p><System common definition></p> <ul style="list-style-type: none"> prf_trace operand Whether performance verification trace information is to be acquired nam_prf_trace_level operand Acquisition level for NAM event trace information <p><Performance verification trace definition></p> <ul style="list-style-type: none"> prf_trace_backup operand Acquires a backup of the trace file 	Maximum 10 megabytes	3 generations	<p>This is a NAM event trace information file. NAM event trace information is acquired when Y is specified in the prf_trace operand and a value other than 00000000 is specified in the nam_prf_trace_level operand.</p> <p>In version 07-02 or later, a backup file is not acquired when N is specified in the prf_trace_backup operand.</p>

No.	Name	Related definition	Size	Max files	Description
22	Process service event trace information file	<p><System common definition></p> <ul style="list-style-type: none"> • <code>prf_trace</code> operand Whether performance verification trace information is to be acquired <p><Performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_trace_backup</code> operand Acquires a backup of the trace file <p><Process service definition></p> <ul style="list-style-type: none"> • <code>prc_prf_trace</code> operand Whether event traces are to be acquired for the process service 	Maximum 1 megabyte	3 generations	<p>This is a process service event trace information file. Process service event trace information is acquired when <code>Y</code> is specified in both <code>prf_trace</code> and <code>prc_prf_trace</code> operands.</p> <p>In version 07-02 or later, a backup file is not acquired when <code>N</code> is specified in the <code>prf_trace_backup</code> operand.</p>
23	<code>prfget</code> command inheritance file	None	128 bytes	<p>Earlier than 06-01: 1 file</p> <p>06-01 to earlier than 07-02: 2 files</p> <p>07-02 or later: 5 files</p>	<p>This file is used to control OpenTP1 operation. This is an internal file used to record the location of trace data that has already been acquired during execution of the <code>prfget</code> command.</p>

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
24	Performance verification trace information file	<p><System common definition></p> <ul style="list-style-type: none"> prf_trace operand Whether performance verification trace information is to be acquired <p><Performance verification trace definition></p> <ul style="list-style-type: none"> prf_file_size operand Size of the trace file prf_information_level operand Output level of trace file-related messages prf_file_count operand Number of trace file generations prf_trace_backup operand Acquires a backup of the trace file 	prf_file_size operand value	Earlier than 06-01: 3 generations 06-01 or later: prf_file_count operand value	This is a performance verification trace information file. Performance verification trace information is acquired when Y is specified in the prf_trace operand. The default is that this file is acquired. In version 07-02 or later, the backup file is not acquired when N is specified in the prf_trace_backup operand.
25	Lock file for RAP-processing	None	16 bytes x (1 + number of RAP-processing servers)	Number of RAP-processing listeners	This file is used to control OpenTP1 operation. It is a lock file used to synchronize the RAP-processing listener and server.
26	Invalid message file for RAP-processing	None	03-05-/T, 03-06-/R, 05-03-/J, 05-04-/G, 06-00 or later: Maximum 1 megabyte. Earlier than 06-00 that is not included in the above versions: Monotonic increase	Depends on the version.#3	This is a troubleshooting file that is created when an invalid message is received by the API service.

No.	Name	Related definition	Size	Max files	Description
27	RTS log file	<p><Real-time statistics service definition></p> <ul style="list-style-type: none"> • <code>rts_log_file</code> operand Whether acquired statistical information is to be output to the RTS log file • <code>rts_log_file_name</code> operand Name of the RTS log file • <code>rts_log_file_size</code> operand Size of the RTS log file • <code>rts_log_file_count</code> operand Number of RTS log file generations • <code>rts_log_file_backup</code> operand Whether a backup of the RTS log file is to be created 	Maximum value is the <code>rts_log_file_size</code> operand value	<code>rts_log_file_count</code> operand value	This is a real-time statistics log file. Real-time statistics are acquired when <code>Y</code> is specified in the <code>rts_log_file</code> operand. In version 07-02 or later, a backup file is acquired at the time <code>RTSSUP</code> starts if <code>Y</code> is specified in the <code>rts_log_file_backup</code> operand.
28	Backup file for RTS log file	Same as for the real-time statistics (RTS) log file (No. 27)	Maximum value is the <code>rts_log_file_size</code> operand value	<code>rts_log_file_count</code> operand value x 1 generation	This is a backup file for the real-time statistics log file. This backup file is not acquired when <code>N</code> is specified in the <code>rts_log_file_backup</code> operand.
29	Server recovery journal	None	Several bytes to kilobytes (variable)	Variable	This file is used to acquire various types of journal information for each system service.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
30	Transaction recovery journal	None	Several bytes to kilobytes (variable)	Variable	This file is used to acquire various types of journal information related to transactions for each transaction branch (UAP process).
31	Undetermined-transaction information file	<p data-bbox="544 506 751 562"><Transaction service definition></p> <ul data-bbox="544 566 842 1048" style="list-style-type: none"> <li data-bbox="544 566 842 723">• <code>trn_tran_recovery_list</code> operand Whether undetermined transaction information is to be acquired during a complete recovery <li data-bbox="544 728 842 913">• <code>trn_recovery_list_remove</code> operand Whether the undetermined-transaction information file is to be deleted during OpenTP1 startup. <li data-bbox="544 918 842 1048">• <code>trn_recovery_list_remove_level</code> operand Deletion level for the undetermined-transaction information file 	Several kilobytes (Monotonic increase)	Variable	This is an undetermined-transaction information file. Undetermined transaction information is acquired when <code>Y</code> is specified in the <code>trn_tran_recovery_list</code> operand. This file is deleted during startup of OpenTP1 if the <code>trn_recovery_list_remove</code> and <code>trn_recovery_list_remove_level</code> operands are specified. The default is that this file is not deleted.

No.	Name	Related definition	Size	Max files	Description
32	TRN event trace information file (PRF trace)	<p><System common definition></p> <ul style="list-style-type: none"> • prf_trace operand Whether performance verification trace information is to be acquired <p><Transaction service definition></p> <ul style="list-style-type: none"> • trn_prf_event_trace_level operand Acquisition level for TRN event trace information • trn_prf_event_trace_condition operand Type of TRN event trace to be acquired <p><Performance verification trace definition></p> <ul style="list-style-type: none"> • prf_trace_backup operand Acquires a backup of the trace file <p><TRN event trace definition></p> <ul style="list-style-type: none"> • prf_file_size operand Size of trace file for TRN event trace information • prf_information_level operand Display level for messages related to TRN event traces • prf_file_count operand Number of trace file generations for TRN event trace information 	prf_file_size operand value	prf_file_count operand value	<p>This is a TRN event trace information file. TRN event trace information (various events associated with the transaction services and the XA functions issued at transaction branches) is acquired when Y is specified in the prf_trace operand. The default is that this information is acquired. In version 07-02 or later, a backup file is not acquired when N is specified in the prf_trace_backup operand.</p>
33	TRN event trace file	None	1024 kilobytes	2 generations	<p>This is a TRN event trace file. It is used to acquire trace information related to transaction processing. Its contents are not disclosed.</p>

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
34	Multiple-file management file (internal file)	None	16 bytes	1 file	This file is used to control OpenTP1 operation. It is a generation management file for recording the current generation of shared-memory dump files.
35	XAR performance verification trace information file	<p><System common definition></p> <ul style="list-style-type: none"> • <code>prf_trace</code> operand Whether performance verification trace information is to be acquired <p><Performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_trace_backup</code> operand Acquires a backup of the trace file <p><XA resource service definition></p> <ul style="list-style-type: none"> • <code>xar_prf_trace_level</code> operand Acquisition level for XAR performance verification trace information <p><XAR performance verification trace definition></p> <ul style="list-style-type: none"> • <code>prf_file_size</code> operand Size of XAR performance verification trace information file • <code>prf_information_level</code> operand Display level for messages related to XAR performance verification traces. • <code>prf_file_count</code> operand Number of generations of XAR performance verification trace information files 	<code>prf_file_size</code> operand value	<code>prf_file_count</code> operand value	This is an XAR performance verification trace information file. XAR performance verification trace information is acquired when <code>Y</code> is specified in the <code>prf_trace</code> operand and a value other than 00000000 is specified in the <code>xar_prf_trace_level</code> operand. When <code>N</code> is specified in the <code>prf_trace_backup</code> operand, a backup file is not acquired.

No.	Name	Related definition	Size	Max files	Description
36	XAR event trace information file	<p><Transaction service definition></p> <ul style="list-style-type: none"> trn_xar_use operand Whether the XA resource service is to be used <p><XA resource service definition></p> <ul style="list-style-type: none"> xar_eventtrace_level operand Output level for the XAR event trace information xar_eventtrace_record operand Maximum number of records output to the XAR event trace information file 	xar_eventtrace_record operand value x 224 + 8	2 generations	This is an XAR event trace information file. XAR event trace information is acquired when γ is specified in the trn_xar_use operand.
37	Backup file for XAR event trace information file	Same as for the XAR event trace information file (No. 36)	xar_eventtrace_record operand value x 224 + 8	2 x 3 generations	This is a backup file of the XAR event trace information file.
38	Error log information	None	Earlier than 06-01: 64 kilobytes 06-01 or later: 1 megabyte	2 generations	This is internal trace information for OpenTP1. Its contents are not disclosed.
39	Directory for system internal lock control	<p><Process service definition></p> <ul style="list-style-type: none"> prc_process_count operand Maximum number of server processes that can be started concurrently 	0	prc_process_count operand value x 2	This file is used to control OpenTP1 operation.
40	Directory for system internal synchronization control	<p><Process service definition></p> <ul style="list-style-type: none"> prc_process_count operand Maximum number of server processes that can be started concurrently 	0	prc_process_count operand value + 83	This file is used to control OpenTP1 operation.
41	Control file for resources for internal control (FIFO file)	None	32 bytes (32 or 64 bits)	1 file	This file is used to control OpenTP1 operation.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
42	RPC log file	None	1 megabyte	2 generations	This is communication trace information for checking design.
43	RPC trace file	<p><System common definition></p> <ul style="list-style-type: none"> • <code>rpc_trace_operand</code> Whether RPC traces are to be acquired • <code>rpc_trace_name</code> Name of the file for acquiring RPC traces • <code>rpc_trace_size</code> Size of the file for acquiring RPC traces 	Maximum value is the <code>rpc_trace_size</code> operand value	2 generations	This is an RPC trace file. When <code>Y</code> is specified in the <code>rpc_trace</code> operand, RPC service request information (event information and the messages that are sent and received when the <code>dc_rpc_call</code> or <code>dc_rpc_call_to</code> function is issued) is acquired.
44	Backup file for FIL event trace information file	Same as for the FIL event trace information file (No. 8)	Maximum 1 megabyte	3 x 2 generations	This is a backup file for the FIL event trace file. It is acquired when <code>Y</code> is specified in the <code>prf_trace</code> operand and 1 in the <code>fil_prf_trace_option</code> operand. When <code>N</code> is specified in the <code>prf_trace_backup</code> operand, the backup file is not acquired.

No.	Name	Related definition	Size	Max files	Description
45	Backup file for JNL performance verification trace information file	Same as for the JNL performance verification trace information file (No. 16)	prf_file_size operand value	prf_file_count operand value x 2 generations	This is a backup file for JNL performance verification trace information file. It is acquired when Y is specified in the prf_trace operand and 00000001 or 00000002 is specified in the jnl_prf_event_trace_level operand. In version 07-03 or later, this backup file is not acquired when N is specified in the prf_trace_backup operand.
46	Backup file for LCK performance verification trace information file	Same as for the LCK performance verification trace information file (No. 18)	prf_file_size operand value	prf_file_count operand value x 2 generations	This is a backup file for LCK performance verification trace information file. It is acquired when Y is specified in the prf_trace operand. In version 07-03 or later, this backup file is not acquired when N is specified in the prf_trace_backup operand.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
47	Backup file for NAM event trace information file	Same as for the NAM event trace information file (No. 21)	Maximum 10 megabytes	3 x 2 generations	<p>This is a backup file for the NAM event trace file. It is acquired when Y is specified in the <code>prf_trace</code> operand and a value other than 00000000 is specified in the <code>nam_prf_trace_level</code> operand.</p> <p>In version 07-02 or later, this backup file is not acquired when N is specified in the <code>prf_trace_backup</code> operand.</p>
48	Backup file for process service event trace information file	Same as for the process service event trace information file (No. 22)	Maximum 1 megabyte	3 x 2 generations	<p>This is a backup file for the process service event trace file. It is acquired when Y is specified in the <code>prf_trace</code> operand and Y in the <code>prc_prf_trace</code> operand.</p> <p>In version 07-02 or later, this backup file is not acquired when N is specified in the <code>prf_trace_backup</code> operand.</p>

No.	Name	Related definition	Size	Max files	Description
49	Backup file for TRN event trace information file (PRF trace)	Same as for the TRN event trace information file (PRF trace) (No. 32)	prf_file_size operand value	prf_file_count operand value x 2 generations	This is a backup file for the TRN event trace file. It is acquired when Y is specified in the prf_trace operand. In version 07-02 or later, this backup file is not acquired when N is specified in the prf_trace_backup operand.
50	Backup file for XAR performance verification trace information file	Same as for the XAR performance verification trace information file (No. 35)	prf_file_size operand value	prf_file_count operand value x 2 generations	This is a backup file for the XAR performance verification trace information file. It is acquired when Y is specified in the prf_trace operand. This backup file is not acquired when N is specified in the prf_trace_backup operand.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
51	Core file	<p><Process service definition></p> <ul style="list-style-type: none"> • <code>prc_coresave_path</code> operand Core file storage path • <code>prc_corecompress</code> operand Core file is to be compressed automatically by OpenTP1 when it is stored <p><User service definition> <User service default definition></p> <ul style="list-style-type: none"> • <code>core_shm_suppress</code> operand Whether output of a memory dump to the core file is to be suppressed <p><Environment variable></p> <ul style="list-style-type: none"> • <code>CORE_NOSHM</code> Suppression of output of shared memory to the core file (applicable to AIX version) For details about suppression of output of shared memory to the core file, see the <i>Release Notes</i>. 	Amount of memory used for process ^{#1}	3 generations	This is a core file for system and user servers that have been shut down. Note that generation management is not applicable to a core file for a user server that was terminated by the <code>dcsvstop</code> command with the <code>-f</code> and <code>-d</code> options specified, or by the <code>prckill</code> command or for a user server that was terminated because realtime monitoring resulted in a timeout.
52	OpenTP1 debug information file	<p><Process service definition></p> <ul style="list-style-type: none"> • <code>prc_coresave_path</code> operand Core file storage path 	Variable	3 generations	This is a debugging information file for system and user servers that have been shut down.
53	Trace information dump file	None	64 kilobytes (module trace), 272 bytes (file for trace control)	3 files (module trace: 2 generations, trace control file: 1 file)	This is a trace information dump file for internal checking.

No.	Name	Related definition	Size	Max files	Description
54	Backup file for performance verification trace information file	Same as for the performance verification trace information file (No. 24)	prf_file_size operand value	Earlier than 06-01: 3 x 2 generations 06-01 or later: prf_file_count operand value x 2 generations	This is a backup file for the performance verification trace information file. It is acquired when Y is specified in the prf_trace operand. In version 07-02 or later, a backup file is not acquired when N is specified in the prf_trace_backup operand. Backup acquisition suppression is applied to the following files: <ul style="list-style-type: none"> • Performance verification trace information file • TRN event trace file • NAM event trace file • XAR performance verification trace information file • Process service event trace
55	Schedule queue information file	<Schedule service definition> <ul style="list-style-type: none"> • scd_server_count operand Maximum number of user servers 	(scd_server_count operand + 4) x 16 + 32 (bytes, maximum)	2 generations	This file is used to control OpenTP1 operation.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
56	Shared memory dump file	None	Depends on the machine status and OpenTP1 system operating status	1 generation# 4	This is a dump of the shared memory allocated by OpenTIP1.
57	Shared memory dump file	None	Depends on the machine status and OpenTP1 system operating status	3 generations #4	This is a dump of the shared memory allocated by OpenTIP1.
58	Transaction control object	None	Several to dozens of kilobytes (variable)	1 or more files	This file is used to control OpenTP1 operation. This is an object (dc_trn_allrm.o) file created at the extension of the trnlncrm command or an object file created by the user at the extension of the trnmkobj command. A user's file is created only when the trnmkobj command is executed.
59	Shared memory information file	None	544 bytes (32 or 64 bits)	1 file	This file is used to control OpenTP1 operation.
60	Resources for internal control (FIFO file)	<Schedule service definition> • scd_server_count Maximum number of user servers	0	scd_server_count operand value	This file is used to control OpenTP1 operation.

No.	Name	Related definition	Size	Max files	Description
61	jnlcolc command inheritance file	None	Depends on the size of recovery journal to be extracted	1 file	This file is used to control OpenTP1 operation. It is an inheritance file that is created when the jnlcolc command is used to extract a recovery journal of multiple journals.
62	Backup file for jnlcolc command inheritance file	None	Size of the inheritance file before renaming	1 file	This file is used to control OpenTP1 operation. If the jnlcolc command is executed and the specified inheritance file already exists, this file is created by renaming the existing file as jnlcolc***.bak.
63	jnlrput command inheritance file	None	Depends on the size of the transaction-undet ermined UJ	1 file	This file is used to control OpenTP1 operation. It is an inheritance file that is created when the jnlrput command is used to extract a transaction-deter mined UJ of multiple journals.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
64	Backup file for jnlrput command inheritance file	None	Size of the inheritance file before renaming	1 file	This file is used to control OpenTP1 operation. If the jnlrput command is executed and the specified inheritance file already exists, this file is created by renaming the existing file as jnlrput***.bak.
65	Current work path control file 2	None	Depends on DCDIR	1 file	This file is used to control OpenTP1 operation.
66	Work file for dcrasget command	None	Depends on the information to be collected	Depends on the information to be collected	This file is used to control OpenTP1 operation.

Legend:

Max files: Maximum number of files

#1

The size of memory used is variable because it depends on the coding. In versions other than UP-UX and Windows, the size requirement is even larger because a shared memory dump is provided.

#2

This file is acquired even when there is no end block. Therefore, if there was no time to write the end block for a reason such as forced termination of OpenTP1, the file is output whether or not a block error occurred.

#3

The maximum number of files depends on the product version, as follows.

For 03-05-/T, 03-06-/R, 05-03-/J, 05-04-/G, or 06-00 or later:

Number of processes for which invalid messages were received by the remote

API service (for the RAP-processing server, information is output to the same file as for the RAP-processing listener).

For a version earlier than 06-00 that is not one of the above versions:

Number of processes for which invalid messages were received by the remote API service.

#4

The uncompressed file is deleted after the file is compressed; however, immediately after compression, both uncompressed and compressed files exist. Therefore, the disk space required is twice the maximum size of the uncompressed file.

(2) TP1/Server Base (applicable to UNIX version only)

The table below lists the files that are output by TP1/Server Base (applicable to UNIX version only). For details about each file, see Table F-4 *Descriptions of the files output by TP1/Server Base (applicable to UNIX version only)*.

