HITACHI
Inspire the Next

JP1 Version 12 for UNIX Systems

JP1/Performance Management - Agent Option for Platform Description, User's Guide and Reference

3021-3-D84-10(E)

Notices

■ Relevant program products

For details about the applicable OS versions, and the service packs and patches required for JP1/Performance Management - Agent Option for Platform, see the *Release Notes*.

JP1/Performance Management - Manager (for Windows Server 2012, Windows Server 2016, Windows Server 2019):

P-2A2C-AACL JP1/Performance Management - Manager 12-50

The above product includes the following:

P-CC2A2C-5ACL JP1/Performance Management - Manager 12-50

P-CC2A2C-5RCL JP1/Performance Management - Web Console 12-50

JP1/Performance Management - Manager (for CentOS 6 (x64), CentOS 7, CentOS 8, Linux 6 (x64), Linux 7, Linux 8, Oracle Linux 6 (x64), Oracle Linux 7, Oracle Linux 8, SUSE Linux 12, SUSE Linux 15):

P-812C-AACL JP1/Performance Management - Manager 12-50

The above product includes the following:

P-CC812C-5ACL JP1/Performance Management - Manager 12-50

P-CC812C-5RCL JP1/Performance Management - Web Console 12-50

JP1/Performance Management - Agent Option for Platform (for HP-UX 11i V3 (IPF)):

P-1J2C-ACCL JP1/Performance Management - Agent Option for Platform 12-50

The above product includes the following:

P-CC1J2C-FCCL JP1/Performance Management - Agent Option for Platform 12-50

P-CC1J2C-AJCL JP1/Performance Management - Base 12-00

JP1/Performance Management - Agent Option for Platform (for CentOS 6 (x64), CentOS 7, CentOS 8, Linux 6 (x64), Linux 7, Linux 8, Oracle Linux 6 (x64), Oracle Linux 7, Oracle Linux 8, SUSE Linux 12, SUSE Linux 15):

P-812C-ACCL JP1/Performance Management - Agent Option for Platform 12-50

The above product includes the following:

P-CC812C-FCCL JP1/Performance Management - Agent Option for Platform 12-50

P-CC812C-AJCL JP1/Performance Management - Base 12-00

JP1/Performance Management - Agent Option for Platform (for Solaris 11):

P-9D2C-ACCL JP1/Performance Management - Agent Option for Platform 12-50

The above product includes the following:

P-CC9D2C-FCCL JP1/Performance Management - Agent Option for Platform 12-50

P-CC9D2C-AJCL JP1/Performance Management - Base 12-00

JP1/Performance Management - Agent Option for Platform (for AIX V7.1, AIX V7.2):

P-1M2C-ACCL JP1/Performance Management - Agent Option for Platform 12-50

The above product includes the following:

P-CC1M2C-FCCL JP1/Performance Management - Agent Option for Platform 12-50
P-CC1M2C-AJCL JP1/Performance Management - Base 12-00

These products include parts that were developed under licenses received from third parties.

■ Trademarks

HITACHI, Cosminexus, HiRDB, JP1, OpenTP1, uCosminexus are either trademarks or registered trademarks of Hitachi, Ltd. in Japan and other countries.

AIX is a trademark of International Business Machines Corporation, registered in many jurisdictions worldwide.

AMD is a trademark (or registered trademark) of Advanced Micro Devices, Inc.

DB2 is a trademark of International Business Machines Corporation, registered in many jurisdictions worldwide.

IBM is a trademark of International Business Machines Corporation, registered in many jurisdictions worldwide.

Intel is a trademark of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

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

Itanium is a trademark of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

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

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

PowerVM is a trademark of International Business Machines Corporation, registered in many jurisdictions worldwide.

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

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

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

UNIX is a trademark of The Open Group.

WebSphere is a trademark of International Business Machines Corporation, registered in many jurisdictions worldwide.

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

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

Other company and product names mentioned in this document may be the trademarks of their respective owners.

■ 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.

■ Microsoft product name abbreviations

This manual uses the following abbreviations for Microsoft product names.

Abbreviation		Full name or meaning
Exchange Server		Microsoft(R) Exchange Server
IIS		Microsoft(R) Internet Information Services
Internet Explorer		Windows(R) Internet Explorer(R)
Windows Server 2012	Windows Server 2012	Microsoft(R) Windows Server(R) 2012 Datacenter
		Microsoft(R) Windows Server(R) 2012 Standard
	Windows Server 2012 R2	Microsoft(R) Windows Server(R) 2012 R2 Datacenter
		Microsoft(R) Windows Server(R) 2012 R2 Standard
Windows Server 2016		Microsoft(R) Windows Server(R) 2016 Datacenter
		Microsoft(R) Windows Server(R) 2016 Standard
Windows Server 2019		Microsoft(R) Windows Server(R) 2019 Datacenter
		Microsoft(R) Windows Server(R) 2019 Standard

Windows Server 2012, Windows Server 2016, and Windows Server 2019 are sometimes referred to collectively as *Windows*.