Table F-3: List of files output by TP1/Server Base (applicable to UNIX version only)

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	Communication control file	\$DCDIR/.tact/*	Initial	J	Local socket	When a process is started	N
2	Backup file for inittab	\$DCDIR/conf/Inittab/inittabN (N: 1-3)	Initial	K	Text	When dcsetup command is executed	Y
3	Control file for dccspool	\$DCDIR/spool/.dccspool_time	06-02	J	Null	When dccspool command is executed	AD (dccspool)
4	Pipe file for prctee	\$DCDIR/spool/.prc_fifo	05-00	J	Binary	When prctee command is started	N
5	PID storage file for prctee process	\$DCDIR/spool/.prc_tee	Initial	J	Binary	When prctee command is started	N

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
6	prctee process start error output file	\$DCDIR/spool/.prcerr.log	07-00	A	Text	When prctee command startup fails	NE
7	PAUSE checking file for prcd	\$DCDIR/spool/.prcnopause	Initial	J	Binary	When OpenTP1 starts	N
8	PID storage file for Linux prcd	\$DCDIR/spool/.prcpid	06-00	J	Binary	When prcd starts (not when dstart command is executed)	N
9	prctee process error output file	\$DCDIR/spool/.prctee.log	07-00	A	Text	When an output error occurs on betarn.log	NE
10	Archive-journal source node connection information file	\$DCDIR/spool/dcjnlinf/coninf	Initial	J	Text	When connection is established with or released from the archive journal node	NE
11	Archive journal node connection information file	\$DCDIR/spool/dcjnlinf/coninf	Initial	J	Text	When connection is established with or released from the archive-journal source node	NE

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
12	Invalid journal information file for archive journal node	\$DCDIR/spool/ dcjnlinf/errinf/ an_X (n: Resource group serial number for the journal in which invalid data was detected. This serial number is assigned sequentially to the resource groups specified in the -a option of the jnldfsv definition command in the global archive journal service definition X: Generation number of the journal in which invalid data was detected)	Initial	J	Text	When OpenTP1 is re-run and an error is detected in a loaded block	N
13	Invalid journal information file for archive journal file	\$DCDIR/spool/ dcjnlinf/errinf/ cuef_N (N: PID of the command that was shut down)	07-03	A	Binary	When the jnlunlfg, jnlchgfg, or jnlis command detects an invalid journal	NE
14	GWF log file	\$DCDIR/spool/ gwflogN (N: 1 or 2)	Initial	C, E, H	Text	When a gateway service-related internal function returns control with an error	N
15	UAP trace edit/output file	prc_coresave_path-op erand-value/ server-nameN.uat (N: 1-3) Default: \$DCDIR/spool/ save/ server-nameN.uat (N: 1-3)	Initial	C, H	Text	When the process is shut down [#]	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
16	Backup file for UAP trace data file	\$DCDIR/spool/save/trc/ <i>server-name</i> _N.uatmap (N: 1-3) When server terminates abnormally: \$DCDIR/spool/save/trc/ <i>server-name</i> N.uatmap (N: 1-3)	07-03	K	Binary	When the user server is shut down or when OpenTP1 starts	NE
17	Backup file for module trace data file	\$DCDIR/spool/save/trc/ <i>server-name</i> _N.umtm ap (N: 1-3) When server terminates abnormally: \$DCDIR/spool/save/trc/ <i>server-name</i> N.umtm ap (N: 1-3)	07-03	K	Binary	When the user server is shut down or when OpenTP1 starts	NE
18	PID file for <i>prcd</i> management process	\$DCDIR/tmp/home/ <i>server-name</i> / <i>server-name</i> .prc.PID	Initial	J	Null	When each process is started	N
19	UAP trace data file	\$DCDIR/tmp/home/ <i>server-name</i> .ID/ <i>server-name</i> .dcuat.map	07-03	C	Binary	When a UAP process is started	NE
20	Module trace data file	\$DCDIR/tmp/home/ <i>server-name</i> .XX/ <i>server-name</i> .dcumt.map (XX: Serial number)	07-03	C	Binary	When a process is started	NE
21	DCDIR management information in the same machine	Linux version: /opt/OpenTP1/etc/.OpenTP1 AIX or HI-UX/WE2 version: /usr/etc/BeTRAN	Initial	J	Text	When <i>dcsetup</i> command is executed	N

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
22	DCDIR management shell in the same machine	Linux version: /opt/OpenTP1/ etc/dcpwon AIX or HI-UX/WE2 version: /usr/bin/dcpwon	Initial	J	Text	When dcsetup command is executed	N
23	Communication control file	1. In Linux version: /opt/OpenTP1/ etc/so_unix 2. In Solaris version whose OS version is 10 or later and whose environment is Zone (virtual server): /etc/OpenTP1/ so_unix 3. Other: /dev/OpenTP1/ so_unix	In case 1: 05-03 In case 2: 07-00 In case 3: 03-02	J	Local socket	When a process is started	N
24	Backup file for dcsetup	/tmp/.admdPID/*	Initial	J	Text	When dcsetup command is executed	AD (dcsetup)
25	Realtime output function (betran.log)	HP-UX, AIX, or Solaris version: /tmp/betran.log Linux (IPF) version: \$DCDIR/spool/ betran.log Linux (x86): \$DCDIR/spool/ prclogN (N: 1 or 2) HI-UX/WE2 version: This file is not used because data is output to the console.	Initial	A	Text	When OpenTP1 is run	N
26	Temporary file used during execution of namdomain setup command	/tmp/ TP1dmfilwork (PID)	Initial	J	Text	When namdomains etup command is executed	AD (namdomai nsetup)

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
27	Lock file for dcsetup	Linux version: /var/spool/ .OpenTP1.HiRDB/ .lock AIX or HI-UX/WE2 version: /usr/spool/ .OpenTP1.HiRDB/ .lock	Initial	J	Binary	When dcsetup command is executed	N
28	Temporary file used during execution of jnlsort command	<i>current-path-where-jnlsort-command-executed/sort_I + process-ID</i> <i>current-path-where-jnlsort-command-executed/sort_o + process-ID</i>	Initial	J	Text	When jnlsort command is executed with the -g option omitted	AD (jnlsort)

Legend:

A: Monotonic increase (with no deletion function)

C: Round-robin (with no backup acquisition function)

E: Round-robin (data is output immediately after a specified amount is reached and then a new file is used)

H: Round-robin (existing data is deleted from the target file and then new data is written from the beginning)

J: Control file or temporary file

K: Backup file

Y: Can be deleted at the user's discretion

NE: Must not be deleted, unless error checking is not required.

AD: Deleted when the command is terminated (parentheses enclose the name of the applicable command)

N: Must not be deleted

#

Includes process shutdown when the command is executed under the following conditions:

- The -f and -d options are specified in the dcstop command.

- The `-f` and `-d` options are specified in the `dcsvstop` command.

The table below describes the files that are output by TP1/Server Base (applicable to UNIX version only).

Table F-4: Descriptions of the files output by TP1/Server Base (applicable to UNIX version only)

No.	Name	Related definition	Size	Max files	Description
1	Communication control file	None	Communication message size	Number of active processes	This file is used when communication is performed on the local machine.
2	Backup file for <code>inittab</code>	None	Size of <code>/etc/inittab</code> file	3 generations	This is a backup file for the <code>/etc/inittab</code> file.
3	Control file for <code>dcsvpool</code>	None	0	1 file	This file is used to control OpenTP1 operation.
4	Pipe file for <code>prctee</code>	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.
5	PID storage file for <code>prctee</code> process	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.
6	<code>prctee</code> process start error output file	None	Monotonic increase	1 file	This file is used to output <code>prcd</code> information in the event of a <code>prctee</code> process start error.
7	PAUSE checking file for <code>prcd</code>	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.
8	PID storage file for Linux <code>prcd</code>	None	Several bytes (fixed)	1 file	This file is used to control OpenTP1 operation.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
9	prctee process error output file	None	Monotonic increase	1 file	This file is used to output information about output errors on betran.log.
10	Archive-journ al source node connection information file	<Global archive journal service definition> • jnldfsv definition command Number of resource groups for the global archive journal service	124 bytes	Number of resource groups specified in -a option of jnldfsv definition command in the global archive journal service definition	This is a connection information file for the archive-journal source node. When the global archive journal function is used, connection information for the archive journal node is acquired.
11	Archive journal node connection information file	None	124 bytes x number of archive-journal source nodes to be connected to the archive journal node	Number of archive-jour nal source nodes to be connected to the archive journal node	This is a connection information file for the archive journal node. When the global archive journal function is used, connection information for the archive-journal source node is acquired.

No.	Name	Related definition	Size	Max files	Description
12	Invalid journal information file for archive journal node	<Archive journal service definition> <ul style="list-style-type: none"> • <code>jnl_arc_max_data_size</code> operand Maximum size of transfer data during archive processing • <code>jnl_dual</code> operand Whether dual journal files are to be used 	<code>jnl_arc_max_data_size</code> operand value	For <code>jnl_dual=N</code> : Number of archive-journal source nodes to be connected to the archive journal node For <code>jnl_dual=Y</code> : Number of archive-journal source nodes to be connected to the archive journal node x 2	This is an invalid journal information file for the archive journal node. It is acquired when there is an error in the blocks read when OpenTP1 started the re-run (abnormal).#
13	Invalid journal information file for archive journal file	<Global archive journal service definition> <ul style="list-style-type: none"> • <code>jnl_arc_max_data_size</code> operand Maximum size of transfer data during archive processing 	<code>jnl_arc_max_data_size</code> operand value	Monotonic increase	This is an invalid journal information file for the archive journal file. It is acquired when the <code>jnlis</code> command detects an invalid journal.
14	GWF log file	None	0.1 megabytes	2 generations	This is communication trace information for checking design. Note that the gateway service is not supported by the Linux and Solaris versions. None of the OSs supports 07-00 or later.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
15	UAP trace edit/output file	<p><Process service definition></p> <ul style="list-style-type: none"> prc_coresave_path operand Core file storage path 	Variable	3 generations	This is a UAP trace information file for system and user servers that have been shut down.
16	Backup file for UAP trace data file	<p><User service definition></p> <p><User service default definition></p> <p><System common definition></p> <p><Individual system service definition></p> <ul style="list-style-type: none"> uap_trace_max operand Number of UAP trace records traceinfo_output_file operand Whether UAP and module trace information are to be output to a file 	<p>32-bit version: ((uap_trace_max operand value + 1) x 256) + 128 (bytes)</p> <p>64-bit version: ((uap_trace_max operand value + 1) x 264) + 144 (bytes)</p>	6 generations (3 generations in the case of normal or abnormal termination)	This is a backup file for the source data file for the UAP trace edit/output file to which UAP trace information for system and user servers is output.
17	Backup file for module trace data file	<p><User service definition></p> <p><User service default definition></p> <p><System common definition></p> <p><Individual system service definition></p> <ul style="list-style-type: none"> traceinfo_output_file operand Whether UAP and module trace information are to be output to a file 	<p>32-bit version: 4096 bytes (for the process server, 73696 bytes)</p> <p>64-bit version: 6128 bytes (for the process server, 110528 bytes)</p>	6 generations (3 generations in the case of normal or abnormal termination)	This is a backup file for the source data file for the module trace edit/output file to which module trace information for system and user servers is output.
18	PID file for prcd management process	None	0	As many as the number of processes	This file is used to control OpenTP1 operation.

No.	Name	Related definition	Size	Max files	Description
19	UAP trace data file	<User service definition> <User service default definition> <System common definition> <Individual system service definition> <ul style="list-style-type: none"> • <code>uap_trace_max</code> operand Number of UAP trace records • <code>traceinfo_output_file</code> operand Whether UAP and module trace information is to be output to a file 	32-bit version: $((uap_trace_max\ operand\ value + 1) \times 256) + 128$ (bytes) 64-bit version: $((uap_trace_max\ operand\ value + 1) \times 264) + 144$ (bytes)	1 generation	This is a source data file for the UAP trace edit/output file to which UAP trace information for system and user servers is output.
20	Module trace data file	<User service definition> <User service default definition> <System common definition> <Individual system service definition> <ul style="list-style-type: none"> • <code>traceinfo_output_file</code> operand Whether UAP and module trace information is to be output to a file 	32-bit version: 4096 bytes (for the process server, 73696 bytes) 64-bit version: 6128 bytes (for the process server, 110528 bytes)	1 generation	This is a source data file for the module trace edit/output file to which module trace information for system and user servers is output.
21	DCDIR management information in the same machine	None	Depends on DCDIR registered in the same machine	1 file	This file is used to control OpenTP1 operation.
22	DCDIR management shell in the same machine	None	Depends on DCDIR registered in the same machine	1 file	This file is used to control OpenTP1 operation.
23	Communication control file	<System common definition> <ul style="list-style-type: none"> • <code>rpc_multi_tpl_in_same_host</code> operand Whether on the same host multiple OpenTP1 systems are to be run as the same global domain (group of node names specified in the <code>all_node</code> operand in the system common definition) 	Communication message size	Number of active processes	This file is used when communication is performed on the local machine.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
24	Backup file for dcsetup	None	Depends on length of DCDIR and size of /etc/inittab	12 files	This file is used to control OpenTP1 operation.
25	Realtime output function (betran.log)	None	Monotonic increase (if changed to the wraparound method, 65535 kilobytes)	1 generation (2 generations if changed to the wraparound method)	This is a log file used to output standard output and standard error output information. In versions earlier than 07-00, information about <code>prctee</code> process errors is also output to this file. You can rename the file and change its storage directory and file size. For details, see the software documentation or <i>3.5 Standard output file operations</i> .
26	Temporary file used during execution of <code>namdomainsetup</code> command	None	Size equivalent to number of lines in the <code>hosts</code> file of host names that were specified in the domain data file	1 file	This file is used to control OpenTP1 operation. When the <code>namdomainsetup</code> command is executed, this file is created temporarily to check if host names specified in the domain data file are already registered.
27	Lock file for dcsetup	None	0	1 file	This file is used to control OpenTP1 operation.

No.	Name	Related definition	Size	Max files	Description
28	Temporary file used during execution of <code>jnl-sort</code> command	None	<ul style="list-style-type: none"> <code>sort_I + process ID</code> 256 bytes x number of blocks to be sorted <code>sort_o + process ID</code> 256 bytes x number of blocks to be sorted 	1 file per <code>sort_I + process ID</code> 1 file per <code>sort_o + process ID</code>	This file is used to control OpenTP1 operation. It is a journal block information file before and after blocks are sorted by the OS's <code>sort</code> command.

Legend:

Max files: Maximum number of files

#

This file is acquired even when there is no end block. Therefore, if there was no time to write the end block for a reason such as forced termination of OpenTP1, this file is output whether or not a block error occurred.

(3) TP1/Server Base (applicable to Windows version only)

The table below lists the files that are output by TP1/Server Base (applicable to Windows version only). For details about each file, see Table F-6 *Descriptions of the files output by TP1/Server Base (applicable to Windows version only)*.

Table F-5: List of files output by TP1/Server Base (applicable to Windows version only)

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	Patch log file	TP1/Server Base: %DCDIR%\patchlog base.txt DAM: %DCDIR%\patchlog dam.txt TAM: %DCDIR%\patchlog tam.txt EXT1: %DCDIR%\patchlog ext1.txt HAF: %DCDIR%\patchlog haf.txt	07-00	A	Text	Acquired at the extension of patch execution	NE
2	prctee process error output file	%DCDIR%\spool\be tran.err	05-00-/E	A	Text	When OpenTP1 starts	N
3	Lock file for raw device	%DCDIR%\spool\dc ntbinf\rawlock_ <i>N</i> (<i>N</i> : Drive letter)	06-50	J	Null	When raw device is opened for the first time	N
4	Backup module (UMT) trace file	%DCDIR%\spool\sa ve\trc_zx.umt (<i>z</i> : server name of 1-8 characters, <i>x</i> : serial number)	Initial	I	Binary	When OpenTP1 starts	Y
5	Backup UAP trace file	%DCDIR%\spool\sa ve\trc\ <i>server-name</i> <i>N</i> .uat (<i>N</i> : 1-10)	Initial	C, I	Binary	When a process is shut down [#] and when OpenTP1 starts	NE

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
6	Shared memory file	%DCDIR%\spool\shm\N (N: numeric character beginning at 1)	Initial	J	Binary	When OpenTP1 starts. The next time online operation begins, the file is deleted and then re-created.	N
7	Lock file for GUI	%DCDIR%\tp1_tools\Control.lck %DCDIR%\tp1_tools\Environment.lck	06-50	J	Null	When the GUI function is used. The file is deleted when the GUI function is terminated.	N
8	admshowpp log file	%temp%\admshowpp.log	07-01	J	Text	Acquired at the extension of the dcsetup command	N
9	dcsetup log file	%temp%\dcsetup.log	Initial	A	Text	Acquired at the extension of the installer and patch operation	NE
10	dcsetupml log file	%temp%\dcsetupml.log	06-50	A	Text	Acquired at the extension of the dcsetupml command and installer operation	NE

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
11	trnlncrm log file	%temp%\trnlncrm.log	Initial	A	Text	Acquired at the extension of the trnlncrm command execution, dcsetupml command execution, installer operation, and patch operation	NE
12	Standard output redirect file	<i>redirect_file_name-op erand-valueN</i> (N: 1 or 2) Default: %DCDIR%\spool\pr clogN (N: 1 or 2)	05-00-/E	C, E, H	Text	When OpenTP1 is run	Y
13	UAP trace file	<i>current-work-path\temp\home_z.x_zPP PPP.uat</i> (z: server name of 1-8 characters, PPPPP: PID, x: serial number)	Initial	I	Binary	Created at the extension of the user server startup	Y
14	Module (UMT) trace file	<i>current-work-path\temp\home_z.x_zPP PPP.umt</i> (z: server name of 1-8 characters, PPPPP: PID, x: serial number) For a process server: %DCDIR%\spool\save_prcPPPPPP.umt (PPPPPP: PID)	Initial	I	Binary	Created at the extension of the OpenTP1 startup and user server startup	Y

Legend:

A: Monotonic increase (with no deletion function)

C: Round-robin (with no backup acquisition function)

E: Round-robin (data is output immediately after a specified amount is reached and then a new file is used)

H: Round-robin (existing data is deleted from the target file and then new data is written from the beginning)

I: Round-robin (a new file is used when OpenTP1 starts)

J: Control file or temporary file

Y: Can be deleted at the user's discretion

NE: Must not be deleted, unless error checking is not required.

N: Must not be deleted.

#

Includes process shutdown when the command is executed under the following conditions:

- The `-f` and `-d` options are specified in the `dcstop` command.
- The `-f` and `-d` options are specified in the `dcsvstop` command.

The table below describes the files that are output by TP1/Server Base (applicable to Windows version only).

Table F-6: Descriptions of the files output by TP1/Server Base (applicable to Windows version only)

No.	Name	Related definition	Size	Max files	Description
1	Patch log file	None	Monotonic increase This file is overwritten each time a patch is executed.	5 files (BASE: 1 generation, DAM: 1 generation, TAM: 1 generation, EXT1: 1 generation, HAF: 1 generation)	This is a patch execution log file. It is acquired when a patch is executed.
2	<code>prctee</code> process error output file	None	Monotonic increase	1 file	This file is used to output <code>prctee</code> process error information.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
3	Lock file for raw device	None	0	Number of raw devices used	This file is used to control OpenTP1 operation. It is an internal file created when a raw device is used for the OpenTP1 file system.
4	Backup module (UMT) trace file	None	4160 bytes (for a process server, 73696 bytes)	10 generations x number of active servers	This is a backup file for the module (UMT) trace file.
5	Backup UAP trace file	None	Variable	10 generations	This is a UAP trace information file for system and user servers that have been shut down.
6	Shared memory file	None	Depends on the machine status and OpenTP1 system operating status	1 generation	This file is used to control OpenTP1 operation. It is a shared memory file allocated by OpenTP1.
7	Lock file for GUI	None	0	2 files	This file is used to control OpenTP1 operation. It is created to prevent duplicate startup when OpenTP1's GUI function is used.

No.	Name	Related definition	Size	Max files	Description
8	admshowpp log file	None	During normal operation: 0 bytes In event of an error: Size varies This file is overwritten each time a command is executed.	1 generation	This file is used to control OpenTP1 operation. It is an error log for internal commands. Internal commands that execute at the extension of the dcsetup command output logs to this file. In the event of a command error, the error details are output to the file.
9	dcsetup log file	None	Monotonic increase This file is overwritten each time a command is executed.	1 generation	This is a trace file for the dcsetup command. It is acquired at the extension of the installer operation and patch operation.
10	dcsetupml log file	None	Monotonic increase This file is overwritten each time a command is executed.	1 generation	This is a trace file for the dcsetup command. It is acquired at the extension of dcsetupml command execution and installer operation.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
11	trnlncrm log file	None	Monotonic increase This file is overwritten each time a command is executed.	1 generation	This is a trace file for the trnlncrm command. It is acquired at the extension of trnlncrm command execution, dcsetupml command execution, installer operation, and patch operation.
12	Standard output redirect file	<p><System environment definition></p> <ul style="list-style-type: none"> • redirect_file operand Destination of standard output and standard error output for a process under OpenTP1 • redirect_file_name operand Name of the standard output and standard error output file • redirect_file_size operand Size of the standard output and standard error output file 	Maximum value is the redirect_file_size operand value (can be changed to a file that increases monotonically)	2 generations (1 generation if changed to a file that increases monotonically)	This is a log file used to output standard output and standard error output information. This file is output only when Y is specified in the redirect_file operand. This file is not output when the prc_port and rpc_port_base operands are specified in the system common definition.
13	UAP trace file	<p><User service definition></p> <p><User service default definition></p> <ul style="list-style-type: none"> • uap_trace_max operand Maximum number of UAP trace records 	Variable	1 generation x number of active servers	This is a UAP trace file for a process during online operation.
14	Module (UMT) trace file	None	4160 bytes (for a process server, 73696 bytes)	1 generation x number of active servers	This is a module (UMT) trace file for a process during online operation.

Legend:

Max files: Maximum number of files

(4) TP1/FS/Direct Access

The table below lists the files that are output by TP1/FS/Direct Access. For details about each file, see Table F-8 *Descriptions of the files output by TP1/FS/Direct Access*.

Table F-7: List of files output by TP1/FS/Direct Access

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	damd life checking file	\$DCDIR/tmp/ damlive	Initial	J	Binary	When damd is started	N
2	Temporary file during logical file recovery	<i>damfrc-command-execution-directory/</i> <i>.dcdamPPPPPPP</i> (PPPPPP: last 7 digits of PID)	Initial	J	Binary	When the damfrc command is executed with the -c option omitted and an online backup file specified	AD (damfrc)
3	Temporary file during acquisition of online DAM backup	<i>dambkup--o--s-execution-directory/</i> <i>bkTTTTPPPP</i> (TTTT: last 4 digits of the total time in seconds PPPP: last 4 digits of the process ID)	Initial	J	Binary	When the dambkup command is executed with the -o and -s options specified	AD (damfrc)

Legend:

J: Control file or temporary file

AD: Deleted when the command is terminated. Parentheses enclose the name of the applicable command.

N: Must not be deleted.

The table below describes the files that are output by TP1/FS/Direct Access.

Table F-8: Descriptions of the files output by TP1/FS/Direct Access

No.	Name	Related definition	Size	Max files	Description
1	damd life checking file	None	136 bytes	1 file	This file is used to control OpenTP1 operation. It determines whether damd is running.
2	Temporary file during logical file recovery	None	96 bytes	1 file	This file is used to control OpenTP1 operation. When the <code>damfrc</code> command is executed with the <code>-c</code> option omitted and an online backup file specified, internal information is output to a temporary file under the directory where the command was executed.
3	Temporary file during acquisition of online DAM backup	None	Size of DAM file that is to be backed up + 128 bytes	1 file	This file is used to control OpenTP1 operation. When the <code>dambkup</code> command is executed with the <code>-o</code> and <code>-s</code> options specified, this file is used for temporary storage of backup data for the DAM file before the backup data is output to standard output.

Legend:

Max files: Maximum number of files

(5) TP1/FS/Table Access

The table below lists the files that are output by TP1/FS/Table Access. For details about each file, see Table F-10 *Descriptions of the files output by TP1/FS/Table Access*.

Table F-9: List of files output by TP1/FS/Table Access

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	Temporary file during acquisition of online TAM backup	\$DCDIR/tmp/.dctamdN (N: 1-9999999)	Initial	J	Binary	When tambkup command is executed with -o operand specified	AD (tambkup)
2	Shared memory information file (for TAM)	\$DCDIR/tmp/tammenfile	Initial	J	Binary	When OpenTP1 starts	N
3	Temporary file during logical file recovery	tamfrc-command-execution-directory/ .dctamPID	Initial	J	Binary	When tamfrc command is executed with -j option omitted and an online backup file specified	AD (tamfrc)

Legend:

J: Control file or temporary file

AD: Deleted when the command is terminated. Parentheses enclose the name of the applicable command.