■ Issued

Jan. 2021: 3021-3-D84-10(E)

■ Copyright

Copyright (C) 2019, 2021, Hitachi, Ltd.
Copyright (C) 2019, 2021, Hitachi Solutions, Ltd.

Summary of amendments

The following table lists changes in this manual (3021-3-D84-10(E)) and product changes related to this manual.

Changes	Location
The functionality of monitoring processes on host and container environments just like the Docker environment when the Podman environment is used in Linux. Based on this change, descriptions common to the Docker and Podman environments were changed to the descriptions on the *Linux container*. In addition, a script file to collect information of processes on the Podman container in Linux was added.	*2.3.14*, *2.3.21(1)*, *2.3.21(2)*, *2.3.21(3)*, *2.3.21(4)*, *2.3.21(5)*, *2.3.21(5)(a)*, *2.3.21(5)(b)*, *2.3.21(6)*, *2.3.21(6)(a)*, *2.3.22*, *2.3.22(1)(c)*, *7.2.4(1)(a)*, *7.2.5(1)(b)*, *12.4.5*, *Appendix F.2*, *Appendix G*, *Appendix H*
The following message was added: `KAVF10713-W`	*11.2*, *11.4*
A note on collecting dump information when troubleshooting was added.	*12.4.1*, *12.4.2*
Commands used to collect system information (Linux) during troubleshooting were added.	*12.4.5*

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

Preface

This manual explains the features of JP1/Performance Management - Agent Option for Platform, and the records it collects.

■ Intended readers

This manual is intended for the following users:

- Users who are interested in designing and building an operation monitoring system
- Users who are interested in defining conditions for collecting performance data
- Users who are interested in defining reports and alarms
- Users who are interested in referencing performance data that is collected for the purpose of system monitoring
- Users who are interested in developing and evaluating corrective measures to take for a system based on monitoring results, or users who are interested in directing the implementation of such measures

This manual assumes that the reader is knowledgeable about the operation of the monitored systems and is familiar with their operating systems.

For details about setting up and operating a system that uses JP1/Performance Management, the reader is directed to the following manuals as well:

- *JP1/Performance Management Planning and Configuration Guide*
- *JP1/Performance Management User's Guide*
- *JP1/Performance Management Reference*

■ Organization of this manual

This manual is organized into the parts listed below. Note that this manual applies to the following OSs: HP-UX, Solaris, AIX, and Linux(R). When there are differences in functionality between OSs, this manual provides a separate description for each OS.

PART 1: Overview

PART 1 provides an overview of JP1/Performance Management - Agent Option for Platform.

PART 2: Setup and Operation

PART 2 explains installation and setup of JP1/Performance Management - Agent Option for Platform, and how to collect user records.

PART 3: Reference

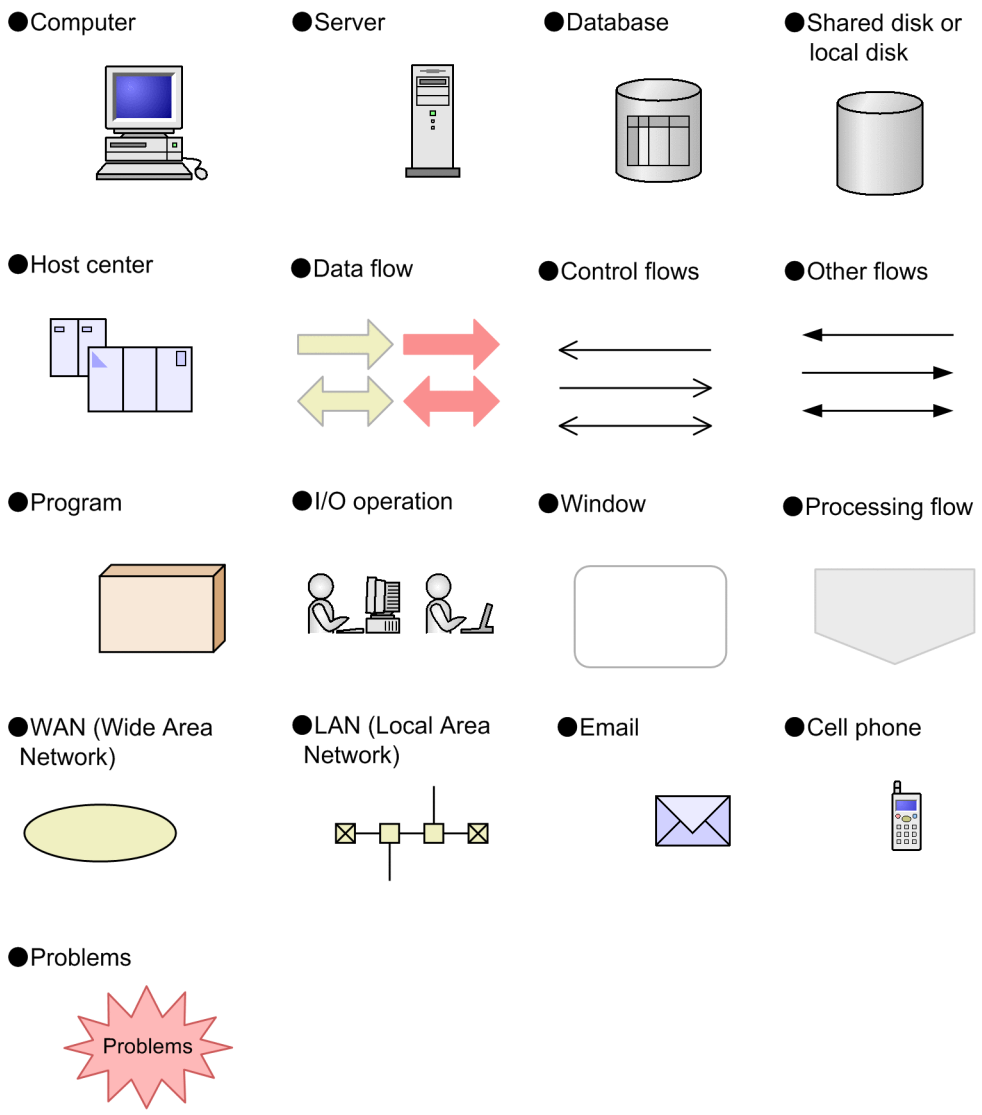
PART 3 explains the monitoring templates, records, commands, and messages associated with JP1/Performance Management - Agent Option for Platform.