N: Must not be deleted.

The table below describes the files that are output by TP1/FS/Table Access.

Table F-10: Descriptions of the files output by TP1/FS/Table Access

No.	Name	Related definition	Size	Max files	Description
1	Temporary file during acquisition of online TAM backup	None	Size of TAM file to be backed up	1 file	This file is used to control OpenTP1 operation. When the <code>tambkup</code> command is executed with the <code>-o</code> operand specified, this file is used for temporary storage of backup data for the TAM file.
2	Shared memory information file (for TAM)	None	4 bytes	1 file	This file is used to control OpenTP1 operation.
3	Temporary file during logical file recovery	None	96 bytes	1 file	This file is used to control OpenTP1 operation. When the <code>tamfrc</code> command is executed with the <code>-j</code> option omitted and an online backup file specified, internal information is output to a temporary file under the directory where the command was executed.

Legend:

Max files: Maximum number of files

(6) TP1/Online Tester

The table below lists the files that are output by TP1/Online Tester. For details about each file, see Table F-12 *Descriptions of the files output by TP1/Online Tester*.

Table F-11: List of files output by TP1/Online Tester

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	XATMI send data file	\$DCDIR/spool/ uto/test-user-ID/ user-service-name/ xsdservice-name	Initial	J	Binary	When a client's XATMI request made by the tpsend function is accepted	Y
2	MCF send message file	\$DCDIR/spool/ uto/test-user-ID/ user-service-name/ sendmsg	Initial	J	Binary	When a client's request made by an MCF function listed in Table F-12 is accepted	Y
3	UTO trace file	\$DCDIR/spool/ uto/test-user-ID/ traceN (N: 1 or 2)	Initial	C, E, H	Binary	When an online tester is used to check UAP operation	Y
4	Temporarily stored data file	\$DCDIR/spool/ uto/test-user-ID/ utotmplogical-termin al-name	Initial	J	Binary	When temporarily stored data is received from a client by issuing the dc_mcf_tempput function	Y

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
5	Test data definition file	File path specified in <code>-o</code> option of <code>utofilcre</code> operand, or File path specified in the test data definition file that was specified in the <code>-e</code> option of the <code>utofilcre</code> operand You can create the following tester files: <ul style="list-style-type: none"> • RPC request data file • RPC response data file • XATMI request data file • XATMI response data file • XATMI receive data file • Asynchronous receive data file • Synchronous receive data file 	Initial	J	Binary	When the <code>utofilcre</code> command is executed	Y
6	MCF receive message file (there are asynchronous and synchronous files)	File path specified in <code>r</code> option of <code>utomsgout</code> command	Initial	J	Binary	When the <code>utomsgout</code> command is executed with the <code>-r</code> operand specified	Y
7	RPC response data file	File path specified for the RPC response data file name in the <code>utosppsvc</code> command	Initial	J	Binary	When the <code>utosppsvc</code> command is executed with the RPC response data file name specified	Y

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
8	UTO trace merge file	File path specified in -o option of utotrcmrg command	Initial	J	Binary	When the utotrcmrg command is executed with the -o operand specified	Y
9	XATMI response data file	File path specified for the XATMI response data file name in the utoxspssvc command	Initial	J	Binary	When the utoxspssvc command is executed with the XATMI response data file name specified	Y

Legend:

C: Round-robin (with no backup acquisition function)

E: Round-robin (data is output immediately after a specified amount is reached and then a new file is used)

H: Round-robin (existing data is deleted from the target file and then new data is written from the beginning)

J: Control file or temporary file

Y: Can be deleted at the user's discretion

The table below describes the files that are output by TP1/Online Tester.

Table F-12: Descriptions of the files output by TP1/Online Tester

No.	Name	Related definition	Size	Max files	Description
1	XATMI send data file	<User service definition> The user server name becomes part of the output path. The service name becomes part of the file name. <Environment variable> The test user ID (DCUTOKEY) setting becomes part of the output path.	XATMI send data length+ 68 bytes	1 file for each set of a test user ID, user server name, and service name	This is an XATMI send data file. When a client issues the tpsend function, the send data is stored in this file.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
2	MCF send message file	<p><Tester service definition></p> <ul style="list-style-type: none"> max_message_file_size operand <p>Maximum size of MCF send message file</p>	max_message_file_size operand value	1 file for each test user ID	<p>This file stores MCF send messages. When the MCF simulation function is used, this file stores send messages when the following functions are issued:</p> <ul style="list-style-type: none"> dc_mcf_execap function dc_mcf_reply function dc_mcf_send function dc_mcf_sendsync function dc_mcf_sendrecv function
3	UTO trace file	<p><Tester service definition></p> <ul style="list-style-type: none"> max_trace_file_size operand <p>Maximum size of file used to collect UAP trace information</p>	max_trace_file_size operand value	2 generations	This is a UTO trace file. It is used to collect UAP trace information.
4	Temporarily stored data file	<p><Tester service definition></p> <p>The utotmp logical terminal name becomes the file name.</p> <p><Environment variable></p> <p>The test user ID (DCUTOKEY) setting becomes part of the output path.</p>	128 bytes	1 file for each set of a test user ID and utotmp logical terminal name	This file is used to control OpenTP1 operation. When a client issues the dc_mcf_temp ut function, the temporarily stored data is stored in the file.

No.	Name	Related definition	Size	Max files	Description
5	<ul style="list-style-type: none"> • RPC request data file • RPC response data file • XATMI request data file • XATMI response data file • XATMI receive data file • Asynchronous receive data file • Synchronous receive data file 	None	Any size	1 file	This is a test data definition file used by the online tester.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
6	MCF receive message file (there are asynchronous and synchronous files)	None	Any size	1 file	<p>This file is used to store MCF receive messages. The MCF receive messages are created by editing the send message information output by the online tester.</p> <ul style="list-style-type: none"> • Asynchronous-type Messages that are passed to MHP's <code>dc_mcf_receive</code> function • Synchronous type Messages that are passed to UAP's functions, such as <code>dc_mcf_recvsync</code> and <code>dc_mcf_sendrecv</code>
7	RPC response data file	None	Any size	1 file	<p>This is an RPC response data file. When the server UAP simulation function with an RPC interface is used, this file is used as the response data that is returned to the program requesting the service.</p>

No.	Name	Related definition	Size	Max files	Description
8	UTO trace merge file	None	Total size of files specified for trace file names	1 file	This is a UTO trace merge file. It is created when the <code>utotrcmrg</code> command is used to merge multiple UTO trace files into one file.
9	XATMI response data file	Overall typed buffer definition	Undefined because the file is created on the basis of the typed buffer definition file and XATMI request data file	1 file	This is an XATMI response data file. It is used as the receive data during service execution and as the response data after service execution.

Legend:

Max files: Maximum number of files

(7) TP1/Message Control

The table below lists the files that are output by TP1/Message Control. For details about each file, see Table F-14 *Descriptions of the files output by TP1/Message Control*.

Table F-13: List of files output by TP1/Message Control

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
1	Backup file for system service common information definition file	<code>\$DCDIR/.mcfdef/mcf</code>	05-05	K	Text	When the <code>dcsetup</code> command is executed with the <code>-d</code> operand specified	N

F. List of Files Output by OpenTP1

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
2	Backup file for system service information definition file	\$DCDIR/.mcfdef/mcfu*	05-05	K	Text	When the dcsetup command is executed with the -d operand specified	N
3	Backup file for executable MCF communication server program	\$DCDIR/.mcfserv/mcfu*	05-05	K	Binary	When the dcsetup command is executed with the -d operand specified	N
4	MCF dump file ^{#1}	\$DCDIR/spool/mcfdKAXZZ (K: process identifier, AXX: MCF identifier, ZZ: 01-99)	Initial	H, I	Binary	When an error (such as an internal conflict) is detected in TP1/Message Control or TP1/Messaging	NE
5	MCF trace file ^{#1}	\$DCDIR/spool/mcftAXZZ (AXX: MCF identifier, ZZ: 01-99)	Initial	H, I	Binary	When a trace buffer becomes full in shared memory while an MCF communication process is running	NE
6	MCF performance verification trace information file ^{#1}	\$DCDIR/spool/dcmcfinf/_mc_NNN (NNN: 000-256)	07-01	D, F	Binary	When a trace buffer becomes full while an MCF communication process or UAP is running	NE

No.	Name	File name or directory name	Version	Type	File format	Acquisition timing	Deletable
7	Backup file when preparation for MCF configuration change is stopped ^{#2}	DCMCFQUEBAK value in the system service common information definition Default: \$DCDIR/spool/mcf/mcfquebak	07-03	K	Binary	When preparation for MCF configuration change is stopped (by executing <code>dstop -b -q</code>)	Y ^{#3}
8	Backup file for MCF performance verification trace information file ^{#1}	\$DCDIR/spool/save/_mc_MMM.bkN (MMM: 001-256, N: 1 or 2)	07-01	K	Binary	When OpenTP1 has stopped	NE
9	Mapping error information file ^{#4}	\$DCDIR/spool/save/dcmap_errXXX (XXX: process ID of TP1/NET/XMAP3 MCF communication process)	Initial	H, I	Text	When an error occurs during mapping	NE

Legend:

D: Round-robin (with a backup acquisition function)

F: Round-robin (data is output immediately before a specified amount is reached and then a new file is used)

H: Round-robin (existing data is deleted from the target file and then new data is written from the beginning)

I: Round-robin (a new file is used when OpenTP1 starts)

K: Backup file

Y: Can be deleted at the user's discretion

NE: Must not be deleted, unless error checking is not required.

N: Must not be deleted.

#1

This file is output for TP1/Message Control, TP1/NET/Library, protocol products, and TP1/Messaging.

#2

This file is output for TP1/Message Control, TP1/NET/Library, and protocol products.

#3

You can delete the file after the MCF configuration change restart is complete.

#4

This file is output for TP1/NET/XMAP3.

The table below describes the files that are output by TP1/Message Control.

Table F-14: Descriptions of the files output by TP1/Message Control

No.	Name	Related definition	Size	Max files	Description
1	Backup file for system service common information definition file	System service common information definition (\$DCDIR/lib/sysconf/mcf)	Size of the \$DCDIR/lib/sysconf/mcf file	1 file	This is a backup file for the system service common information definition.
2	Backup file for system service information definition file	System service information definition	Size of the system service information definition file	Number of system service information definition files	This is a backup file for the system service information definition.
3	Backup file for executable MCF communication server program	None	Size of the MCF communication process program or MCF application start process program (\$DCDIR/lib/servers/mcfu*)	Number of executable MCF communication server programs	This is a backup file for the executable MCF communication server program.

No.	Name	Related definition	Size	Max files	Description
4	MCF dump file ^{#1}	None	Size determined using the formula for estimating the size of an MCF dump file in the <i>MCF Release Notes</i> .	Following number of files output by MCF manager process, MCF communication process, and UAP: TP1/NET/Library version earlier than 07-03: 99 files TP1/NET/Library 07-04 or later: 3 files	This file is used to obtain memory information in the event of an MCF failure. It is used for troubleshooting purposes. This file is not output during normal processing.
5	MCF trace file ^{#1}	<p><Trace definition in MCF communication configuration definition></p> <p>-t option in <code>mcftrc</code> definition command</p> <ul style="list-style-type: none"> • size operand Trace buffer size • bufcnt operand Number of trace buffers • trccnt operand Number of trace files • disk operand Whether disk output feature is to be used <p>-m option in <code>mcftrc</code> definition command Handling of the number of files</p>	size value x bufcnt value	When <code>del</code> is specified in -m option: trccnt operand value When <code>off</code> is specified in -m option: 99 files	This file is used to collect send/receive data and events that occur within MCF. It is used for troubleshooting purposes. If <code>disk=no</code> is specified (or omitted), this trace file is not created. In version 07-00 or later, <code>disk=yes</code> is the default setting.

F. List of Files Output by OpenTP1

No.	Name	Related definition	Size	Max files	Description
6	MCF performance verification trace information file ^{#1}	<p><System common definition></p> <ul style="list-style-type: none"> prf_trace operand Whether PRF traces are to be acquired <p><User service default definition></p> <ul style="list-style-type: none"> mcf_prf_trace operand <p><User service definition></p> <ul style="list-style-type: none"> mcf_prf_trace operand Whether PRF traces are to be acquired for each user server <p><MCF performance verification trace definition></p> <ul style="list-style-type: none"> prf_file_size operand Trace file size prf_file_count operand Number of trace file generations <p><System service definition></p> <ul style="list-style-type: none"> mcf_prf_trace operand Whether PRF traces are to be acquired for each MCF communication process <p><System service common information definition></p> <ul style="list-style-type: none"> mcf_prf_trace_level operand Acquisition level for the MCF performance verification trace information 	prf_file_size operand value	prf_file_count operand value	This is an MCF performance verification trace information file. It is used for performance verification and troubleshooting purposes. Performance verification trace information is acquired when Y is specified in the prf_trace operand in the system common definition and 00000001 is specified in the mcf_prf_trace_level operand in the system service common information definition. The default is that this file is not acquired.
7	Backup file when the preparation for MCF configuration change is stopped ^{#2}	<p><System service common information definition></p> <pre>putenv DCMCFQUEBAK backup-file-name~ <path name><<\$DCDIR/spool/mcf/mcfquebak>></pre>	296 + number of queue groups x 16 + number of logical terminals x 20 + number of service groups x 44 + number of untransmitted or unprocessed messages x (average message length + 1036) (bytes)	1 file	This is a backup file for untransmitted and unprocessed messages when the MCF configuration change restart function is used.

No.	Name	Related definition	Size	Max files	Description
8	Backup file for MCF performance verification trace information file ^{#1}	Same as for MCF performance verification trace information file (No. 6)	prf_file_size operand value	prf_file_count operand value x 2 generations	This is a backup file for the MCF performance verification trace information file. It is obtained when Y is specified in the prf_trace operand and 00000001 is specified in the mcf_prf_trace_level operand in the system common definition. In version 07-02 or later, this backup file is not obtained when prf_trace_backup=N is specified in the performance verification trace definition.
9	Mapping error information file ^{#3}	<Mapping service attribute definition> ERRLOG4 = { YES NO } <ul style="list-style-type: none"> • YES Acquires error information. • NO Does not acquire error information if a mapping error with error code 4 (minor error) occurs. 	Depends on size of logical map area and size of physical map file	Number of MCF communication processes ^{#4}	This file is used to acquire error information in the event of a mapping error. It is used for troubleshooting purposes.

Legend:

Max files: Maximum number of files

#1

This file is output for TP1/Message Control, TP1/NET/Library, protocol products, and TP1/Messaging.

F. List of Files Output by OpenTP1

#2

This file is output for TP1/Message Control, TP1/NET/Library, and protocol products.

#3

This file is output for TP1/NET/XMAP3.

#4

This is the number of TP1/NET/XMAP3 MCF communication processes in which a mapping error has occurred. This value increases each time the process ID of the mapping service process changes at the start of OpenTP1.

G. Estimating the Size of the Physical File for a Message Queue

The UAP can use a disk queue for the input/output queue for exchanging logical messages under the message control facility (TP1/Message Control). The disk queue causes the logical messages to be written on a physical file for message queue. The record length and number of records specified in the queinit command determine the size of this file. This section provides the estimation formulae for the physical file for message queue.

G.1 Estimation formula of the record length

$$\text{Record length} = \lceil (\lceil a + 976 + 8 \times b + 36 \times c \rceil / c \lceil) / d \lceil \times d$$

\lceil \lceil : Rounded up to the nearest whole integer.

a: Size of a logical message

b: Number of segments of a logical message

c: Number of records for storing a logical message

d: Sector length specified during initialization of the file system

G.2 Estimation formula of the number of records

$$\text{Number of records} = e \times c + 1$$

e: Maximum number of logical messages stored in the physical file

c: Number of records for storing a logical message

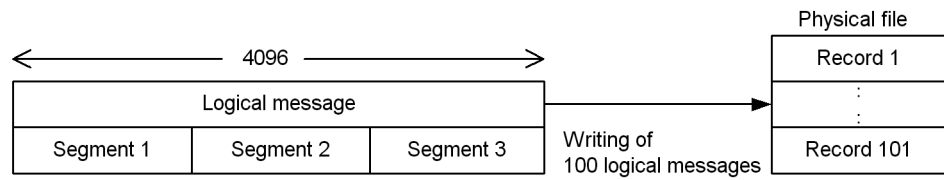
Note

One logical message can be stored in one record or over several records. One record can store only one logical message. When handling logical messages with different sizes, we recommend to assign two or more records for storing a logical message in the above estimation so that a logical message can be stored in multiple records.

G.3 Estimation example

Estimate the record length and number of records of the physical file for storing 100 logical messages, where the message size is 4096 bytes and the number of segments of a logical message is 3. One record should store one logical message. The sector length of the file system is 512 bytes.

G. Estimating the Size of the Physical File for a Message Queue



- Record length= $\lceil (\lceil (4096 + 976 + 8 \times 3 + 36 \times 1) / 1 \rceil) / 512 \rceil \times 512 = 5632$
 - Number of records = $100 \times 1 + 1 = 101$
- $\lceil \lceil \rceil$: Rounded up to the nearest integer.

H. Estimating the Size of OpenTP1 Files

This appendix describes the formulae for estimating the size of OpenTP1 files.

H.1 Estimating the size of a status file

To estimate the size of a status file, you need to know the number of records in the target status file. To find this number, determine the number of bytes in the status file used by each key and the number of keys. Add the calculated number of records to the number of records used by other services.

Determine the number of bytes in the status file used by each key and the number keys for each archive source node, archive node, and MCF service. Specify the results in the `-c` option in the `stsinit` command for initializing a status file. Note that the calculated results indicate the minimum requirements. For safety, determine a value that is 1.2 times larger than the calculated size.

The formula for determining the number of records in a status file is as follows:

Total number of records required for all services + number of records required for managing the status file

Legend:

Number of records required for each service

Use the following formula to determine the number of records required for each service:

$$\uparrow (\text{Number of bytes in the status file used by each key} / (\text{record length}^{\#} - 40)) \uparrow \times \text{number of keys}$$

\uparrow \uparrow : Decimals are rounded up to a whole number.

Number of records required for managing the status file

Use the following formula to determine the number of records required to manage the status file:

When the record length is 512 or less

$$\uparrow (\text{number of records required for each service} / 472) \uparrow + 23$$

When the record length is 513 or greater

$$\uparrow (\text{total number of records required for all services} + 8776 + \uparrow (8192 / (\text{record length}^{\#} - 40)) \uparrow \times 16) / (\text{record length}^{\#} - 40) \uparrow$$

\uparrow \uparrow : Decimals are rounded up to a whole number.

#

Unit for record length: bytes

The following subsections indicate the number of bytes in the status file used by each key and the number of keys.

■ **For archive source nodes**

The following table shows the number of bytes in the status file used by each key and the number of keys.

Table H-1: Number of bytes in the status file used by each key of each service and the number of keys (for an archive source node)

Service	Number of bytes in the status file used by each key (Unit: bytes)	Number of keys
System manager	$A \times 128 + 2432$	1
Schedule service	$B \times 144 + 128$	1
Transaction journal service	2048	1
Transaction service	$1280 \times (\text{number of RMs} + \text{RM extensions}) + 2048$	1
DAM service	$(C + D) \times 288 + 256$	1
Message queue service	$E \times 120 + 36$	1
	$(F + G) \times 40$	1
	$(G + H + 1) \times 48$	1
Journal service	8192	1
	8192	J
	17008	L
	$\sum^L (M \times 384 + 592)$	L
Checkpoint dump service	90	L
	126	L
	32	L
TAM service	$N \times 136 + 64$	1
Online tester	$P \times 160 + 132$	1

Service	Number of bytes in the status file used by each key (Unit: bytes)	Number of keys
Server recovery journal service	$512 \times Q + 128$	1
IST service	$V \times 8$	1
XA resource service	64	1

Legend:

A: Value specified in the `server_count` operand in the system environment definition

B: Value specified in the `scd_hold_recovery_count` operand in the schedule service definition

C: Total number of logical files specified in the `damfile` definition command in the DAM service definition

D: Value specified in the `dam_added_file` operand in the DAM service definition

E: Total number of physical files specified in the `quegrp` definition command in the message queue service definition

F: Value specified in the `-g` option of the `mcfmexp` command in the MCF manager definition

G: Value specified in the `-l` option of the `mcfmexp` command in the MCF manager definition

H: Total number of MCF communication services specified in the `mcfmname` command in the MCF manager definition

J: $\lceil (5 + 3 \times K) / 63 \rceil - 1$ ($\lceil \rceil$: Decimals are rounded up to a whole number.)

K: Total number of journal file groups specified in the `jnladdfg` definition command in the system journal service definition

L: Total number of files containing the checkpoint dump service definition specified in `jnldfs -c` in the journal service definition

M: Total number of journal file groups specified in the `jnladdfg` definition command in the checkpoint dump service definition

N: Value specified in the `tam_max_tblnum` operand in the TAM service definition

P: Value specified in the `uto_server_count` operand in the tester service definition

Q: Total number of files containing the checkpoint dump service definition specified in `jnldfs -c` in the journal service definition

V: $\downarrow ((2347 + \Sigma(W \times Y) + \Sigma(16 \times Y)) / 8) \downarrow$ (\downarrow : Decimals are truncated.)

W: Record length in each IST table specified in the IST service definition (unit: bytes)

Y: Number of records in each IST table specified in the IST service definition

■ **For archive nodes**

Table H-2: Number of bytes in the status file used by each key of each service and the number of keys (for an archive node)

Service	Number of bytes in the status file used by each key (Unit: bytes)	Number of keys
System manager	$A \times 128 + 2432$	1
Journal service	8192	1
	8192	R

Legend:

A: Value specified in the `server_count` operand in the system environment definition

R: $S + \uparrow (\sum_1^S (5 + T \times (1 + 2 \times U))) / 63 \uparrow$ (\uparrow : Decimals are raised to a unit.)

S: Total number of files containing the archive journal service definition specified in `jnldfs -a` in the global archive journal service definition

T: Total number of journal-related file groups specified in `jnladdfg` in the archive journal service definition

U: Value specified in the `jnl_max_file_dispersion` operand in the archive journal service definition

To determine the number of records required by the MCF service, use the number of bytes in the status file used by each key and the number of keys.

The following table shows the number of bytes in the status file used by each key of the MCF service and the number of keys.

■ For MCF services

Table H-3: Number of bytes in the status file used by each key and the number of keys (for an MCF service)

Item	Number of bytes in the status file used by each key (Unit: bytes)		Number of keys
MCF service	L x 128		L
	224		L
	Different communication processes	80 x $\uparrow M / 1818 \uparrow$ x 48	$\uparrow (\uparrow M / 1818 \uparrow) / 1363 \uparrow$
		80 x $\uparrow N / 1818 \uparrow$ x 48	$\uparrow (\uparrow N / 1818 \uparrow) / 1363 \uparrow$
		MIN (48 + M x 36: 65536)	$\uparrow M / 1818 \uparrow$
		MIN (48 + N x 36: 65536)	$\uparrow N / 1818 \uparrow$
		80 + $\uparrow O / 4092 \uparrow$ x 48	$\uparrow (\uparrow O / 4092 \uparrow) / 1363 \uparrow$
		80+ $\uparrow P / 4092 \uparrow$ x 48	$\uparrow (\uparrow P / 4092 \uparrow) / 1363 \uparrow$
		MIN (48 + O x 36: 65536)	$\uparrow O / 4092 \uparrow$
		MIN (48 + P x 36: 65536)	$\uparrow P / 4092 \uparrow$
		4	1
		64	1
48 + 1024 x Q	1		

Legend:

L: Number of MCF communication service names specified in the communication service definition (`mcfmcname` definition command) in the MCF manager definition

M: Upper limit for the number of service groups specified in the status inheritance definition (`mcfmsts` definition command) in the MCF manager definition

N: Upper limit for the number of services specified in the status inheritance definition (`mcfmsts` definition command) in the MCF manager definition

O: Upper limit for the number of applications specified in the status inheritance definition (`mcfsts` definition command) in the MCF communication definition

P: Upper limit for the number of logical terminals specified in the status inheritance definition (`mcf tsts` definition command) in the MCF communication definition

Q: Number of descriptions in the mapping service definition for the mapping service identifier corresponding to the mapping service attribute definition in which YES is specified for the `PATHSTS` operand

$\text{MIN}(x : y) : x \text{ or } y, \text{ whichever is smaller}$

↑ ↑: Decimals are rounded up to whole numbers.

Note

For details about the size of the status file of the MQA service and the ISAM service, see the manual *OpenTP1 TPI/Message Queue User's Guide*, the manual *Indexed Sequential Access Method ISAM*, or the *Release Notes* of the applicable product, and determine the size.

H.2 Estimating the size of a system journal file

To estimate the size of a system journal file, determine the total number of journals that will be generated from the beginning to the end of online operation. Use the following formula to determine this value:

(Average number of journals per transaction) × (total number of transactions from the start to end of online operation)

+ number of journals for the statistics that are acquired by the `dcstats` command between the start and end of online operation

+ (number of journals per OpenTP1 RPC issued by a UAP)

× (total number of OpenTP1 RPCs issued from a UAP between the start and end of online operation) (unit: bytes)

The details of the values used in the formula are as follows:

- Average number of journals per transaction

Use the total number of journal records that are generated based on the conditions listed in *Table H-4* and the following formula to determine this value:

$$\{ \uparrow (j + 240) / 4096 \uparrow \times n_a + \uparrow ((a - j \times n_a) + 240) / 4096 \uparrow \} \times 4096 \text{ (unit: bytes)}$$

Legend:

j: Value specified in the `jnl_max_datasize` operand in the system journal service definition

a: Total number of journal records that are generated based on the applicable conditions listed in *Table H-4*

$$n_a: \downarrow a / j \downarrow$$

↑ ↑: Decimals are rounded up to a whole number.

↓ ↓: Decimals are truncated.

For details about the number of journals per transaction to be used by TP1/Message Queue, see the manual *OpenTP1 TP1/Message Queue User's Guide* or the *Release Note*.

- Number of journals for the statistics that are acquired using the `dcstats` command between the start and end of online operation

Use the value shown in *Table H-5* and the following formula to determine this value:

$$(\uparrow (j + 240) / 4096 \uparrow \times n_b + \uparrow ((b - j \times n_b) + 240) / 4096 \uparrow) \times 4096 \times N \text{ (unit: bytes)}$$

Legend:

j: Value specified in the `jnl_max_datasize` operand in the system journal service definition

b: Length of one statistics journal determined from *Table H-5*

$$n_b: \downarrow b / j \downarrow$$

N: Length of time from the start to the end of system statistics acquisition divided by the length of time specified in the `-m` option of the `dcstats` command (number of times statistics are acquired)

↑ ↑: Decimals are rounded up to a whole number.

- Number of journals per OpenTP1 RPC issued by a UAP

Use the value shown in *Table H-6* and the formula shown below to determine this value. This value is 0 when the UAP does not acquire statistics using an RPC.

$$\{ \uparrow (j + 240) / 4096 \uparrow \times n_c + \uparrow ((c - j \times n_c) + 240) / 4096 \uparrow \} \times 4096 \text{ (unit: bytes)}$$

Legend:

j: Value specified in the `jnl_max_datasize` operand in the system journal service definition

c: Length of one statistics journal determined from *Table H-6*

$$n_c: \downarrow c / j \downarrow$$

↑ ↑: Decimals are rounded up to a whole number.

↓ ↓: Decimals are truncated.

The total size of journal files actually prepared is based on the result of the following formula:

Total number of journals that are generated during online operation x 1.2 (safety factor) + 81920 (unit: bytes)

Note, however, that the size of one journal file must be larger than the result of the following formula:

$(12 + \uparrow (j + 336) / 4096) \uparrow \times 4096$ (unit: bytes)

Legend:

j: Value specified in the `jnl_max_datasize` operand in the system journal service definition

↑ ↑: Decimals are rounded up to a whole number.

When you use two or more journal files to accommodate the total journal file size calculated above, you do not need to unload journals until online operation ends. If you cannot allocate the calculated total journal file size, use $1/n$ ($n > 0$) of the total size as your total size and allocate the size among two or more journal files.

When you do this, note that the size of one journal file must be greater than the size determined in the above formula. In this case, one journal file is reused n times until the end of online operation. This means the journal file must be unloaded.

To protect against journal file failures, allocate reserve journal files, if possible. A reserve journal file must also have the size shown in the above formula.

If you duplicate journal files by specifying Y in the `jnl_dual` operand in the system journal service definition, two files are required for each journal file described above, doubling the total size of journal files.

The paired journal files must have the same size. Although OpenTP1 will run if the sizes of the paired journal files are different, OpenTP1 uses the smaller of the sizes as the journal files size. This wastes space.

Table H-4: Number of journals per transaction

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
Fixed part in transaction	Journals always generated for each transaction branch	$672 + 56 \times tb + 40 \times tr + 1216 \times (tc + tu) + 8 \times (ts + tt + tu) + 224 \times tx + 208 \times ty + 1384 \times (tz + tv)$	tb: Number of transactional RPCs already issued tr: Number of RMs already accessed tc: Number of RPCs already issued via CRM ts: 0 when $tb = 0$, 1 when $tb > 0$ tt: 0 when $tr = 0$, 1 when $tr > 0$ tu: 0 when $tc = 0$ and the branch that started a transaction was not called by an RPC via CRM, 1 when $tc > 0$ or the branch that started a transaction that was called by an RPC via CRM tx: Number of times RPCs are issued after the UJ is acquired ty: Number of first RMs that are accessed after the UJ is acquired tz: Number of RPCs issued via CRM after the UJ is acquired tv: 0 when $tu = 1$ or $tz = 0$, 1 when $tu = 0$ and $tz > 0$	cj, hj, pj, tj
When UAP history information is acquired	For each acquisition for a transaction branch that acquires UAP history information	$114 + ul$	ul: Length of UAP history information	uj
When a DAM is used	For each transaction branch when a DAM file is updated within the transaction	$128 + \Sigma (24 + du)$	du: Block length in the DAM file Σ : Total number of updates in the applicable transaction branch	fj

H. Estimating the Size of OpenTP1 Files

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
	When the number of DAM file updates in a transaction branch exceeds the number of batch update blocks (value specified in the <code>dam_update_block</code> operand in the DAM service definition) if <code>flush</code> is specified in the <code>dam_update_block_over</code> operand in the DAM service definition and the DAM file is updated in the transaction	$128 + \Sigma (24 + du \times 2)$	du: Block length in the DAM file Σ : Number of updates in the applicable transaction branch (number of updates after the previous journal output)	fj
	When a DAM file specified with the <code>-d</code> option is updated if the <code>-d</code> option is specified in the <code>damfile</code> definition command in the DAM service definition At each interval specified in the <code>dam_io_interval</code> operand in the DAM service definition and each time validation of a checkpoint dump is completed	64×2	--	xj
	Each time online backup (<code>dambkup</code> command with the <code>-o</code> option specified) is executed	64	--	xj
When a TAM is used	For each transaction branch when a TAM file is updated in the transaction	$128 + \Sigma (64 + tu \times 2)$	tu: Record length in the TAM file Σ : Total number of updates in the applicable transaction branch	cj
When MCF is used	Each time a message is sent to a logical terminal for which <code>mcfalcle -o aj=yes</code> is specified in the MCF communication configuration definition and the transmission is completed (in a communication process)	176	--	aj

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
	Each time a logical message with a serial number is sent from a UAP process within a transaction to an output logical terminal	160	--	cj
	Each time a logical message with a serial number is sent from a UAP process outside a transaction (mcfallcap -n trnmode=notrn is specified) to an output terminal	128	--	
	Each time a communication process receives a logical message	$\uparrow \{88 + 24 \times (\uparrow mc / ql \uparrow) + mc^{#1}\} / 4 \uparrow \times 4$	mc: Length of the received logical message ql: Length of the queue file physical record	
	Each time a communication process sends a logical message	112	--	
	Each time a transaction branch is completed	$\uparrow \{260 + 24 \times Sn + \sum (24 \times (\uparrow mc / ql \uparrow)) + \sum mc^{#2}\} / 4 \uparrow \times 4$	mc: Length of the sent logical message ql: Length of the queue file physical record Sn: Number of logical messages received by the applicable transaction \sum : Total number of logical messages sent by the applicable transaction	
	Each time a communication process uses the disk queue to send a logical message with a serial number	144	--	--
	Each time a message is received from an application (MHP service) for which mcfallcap -j gj=yes is specified	$\uparrow (204 + mg) / 4 \uparrow \times 4$	mg: Length of the received message segment	gj

H. Estimating the Size of OpenTP1 Files

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
	Each time a communication process receives a message from an application for which <code>mcfallcap -j ij=yes</code> is specified	$\Sigma (\uparrow (172 + si) / 4 \uparrow \times 4)$	<p>si: Same as the input message length for a single segment</p> <p>Same as each applicable segment length for multiple segments</p> <p>Σ: Total number of journals acquired for all the segments</p>	ij
	Each time a communication process receives a message from a logical terminal or sends a message to the logical terminal before the <code>mcftdctmj</code> command is executed on the logical terminal on which the <code>mftactmj</code> command was previously executed	$\Sigma (\uparrow (180 + sm) / 4 \uparrow \times 4)$	<p>sm: Same as the length of the received or sent message for a single segment</p> <p>Same as the length of each applicable segment for multiple segments</p> <p>Σ: Total number of journals acquired for all the segments</p>	mj
	<p>For an SPP: Each time an application program is started, a message is resent, or a message is sent</p> <p>For an MHP: Each time an application (MHP service) for which <code>mcfallcap -j oj=yes</code> is specified starts an application program, sends a response message, resends a message, or sends a message</p>	$\uparrow (204 + mo) / 4 \uparrow \times 4$	mo: Length of the sent message segment	oj

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
Statistics	For each transaction for which a value other than none is specified in the <code>trn_statistics_item</code> operand in the user service definition when <code>Y</code> is specified in the <code>trn_tran_statistics</code> operand in the transaction service definition or the <code>trnstics</code> command is executed with the <code>-s</code> option specified (acquisition of statistics starts from the next transaction)	276	--	sj
When ISAM is used	For each transaction branch when an ISAM file is updated in the transaction	$104 + \sum 1 (52)$ $+ \sum 2 (40 + iu \times 2) + \sum 3 (40)$ $+ \sum 4 (32 + iu)$ $+ \sum 5 (24)$	iu: Record length in the ISAM file $\sum 1$: Total number of times open and close functions are issued in the applicable transaction branch $\sum 2$: Total number of times the record update function is issued in the applicable transaction branch $\sum 3$: Total number of times the lock function is issued in the applicable transaction branch $\sum 4$: Total number of times the record addition function is issued in the applicable transaction branch $\sum 5$: Total number of times the record deletion function is issued in the applicable transaction branch	fj

Legend:

--: Not applicable.

#1

Add this value if the value specified in the `que_io_maxreclsize` operand in the message queue service definition is greater than `mc`.

#2

Add this value if the value specified in the `que_io_maxreclsize` operand in the message queue service definition is greater than Σmc .

Table H-5: Number of journals for statistics to be acquired by using the `dcstats` command

Type	Condition for generating journals	Journal record length	Description of variables	Type of journal record acquired when condition is satisfied
Statistics	At each interval specified in the <code>m</code> option of the <code>dcstats</code> command after the start of journal output for system statistics by execution of the <code>dcstats</code> command before execution of the <code>dcstats</code> command with the <code>-r</code> option specified	$(80 + ak \times 24) \times as$	<p><code>ak</code>: Sum of the following values when the types of statistics specified in <code>-k</code> of the <code>dcstats</code> command are replaced by the following values (117 when <code>-k</code> is omitted):</p> <p><code>rpc</code>: 4 <code>lck</code>: 3 <code>prc</code>: 3 <code>nam</code>: 3 <code>que</code>: 11 <code>scd</code>: 4 <code>mcf</code>: 5 <code>mqa</code>: 12 <code>dam</code>: 9 <code>tam</code>: 6 <code>trn</code>: 2 <code>cpd</code>: 2 <code>jnl</code>: 21 <code>osl</code>: 19 <code>ist</code>: 7 <code>xat</code>: 3</p> <p><code>as</code>: Total number of user servers + 1 when <code>-a</code> is specified in the <code>dcstats</code> command</p> <p>Number of user servers specified in the <code>dcstats</code> command + 1 when <code>-s</code> is specified in the <code>dcstats</code> command</p> <p>Specified number of user servers when user servers are specified in the <code>dcstats</code> command without the <code>-a</code> or <code>-s</code> option specified, or 1 when no user servers are specified</p>	sj

Table H-6: Number of journals per RPC

Type	Condition for generating journals	Journal record length	Description of variables	Type for journal record acquired when condition is satisfied
Statistics	For each RPC when γ is specified in the <code>rpc_delay_statistics</code> operand in the system common definition	80	--	sj
	For each RPC when γ is specified in the <code>rpc_response_statistics</code> operand in the user service definition	272 x 2	--	

Legend:

--: Not applicable.

H.3 Estimating the size of a server recovery journal file

When you use the independent restart facility and the system switchover facility of the MCF service, you must create the server recovery journal file for the MCF service as an OpenTP1 file beforehand. When you do so, use the following formula to determine the size of the server recovery journal file:

$J \times C_i \times C_g$ (unit: bytes)

Legend:

J: Value specified in the `jnl_max_datasize` operand in the system journal service definition

C_i : Value specified in the `jnl_cdinterval` operand in the system journal service definition

C_g : Value specified in the `assurance_count` operand in the checkpoint dump service definition

H.4 Estimating the size of a checkpoint dump file

The following table shows how to estimate the size of a checkpoint dump file.

Table H-7: Size of a checkpoint dump file

Item	Size of checkpoint dump file (Unit: bytes)
MCF service	$(\lceil Sz1 / 4096 \rceil + 3) \times 4096$
Transaction journal service	$(\lceil Sz2 / 4096 \rceil + 3) \times 4096$

Legend:

$\lceil \rceil$: Decimals are rounded up to a whole number.

$$Sz1: (\lceil (Cn \times 68 + 84) / 32 \rceil \times 32 + \sum^{Pn} (Pr \times 32))$$

$$Sz2: (640 \times (rm \text{ count} + 1) + 64 \times Br) \times Tr$$

Cn: Number of logical terminals with serial numbers specified in the `-n` option of the `mcfmcomn` definition command in the MCF manager definition

Pn: Number of physical files specified in the message queue service definition

Pr: Number of records specified in the `-n` option of the `queinit` command

Br: Maximum number of connection branches per transaction specified in the `trn_max_subordinate_count` operand in the transaction service definition

rm count: Number of resource managers (only RMs of OpenTP1)

Tr: Number of transaction branches specified in the `trn_tran_process_count` operand in the transaction service definition

Note that archive journal nodes do not require checkpoint dump files.

For details about the size of a checkpoint dump file used by TP1/Message Queue, see the manual *OpenTP1 TP1/Message Queue User's Guide* or the *Release Note*.

H.5 Estimating the size of an archive journal file

This section describes the formula for estimating the size of an archive journal file.

First, use the following formula to determine the total number of journals to be archived in an archive journal file:

Σ (Total number of journals that are generated by each archive source node during online operation) (unit: bytes)

For details about the total number of journals generated by each archive source node during online operation, see *H.2 Estimating the size of a system journal file*. Use the estimated size of the system journal file of each node.

The formula for determining the total size of archive journal files that are actually prepared is as follows:

Total number of journals to be archived in archive journal files x 1.2 (safety factor) + 16384 (unit: bytes)

Note, however, that the size of one archive journal file must be larger than the result of the following formula:

$(4 + 256 \times \text{number of archive source nodes whose data is to be archived in this archive journal file}) \times 4096$ (unit: bytes)

When you use two or more journal files to accommodate the total journal file size calculated above, you do not need to unload journals until online operation ends. If you cannot allocate the calculated total archive journal file size, use $1/n$ ($n > 0$) of the total size as your total size and allocate the size among two or more archive journal files. When you do this, one archive journal file is reused for n times until the end of online operation. This means the archive journal file must be unloaded.

To protect against archive journal file failures, allocate reserve archive journal files, if possible. A reserve archive journal file must also have the size shown in the above formula for determining the size of one archive journal file.

Note that one archive journal file makes up a file group. This means that the size of one archive journal file also indicates the size of the file group.

When you duplicate archive journal files by specifying \forall in the `jnl_dual` operand in the archive journal service definition, two files are required for each archive journal file described above, doubling the total size of archive journal files. The paired archive journal files must have the same size. Although OpenTP1 will run if the sizes of the paired archive journal files are different, OpenTP1 uses the smaller of the sizes as the archive journal file size. This wastes space.

If you do not use the parallel access facility of the archive journal file, the size of the file group is the size of the physical file.

When you use the parallel access facility of the archive journal file, you must divide the size of the file group by $m^\#$ ($m^\# > 1$), use the divided size as the size of an archive journal file, and prepare archive journal files for at least the minimum number of dispersions for each file group. In this case, the size of the physical file is the size of the element file.

#

m is the value specified in the `jnl_max_file_dispersion` operand (maximum number of dispersions for enabling parallel access) in the archive journal service definition.

H.6 Estimating the size of a DAM file

This section describes the formula for estimating the size of a DAM file.

The maximum size of a DAM file is determined as follows:

Maximum size of a DAM file = maximum value satisfying $F - D > 0$

- $F = Msize \times 1048576 - [4096 + 2 \times \{ \uparrow 64 \times Fnum / Ssize \uparrow \times Ssize + \uparrow (12 \times Fnum + 28) / Ssize \uparrow \times Ssize + \uparrow 16008 / Ssize \uparrow \times Ssize \}]$
- $D = (Blen + 8) \times (Bnum + 1)$

Legend:

Msize: Value specified in the `-n` option of the `filmkfs` command

Fnum: Value specified in the `-l` option of the `filmkfs` command

Ssize: Value specified in the `-s` option of the `filmkfs` command

Blen: Length of a DAM file block to be created (unit: bytes)

Bnum: Number of DAM file blocks to be created

\uparrow \uparrow : Decimals are rounded up to a whole number.

For the command arguments of the `damload` command, specify values of `Blen` (block length) and `Bnum` (number of blocks) that satisfy the above formula.

In a DAM file, the size available to users is determined by `Blen` (block length) \times `Bnum` (number of blocks) (unit: bytes).

H.7 Estimating the size of a TAM file

This section describes the formula for estimating the size of a TAM file.

A TAM file contains a data section for records, a management section, and an index section. Therefore, when you calculate the size of a TAM file, you need to consider the sizes of the management section and the index section as well. Note that the size also differs depending on the index type used (tree format or hash format).

The formula for determining the size of a TAM file is as follows:

Management section size + index section size + data section size

Legend:

Management section size

Sector length of the OpenTP1 file system

Index section size (for tree format)

$\uparrow s ((\uparrow w (152 + \text{key length}) \uparrow w) \times \text{maximum number of records}) \uparrow s$

Index section size (for hash format)

$$\uparrow s ((\uparrow w (152 + \text{key length}) \uparrow w) \times (M + N)) \uparrow s$$

Data section size (for tree format)

$$\uparrow s ((\text{record length}^{\#} + 5) \times \text{maximum number of records}) \uparrow s$$

Data section size (for hash format)

$$\uparrow s ((\text{record length}^{\#} + 5) \times (M + N)) \uparrow s$$

M

Result of multiplying the maximum number of records by the hash entry usage rate and truncating the value to a whole number (when the result is less than 1, 1 is assumed)

N

Maximum number of records - 1

S

Sector length of the OpenTP1 file system

$$\uparrow s \uparrow s$$

Value rounded up to an integral multiple of the sector length of the OpenTP1 file system

$$\uparrow w \uparrow w$$

For 32 bits: Value rounded up to a multiple of 4

For 64 bits: Value rounded up to a multiple of 8

#

Unit of record length: bytes

I. Estimating the Number of Record Locks

OpenTP1 locks files to ensure exclusive access during online operation. When locking is used, a record lock is registered in the OpenTP1 system. The maximum number of record locks that can be registered is determined by $\text{maxuprc} \times \text{nofiles} \times \text{number of user licenses}$.