PART 4: Troubleshooting

PART 4 explains the actions to take when a problem occurs with JP1/Performance Management - Agent Option for Platform.

■ Conventions: Diagrams

This manual uses the following conventions in diagrams:



■ Conventions: Fonts and symbols

The following table explains the text formatting conventions used in this manual:

Text formatting	Convention
Bold	Bold characters indicate text in a window, other than the window title. Such text includes menus, menu options, buttons, radio box options, or explanatory labels. For example: • From the **File** menu, choose **Open**. • Click the **Cancel** button. • In the **Enter name** entry box, type your name.
Italic	Italic characters indicate a placeholder for some actual text to be provided by the user or system. For example: • Write the command as follows: `copy` *source-file target-file*

Text formatting	Convention
Italic	• The following message appears: `A file was not found. (file =` *file-name*`)` Italic characters are also used for emphasis. For example: • Do *not* delete the configuration file.
`Monospace`	Monospace characters indicate text that the user enters without change, or text (such as messages) output by the system. For example: • At the prompt, enter `dir`. • Use the `send` 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 have been omitted. 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.
`( )`	Parentheses indicate the range of items to which the vertical bar (`|`) or ellipsis (`. . .`) is applicable.

Conventions for mathematical expressions

The following table lists the conventions used in mathematical formulas:

Symbol	Meaning
×	Multiplication sign
÷	Division sign

■ 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*.

Contents

Notices 2
Summary of amendments 5
Preface 6

Part 1: Overview

1 Overview of PFM - Agent for Platform 19
1.1 Features of PFM - Agent for Platform 20
1.1.1 Collect UNIX performance data 20
1.1.2 Collect performance data according to its characteristics 21
1.1.3 Save performance data 21
1.1.4 Report problems in UNIX operations 21
1.1.5 Easily define alarms and reports 22

2 Performance Monitoring 23
2.1 Overview of performance monitoring 24
2.1.1 Objective of performance monitoring 24
2.1.2 Most important system resources in need of performance monitoring 25
2.1.3 Determining baselines 25
2.2 Overview of user-defined record collection 26
2.2.1 Workgroup information 26
2.2.2 Process operating status information 28
2.2.3 Application operating status information 28
2.2.4 User-specific performance data 28
2.3 Overview of operations on virtualized systems 29
2.3.1 Advantages of virtualized systems 29
2.3.2 Objectives of performance monitoring on virtualized systems 29
2.3.3 System resources most in need of performance monitoring on virtualized systems 30
2.3.4 Installing PFM - Agent for Platform in a virtualized system 30
2.3.5 Setting up PFM - Agent for Platform in a virtualized system 30
2.3.6 Examples of using Agent for Platform - PFM in a virtualized system: Determining whether to change virtualized system settings 31
2.3.7 Examples of using Agent for Platform - PFM in a virtualized system: Determining whether additional physical resources are needed 32
2.3.8 Notes regarding usage of PFM - Agent for Platform in virtualized systems 32
2.3.9 Notes regarding collection of processor information 33
2.3.10 Notes regarding collection of memory information 33
2.3.11 Notes regarding collection of disk information 34