This appendix describes the formula for estimating the number of record locks to be registered in OpenTP1.

For nodes other than archive journal nodes

Number of journal files + number of status files
 + total number of DAM files opened by user server processes
 + total number of queue physical files accessed by processes
 + number of user servers that are simultaneously executed
 + value specified in the `trn_recovery_process_count` operand in the transaction service definition + 3
 + total number of TAM files $\times 2$
 + number of queue files of TP1/Message Queue $\times 2$

For an archive journal node

Number of journal files + number of journal server resource groups + number of status files + 3

Legend:

Number of journal files

Sum of the number of `jnladdpf` commands in the system journal service definition or the archive journal service definition and the checkpoint dump service definition (double this value if files are duplicated)

Number of status files

Number specified in the `sts_file_name_[1-7]` operand in the status service definition

Total number of DAM files opened by user server processes (i: number of user server processes)

\sum_1^i (Number of DAM files opened by user server processes)

I. Estimating the Number of Record Locks

Total number of queue physical files accessed by processes (j: number of MCF communication servers, k: number of user servers)

$$\sum_1^j (\text{Number of queue physical files accessed by the MCF communication server})$$
$$+ \sum_1^k (\text{number of queue physical files accessed by user servers})$$

+ number of queue physical files

Number of resource groups for the journal server

Number of resource groups specified in the `jnldfs -a` definition command in the global archive journal service definition

J. Estimating the Resources Used by UNIX Message Exchange Functions

OpenTP1 uses the UNIX message exchange functions (inter-process communication functionality enabled by system calls issued by the `msgsnd` and `msgrcv` functions) to input and output schedules of user servers, input and output performance verification trace data, and input journal data to and output journal data from journal files.

The following table shows how to estimate the size of each resource used by OpenTP1. For details about the system parameters, see the *Release Notes* and OS documentation.

Resource	Archive source node	Archive node
Number of message IDs	Number of dispersions \times JDUAL + SVMAX + PRFNUM + 7 When using multiple OpenTP1 installations, multiply the result by the number of OpenTP1 environments.	r $\sum_{i=1} (\text{Number of dispersions} \times \text{JDUAL}) + \text{PRFNUM} + 1$
Size of one message	56 bytes	56 bytes
Total number of bytes in the maximum queued message per message ID	Larger of the values determined from formulas A and B: Formula A 56 \times number of file groups for system journal files \times number of dispersions \times JDUAL Formula B (PROC + maximum number of service requests that are simultaneously generated for one service group + number of server processes that simultaneously end abnormally + 4) \times 56	56 \times number of file groups for archive journal files \times number of dispersions \times JDUAL
Maximum number of queued OpenTP1 messages	(PROC \times 2 + a + 2 + PRFNUM \times 8) + number of server processes that terminate abnormally simultaneously When using multiple OpenTP1 installations, multiply the result by the number of OpenTP1 environments.	Number of file groups for archive journal files \times number of dispersions \times JDUAL + 19

Legend:

Number of dispersions: Value of the `jnl_max_file_dispersion` operand in the system journal service definition or the archive journal service definition

`SVMAX`: Value specified in the `scd_server_count` operand in the schedule service definition

`PRFNUM`: 9 when Y is specified in the `prf_trace` operand in the system common definition; 0 when N is specified in the `prf_trace` operand in the system common definition

`JDUAL`: 2 when Y is specified in the `jnl_dual` operand in the system journal service definition; 1 when N is specified in the `jnl_dual` operand in the system journal service definition

`r`: Number of resource groups specified in the `-a` option of the `jnldfs` definition command in the global archive journal service definition

`PROC`: 1 when Y is specified in the `prc_recovery_resident` operand in the process service definition; number of user servers to be started when N is specified in the `prc_recovery_resident` operand in the process service definition

`a`: Maximum number of service requests that can be generated for OpenTP1 concurrently = (maximum number of concurrently generated service requests x number of SPPs) + (number of file groups for system journal files x number of dispersions x `JDUAL`)

K. Estimating the Size of an OpenTP1 File System

You specify the size of an OpenTP1 file system in the `-n` option of the `filmkfs` command. The formula for estimating the size of an OpenTP1 file system is as follows:

$$\lceil ((T + (t \times 2 \times s) + 4096 + 2 \times (\lceil s \rceil (64 \times L) \lceil s \rceil + \lceil s \rceil (12 \times L + 28) \lceil s \rceil + \lceil s \rceil (16008) \lceil s \rceil)) / (1024 \times 1024)) \rceil$$

Legend:

$\lceil \lceil \rceil$: Decimals are rounded up to a whole number.

T : Total size of all OpenTP1 files

t : Number of T files

s : Value specified in the `-s` option of the `filmkfs` command (512 for a UNIX file)

$\lceil s \rceil$: Value rounded up to a multiple of the value specified in the `-s` option of the `filmkfs` command (512 for a UNIX file)

L : Value specified in the `-l` option of the `filmkfs` command

The total size of OpenTP1 files consists of the sizes of the different types of OpenTP1 files shown in the following table.

The formulas shown below do not contain the management information (2-sector area) that is added when an OpenTP1 file is created. Therefore, the actual size of a created file is the result of any of the following formulas plus the length of the two sectors.

Table K-1: Formulas for estimating the sizes of OpenTP1 files

OpenTP1 file name	Formula for estimating the file size (Unit: bytes)
Status file	Value specified in the <code>-s</code> option of <code>stsinit</code> x value specified in the <code>-c</code> option of <code>stsinit</code>
<ul style="list-style-type: none"> • System journal file • Checkpoint dump file • Archive journal file • Transaction recovery file • Server recovery journal file 	Value specified in the <code>-n</code> option of the <code>jnlinit</code> command x 4096 + 12288
DAM file	(Number of blocks specified in the <code>damload</code> command + 1) x (block length specified in the <code>damload</code> command + 8)

OpenTP1 file name	Formula for estimating the file size (Unit: bytes)
TAM file	For tree format $S + \lceil s \lceil (\lceil w (152 + L) \lceil w) \times M \lceil s + \lceil s ((R + 5) \times M) \lceil s$ For hash format $S + \lceil s (\lceil w (152 + L) \lceil w) \times (T + M - 1) \lceil s + \lceil s ((R + 5) \times (T + M - 1)) \lceil s$
Message queue file	Value specified in the <code>-s</code> option of the <code>queinit</code> command \times value specified in the <code>-n</code> option of the <code>queinit</code> command
XAR file	Value specified in the <code>-n</code> option of the <code>xarinit</code> command \times value specified in the <code>-s</code> option of the <code>xarinit</code> command (if the <code>-s</code> option is omitted, the sector length in the OpenTP1 file system)
MQA file	For details about how to estimate the size of a queue file, see the manual <i>OpenTP1 TP1/Message Queue User's Guide</i> .
ISAM file	For details about how to estimate the size of an ISAM file, see the manual <i>Indexed Sequential Access Method ISAM</i> .

Legend:

S: Sector length of the OpenTP1 file system (512 for a UNIX file)

$\lceil s \lceil s$: Value rounded up to a multiple of S

$\lceil w \lceil w$: Value rounded up to a multiple of 4

L: Value specified in the `-l` option of the `tamcre` command

M: Value specified in the `-m` option of the `tamcre` command

R: Value specified in the `-r` option of the `tamcre` command

T: Result of $M \times U / 100$ rounded truncated to a whole number (M = 1 when the result is less than 1)

U: Value specified in the `-u` option of the `tamcre` command

L. Acquiring Performance Verification Traces

This appendix describes the timing for acquisition of performance verification trace information and MCF performance verification trace information, as well as the actual information that is acquired for each event ID. For MCF performance verification trace information, this appendix also describes the MCF-specific dump output information.

(1) Acquiring performance verification trace information

OpenTP1 acquires the trace information at the timings of the event IDs listed in the following table. Also listed is the trace data length that is required for one transaction.

Table L-1: Timing to acquire the performance verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0x1000	Immediately before sending the service request by <code>dc_rpc_call()</code>	128
0x1001	Before calling the service function by <code>dc_rpc_mainloop()</code>	128
0x1002	After the service function is returned by <code>dc_rpc_mainloop()</code>	64
0x1003	Immediately after responding to the client by <code>dc_rpc_mainloop()</code>	64
0x1004	Immediately before <code>dc_rpc_call()</code> returns. Acquires the traces regardless of the synchronous, asynchronous or non-response type.	192
0x1005	When accepting the asynchronous rpc response message	128
0x1006	Entry to the <code>dc_rpc_poll_any_replies()</code> function	64
0x1007	Immediately before <code>dc_rpc_poll_any_replies()</code> returns	64
0x1008	When discarding the <code>dc_rpc_discard_further_replies()</code> receive message	128
0x1009	When discarding the <code>dc_rpc_discard_specific_reply()</code> receive message	128
0x2000	Before notifying (<code>msgsnd</code>) the service function to a process when the service function is in the same node	64
0x2001	Immediately before sending the service function to the server node when the service function is in another node	64
0x2002	When receiving the service request that has been sent	128
0x2003	Before notifying (<code>msgsnd</code>) the service function to a process	64

L. Acquiring Performance Verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0x2004	When transferring the service function to another node (A service is requested to this node, but the service function is not executable so that it is sent to another node.)	64
0x2005	Immediately after getting the service request from the scheduling queue	128
0x2007	When receiving the service request that has been transferred from another node	128
0x4000	When sending the prepare message	192
0x4001	When receiving the prepare message	192
0x4002	When sending the prepare completion message	192
0x4003	When receiving the prepare completion message	192
0x4004	When sending the commit message	192
0x4005	When receiving the commit message	192
0x4006	When sending the commit completion message	192
0x4007	When receiving the commit completion message	192
0x4008	When sending the rollback message	192
0x4009	When receiving the rollback message	192
0x400a	When sending the rollback completion message	192
0x400b	When receiving the rollback completion message	192
0x400c	When sending the read-only message	192
0x400d	When receiving the read-only message	192
0x400e	When sending the recovery message	192
0x400f	When receiving the recovery message	192
0x4010	When sending the recovery response message	192
0x4011	When receiving the recovery response message	192
0x4012	When sending the address resolve message	192
0x4013	When receiving the address resolve message	192
0x4014	When sending the RM thread message	192

Event ID	Timing	Trace data length (unit: bytes)
0x4015	When receiving the RM thread message	192
0x4016	When sending the heuristic message	192
0x4017	When receiving the heuristic message	192
0x4100	When starting the transaction	192
0x4150	When terminating the transaction	192
0x4a00	Immediately after issuing a request to start a transaction branch	128
0x4a01	Immediately before the request to start a transaction branch returns	128
0x4a02	Immediately after issuing a request for RPC execution from inside a transaction branch	128
0x4a03	Immediately before the request for RPC execution from inside a transaction branch returns	128
0x4a04	Immediately after issuing a request to end a transaction branch	128
0x4a05	Immediately before the request to end a transaction branch returns	128
0x4a06	Immediately after issuing a request to prepare a transaction branch for commitment	128
0x4a07	Immediately before a request to prepare a transaction branch for commitment returns	128
0x4a08	Immediately after issuing a request to commit a transaction branch	128
0x4a09	Immediately before a commit request for a transaction branch returns	128
0x4a0a	Immediately after issuing a request to roll back a transaction branch	128
0x4a0b	Immediately before a rollback request for a transaction branch returns	128
0x4a0c	Immediately after issuing a request to notify about a transaction branch in Prepared or Heuristically Completed status	64
0x4a0d	Immediately before the request to notify about a transaction branch in Prepared or Heuristically Completed status returns	64
0x4a0e	Immediately after issuing a request to discard a transaction branch in Heuristically Completed status	128
0x4a0f	Immediately before a request to discard a transaction branch in Heuristically Completed status returns	128
0x4b00	Immediately before the transaction branch starts	64

L. Acquiring Performance Verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0x4b01	Immediately after the transaction branch starts	64
0x4b02	Immediately before executing an RPC call from inside a transaction branch	64
0x4b03	Immediately after executing an RPC call from inside a transaction branch	64
0x4b04	Immediately before ending a transaction branch	64
0x4b05	Immediately after ending a transaction branch	64
0x4b06	Immediately before a transaction branch is prepared for commitment	64
0x4b07	Immediately after a transaction branch is prepared for commitment	64
0x4b08	Immediately before a transaction branch is committed	64
0x4b09	Immediately after a transaction branch is committed	64
0x4b0a	Immediately before a transaction branch is rolled back	64
0x4b0b	Immediately after a transaction branch is rolled back	64
0x4b0c	Immediately before notifying a transaction branch in Prepared or Heuristically Completed status	64
0x4b0d	Immediately after notifying a transaction branch in Prepared or Heuristically Completed status	64
0x4b0e	Immediately before discarding a transaction branch in Heuristically Completed status	64
0x4b0f	Immediately after discarding a transaction branch in Heuristically Completed status	64
0x5001	When receiving the connection establishment request from the client when using the remote API facility	64
0x5002	Before sending the RAP-processing server allocation request from the RAP-processing listener to the RAP-processing server when using the remote API facility	64
0x5003	After receiving the inheritance request for the connection from the RAP-processing listener to the RAP-processing server	64
0x5004	When receiving the API substitution execution request from the client when using the remote API facility	64
0x5005	Before sending the API substitution execution response when using the remote API facility	64

Event ID	Timing	Trace data length (unit: bytes)
0x5006	When receiving the RAP-processing server deallocation or disconnection request from the client when using the remote API facility	64
0x5007	When receiving the RAP-processing server deallocation response from the RAP-processing server when using the remote API facility	64
0x5008	When sending the RAP-processing server deallocation response from the RAP-processing listener to the RAP-processing client when using the remote API facility	64
0x6400	When accepting a resource lock request	128
0x6401	When a resource lock request returns	128
0x6410	Immediately before a lock wait	128
0x6411	Immediately after a lock wait is released	128
0x6420	When accepting a request to release lock with a resource name specified	128
0x6421	When a request to release lock with a resource name specified returns	128
0x6430	When accepting a request to release lock for all resources	128
0x6431	When a request to release lock for all resources returns	128
0x6805	When input from an OpenTP1 file is completed	192
0x6807	When output to an OpenTP1 file is completed	192
0x6905	When <code>read()</code> system call processing on an OpenTP1 file is completed	192
0x6907	When <code>write()</code> system call processing on an OpenTP1 file is completed	192
0x6909	When <code>lseek()</code> system call processing on an OpenTP1 file is completed	192
0xb001	When a process is generated (child process)	64
0xb002	When a process expires	64
0xb003	When a process is generated (parent process)	64
0xb010	Entry to process generation request	64
0xb012		
0xb011	Exit from process generation request	64
0xb013		
0xb014	Entry to process information registration request	64

L. Acquiring Performance Verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0xb015	Exit from process information registration request	64
0xb016	Entry to process information registration request (for TP1/Resource Manager Monitor)	64
0xb017	Exit from process information registration request (for TP1/Resource Manager Monitor)	64
0xb018	Entry to notification of process startup completion	64
0xb019	Exit from notification of process startup completion	64
0xb01a	Entry to waiting for notification of process startup completion	64
0xb01b	Exit from waiting for notification of process startup completion	64
0xb01c	Entry to notification of process termination	64
0xb01d	Exit from notification of process termination	64
0xb01e	Entry to notification of process termination (for TP1/Resource Manager Monitor)	64
0xb01f	Exit from notification of process termination (for TP1/Resource Manager Monitor)	64
0xb020	Entry to waiting for notification of process termination	64
0xb021	Exit from waiting for notification of process termination	64
0xb022	Entry to request to forcibly terminate a process	64
0xb024		
0xb023	Exit from request to forcibly terminate a process	64
0xb025		
0xb026	Entry to instruction to terminate a process service	64
0xb027	Exit from instruction to terminate a process service	64
0xb110	Entry to process-generating processing of process server	64
0xb111	Exit from process-generating processing of process server	64
0xb114	Entry to process information registration processing of process server	64
0xb115	Exit from process information registration processing of process server	64

Event ID	Timing	Trace data length (unit: bytes)
0xb118	Entry to acceptance of process startup completion notification process server	64
0xb119	Exit from acceptance of process startup completion notification process server	64
0xb11a	Entry to waiting for process startup completion notification processing of process server	64
0xb11b	Exit from waiting for process startup completion notification processing of process server	64
0xb11c	Entry to reception of process termination notification process server	64
0xb11d	Exit from reception of process termination notification process server	64
0xb120	Entry to process termination notification wait processing of process server	64
0xb121	Exit from process termination notification wait processing of process server	64
0xb122	Entry to forcible process termination processing of process server	64
0xb124		
0xb123	Exit from forcible process termination processing of process server	64
0xb125		
0xb126	Entry to process service termination processing	64
0xb127	Exit from process service termination processing	64
0xb130	Entry to regular fixed-interval processing by process service	64
0xb131	Exit from regular fixed-interval processing by process service	64
0xb132	Entry to process termination detection processing by process service	64
0xb133	Exit from process termination detection processing by process service	64
0xc001	After creating a journal record header	128
0xc002	After creating a journal block header	128
0xc101	Immediately before sending a message to an I/O process	128
0xc102	Immediately after receiving a response message from an I/O process	64
0xc103	Immediately before issuing a service request to an I/O process by RPC	128

L. Acquiring Performance Verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0xc104	Immediately after receiving an RPC response from an I/O process	64
0xc201	Immediately after receiving an I/O request message	128
0xc202	Immediately before performing output to a journal file	64
0xc203	Immediately after performing output to a journal file	64
0xc204	Immediately before sending an I/O termination message to the I/O-requesting process	64
0xc205	Immediately after receiving an I/O-requesting RPC	128
0xc206	Immediately before sending an I/O-termination RPC response to the I/O-requesting process	128
0xc301	When journal buffering processing starts	64
0xc302	When journal buffering processing ends	64
0xc401	Immediately before waiting for an available journal buffer	64
0xc402	Immediately after waiting for an available journal buffer	64
0xd000	Immediately before the return of the MQPUT call of TP1/Message Queue	192
0xd001	Immediately before the return of the MQGET call of TP1/Message Queue	256
0xd002	Immediately before the return of the function of TP1/Message Queue for registering transferred messages	192
0xd003	Immediately before the return of the function of TP1/Message Queue for collecting transferred messages	256
0xf000	After the name server receives a request to send service information from a process on the local node	128
0xf001	Before the name server sends service information to a remote node	128
0xf002	After the name server sends service information to remote node	128
0xf003	After the name server receives service information from a remote node	128
0xf004	After the name server receives a request to delete service information from a process on the local node	128
0xf005	Before the name server sends a request to delete service information to a remote node	128
0xf006	After the name server sends a request to delete service information to a remote node	128

Event ID	Timing	Trace data length (unit: bytes)
0xf007	After the name server receives a request to delete service information from a remote node	128
0xf008	After the name server receives a request to delete service information from a process on the local node	128
0xf009	Before the name server sends a request to delete service information to a remote node	128
0xf00a	After the name server sends a request to delete service information to a remote node	128
0xf00b	After the name server receives a request to modify user server load information from a process on the local node	128
0xf00c	Before the name server sends a request to modify user server load information to a remote node	128
0xf00d	After the name server sends a request to modify user server load information to a remote node	128
0xf00e	After the name server receives a request to modify user server load information from a remote node	128
0xf010	After the name server receives a request to search service information	128
0xf011	Before the name server sends a request to search service information to a remote node	128
0xf012	After the name server sends a request to search service information to a remote node	128
0xf013	After the name server receives a request to search service information from a remote node	128
0xf014	Before the name server sends the results of searching service information to a remote node	128
0xf015	After the name server receives the results of searching service information from a remote node	128
0xf016	Before the name server sends the results of searching service information	128
0xf017	After the name server receives the results of searching service information from a remote node	128
0xf018	Before the name server sends a request to search service information to a remote node	128
0xf019	After the name server sends a request to search service information to a remote node	128

L. Acquiring Performance Verification Traces

Event ID	Timing	Trace data length (unit: bytes)
0xf01a	After the name server receives the results of searching service information from a remote node	128
0xf020	After the name server receives a request to search service information during RPC execution	128
0xf021	Before the name server sends a request to search service information to a remote node during RPC execution	128
0xf022	After the name server sends a request to search service information to a remote node during RPC execution	128
0xf023	After the name server receives a request to search service information from a remote node during RPC execution	128
0xf024	Before the name server sends a request to search service information to a remote node during RPC execution	128
0xf025	After the name server receives the results of searching service information from a remote node during RPC execution	128
0xf026	Before the name server sends the results of searching service information during RPC execution	128
0xf027	Before the name server sends a request to search service information to a remote node during RPC execution, when γ is specified for the <code>name_global_lookup</code> operand	128
0xf028	After the name server sends a request to search service information to a remote node during RPC execution, when γ is specified for the <code>name_global_lookup</code> operand	128
0xf029	Before the name server sends the results of searching service information to a remote node during RPC execution	128
0xf100	After a node is deleted from the RPC suppression list	128
0xf101	After a node is added to the RPC suppression list	128
0xf102	After service information is deleted from the global cache (when a node is detected to have gone down)	128
0xf103	After service information is added to the global cache	128
0xf104	After service information is deleted from the global cache (situations not related to detection of nodes going down)	128
0xf105	After modifying the load information of a user server operating on a remote node	128
0xf106	After starting service information search processing	128

Event ID	Timing	Trace data length (unit: bytes)
0xf107	Before ending service information search processing	128
0xf108	After starting service information search processing during RPC execution	128
0xf109	Before ending service information search processing during RPC execution	128
0xf200	Before sending a request to send service information to the name server on the local node	128
0xf201	After sending a request to send service information to the name server on the local node	128
0xf202	Before sending a request to delete service information to the name server on the local node	128
0xf203	After sending a request to delete service information to the name server on the local node	128
0xf204	Before sending a request to delete service information to the name server on the local node	64
0xf205	After sending a request to delete service information to the name server on the local node	64
0xf206	Before sending a request to modify user server load information to the name server on the local node	128
0xf207	After sending a request to modify user server load information to the name server on the local node	128
0xf210	Before sending a request to search service information to the name server on the local node	128
0xf211	After sending a request to search service information to the name server on the local node	128
0xf212	After receiving the results of searching service information from the name server on the local node	128
0xf213	Before sending a request to search service information to the name server on the local node	128
0xf214	After sending a request to search service information to the name server on the local node	128
0xf215	After receiving the results of searching service information from the name server on the local node	128
0xf216	Before sending a request to search service information to the name server on the local node at RPC execution	128

Event ID	Timing	Trace data length (unit: bytes)
0xf217	After sending a request to search service information to the name server on the local node at RPC execution	128
0xf218	After receiving the results of searching service information from the name server on the local node at RPC execution	128

The following table lists the contents acquired for each event ID.