2.3.12 Notes regarding collection of network information 34
2.3.13 Notes regarding collection of process information 34
2.3.14 Virtualized system functionality and collected performance data 34
2.3.15 Data that can be collected with virtualized systems that use zones 35
2.3.16 Data that can be collected with virtualized systems that use PowerVM 38
2.3.17 Data that can be collected on virtualized systems that use HP Integrity Virtual Machines (I-VM) 40
2.3.18 Data that can be collected with virtualized systems that use VMware/KVM 42
2.3.19 Data that can be collected with virtualization systems that use Hitachi Compute Blade logical partitioning feature 44
2.3.20 Data that can be collected with virtualized systems that use WPAR 45
2.3.21 Data that can be collected on a virtualized system based on the Linux container 48
2.3.22 Range of data collected for each virtualized system 52
2.3.23 Monitoring processor performance based on results output by the sar command in an AIX environment 82
2.4 Overview of monitoring templates 94

3 Introduction to Performance Monitoring Operations by Example 95
3.1 Performance monitoring examples 96
3.1.1 Processor monitoring examples 96
3.1.2 Memory monitoring examples 98
3.1.3 Disk monitoring examples 101
3.1.4 Network monitoring examples 104
3.1.5 Process monitoring examples 105
3.1.6 Examples of collecting information about used ports 106
3.1.7 Examples of collecting performance data from multiple hosts on which PFM products are not installed 108
3.2 Performance monitoring definition examples 111
3.2.1 Processor monitoring definition examples 111
3.2.2 Memory monitoring definition examples 114
3.2.3 Disk monitoring definition examples 116
3.2.4 Network monitoring definition examples 118
3.2.5 Process monitoring definition examples 119

Part 2: Setup and Operation

4 Installation and Setup in UNIX 121
4.1 Installation and setup workflow 122
4.2 Preparations required before installation 124
4.2.1 Required OS 124
4.2.2 Network environment setup 124
4.2.3 OS user permissions needed for installation 126
4.2.4 Prerequisite programs 126
4.2.5 Prerequisite for specifying 4,096-byte monitoring conditions for processes 127

4.2.6 Preparations for collecting data in the event of a failure 127
4.2.7 Points to note before installing PFM - Agent for Platform 128
4.3 Installation 130
4.3.1 Order in which to install the programs 130
4.3.2 Installation procedure for PFM - Agent for Platform 130
4.3.3 Notes on installation 133
4.4 Setup 135
4.4.1 Setting the LANG environment variable 135
4.4.2 Registering PFM - Agent for Platform in PFM - Manager and PFM - Web Console Options 137
4.4.3 Network setup Options 139
4.4.4 Changing the log file size Options 139
4.4.5 Changing the performance data storage destination Options 139
4.4.6 Setting up PFM - Manager for the connection destination of PFM - Agent for Platform 140
4.4.7 Setting up the action log Options 140
4.5 Uninstallation 141
4.5.1 Points to note before uninstalling PFM - Agent for Platform 141
4.5.2 Releasing the connection to PFM - Manager 142
4.5.3 Uninstallation procedure 142
4.6 Changing the system configuration of PFM - Agent for Platform 143
4.7 Changing the operation of PFM - Agent for Platform 144
4.7.1 Changing the storage location of performance data 144
4.7.2 Updating the Store version to 2.0 146
4.8 Backup and restoration 149
4.8.1 Backup 149
4.8.2 Restoration 150
4.9 Settings for browsing manuals in a Web browser 151
4.9.1 Settings for browsing manuals 151
4.9.2 Browsing a manual 152
4.10 Notes regarding operation 153

5 Collecting Log Information 155
5.1 Overview of collecting log information 156
5.2 Setup for collecting log information 158
5.2.1 Setting up the event file 158
5.2.2 Setting up PFM - Web Console 159
5.2.3 Notes on collecting log information 159

6 Operation in a Cluster System 161
6.1 Configuration of PFM - Agent for Platform in a cluster system 162
6.2 Processing at failover 164
6.2.1 Failover when an error occurs on a PFM - Agent host 164

6.2.2 Effects of PFM - Manager stopping, and corrective action 165
6.3 Installation and setup in a cluster system 166
6.3.1 Flow for installation and setup in a cluster system 166
6.3.2 Items to be checked before installation in a cluster system 168
6.3.3 Installation procedure in a cluster system 170
6.3.4 Setup procedure in a cluster system 170
6.4 Uninstallation and unsetup in a cluster system 177
6.4.1 Flow of uninstallation and unsetup in a cluster system 177
6.4.2 Unsetup procedure in a cluster system 178
6.4.3 Uninstallation procedure in a cluster system 181
6.5 Changing the PFM - Agent for Platform operation method in a cluster system 182
6.5.1 Exporting and importing the logical host environment definition file in a cluster system 182
6.6 Notes on operating PFM - Agent for Platform in a cluster system 183
6.6.1 Notes on specifying the settings for collecting information about the operating status of applications and processes 183
6.6.2 Notes on collecting user-specific performance data 183
6.6.3 Note on continuity of performance data before and after failover 184
6.6.4 Note on collecting the same records from multiple hosts in the same machine (only in AIX) 184

7 User-Defined Record Collection 185
7.1 Settings for collecting workgroup information 186
7.1.1 Setting a workgroup file for collecting workgroup information 186
7.1.2 Examples of alarm operation when collecting workgroup information 192
7.1.3 Setting up PFM - Web Console 192
7.2 Settings for collecting information about the operating status of processes 193
7.2.1 Overview of collecting information about the operating status of processes 193
7.2.2 Differences in functionality from previous versions 193
7.2.3 Records used to collect information about the operating status of processes 194
7.2.4 Settings in the Agents tree for collecting information about the operating status of processes 195
7.2.5 Settings in the Services tree for collecting information about the operating status of processes 206
7.2.6 Command-based settings for collecting information about the operating status of processes 212
7.2.7 Distinguishing the letter case of monitoring targets 215
7.2.8 Example of responding to an alarm while collecting information about the operating status of processes 217
7.2.9 Notes on collecting information about the operating status of processes 219
7.3 Settings for collecting information about the application operating status 220
7.3.1 User-defined record settings for collecting application operating status information settings 220
7.3.2 Checking or changing user-defined records for collecting application operating status information 223
7.3.3 Deleting user-defined records for collecting application operating status information 224
7.3.4 Example of using an alarm when collecting information about the application operating status 224
7.4 Settings for collecting user-specific performance data 231
7.4.1 Overview of the functionality for collecting user-specific performance data 231

7.4.2 Settings for collecting user-specific performance data 237
7.4.3 Example of collecting user-specific performance data (for a physical host environment) 241
7.4.4 Example of collecting user-specific performance data (for a logical host environment) 243
7.4.5 Format of the jpcuser command 245
7.4.6 Format of user-created data files 247
7.4.7 Checking the debug log to determine whether user-created data is correct 251

Part 3: Reference

8 **Monitoring Templates 258**
Format of alarm explanations 259
List of alarms 260
Disk Service Time 262
I/O Wait Time 263
Kernel CPU 264
Pagescans 265
Run Queue 266
Swap Outs 267
User CPU 268
CPU Per Processor(K) 269
CPU Per Processor(U) 271
Alloc Mem Mbytes 273
File System Free(L) 275
File System Free(R) 277
Disk Busy % 279
Disk Queue 281
Network Rcvd/sec 283
Process End 285
Process Alive 286
Process Existence 288
Application Status 290
Format of report explanations 292
Organization of report directories 293
List of reports 296
Application Status 299
Application Process Count 300
Application Process Status 301
Avg Service Time - Top 10 Devices 302
Avg Service Time Status - Top 10 Devices 303
Console Messages 304
CPU Per Processor Status 305
CPU Per Processor Usage 306
CPU Status 307
CPU Status (Multi-Agent) 309
CPU Trend 310
CPU Trend (Multi-Agent) 311

CPU Usage - Top 10 Processes 312
CPU Usage Summary 313
Device Detail 314
Device Usage Status 315
Device Usage Status (Multi-Agent) 316
Free Space Mbytes - Top 10 Local File Systems 317
I/O Activity - Top 10 Processes 318
I/O Overview 319
Local File System Detail 320
Major Page Faults - Top 10 Processes 321
Memory Paging 322
Memory Paging (8.4) 323
Memory Paging Status 324
Memory Paging Status (8.4) 325
Memory Paging Status (Multi-Agent) 326
Memory Paging Status (Multi-Agent) (8.4) 327
Monitoring Process Detail 328
Network Interface Detail 329
Network Interface Summary (real-time report indicating network usage status) 330
Network Interface Summary (historical report indicating network usage status) 331
Network Overview 332
Network Status 333
Network Status (Multi-Agent) 334
NFS Activity Overview 335
NFS Load Trend 336
NFS Usage Status 337
NFS Usage Status (Multi-Agent) 338
Paging Trend (Multi-Agent) 339
Paging Trend (Multi-Agent) (8.4) 340
Process Detail 341
Process Overview 343
Process Summary Status 344
Process Trend 345
Remote File System Detail 346
Space Usage - Top 10 Local File Systems 347
Space Usage - Top 10 Remote File Systems 348
System Overview (real-time report indicating system operation status) 349
System Overview (historical report indicating system operation status) 351
System Overview (8.4) (historical report indicating system operation status) 353
System Utilization Status 355
Workload Status 356
Workload Status (Multi-Agent) 357