Table L-2: Information acquired for each event ID (performance verification traces)

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0x1000	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N
0x1001	Y	Y	N	Y	Y	Y	Y	N	N	N	N	N
0x1002	Y	Y	N	Y	Y	N	N	Y	N	N	N	N
0x1003	Y	Y	N	Y	Y	N	N	Y	N	N	N	N
0x1004	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N
0x1005	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N
0x1006	Y	N	N	Y	Y	N	N	N	N	N	N	N
0x1007	Y	Y	N	Y	Y	N	N	Y	N	N	N	N
0x1008	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N
0x1009	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N
0x2000	Y	Y	N	Y	Y	N	N	N	N	N	N	N
0x2001	Y	Y	Y	Y	Y	N	N	N	N	N	N	N
0x2002	Y	Y	N	Y	Y	Y	Y	N	N	N	N	N
0x2003	Y	Y	N	Y	Y	N	N	N	N	N	N	N
0x2004	Y	Y	Y	Y	Y	N	N	N	N	N	N	N
0x2005	Y	Y	N	Y	Y	N	N	N	N	N	N	N
0x2007	Y	Y	N	Y	Y	Y	Y	N	N	N	N	N
0x4000	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0x4001	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4002	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4003	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4004	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4005	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4006	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4007	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4008	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4009	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x400a	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x400b	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x400c	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x400d	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x400e	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x400f	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4010	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4011	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4012	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4013	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4014	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4015	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4016	Y	Y	Y	Y	N	N	N	Y	Y	N	N	N
0x4017	Y	Y	N	Y	N	N	N	Y	Y	N	N	N
0x4100	Y	N	N	Y	N	N	N	N	Y	N	N	N
0x4150	Y	N	N	Y	N	N	N	Y	Y	N	N	N
0x4a00	N	N	N	N	N	N	N	N	N	N	N	N

L. Acquiring Performance Verification Traces

Event ID	Clf ID	Clf com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0x4a01	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a02	N	N	N	N	N	N	N	N	N	N	N	N
0x4a03	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a04	N	N	N	N	N	N	N	N	N	N	N	N
0x4a05	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a06	N	N	N	N	N	N	N	N	N	N	N	N
0x4a07	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a08	N	N	N	N	N	N	N	N	N	N	N	N
0x4a09	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a0a	N	N	N	N	N	N	N	N	N	N	N	N
0x4a0b	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a0c	N	N	N	N	N	N	N	N	N	N	N	N
0x4a0d	N	N	N	N	N	N	N	Y	N	N	N	N
0x4a0e	N	N	N	N	N	N	N	N	N	N	N	N
0x4a0f	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b00	N	N	N	N	N	N	N	N	N	N	N	N
0x4b01	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b02	N	N	N	N	N	N	N	N	N	N	N	N
0x4b03	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b04	N	N	N	N	N	N	N	N	N	N	N	N
0x4b05	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b06	N	N	N	N	N	N	N	N	N	N	N	N
0x4b07	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b08	N	N	N	N	N	N	N	N	N	N	N	N
0x4b09	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b0a	N	N	N	N	N	N	N	N	N	N	N	N

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0x4b0b	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b0c	N	N	N	N	N	N	N	N	N	N	N	N
0x4b0d	N	N	N	N	N	N	N	Y	N	N	N	N
0x4b0e	N	N	N	N	N	N	N	N	N	N	N	N
0x4b0f	N	N	N	N	N	N	N	Y	N	N	N	N
0x5001	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5002	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5003	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5004	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5005	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5006	N	Y	N	Y	Y	N	N	N	N	N	N	N
0x5007	N	Y	N	Y	Y	N	N	Y	N	N	N	N
0x5008	N	Y	N	Y	Y	N	N	Y	N	N	N	N
0x6400	N	N	N	N	N	N	N	N	Y	N	N	N
0x6401	N	N	N	N	N	N	N	Y	Y	N	N	N
0x6410	N	N	N	N	N	N	N	N	N	N	N	N
0x6411	N	N	N	N	N	N	N	Y	N	N	N	N
0x6420	N	N	N	N	N	N	N	N	Y	N	N	N
0x6421	N	N	N	N	N	N	N	Y	Y	N	N	N
0x6430	N	N	N	N	N	N	N	N	Y	N	N	N
0x6431	N	N	N	N	N	N	N	Y	Y	N	N	N
0x6805	N	N	N	N	N	N	N	Y	N	N	N	N
0x6807	N	N	N	N	N	N	N	Y	N	N	N	N
0x6905	N	N	N	N	N	N	N	Y	N	N	N	N
0x6907	N	N	N	N	N	N	N	Y	N	N	N	N
0x6909	N	N	N	N	N	N	N	Y	N	N	N	N

L. Acquiring Performance Verification Traces

Event ID	Clf ID	Clf com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xb001	N	N	N	N	N	N	N	N	N	N	N	N
0xb002	N	N	N	N	N	N	N	N	N	N	N	N
0xb003	N	N	N	N	N	N	N	N	N	N	N	N
0xb010	N	N	N	N	N	N	N	N	N	N	N	N
0xb011	N	N	N	N	N	N	N	N	N	N	N	N
0xb012	N	N	N	N	N	N	N	N	N	N	N	N
0xb013	N	N	N	N	N	N	N	N	N	N	N	N
0xb014	N	N	N	N	N	N	N	N	N	N	N	N
0xb015	N	N	N	N	N	N	N	N	N	N	N	N
0xb016	N	N	N	N	N	N	N	N	N	N	N	N
0xb017	N	N	N	N	N	N	N	N	N	N	N	N
0xb018	N	N	N	N	N	N	N	N	N	N	N	N
0xb019	N	N	N	N	N	N	N	N	N	N	N	N
0xb01a	N	N	N	N	N	N	N	N	N	N	N	N
0xb01b	N	N	N	N	N	N	N	N	N	N	N	N
0xb01c	N	N	N	N	N	N	N	N	N	N	N	N
0xb01d	N	N	N	N	N	N	N	N	N	N	N	N
0xb01e	N	N	N	N	N	N	N	N	N	N	N	N
0xb01f	N	N	N	N	N	N	N	N	N	N	N	N
0xb020	N	N	N	N	N	N	N	N	N	N	N	N
0xb021	N	N	N	N	N	N	N	N	N	N	N	N
0xb022	N	N	N	N	N	N	N	N	N	N	N	N
0xb023	N	N	N	N	N	N	N	N	N	N	N	N
0xb024	N	N	N	N	N	N	N	N	N	N	N	N
0xb025	N	N	N	N	N	N	N	N	N	N	N	N
0xb026	N	N	N	N	N	N	N	N	N	N	N	N

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xb027	N	N	N	N	N	N	N	N	N	N	N	N
0xb110	N	N	N	N	N	N	N	N	N	N	N	N
0xb111	N	N	N	N	N	N	N	N	N	N	N	N
0xb114	N	N	N	N	N	N	N	N	N	N	N	N
0xb115	N	N	N	N	N	N	N	N	N	N	N	N
0xb118	N	N	N	N	N	N	N	N	N	N	N	N
0xb119	N	N	N	N	N	N	N	N	N	N	N	N
0xb11a	N	N	N	N	N	N	N	N	N	N	N	N
0xb11b	N	N	N	N	N	N	N	N	N	N	N	N
0xb11c	N	N	N	N	N	N	N	N	N	N	N	N
0xb11d	N	N	N	N	N	N	N	N	N	N	N	N
0xb120	N	N	N	N	N	N	N	N	N	N	N	N
0xb121	N	N	N	N	N	N	N	N	N	N	N	N
0xb122	N	N	N	N	N	N	N	N	N	N	N	N
0xb123	N	N	N	N	N	N	N	N	N	N	N	N
0xb124	N	N	N	N	N	N	N	N	N	N	N	N
0xb125	N	N	N	N	N	N	N	N	N	N	N	N
0xb126	N	N	N	N	N	N	N	N	N	N	N	N
0xb127	N	N	N	N	N	N	N	N	N	N	N	N
0xb130	N	N	N	N	N	N	N	N	N	N	N	N
0xb131	N	N	N	N	N	N	N	N	N	N	N	N
0xb132	N	N	N	N	N	N	N	N	N	N	N	N
0xb133	N	N	N	N	N	N	N	N	N	N	N	N
0xc001	N	N	N	N	N	N	N	N	N	N	N	N
0xc002	N	N	N	N	N	N	N	N	N	N	N	N
0xc101	N	N	N	N	N	N	N	N	N	N	N	N

L. Acquiring Performance Verification Traces

Event ID	Clf ID	Clf com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xc102	N	N	N	N	N	N	N	N	N	N	N	N
0xc103	N	N	N	N	N	N	N	N	N	N	N	N
0xc104	N	N	N	N	N	N	N	N	N	N	N	N
0xc201	N	N	N	N	N	N	N	N	N	N	N	N
0xc202	N	N	N	N	N	N	N	N	N	N	N	N
0xc203	N	N	N	N	N	N	N	N	N	N	N	N
0xc204	N	N	N	N	N	N	N	N	N	N	N	N
0xc205	N	N	N	N	N	N	N	N	N	N	N	N
0xc206	N	N	N	N	N	N	N	N	N	N	N	N
0xc301	N	N	N	N	N	N	N	N	N	N	N	N
0xc302	N	N	N	N	N	N	N	N	N	N	N	N
0xc401	N	N	N	N	N	N	N	N	N	N	N	N
0xc402	N	N	N	N	N	N	N	N	N	N	N	N
0xd000	N	N	N	N	N	N	N	Y	N	Y	Y	Y
0xd001	N	N	N	N	N	N	N	Y	N	Y	Y	Y
0xd002	N	N	N	N	N	N	N	Y	N	Y	Y	Y
0xd003	N	N	N	N	N	N	N	Y	N	Y	Y	Y
0xf000	N	N	N	N	N	N	N	N	N	N	N	N
0xf001	N	N	N	N	N	N	N	N	N	N	N	N
0xf002	N	N	N	N	N	N	N	Y	N	N	N	N
0xf003	N	N	N	N	N	N	N	N	N	N	N	N
0xf004	N	N	N	N	N	N	N	N	N	N	N	N
0xf005	N	N	N	N	N	N	N	N	N	N	N	N
0xf006	N	N	N	N	N	N	N	Y	N	N	N	N
0xf007	N	N	N	N	N	N	N	N	N	N	N	N
0xf008	N	N	N	N	N	N	N	N	N	N	N	N

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xf009	N	N	N	N	N	N	N	N	N	N	N	N
0xf00a	N	N	N	N	N	N	N	Y	N	N	N	N
0xf00b	N	N	N	N	N	N	N	N	N	N	N	N
0xf00c	N	N	N	N	N	N	N	N	N	N	N	N
0xf00d	N	N	N	N	N	N	N	Y	N	N	N	N
0xf00e	N	N	N	N	N	N	N	N	N	N	N	N
0xf010	N	N	N	N	N	N	N	N	N	N	N	N
0xf011	N	N	N	N	N	N	N	N	N	N	N	N
0xf012	N	N	N	N	N	N	N	Y	N	N	N	N
0xf013	N	N	N	N	N	N	N	N	N	N	N	N
0xf014	N	N	N	N	N	N	N	N	N	N	N	N
0xf015	N	N	N	N	N	N	N	N	N	N	N	N
0xf016	N	N	N	N	N	N	N	N	N	N	N	N
0xf017	N	N	N	N	N	N	N	N	N	N	N	N
0xf018	N	N	N	N	N	N	N	N	N	N	N	N
0xf019	N	N	N	N	N	N	N	Y	N	N	N	N
0xf01a	N	N	N	N	N	N	N	N	N	N	N	N
0xf020	N	N	N	N	N	N	N	N	N	N	N	N
0xf021	N	N	N	N	N	N	N	N	N	N	N	N
0xf022	N	N	N	N	N	N	N	Y	N	N	N	N
0xf023	N	N	N	N	N	N	N	N	N	N	N	N
0xf024	N	N	N	N	N	N	N	N	N	N	N	N
0xf025	N	N	N	N	N	N	N	N	N	N	N	N
0xf026	N	N	N	N	N	N	N	N	N	N	N	N
0xf027	N	N	N	N	N	N	N	N	N	N	N	N
0xf028	N	N	N	N	N	N	N	Y	N	N	N	N

L. Acquiring Performance Verification Traces

Event ID	Clf ID	Clf com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xf029	N	N	N	N	N	N	N	N	N	N	N	N
0xf100	N	N	N	N	N	N	N	N	N	N	N	N
0xf101	N	N	N	N	N	N	N	N	N	N	N	N
0xf102	N	N	N	N	N	N	N	N	N	N	N	N
0xf103	N	N	N	N	N	N	N	N	N	N	N	N
0xf104	N	N	N	N	N	N	N	N	N	N	N	N
0xf105	N	N	N	N	N	N	N	N	N	N	N	N
0xf106	N	N	N	N	N	N	N	N	N	N	N	N
0xf107	N	N	N	N	N	N	N	Y	N	N	N	N
0xf108	N	N	N	N	N	N	N	N	N	N	N	N
0xf109	N	N	N	N	N	N	N	Y	N	N	N	N
0xf200	N	N	N	N	N	N	N	N	N	N	N	N
0xf201	N	N	N	N	N	N	N	Y	N	N	N	N
0xf202	N	N	N	N	N	N	N	N	N	N	N	N
0xf203	N	N	N	N	N	N	N	Y	N	N	N	N
0xf204	N	N	N	N	N	N	N	N	N	N	N	N
0xf205	N	N	N	N	N	N	N	Y	N	N	N	N
0xf206	N	N	N	N	N	N	N	N	N	N	N	N
0xf207	N	N	N	N	N	N	N	Y	N	N	N	N
0xf210	N	N	N	N	N	N	N	N	N	N	N	N
0xf211	N	N	N	N	N	N	N	Y	N	N	N	N
0xf212	N	N	N	N	N	N	N	Y	N	N	N	N
0xf213	N	N	N	N	N	N	N	N	N	N	N	N
0xf214	N	N	N	N	N	N	N	Y	N	N	N	N
0xf215	N	N	N	N	N	N	N	Y	N	N	N	N
0xf216	N	N	N	N	N	N	N	N	N	N	N	N

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID
0xf217	N	N	N	N	N	N	N	Y	N	N	N	N
0xf218	N	N	N	N	N	N	N	Y	N	N	N	N

Legend:

Clt ID: Client OpenTP1 identifier

Clt com num: Client communication number

Svr ID: Server OpenTP1 identifier

Rt ID: Root OpenTP1 identifier

Rt com num: Root communication number

Svc grp: Service group name

Svc: Service name

Ret cd: Return code

Gbl ID: Global transaction identifier

Qu: Queue name

Msg tkn: Message token

Msg ID: Message identifier

Y: Acquires the information.

N: Does not acquire the information.

(2) Acquiring MCF performance verification trace information

OpenTP1 acquires message transfer trace information at the times listed in the table below for each event ID. This table also shows the length of the trace data that is required when each events occurs.

Table L-3: Timing for acquisition of MCF performance verification trace information

Event ID	Timing	Acquisition process	Trace data length (units: bytes)
0xa000	Immediately after receiving a message	C	128
0xa001	Immediately before sending a message	C	128

Event ID	Timing	Acquisition process	Trace data length (units: bytes)
0xa020	Immediately before storing a receive message in the input queue (if an IJ is acquired, immediately before acquiring the IJ)	C	128
0xa021	Immediately after a message is received by an MHP (if a GJ is acquired, immediately before acquiring the GJ)	U	128
0xa022	Immediately after storing a send message in the output queue (if an OJ is acquired, immediately before acquiring the OJ)	U	128
0xa023	Immediately before starting transaction commit processing (immediately before acquiring a PJ)	U	256
0xa024	Immediately after preparations for transaction commit processing are completed (immediately before acquiring an HJ)	U	256
0xa025	Immediately before rolling back a transaction (immediately before acquiring a BJ)	U	256
0xa026	Immediately after synchronization point processing on transaction is finished (immediately before acquiring a TJ)	U	128
0xa027	Immediately after message transmission is completed (if an AJ is acquired, immediately before acquiring the AJ)	C	128
0xa050	Immediately before calling an MHP service function	U	320 ^{#1}
0xa051	Immediately after an MHP service function returns	U	320 ^{#1}
0xa060	Immediately after a function is called ^{#2}	U	128
0xa061	Immediately before a function returns ^{#2}	U	128
0xa070	Immediately before calling a user exit routine	C ^{#3}	128
0xa071	Immediately after a user exit routine returns	C ^{#3}	128

Legend:

C: Acquired by the MCF communication service process

U: Acquired by the user server process

#1

This information is not acquired if the MHP has a non-transaction attribute. In such a case, the trace data length is 192 bytes.

#2

This information might not be acquired if there is a syntax error in a UAP program written in COBOL.

#3

In the case of a user exit routine that edits sequential numbers of send messages, the information is acquired by the user server process.

The following table lists the information acquired for each event ID.

Table L-4: Information acquired for each event ID (MCF performance verification traces)

Event ID	Clf ID	Clf com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID	MCF #1
0xa000	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa001	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa020	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa021	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa022	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa023	--	--	--	--	--	--	--	--	Y	--	--	--	Y
0xa024	--	--	--	--	--	--	--	--	Y	--	--	--	Y
0xa025	--	--	--	--	--	--	--	--	Y	--	--	--	Y
0xa026	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa027	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa050	--	--	--	--	--	Y	Y	--	Y#1	--	--	--	--
0xa051	--	--	--	--	--	Y	Y	--	Y#1	--	--	--	--
0xa060	--	--	--	--	--	--	--	--	--	--	--	--	Y
0xa061	--	--	--	--	--	--	--	Y	--	--	--	--	Y
0xa070	--	--	--	--	--	--	--	--	--	--	--	--	Y

Event ID	Clt ID	Clt com num	Svr ID	Rt ID	Rt com num	Svc grp	Svc	Ret cd	Gbl ID	Qu	Msg tkn	Msg ID	MCF #1
0xa071	--	--	--	--	--	--	--	Y#2	--	--	--	--	Y

Legend:

Clt ID: Client OpenTP1 identifier

Clt com num: Client communication number

Svr ID: Server OpenTP1 identifier

Rt ID: Root OpenTP1 identifier

Rt com num: Root communication number

Svc grp: Service group name

Svc: Service name

Ret cd: Return code

Gbl ID: Global transaction identifier

Qu: Queue name

Msg tkn: Message token

Msg ID: Message identifier

MCF: MCF-specified information

Y: Acquires the information.

--: Does not acquire the information.

#1

For details about the MCF-specific information, see *Table L-5*.

#2

In the case of a user exit routine that edits sequential numbers of send messages, the information is not acquired.

The following table shows the MCF-specific dump output information for each event ID.

Table L-5: Dump output information for each event ID (MCF performance verification trace)

Event ID	Offset				
	0x0000 to 0x0003	0x0004 to 0x0007	0x0008 to 0x000f	0x0010 to 0x0017	0x0018 to 0x001f
0xa000	MCF identifier	Thread ID	--	Input source logical terminal name	Protocol data type ^{#1}
0xa001	MCF identifier	Thread ID	--	Output destination logical terminal name	MAP name, ^{#2} Protocol data type ^{#1}
0xa020	MCF identifier	Thread ID	Application name	Input source logical terminal name	IJ
0xa021	--	--	Application name	Input source logical terminal name	GJ
0xa022	--	--	Application name	Output destination logical terminal name	OJ
0xa023	--	--	--	--	PJ
0xa024	--	--	--	--	HJ
0xa025	--	--	--	--	BJ
0xa026	--	--	--	--	TJ
0xa027	MCF identifier	Thread ID	Application name	Output destination logical terminal name	AJ
0xa050	--	--	--	--	--
0xa051	--	--	--	--	--
0xa060	MCF identifier	Thread ID	--	--	API name ^{#3}
0xa061	MCF identifier	Thread ID	Application name	Logical terminal name	API name ^{#3}

Event ID	Offset				
	0x0000 to 0x0003	0x0004 to 0x0007	0x0008 to 0x000f	0x0010 to 0x0017	0x0018 to 0x001f
0xa070	MCF identifier	Thread ID	--	--	User exit routine name ^{#3}
0xa071	MCF identifier	Thread ID	--	--	User exit routine name ^{#3}

Legend:

--: Information is not acquired.

#1

The protocol data type is acquired only when TP1/NET/OSAS-NIF is used.

#2

The MAP name (map name) is acquired only when TP1/NET/XMAP3 is used.

#3

For details about the API name that is output for each MCF-provided function, see *Table L-6*. For details about the user exit routine name that is output for each user exit routine, see *Table L-7*.

The following table lists the API name that is output for each MCF function.