9 Records 358

Data model 359
Format of record explanations 360
List of ODBC key fields 363

Summarization rules 364
List of data types 366
Field values 367
Fields added only when data is stored in the Store database 370
Fields that are output when data stored in the Store database is exported 371
Notes on records 372
List of records 377
Application Process Detail (PD_APPD) 379
Application Process Interval (PD_APSI) 381
Application Process Overview (PD_APS) 388
Application Summary (PD_APP) 395
Application Summary Extension (PD_APP2) 403
CPU - Per Processor Detail (PI_CPUP) 406
Device Detail (PI_DEVD) 411
Device Summary (PI_DEVS) 415
File System Detail - Local (PD_FSL) 419
File System Detail - Remote (PD_FSR) 423
Logged Messages (PL_MESS) 427
Network Interface Detail (PI_NIND) 429
Network Interface Summary (PI_NINS) 435
Process Detail (PD) 438
Process Detail Interval (PD_PDI) 447
Process Summary (PD_PDS) 454
Program Summary (PD_PGM) 457
System Summary Overview (PI) 460
Terminal Summary (PD_TERM) 475
User Data Detail (PD_UPD) 478
User Data Detail - Extended (PD_UPDB) 480
User Data Interval (PI_UPI) 483
User Data Interval - Expanded 1 (PI_XUI1) 486
User Data Interval - Expanded 2 (PI_XUI2) 491
User Data Interval - Expanded 3 (PI_XUI3) 492
User Data Interval - Expanded 4 (PI_XUI4) 493
User Data Interval - Expanded 5 (PI_XUI5) 494
User Data Interval - Extended (PI_UPIB) 495
User Summary (PD_USER) 499
Workgroup Summary (PI_WGRP) 502

10 Commands 506
Command description format 507
jpcappcvt 508

11 Messages 510
11.1 Message format 511
11.1.1 Format of output messages 511
11.1.2 Format of message explanations 512

11.2 Message output destinations 513
11.3 List of messages output to syslog 518
11.4 Messages 519

Part 4: Troubleshooting

12 Error Handling Procedures 549
12.1 Error handling procedures 550
12.2 Troubleshooting 551
12.3 Log information to be collected for troubleshooting 552
12.3.1 Types of log information to be collected 552
12.3.2 Log files and directories to check 553
12.4 UNIX-related data to be collected for troubleshooting 554
12.4.1 OS log information to be collected 554
12.4.2 Performance Management information to be collected 555
12.4.3 Operation information to be collected 557
12.4.4 Error information to be collected 558
12.4.5 Performance data information to be collected 558
12.4.6 Other information to be collected 563
12.5 Procedures for collecting UNIX-related data for troubleshooting 564
12.6 Performance Management error detection 565
12.7 Performance Management system error recovery 566

Appendixes 567
A Estimating System Requirements 568
A.1 Estimating memory requirements 568
A.2 Estimating disk space requirements 568
B List of System Resources for Which Kernel Parameters Need to Be Adjusted 569
B.1 For HP-UX 569
B.2 For Solaris 570
B.3 For AIX 570
B.4 For Linux 571
C List of Identifiers 572
D List of Processes 573
E List of Port Numbers 574
E.1 Port numbers of PFM - Agent for Platform 574
E.2 Firewall passage directions 574
F Properties of PFM - Agent for Platform 575
F.1 Agent Store service properties 575
F.2 Agent Collector service properties 578
G List of Files and Directories 592
H Upgrade Procedure and Notes on Upgrading 595

I Version Compatibility 597
J Outputting Action Log Data 598
J.1 Types of events output to the action log 598
J.2 Format for saving the action log files 598
J.3 Format of output action log data 599
J.4 Settings for outputting action log data 603
K Linking to JP1/SLM 606
L Communication in IPv4 and IPv6 Environments 607
M Version Changes 608
M.1 Changes in 12-50 608
M.2 Changes in 12-00# 608
M.3 Changes in 11-50 609
M.4 Changes in 11-10 610
M.5 Changes in 11-01 610
M.6 Changes in 11-00 610
M.7 Changes in 10-00 614
M.8 Changes in 09-10 615
M.9 Changes in 09-00 616
N Reference Material for This Manual 619
N.1 Related publications 619
N.2 Conventions: Abbreviations for product names 619
N.3 Conventions: Acronyms 622
N.4 Conventions: Format of product names, service IDs, and service keys in this manual 623
N.5 Conventions: KB, MB, GB and TB 624
N.6 Conventions: Performance Management installation folders 624
O Glossary 626

Index 633

1 Overview of PFM - Agent for Platform

This chapter provides an overview of PFM - Agent for Platform.

1.1 Features of PFM - Agent for Platform

PFM - Agent for Platform is a program for collecting and managing performance data to monitor the performance of a system running under UNIX. This program is installed and used on hosts to be monitored.