Table L-6: API name that is output for each MCF function

MCF-provided function	C language library function name	API name
Deletes application timer start requests	dc_mcf_adltap	"TDLTAP"
Reports application information	dc_mcf_ap_info	"APINFO"
Reports application information to UOCs	dc_mcf_ap_info_uoc	"APINFO_U"
Executes a commit operation on an MHP	dc_mcf_commit	"COMMIT"
Stops continuous-inquiry response	dc_mcf_contend	"CONTEND"
Starts an application program	dc_mcf_execap	"EXECAP"
Receives messages	dc_mcf_receive	"RECEIVE"
Receives a synchronous message	dc_mcf_recvsync	"RECVSYNC"

MCF-provided function	C language library function name	API name
Sends a response message	dc_mcf_reply	"REPLY"
Resends a message	dc_mcf_resend	"RESEND"
Executes a rollback operation on an MHP	dc_mcf_rollback	"ROLLBACK"
Sends a message	dc_mcf_send	"SEND"
Sends and receives synchronous messages	dc_mcf_sendrecv	"SENDRECV"
Sends a synchronous message	dc_mcf_sendsync	"SENDSYNC"
Establishes a connection	dc_mcf_tactcn	"TACTCN"
Releases a logical terminal from shutdown status	dc_mcf_tactle	"TACTLE"
Releases a connection	dc_mcf_tdctcn	"TDCTCN"
Shuts down a logical terminal	dc_mcf_tdctle	"TDCTLE"
Deletes a logical terminal's output queue	dc_mcf_tdlqle	"TDLQLE"
Receives temporary stored data	dc_mcf_tempget	"TEMPGET"
Modifies temporary stored data	dc_mcf_tempput	"TEMPPUT"
Cancels user timer monitoring	dc_mcf_timer_cancel	"TIMERCAN"
Sets user timer monitoring	dc_mcf_timer_set	"TIMERSET"
Acquires the connection status	dc_mcf_tlscn	"TLSCN"
Acquires the status of the MCF communication service	dc_mcf_tlscm	"TLSCOM"
Acquires the status of a logical terminal	dc_mcf_tlsle	"TLSLE"
Acquires the acceptance status for a server-type connection establishment request	dc_mcf_tlsln	"TSLN"
Stops accepting a server-type connection establishment request	dc_mcf_tofln	"TOFLN"
Starts accepting a server-type connection establishment request	dc_mcf_tonln	"TONLN"

The following table shows the user exit routine name that is output for each MCF user exit routine.

Table L-7: User exit routine name that is output for each MCF user exit routine

UOC type	UOC name
User exit routine that edits sequential numbers of send messages	"SEND_UOC"
Any other user exit routine	Name of the system-provided variable that sets the address of the user exit routine function

M. Details About Scenario Templates

The details of scenario templates are described in alphabetical order. The last section in this appendix describes the sample scenario template of the Scale Out scenario.

Set 3 as the priority level of scenario templates to prevent the priority level of OpenTP1 commands executed from scenario templates from being lowered. For details about how to set priority levels, see the relevant JP1/AJS2 - Scenario Operation manuals.

Action to be taken if an error occurs

If an error occurs while a scenario is being executed, an error message may appear. Follow the instructions in the error message. For details about error messages, see the manual *OpenTP1 Messages*.

M.1 OpenTP1_AddNode (Add an OpenTP1 node to the domain definition file)

Function

The `OpenTP1_AddNode` scenario template adds the host name and the port number of a newly added OpenTP1 node to the domain definition file on the node on which the domain configuration is to be updated.

To add the host name and the port number of a newly added OpenTP1 node to the domain definition file on the local host, copy the `OpenTP1_AddNode` scenario template to a scenario and execute it as part of the scenario.

When you execute the `OpenTP1_AddNode` scenario template and then the `OpenTP1_UpdateDomain` scenario template, the domain configuration is updated.

To execute the `OpenTP1_AddNode` scenario template, you must specify `Y` for the `name_domain_file_use` operand in the system common definition beforehand.

Input scenario variables

- `DCDIR~<character string of 1-50 characters>`
Specifies an OpenTP1 directory name.
- `DCCONFPATH`
Specifies an OpenTP1 definition file storage directory.
- `HOST_NAME`
Specifies the host name of the OpenTP1 node where scenario jobs are

executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- USER_NAME

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

- ADD_HOST_NAME~<character string of 1-255 characters>

Specifies the host name of the new OpenTP1 node to be added to the OpenTP1 system.

- PORT_NUMBER~((5001-65535))

Specifies the port number of the new OpenTP1 node to be added. This port number is used by the name server.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable

USER_NAME

M.2 OpenTP1_ChangeNodeID (Set the node ID of an OpenTP1 node)

Function

The OpenTP1_ChangeNodeID scenario template sets the node ID of a new OpenTP1 node to be added.

When you execute the OpenTP1_ChangeNodeID scenario template, the node ID specified in the node_id operand in the system common definition is set. Before you execute the OpenTP1_ChangeNodeID scenario template, specify the node_id operand in the system common definition as follows:

```
set node_id = @DCNODE_ID@
```

When you want to change the node ID in the Scale Out scenario or the scenario for setting the OpenTP1 environment, copy the OpenTP1_ChangeNodeID scenario template to the desired scenario and execute it as part of the scenario.

Input scenario variables

- DCDIR~<character string of 1-50 characters>

Specifies an OpenTP1 directory name.

- DCCONFPATH

Specifies an OpenTP1 definition file storage directory.

- HOST_NAME

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- USER_NAME

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

- NODE_ID~<character string of 4 characters>

Specifies the node ID of the OpenTP1 node to be set.

Specify a node ID that is unique in the OpenTP1 system.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable

USER_NAME

M.3 OpenTP1_Deploy (Register OpenTP1)

Function

The `OpenTP1_Deploy` scenario template registers OpenTP1 in the specified OpenTP1 directory in the OS.

The Scale Out scenario and the Rolling Update scenario contain the `OpenTP1_Deploy` scenario template for registration of OpenTP1 in the OS. The `OpenTP1_Deploy` scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>

Specifies the name of the OpenTP1 directory containing OpenTP1 that is to be registered in the OS.

Specify an OpenTP1 directory name that is unique in the OpenTP1 system.

- DCCONFPATH

Specifies an OpenTP1 definition file storage directory.

- HOST_NAME

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

Output scenario variables

None.

Executing user

Superuser

M.4 OpenTP1_ScenarioAddNode (Add a new OpenTP1 node to the domain)

Function

The `OpenTP1_ScenarioAddNode` scenario template adds a new OpenTP1 node to the domain definition file on the node in the OpenTP1 system on which the domain configuration is to be updated.

When you use the `OpenTP1_ScenarioAddNode` scenario template to add a newly added OpenTP1 node to the node on which you update the domain configuration, clients can use services on the newly added OpenTP1 node.

The `OpenTP1_ScenarioAddNode` scenario template consists of the following lower-level scenario templates:

- `OpenTP1_AddNode`
- `OpenTP1_UpdateDomain`

The prerequisites for executing the `OpenTP1_ScenarioAddNode` scenario template are as follows:

- Y is specified in the `name_domain_file_use` operand in the system common definition.
- The node on which you update the domain configuration is online.

Input scenario variables

- `DCDIR~<character string of 1-50 characters>`

Specifies an OpenTP1 directory name.

- `DCCONFPATH`
Specifies an OpenTP1 definition file storage directory.
- `HOST_NAME`
Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.
- `USER_NAME`
Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.
- `ADD_HOST_NAME~<character string of 1-255 characters>`
Specifies the host name of the new OpenTP1 node to be added to the OpenTP1 system.
- `PORT_NUMBER~((5001-65535))`
Specifies the port number of the new OpenTP1 node to be added to the OpenTP1 system. This port number is used by the name server.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable

`USER_NAME`

M.5 OpenTP1_Start (Start OpenTP1)

Function

The `OpenTP1_Start` scenario template starts a new OpenTP1 that has been added. If the previous termination mode was normal termination, OpenTP1 starts normally. If the previous termination mode was abnormal termination, OpenTP1 restarts.

The Scale Out scenario and the Rolling Update scenario contain the `OpenTP1_Start` scenario template for startup of OpenTP1. The `OpenTP1_Start` scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>
Specifies an OpenTP1 directory name.
- DCCONFPATH
Specifies an OpenTP1 definition file storage directory.
- HOST_NAME
Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.
- USER_NAME
Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable
USER_NAME

M.6 OpenTP1_StartUAP (Start a UAP)

Function

The OpenTP1_StartUAP scenario template starts a user server.

The Scale Out scenario contains the OpenTP1_StartUAP scenario template for startup of a user server. The OpenTP1_StartUAP scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>
Specifies an OpenTP1 directory name.
- DCCONFPATH
Specifies an OpenTP1 definition file storage directory.
- HOST_NAME
Specifies the host name of the OpenTP1 node where scenario jobs are

executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- USER_NAME

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

- USER_SERVER_NAME~<character string of 1-8 characters>

Specifies the name of the user server to be started.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable

USER_NAME

M.7 OpenTP1_Stop (Stop OpenTP1)

Function

The `OpenTP1_Stop` scenario template stops OpenTP1 so that the unused resources of an OpenTP1 node can be released.

The Scale In scenario and the Rolling Update scenario contain the `OpenTP1_Stop` scenario template for stopping of OpenTP1. The `OpenTP1_Stop` scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>

Specifies an OpenTP1 directory name.

- DCCONFPATH

Specifies an OpenTP1 definition file storage directory.

- HOST_NAME

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- USER_NAME

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable
USER_NAME

M.8 OpenTP1_StopUAP (Stop a UAP)

Function

The OpenTP1_StopUAP scenario template stops a user server.

The Scale In scenario contains the OpenTP1_StopUAP scenario template for stopping of a user server. The OpenTP1_StopUAP scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>

Specifies an OpenTP1 directory name.

- DCCONFPATH

Specifies an OpenTP1 definition file storage directory.

- HOST_NAME

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- USER_NAME

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

- USER_SERVER_NAME~<character string of 1-8 characters>

Specifies the name of the user server to be stopped.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable
USER_NAME

M.9 OpenTP1_Undeploy (Delete OpenTP1)

Function

The `OpenTP1_Undeploy` scenario template deletes OpenTP1 in the specified OpenTP1 directory from the OS.

The Rolling Update scenario contains the `OpenTP1_Undeploy` scenario template for deletion of OpenTP1 from the OS. The `OpenTP1_Undeploy` scenario template can be copied to other scenarios as well and used as part of those scenarios.

Input scenario variables

- DCDIR~<character string of 1-50 characters>

Specifies the name of the OpenTP1 directory containing OpenTP1 that is to be deleted from the OS.

- DCCONFPATH

Specifies an OpenTP1 definition file storage directory.

- HOST_NAME

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

Output scenario variables

None.

Executing user

Superuser

M.10 OpenTP1_UpdateDomain (Update the domain configuration)

Function

The `OpenTP1_UpdateDomain` scenario template updates the domain

configuration of the OpenTP1 system while OpenTP1 is running.

When you want to add to the system a new node that has been added to the domain definition file by using the `OpenTP1_AddNode` scenario template or another method, copy the `OpenTP1_UpdateDomain` scenario template to the desired scenario and use it as part of that scenario.

The prerequisites for executing the `OpenTP1_UpdateDomain` scenario template are as follows:

- Y is specified in the `name_domain_file_use` operand in the system common definition.
- OpenTP1 is online.

Input scenario variables

- `DCDIR~<character string of 1-50 characters>`

Specifies an OpenTP1 directory name.

- `DCCONFPATH`

Specifies an OpenTP1 definition file storage directory.

- `HOST_NAME`

Specifies the host name of the OpenTP1 node where scenario jobs are executed.

This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.

- `USER_NAME`

Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.

This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.

Output scenario variables

None.

Executing user

OpenTP1 system administrator specified in the input scenario variable `USER_NAME`

M.11 OpenTP1_ScenarioScaleout (Scale Out sample scenario template)

Function

When executed after OpenTP1 is installed, the `OpenTP1_ScenarioScaleout` sample scenario template sets the environment of OpenTP1, and starts OpenTP1 and the sample SPP.

Input scenario variables

- `DCDIR~<character string of 1-50 characters>`
 Specifies the name of the OpenTP1 directory where the `OpenTP1_ScenarioScaleout` sample scenario template is to be registered.
 Specify an OpenTP1 directory name that is unique in the OpenTP1 system.
- `DCCONFPATH`
 Specifies an OpenTP1 definition file storage directory.
- `HOST_NAME`
 Specifies the host name of the OpenTP1 node where scenario jobs are executed.
 This input scenario variable can be omitted. If you omit it, scenario jobs are executed on the host where the scenario is executed.
- `USER_NAME`
 Specifies the user name of the OpenTP1 system administrator who executes scenario jobs.
 This input scenario variable can be omitted. If you omit it, the mapped primary user (user mapping) defined by JP1/Base executes scenario jobs.
- `GROUP_NAME`
 Specifies the name of an OpenTP1 group.
 When you specify the input scenario variable `USER_NAME`, you must also specify the input scenario variable `GROUP_NAME`.
- `BETRAN_FILE1`
 Specifies the path of the OpenTP1 file system in system A.
 When you specify a character type special file, allocate a partition before you execute the `OpenTP1_ScenarioScaleout` sample scenario template.
- `BETRAN_FILE2`
 Specifies the path of the OpenTP1 file system in system B.
 When you specify a character type special file, allocate a partition before you execute the `OpenTP1_ScenarioScaleout` sample scenario template.

- `USER_SERVER_NAME`~<character string of 1-8 characters>
Specifies the name of the user server to be started.
When you use the `OpenTP1_ScenarioScaleout` sample scenario template, specify `basespp`.
- `NODE_ID`~<character string of 4 characters>
Specifies the node ID of the OpenTP1 node to be set.
Specify a node name that is unique in the OpenTP1 system.
- `PORT_NUMBER`~((5001-65535))
Specifies the port number of the new OpenTP1 node to be added to the OpenTP1 system. This port number is used by the name server.

Output scenario variables

None.

Executing user

The superuser executes the scenario job for creating the directory for OpenTP1 whose environment is to be set and the scenario job for registering OpenTP1 in the OS. Other scenario jobs are executed by the OpenTP1 system administrator that is specified in the input scenario variable `USER_NAME`.

Index

Symbols

/etc/inittab, changing 335

A

abbreviations for products v
abnormal termination 48, 317
abnormal terminations count for application, initializing 267, 704
acceptance of requests to establish server-type connection
 starting 819
 terminating 817
acquiring MCF trace
 ending 837
 starting 839
acquiring transaction statistics when XA resource service is used 78
acronyms x
AJ record format 1100
alternate PAGEC module input path 279
alternate physical map input path 279
alternate sending
 starting 830
 starting and terminating 273
 terminating 776
application
 displaying status of 267, 713
 releasing shutdown of 701
 shutting down 707
 shutting down and releasing shutdown status of 267
application operation 267
application program, starting 268, 842
application timer startup request
 deleting 268
 displaying 268
application timer startup requests, displaying 716
archive journal file

 changing status of 326
 configuring, creating, and defining 320
 displaying information about 325
 errors in 358
 estimating size of 1287
 opening and closing 325
 reusing 324
 status changes by open and close commands 326
 status changes of 327
 status changes of online 325, 327
 swapping 326
 unloading 323
 using 321
archive status, displaying 325, 583
archive-journal source node 303
archive-journal target node 303
attribute symbol 1100
audit log
 estimation example of 89
 file rotation 87
 information output to 90
 operation 85
 operation example of 111
 output format 87
 setting up environment for 85, 491
audited events
 information output for 1082
 list of 95
automatic startup 44
automatic system switchover 344
automatic unloading facility
 controlling 586
 unloading using 217

B

backup file
 larger than journal requirement for recovery 330

- smaller than journal requirement for recovery 331
- buffer group utilization status, displaying 279
- buffer pool for storing message 124

C

- cautions about amount of log output 81
- checkpoint dump file
 - automatic opening of 243
 - creating and defining 240
 - deleting 242
 - displaying information about 243
 - duplicated 243
 - errors in 360
 - estimating size of 1286
 - opening and closing 243
 - operation 240
 - sequence of using 241
 - status changes by open and close commands 243
 - status changes for 245
 - using 241
- collecting system statistics, starting and terminating 547
- collecting transaction statistics, starting and terminating 1048
- combinations of host names OpenTP1 uses in running and standby systems 341
- combinations of `scd_announce_server_status` and `scd_this_node_first` operands 62
- command log 82
 - preparation for acquiring 41
- command log directory 19
- common definition for system, changing 10
- communication control unit, terminal, and line errors 368
- communication error 368
 - under multinode facility 368
- connection
 - displaying status of 265, 789
 - establishing 730
 - establishing and releasing 265
 - exchanging 265
 - releasing 749

- switching 747
- connection operation 265
- connection to resource group, forcing to release 582
- contents of input/output queue, copying 773
- continuous inquiry-response processing 273
 - terminating forcibly, for logical terminal 778
- conventions
 - abbreviations for products v
 - acronyms x
 - diagrams xi
 - fonts and symbols xii
 - KB, MB, GB, and TB xiv
 - permitted characters xiv
 - version numbers xv
- copy file of input/output queue contents 27
- core file 27
- core file directory 19
- correspondence between physical file and logical file 204
- CPU error 376
- current 204
- current status 322

D

- DAM file
 - adding and deleting 250
 - addition in online mode 253
 - backing up 251
 - backing up and restoring 251
 - creating 249
 - displaying status of 249
 - errors in 362
 - estimating size of 1289
 - extending block length for 254
 - extracting user data from 255
 - lock 252
 - operation 249
 - recovering 251
 - restoring 251
 - setting number of cache blocks for 253
 - status display by `damls` command 250
 - status display by `fills` command 249
 - status management of 249
- DAM FRC 251

- execution in multiple steps 252
 - execution in one step 252
 - execution using integrating journal file 252
 - execution using intermediate generation of system journal file 252
 - procedure 362
 - simultaneous execution of multiple steps 252
 - suspended by error 363
 - damadd 462
 - dambkup 465
 - damchdef 467
 - damchinf 468
 - damdel 471
 - damfrc 472
 - damhold 477
 - damload 479
 - daml 481
 - damrles 484
 - damrm 486
 - damrstr 488
 - data that can be output by jnlrput command 228
 - dcauditsetup 491
 - dccspool 494
 - dcdefchk 497
 - dcjchconf 501
 - dcjcmdex 504
 - dcjnamch 506
 - dcmakeup 508
 - dcmaphcg 509
 - dcmapl 511
 - dcmstart 517
 - dcmstop 521
 - dcndls 524
 - dcpllist 527
 - dcrasget 529
 - dcreport 531
 - dcreset 535
 - dcsetup 536
 - dcshmls 539
 - dcstart 544
 - dcstats 547
 - dcstop 550
 - dcsvstart 553
 - dcsvstop 555
 - deadlock and timeout information file 27
 - deadlock information file and timeout information file, deleting 79, 695
 - deadlock information file directory 19
 - deferred update facility 250
 - definition analysis file 18
 - definition file directory 19
 - definition object file, creating 38, 39
 - definition, changing 336
 - diagram conventions xi
 - difference in file group status at error between when one-system operation is available and when unavailable 244
 - difference in online operation for only one system 245
 - directory for
 - online testers 19
 - storing transaction control objects 19
 - system internal lock control 19
 - system internal synchronization control 19
 - disk queue message, handling 58
 - domain communication, setting up environment for 8
 - domain configuration
 - changing 126
 - changing (by using domain definition file) 848
 - changing (by using system common definition) 854
 - domain definition file 127
 - updating 506
 - domain representative schedule service 129
 - registering 9
 - registering and deleting 853
 - registering port number of 9
 - dump file format 1076
 - of input/output queue 1076
- E**
- element file 320
 - available 321
 - close 213
 - closed 321
 - open 213
 - opened 321

- unavailable 321
- unusable 213
- usable 213
- environment setup 1
 - by OpenTP1 administrator 11
 - by superuser 5
 - multiOpenTP1 334
 - OpenTP1 302
 - overview of 2
 - procedure for TP1/message control 4
 - procedure for TP1/message queue 4
 - user 334
- environment variable, changing 335
- error events and causes 350
 - list of 350
- error handling procedure 349
- estimating resources used by UNIX message exchange function 1293
- estimation formula of
 - number of records 1269
 - record length 1269
- execution environment
 - for real-time statistics service, setting 942
 - setting up 132
- execution status of temporary closing, displaying 918
- expression for judging congestion of schedule queue 73
- extended inter-node load-balancing facility 63

F

- FIL event trace information file 22
 - backup file for 22
- filbkup 557
- filchgrp 559
- filchmod 561
- filchown 564
- file error 365, 367
- file group 212, 320
 - available 214, 322
 - closed 213, 322
 - opened 213, 322
 - unavailable 214, 322
- file recovery at archive-journal source node 330
- file recovery journal, integrating 596

- files and directories
 - created at installation or cataloging into OS 18, 20, 307
 - created by user 17, 306
 - created during execution of OpenTP1 308
- files created by user 17, 306
- files created during normal operation 21
- files created during OpenTP1 execution 21
- files created in the event of error 26
- fills 566
- filmkfs 571
- filrstr 573
- filstatfs 576
- flow of checks by dcdefchk command 13
- font conventions xii
- forced normal termination 47
- forced termination 48, 52, 316
- forcible determination of transaction 76
- forcible termination of transaction 76
- format of standard output of process ID acquisition command 294
- format of structure dc_mcf_dump_info 1078

G

- garbage collection in OpenTP1 file system 202
- GB meaning xiv
- GJ record format 1101
- global archive unload journal file 323
 - editing and outputting 329
 - outputting records of 329
- global archive unload journal file and unload journal file
 - chronological sorting and merging 328
 - editing and outputting 607
 - outputting records of 647
 - sorting and merging chronologically 662
- global journal operation 320
- global journal service facility 320
- grouped system switchover 345

H

- handling
 - application startup requests when user server (MHP) is terminated 53

service requests in scheduling queue and
messages in input and output queues 49
heuristic decision 76

I

I/O failure continuation table 258, 364
IJ record format 1102
incorrect journal information file 27
information collection when error occurs 377
information inherited during full recovery 382
inhibiting unload check on journal 221, 323
input scenario variable 164
 environment variable whose value is to be
 determined from 165
integrating journal file 330
inter-node load balancing 59
 operation mode of 60
 prerequisite for 59
invalid 204