PFM - Agent for Platform has the following features:

- Analysis of UNIX operating status

  From the UNIX system being monitored, you can use PFM - Agent for Platform to collect and summarize performance data, such as CPU usage and the number of active processes. By displaying the trends in graphs, you can easily analyze UNIX operating status.

- Early detection of problems in UNIX operation and provision of information for troubleshooting

  If a problem, such as a system resource shortage, occurs in the UNIX system being monitored, PFM - Agent for Platform can send an email notification, for example, to the user to ensure early detection of the problem. Furthermore, by graphically displaying the information related to the problem, PFM - Agent for Platform can provide troubleshooting information.

PFM - Agent for Platform requires PFM - Manager and PFM - Web Console.

Note

In addition to PFM - Agent for Platform, PFM - RM for Platform is another product that can analyze UNIX operation statuses in Performance Management. PFM - RM for Platform does not need to be installed on the monitored host, allowing host performance data to be monitored without any modification to the monitored host environment (*remote monitoring*). Note that PFM - RM for Platform collects fewer types of performance data when compared to PFM - Agent for Platform, but a single instance of PFM - RM for Platform can monitor performance data for multiple hosts.

If you need to perform remote monitoring or to collect and manage performance data for multiple monitored hosts, consider installing PFM - RM for Platform.

1.1.1 Collect UNIX performance data

You can use PFM - Agent for Platform to collect UNIX performance data, such as system resource usage.

PFM - Agent for Platform lets you use performance data in the following ways:

- Displaying the operating status of UNIX

  Using PFM - Web Console, you can process performance data into a graphical format called *report* and display it. Reports make it easy to analyze the UNIX operating status.

  The following types of reports are available:

  - Real-time report

    This report shows the current status of the UNIX system being monitored. It is used primarily for checking the system's current status and problems. The real-time performance data collected is used to directly display a real-time report.

  - Historical report

This report shows the status of the UNIX system being monitored from past to present. The report is used primarily for analyzing system trends. Performance data stored in the database of PFM - Agent for Platform is used to display a historical report.

- Using performance data to determine whether a problem has occurred

  If the performance data value collected indicates some sort of error, an action can be taken, such as notifying the user.

1.1.2 Collect performance data according to its characteristics

Performance data is collected in the form of a *record*. Each record is further divided into units called *fields*. Records and fields are generically referred to as *data models*.

According to their characteristics, records can be classified into three types. The kind of performance data to be collected by each type of record is defined by PFM - Agent for Platform. Using PFM - Web Console, the user can select the kinds of performance data to be collected as records.

The following record types are available in PFM - Agent for Platform:

- Product Interval record type (hereafter referred to as *PI record type*)

  Performance data for a specific duration (interval), such as the number of active processes during every minute, is collected as a PI record. Use the PI record type when you want to analyze the changes and trends in the system status over time.
- Product Detail record type (hereafter referred to as *PD record type*)

  Performance data that indicates the system status at a given point in time, such as detailed information on the processes that are currently active, is collected as a PD record. Use the PD record type when you want to determine the system status at a given point in time.
- Product Log record type (hereafter referred to as *PL record type*)

  Log information from applications being run on UNIX or databases is collected as a PL record.

For details about these record types, see *9. Records*.

1.1.3 Save performance data

By saving the collected performance data in the database of PFM - Agent for Platform called the *Store database*, you can save the performance data up to the present and analyze the trends in the UNIX operating status. To analyze trends, you use historical reports.

Using PFM - Web Console, the user can select the performance data records to be saved in the Store database. For details about how to select records in PFM - Web Console, see the chapter explaining how to manage operation monitoring data in the *JP1/Performance Management User's Guide*.

1.1.4 Report problems in UNIX operations

Performance data collected by PFM - Agent for Platform can be displayed as a report on UNIX performance, and can warn the user when a UNIX operation problem or error occurs.

For example, when the amount of available system resources falls to 10% of its capacity, an email notification can be sent to the user. To accomplish this, available system resource level of less than 10% is set as the threshold, and PFM - Agent for Platform can be set to send an e-mail notification when this threshold is reached. The operation that occurs when the threshold is reached is called an *action*. The following types of actions are available:

- Email transmission
- Command execution
- SNMP trap issuance
- JP1 event issuance

Thresholds and actions are defined in a unit called an *alarm*. A table in which one or more alarms are collected is called an *alarm table*. After you define an alarm table, you associate it with PFM - Agent for Platform. The operation of associating an alarm table with PFM - Agent for Platform is called *binding*. Once binding occurs, when the performance data collected by PFM - Agent for Platform reaches the threshold defined in the alarm, the user can be notified.

By defining alarms and actions in this way, you can detect UNIX operational problems early and take the necessary corrective measures.

For details about how to specify alarms and actions, see the chapter explaining how to use alarms to monitor operations in the *JP1/Performance Management User's Guide*.