J

JNL performance verification trace information
file 21
 backup file for 21
jnladdpf 580
jnlardis 582
jnlarls 583
jnlatusl 586
jnlchgfg 590
jnlclsfg 593
jnlcolc 596
jnlcopy 601
jnldepf 606
jnledit 607
jnlinit 618
jnlls 620
jnlmctst 633
jnlmkrf 640
jnlopnfg 641
jnlrinf 644
jnllrm 646
jnllrput 647
jnllsort 662
jnllstts 665

jnlswpfg 683
jnlunlfg 685
journal file
 changing status of 590
 closing 593
 deleting 646
 displaying information about 620
 initializing 618
 opening 641
 recovering 640
 swapping 683
 unloading 685
journal fileless mode 234, 235
journal information directory 19
journal information that is read during restart,
displaying 644
journal integration for file recovery 234, 329
journal physical file
 allocating 580
 deleting 606

K

KB meaning xiv

L

LAN error 368
LCK performance verification trace information
file 22
 backup file for 22
lckls 691
lckpool 693
lckrminf 695
loaded resources of map file, displaying 511
lock control table pool information, displaying 79,
693
lock information, displaying 79, 691
lock operation 79
logcat 696
logcon 699
logging 81
logical file 249
 adding 462
 displaying status of 481

- logical shutdown and release of shutdown status 250
 - recovering 472
 - releasing, from shutdown 484
 - removing 486
 - shutting down logically 477
 - logical terminal
 - deleting output queue for 767
 - displaying status of 269, 795
 - forced termination of continuous inquiry-response processing for 273
 - message journal collection, starting and terminating 272
 - message journal collection, terminating 756
 - operation 269
 - releasing shutdown of 734
 - shutting down 753
 - shutting down and releasing shutdown status of 269
 - starting message journal collection for 737
 - logical terminal message queue
 - discarding initial message of 270
 - skipping first message in 827
 - logical terminal output queue
 - copying contents of 271
 - deleting 272
 - holding process of 784
 - holding process of, and releasing held process of 271
 - outputting contents of 272
 - releasing held process of 824
- M**
- maintenance data, acquiring 529
 - manual startup 44
 - map file 279
 - changing path name of 509
 - MB meaning xiv
 - MCF communication process program file and MCF application startup process program file 17, 306
 - MCF communication service
 - MCF main function of, creating 32
 - operation 264
 - partial start of 832
 - partial stop of 834
 - replacing part of 141
 - MCF communication service status, referencing 793
 - MCF definition file 366
 - MCF dump file 27
 - MCF event trace file 21
 - MCF main function of application startup service
 - coding (for ANSI C and C++) 35
 - coding (for K&R C) 36
 - creating 34
 - MCF main function, installing into directory 36, 37
 - MCF performance verification trace information file 22
 - backup file for 22
 - MCF performance verification trace information, acquiring 1317
 - MCF service name, cataloging 37
 - MCF shared memory dump file 27
 - MCF statistics
 - edited contents of (mcfreport command) 234
 - editing 720
 - outputting 726
 - types of 233
 - MCF statistics file 23
 - MCF trace file, swapping forcibly 840
 - MCF trace information
 - swapping 118
 - temporarily outputting 118
 - MCF uptime statistics
 - edited contents of 231
 - outputting 633
 - mcfactap 701
 - mcfactcap 704
 - mcfadctap 707
 - mcfadltap 710
 - mcfalsap 713
 - mcfalstap 716
 - mcfreport 720
 - mcfstats 726
 - mcfactcn 730
 - mcfactle 734
 - mcfactmj 737
 - mcfactsg 739
 - mcfactss 742

- mcfactsv 745
 - mcfatchn 747
 - mcfidctcn 749
 - mcfidctle 753
 - mcfidctmj 756
 - mcfidctsg 759
 - mcfidctss 762
 - mcfidctsv 765
 - mcfidqlq 767
 - mcfidqlsq 770
 - mcfidmpqu 773
 - mcfidalt 776
 - mcfidct 778
 - mcfihldiq 781
 - mcfihldoq 784
 - mcfilsubuf 787
 - mcfilscn 789
 - mcfilscm 793
 - mcfilsle 795
 - mcfilsln 800
 - mcfilssg 802
 - mcfilssv 806
 - mcfilstrd 809
 - mcfilsum 812
 - mcfiofln 817
 - mcfionln 819
 - mcfirliq 821
 - mcfirliq 824
 - mcfirliq 827
 - mcfistalt 830
 - mcfistart 832
 - mcfistop 834
 - mcfistptr 837
 - mcfistrtr 839
 - mcfiswptr 840
 - mcfuevt 842
 - memory dump, outputting 1050
 - memory queue message, handling 58
 - message
 - receiving 263
 - sending 263
 - message arrived after service group automatic shutdown 57
 - message at abnormal termination 57
 - message at application shutdown and message at service shutdown 58
 - message at application shutdown, handling 58
 - message at service shutdown, handling 59
 - message control facility
 - calculation formula of necessary amount of journal collected by 1107
 - conditions for collecting journals by 1106
 - journal information collected by 1100
 - message handling program (MHP) 55
 - message log 81
 - editing and outputting 81
 - message log file 21, 367
 - displaying 696
 - outputting to 81
 - outputting to, and editing 81
 - message log realtime output facility, switching 699
 - message object file 18
 - message queue
 - allocating physical file for 288
 - deleting physical file for 288
 - monitoring 280
 - message queue file 365
 - MJ record format 1103
 - more than one schedule service in same host, when there is 10
 - multi-node facility, operation under 301
 - multinode area or subarea
 - starting 517
 - terminating 521
 - multiOpenTP1 operation 333
- N**
- NAM event trace information file 22
 - backup file for 22
 - namalivechk 844
 - namblad 846
 - namchgfl 848
 - namdomainsetup 853
 - namndchg 854
 - namsvinf 857
 - namunavl 862
 - network status, displaying 800
 - no checkpoint dump collection destination 243

non-OpenTP1 file, creating 17, 306
 non-overwritable status 242
 nonswappable file group 224
 nonswappable target file group 326
 normal startup 44
 normal termination 47, 52, 316
 notes

- on changing configuration 140
- on creating command for resource manager to be monitored 291
- on entering 459
- on termination 49

 number of cache blocks, acquiring 468
 number of processes, changing 64, 950
 number of record locks, estimating 1291

O

OJ record format 1104
 one-system operation to be available or unavailable 320
 online backup 251, 259
 OpenTP1

- cannot be started 370
- cannot be terminated 372
- cataloging, into OS 6, 303, 334, 536
- deleting, from OS 7, 536
- disk space used by 138
- environment setup procedure 3
- errors in 370
- errors under multinode facility 374
- file operation 199
- incorrectly installed or set up 370
- installing 5, 302
- nonstop operation of 138
- online operation 51
- operation command does not terminate normally 374
- starting 44, 316, 342, 544
- starting and terminating 43, 316, 342
- terminates abnormally 373
- terminating 47, 316, 342, 550
- terminating, in running system 342
- terminating, in standby system 342

 OpenTP1 administrator

cataloging 5, 302
 environment setup 14, 304
 setting up 334
 OpenTP1 debug information file 27
 OpenTP1 directory, creating 6, 302, 334
 OpenTP1 domain operation 126
 OpenTP1 event 1080

- cataloging 1080
- that can be cataloged 1080
- that can be cataloged in JP1 event service facility 1080

 OpenTP1 file

- changing owner of 201
- creating 15, 305, 336
- errors in 353
- estimating size of 1271
- initializing 336
- re-creating 203

 OpenTP1 file access authorization mode, changing 201, 561
 OpenTP1 file group, changing 201, 559
 OpenTP1 file owner, changing 564
 OpenTP1 file system

- backing up 201, 557
- changing attribute of 202
- creating 200
- displaying contents of 200, 566
- displaying status of 200, 576
- estimating size of 1295
- garbage collection in 202
- initializing 15, 305, 334, 571
- operation 200
- restoring 201, 573

 OpenTP1 file system area 200, 303

- creating 8, 303
- owner and access permission of 304
- owner and access right of 8

 OpenTP1 files, and operation commands for their creation and initialization 16, 305
 OpenTP1 group, defining 5, 302
 OpenTP1 internal control

- reserving or releasing resources for 508
- reserving resources for 15, 304

 OpenTP1 internal processing, files for 30

OpenTP1 journal 210
 OpenTP1 node
 displaying status of 319, 524
 manipulating RPC suppression list of 846
 OpenTP1 process, forcibly terminating 864
 OpenTP1 program file 18
 OpenTP1 server information, displaying 857
 OpenTP1 start notification information, forcibly
 invalidating 862
 OpenTP1_Undeploy 1333
 OpenTP1_AddNode 1325
 OpenTP1_ChangeNodeID 1326
 OpenTP1_Deploy 1327
 OpenTP1_ScenarioAddNode 1328
 OpenTP1_ScenarioScaleout 1334
 OpenTP1_Start 1329
 OpenTP1_StartUAP 1330
 OpenTP1_Stop 1331
 OpenTP1_StopUAP 1332
 OpenTP1_UpdateDomain 1333
 operation command execution environment 338
 operation commands 347, 427
 details of 461
 displaying description of how to use 430
 input method of 428
 list of 431, 432
 overview of 428
 specification format for 428
 operation using Job Management Partner 1 163
 operation when inter-node load balancing facility is
 used with other facilities 62
 operation when mcftdctl command is entered while
 message is being received or sent 269
 optional facility 289
 operation 289
 other files, creating 337
 overwritable status or writing status 242

P

parallel access facility 320
 path name, changing 279
 performance verification trace
 acquiring 1297
 timing to acquire 1297

performance verification trace information file 21
 backup file for 21
 performance verification trace information,
 acquiring 1297
 performance verification trace operation 119
 permitted character conventions xiv
 physical file
 backing up 465
 creating 204
 deleting 471
 initializing 479
 restoring 488
 physical file for message queue
 allocating 894
 deleting 898
 estimating size of 1269
 physical file space shortage 365
 planned system switchover 344
 planned termination 316
 planned termination A 47
 planned termination B 48
 prckill 864
 prcls 865
 prcpth 868
 prcpthls 870
 prctctrl 871
 prctee 873
 pre-online transaction recovery 298
 precedence of local node 61
 prfed 875
 prfget 892
 process ID output format 293
 process service event trace information file 22
 backup file for 22
 process service information directory 19
 process, stopping and restarting 968
 product information, displaying 527

Q

queinit 894
 quels 895
 querm 898
 queue group, displaying status of 288
 queue operation 288

R

- RAP-processing client manager, starting and stopping 135
- RAP-processing listener or RAP-processing server
 - displaying status of 133, 901
 - starting and stopping 134
- RAP-processing listener, establishing connection with 136
- rapdfgen 899
- rapls 901
- rapsetup 904
- real-time statistics 1146
 - acquiring 190
 - editing and outputting, in RTS log file 196
 - outputting 194
 - outputting, to RTS log file 194
 - outputting, to standard output 194, 931
- real-time statistics service
 - operation for using 186
 - preparation for using 186
 - starting and stopping 186
- realtime output facility 81
- recovery journal file
 - errors in 359
 - operation 239
 - recovering 332
- referencing load status 61
- relationship between element file status and physical file status 213
- relationship between file group status and physical file status 321
- relationship between TAM data file format and record format of data section of TAM file (-s option not specified) 256
- relationship between TAM data file format and record format of data section of TAM file (-s option specified) 257
- relationship between user server's termination form and OpenTP1's termination form 53
- remote API facility
 - automatically generating definition for using 132, 899
 - cautions when using 136
 - creating definitions 132
 - operation using 132
 - performance improvement of 133
 - preparation for 132
 - setting up execution environment for 904
- reserve 242
- reserved status 322
- resource handling 138
- resource manager
 - cataloging 30
 - cataloging and deleting 290, 1028
 - displaying information about 290, 1041
 - suspending OpenTP1 startup until startup of 296
- resource manager monitor 291
 - operation 291
- resource manager operation 290
- resource, displaying 279
- response wait overtime occurs during OpenTP1 operation command execution 374
- restart following full recovery 44
- restarting process service and reflecting definition 535
- RI 145
- RMM 291
- rolling update, using 175
- RPC suppression list 844
- RPC trace file 21
- RPC trace information, outputting 905
- RPC trace operation 117
- RPC traces
 - merging 117, 916
 - outputting 117
- rpcdump 905
- rpcmrg 916
- rpestat 918
- RTS log file 23
 - editing and outputting 920
- rtsedit 920
- rtsls 931
- rtsssetup 942
- rtsstats 944

S

sample scenario template

- customizing 182
 - registering 181
 - using 178
- scale in, using 173
- scale out and DPM for OpenTP1 environment setup procedure, using 172
- scale out, using 166
- scdchprc 950
- scdhold 954
- scdls 956
- scdrls 966
- scdrsprc 968
- scenario
 - registering 164
 - variable 164
- scenario template 163
 - executing OpenTP1 command from 504
- schedule queue congestion monitoring 68
- schedule queue, monitoring 64
- scheduling
 - automatic shutdown of 57
 - displaying status of 956
 - releasing shutdown of 966
 - restarting 966
 - shutting down 954
 - shutting down and restarting 56
- scheduling buffer group 124
- scheduling operation 56
- scheduling queue information file 21
- scheduling status, displaying 56
- search path
 - displaying and changing 54
 - inheriting 54
- search path name for user server and for command activated from user server
 - changing 868
 - displaying 870
- server
 - displaying status of 53, 865
 - starting 553
 - terminating 555
- server operation 52
- server recovery journal directory 19
- server recovery journal file
 - estimating size of 1286
 - operation 239
 - recovering 239
- service
 - displaying status of 278, 806
 - operation 278
 - releasing, from shutdown 745
 - shutting down 765
 - shutting down and releasing shutdown status of 278
- service group
 - deleting input queue of 770
 - displaying status of 274, 802
 - holding process of input queue for 781
 - operation 274
 - releasing, from shutdown 739
 - shutting down 759
 - shutting down and releasing shutdown status of 274
- service group input queue
 - copying contents of 275
 - deleting 276
 - holding process of, and releasing held process of 275
 - outputting contents of 276
 - releasing held process of 821
- session
 - starting 742
 - starting and terminating 279
 - terminating 762
- settings of target and item to be acquired in real time, changing 196
- shared library, changing 335
- shared memory dump file 26
- shared memory operation 124
- shared memory utilization status, displaying 124
- shutdown 205
- stack trace information, checking 389
- standard output and standard error output of OpenTP1, redirecting 873
- standard output file operation 80
- standard PAGEC module input path 279
- standard physical map input path 279
- standby 204

Index

- standby status 322
- standby user server 346
- startup form 45
 - selecting 44
- startup method 44
 - selecting 44
- startup mode 44
- startup notification information, invalidating 130
- statistics 1109
 - statistics about communication delay 230, 1146
 - details of 1146
 - edited contents of 231
 - statistics about response 1131
 - details of 1131
 - edited contents of 230
 - statistics about transaction 1130
 - details of 1131
 - statistics editable with `jnlstts` command 229
 - statistics that `dcreport` command can acquire 232
 - statistics that `jnlmst` command can acquire 231
 - statistics that `mcfstats` and `mcfreport` commands can acquire 233
- status changes for checkpoint dump files
 - in online mode 246
 - in online mode when one-system operation is available 246
 - in online mode when one-system operation is unavailable 248
- status file
 - closing 970
 - creating and defining 204
 - creating and initializing 975
 - deleting 207, 983
 - displaying contents of 206, 972
 - displaying status of 206, 977
 - errors in 353
 - estimating size of 1271
 - opening 981
 - opening and closing 206
 - operation 204
 - space shortage 207
 - status changes for 207
 - status changes for, in online mode 205, 207
 - swapping 984
 - using 204
- status of multi-processing of message, displaying 809
- status of queue group, displaying 895
- stsclose 970
- stsfills 972
- stsininit 975
- stsls 977
- stsoopen 981
- stsrn 983
- stsswap 984
- swapping 152, 205
- symbol conventions xii
- synonym information, displaying 1001
- system common definition, changing 31
- system definition 9
 - checking 497
 - creating 11, 303
 - setting value of operand in 501
- system environment variable 165
- system journal file
 - changing status of 224
 - creating and defining 211
 - displaying information about 223
 - errors in 356
 - estimating size of 1276
 - opening and closing 223
 - operation 210
 - reusing 222
 - status changes by open and close commands 224
 - status changes for 225
 - status changes for, in online mode 223
 - swapping 224
 - unloading 216
 - unloading, at archive-journal source node 330
 - using 213
- system management information directory 19
- system operation using scenario template 163
- system service information definition file, creating 37
- system statistics 1109
 - details of 1109
 - items to be edited in 1122
 - outputting, to journal 227

- system statistics to standard output in real time, editing and outputting 531
 - system switchover facility 340
 - operation under 339
 - preparation for using 340
 - system switchover procedure 344
- T**
- TAM data file 256, 259
 - TAM file
 - adding in online mode 261
 - backing up 259, 989
 - backing up and restoring 259
 - creating 256
 - creating TAM data file from 259
 - deleting 260, 995
 - displaying synonym information after creating 261
 - errors in 363
 - estimating size of 1289
 - increasing number of records in 261
 - initializing 992
 - lock 260
 - operation 256
 - recovering 260, 996
 - restoring 259, 1017
 - TAM FRC 260
 - execution in multiple steps 260
 - execution in one step 260
 - execution using integrating journal file 260
 - procedure 363
 - suspended by error 363
 - TAM locked resource name, converting 1004
 - TAM table
 - adding 985
 - adding and removing 257
 - displaying status of 257, 1008
 - loading 1006
 - loading and unloading 258
 - logical shutdown and release of shutdown status 258
 - releasing, from shutdown 1012
 - removing 1015
 - shutting down logically 1000
 - status management 257
 - unloading 1018
 - tamadd 985
 - tambkup 989
 - tamcre 992
 - tamdel 995
 - tamfrc 996
 - tamhold 1000
 - tamhsls 1001
 - tamlckls 1004
 - tamload 1006
 - tamls 1008
 - tamrles 1012
 - tamrm 1015
 - tamrstr 1017
 - tamunload 1018
 - target and item for real-time statistics acquisition, changing 944
 - TB meaning xiv
 - terminating process 55
 - termination method 49, 53, 317
 - termination mode 47, 52, 316
 - threshold for number of cache blocks, setting 467
 - timer activation request for application, deleting 710
 - TP1/message control execution, preparations for 32
 - TP1/message control, setup for 3
 - TP1/message queue, setup for 4
 - TP1/Multi 302
 - TP1/NET1/OSI-TP-Extended, operation using 131
 - tptrnls 1020
 - trace data length required for one transaction 1297
 - trace information dump file 21
 - trace information file
 - editing and outputting 875
 - getting 892
 - trace operation 117
 - transaction
 - committing 1023
 - displaying status of 1033
 - forcibly terminating 1026
 - operation 76
 - rolling back 1046
 - transaction control object file, creating 30, 290, 1043
 - transaction information directory 19

transaction information, managing 147
 transaction journal directory 19
 transaction recovery journal file
 operation 239
 recovering 239
 transaction recovery, suspending (when Oracle9i RAC
 is used) 298
 transaction statistics
 edited contents of 229
 startup and termination of collecting 77
 transaction status, displaying 76
 TRN event trace information file 22
 backup file for 22
 trncmt 1023
 trndlinf 1025
 trnfgt 1026
 trnlkrm 1028
 trnls 1033
 trnlsrm 1041
 trnmkobj 1043
 trnrbk 1046
 trnstics 1048
 troubleshooting 383
 deleting information of 494
 information to acquire 384
 items to check 384
 procedure for 391

U

UAP
 cannot be started 369
 deadlock 369
 does not terminate 369
 errors in 369
 terminates abnormally 369
 UAP shared library 75
 UAP trace
 checking 389
 overview of 388
 UAP trace data file 23
 backup file for 23
 UAP trace edit/output file 27
 UAP trace information, outputting 117

undecided transaction information for OSI TP
 communication, displaying 1020
 undecided-transaction information file 27
 deleting 77, 1025
 UNIX message exchange function, estimating
 resources used by 1293
 unload journal file 216
 chronological sorting and merging 227
 commands that input 222, 324
 copying 227, 601
 editing and outputting 227
 outputting records of 228
 unload journal file storage directory, using 217, 219
 unload wait status 216
 unloading 216, 323
 uptime statistics, outputting 228, 329, 665
 user program file 17, 306
 user program file directory 18
 user server
 command activated from 54
 process 55
 replacing 54
 search path for 54
 starting 52
 terminating 52
 user server process, refreshing 74
 user timer monitoring status, displaying 812
 usmdump 1050
 utilization status of buffer group, displaying 787
 utilization status of common memory, displaying 539

V

various definition files 17, 306
 verifying OpenTP1 startup and clearing cache 130,
 844
 version number conventions xv

X

XA resource service
 operation of 146
 operation using 145
 preparation for using 145
 prerequisite definitions for 145
 prerequisite facilities for 145

- releasing, from shutdown status 1071
- restarting 151
- shutting down 1062
- starting 150
- starting and stopping 150
- terminating 152
- trace information acquired by 162
- transaction management using 145

XAR event trace information, displaying 1052

XAR file

- creating 1063
- deleting 1072
- displaying status of 1056
- errors in 366

XAR performance verification trace information file 21

- backup file for 21

XAR transaction

- changing status of 1059
- displaying information about 1065

xarevtr 1052

xarfills 1056

xarforce 1059

xarhold 1062

xarinit 1063

xarls 1065

xarrles 1071

xarrm 1072

Reader's Comment Form

We would appreciate your comments and suggestions on this manual. We will use these comments to improve our manuals. When you send a comment or suggestion, please include the manual name and manual number. You can send your comments by any of the following methods:

- Send email to your local Hitachi representative.
- Send email to the following address:
WWW-mk@itg.hitachi.co.jp
- If you do not have access to email, please fill out the following information and submit this form to your Hitachi representative:

Manual name:	
Manual number:	
Your name:	
Company or organization:	
Street address:	
Comment:	

(For Hitachi use)