1.1.5 Easily define alarms and reports

PFM - Agent for Platform provides a *monitoring template*, which contains reports and alarms in which necessary information is already defined. The monitoring template facilitates preparation for monitoring UNIX operating statuses without complex definitions. The monitoring template can also be customized to suit the user environment. For details about how to use the monitoring template, see the chapter explaining the creation of reports for operation analysis or operation monitoring by alarms in the *JP1/Performance Management User's Guide*. For details about the monitoring template, see *8. Monitoring Templates*.

2 Performance Monitoring

This chapter describes the use of PFM - Agent for Platform to monitor performance.

2.1 Overview of performance monitoring

- Performance data collection methods

  For details about performance data collection methods, see the chapter explaining Performance Management functions in the *JP1/Performance Management Planning and Configuration Guide*.

  For details about the performance data values that are collected, see *9. Records*.

- Performance data management methods

  For details about the PI and PD record type performance data management methods, see the chapter explaining operation monitoring data management in the *JP1/Performance Management User's Guide*.

  When the PL record exceeds the upper limit of the set number of record items, the record is overwritten, starting with the oldest one.

  You can use PFM - Web console to change the upper limit of the number of record items. For details about how to set the upper limit of the number of record items, see the chapter explaining operation monitoring data management in the *JP1/Performance Management User's Guide*.

You use PFM - Web Console to select which records collected and managed by PFM - Agent are to be used as performance data. For details about how to select performance data, see the chapter explaining operation monitoring data management in the *JP1/Performance Management User's Guide*.

2.1.1 Objective of performance monitoring

The monitoring of performance is an important task in the creation and management of a system environment.

You can achieve the following major performance monitoring objectives by using PFM - Agent for Platform:

- Determine the causes of bottlenecks (by analyzing performance data).
- Understand load characteristics and their impact on the corresponding system resources (by analyzing performance data trends).
- Verify whether the system is operating correctly (by monitoring it).

Certain types of bottlenecks can affect the operating performance of the entire system, such as bottlenecks produced by the following causes:

- Insufficient memory
- Monopolization of specific resources by programs
- A damaged subsystem
- An incorrect subsystem configuration (for example, 10 Mbps is set for a 100-Mbps NIC)
- Unbalanced subsystem loads

By monitoring performance under different conditions (such as different numbers of users connected concurrently) and by monitoring performance continuously, you can check the load characteristics and the impact on the corresponding system resources in the system environment. The benefits of such monitoring are as follows:

- Changing and adjusting the system configuration are simplified.
- System resource upgrades can be planned.

Checking whether a production system is operating correctly is extremely important. You can check whether the system environment is operating correctly by using the following types of monitoring together with performance monitoring:

- Monitoring of the operation of processes that provide system functionality
- Monitoring of the operation of services that are prerequisite for the system
- Monitoring the system for invalid processes

Using PFM - Agent for Platform as described above to correctly monitor performance is vital for stable system operation.

2.1.2 Most important system resources in need of performance monitoring

The performance monitoring performed by PFM - Agent for Platform treats the following system resources as being the most important:

- Processor
- Memory
- Disks
- Network
- Processes

PFM - Agent for Platform provides a monitoring template containing these important items.

2.1.3 Determining baselines

Determining baselines is a task in which system values unlikely to affect normal system operation are determined from the performance measurement results.

PFM products use baseline values as thresholds for monitoring system operation. The determination of baselines is therefore an important task for successfully monitoring performance.

Note the following recommendations when you determine baselines:

- Measure system performance at peak load (for example, when a high-load test is run for the operating environment).
- Thresholds depend greatly on the system configuration. Therefore, performance should be measured again to redetermine the baselines if system resources and the operating environment have been changed.

2.2 Overview of user-defined record collection

PFM - Agent for Platform can collect performance data not provided by default and store it in a record. This record for storing performance data is called a *user-defined record*. For details about how to collect user-defined records, see *7. User-Defined Record Collection*.

The following table shows the information for which user-defined records can be set and the records corresponding to type of information.

Table 2–1: Information that can be set for user records, the corresponding records, and how user records are set

Information supporting user-defined record collection	Records
Workgroup information	PI_WGRP record
Process operating status information	PD_APP2 record
	PD_APPD record
	PD_APS record
	PD_APSI record
Application operating status information	PD_APP record
User-specific performance data	PD_UPD record
	PD_UPDB record
	PI_UPI record
	PI_UPIB record
	PI_XUI1 record
	PI_XUI2 record
	PI_XUI3 record
	PI_XUI4 record
	PI_XUI5 record

Like other records, user-defined records specified on each host can be used for displaying reports in PFM - Web Console and issuing alarms based on monitoring.

When multiple pieces of performance data are collected into each record, a new line is added for each field in the user-defined record as each piece of performance data is collected. As a result, each user-defined record becomes a multi-line record. A multi-line record is a multi-instance record.

2.2.1 Workgroup information

When multiple users use UNIX system resources or operate a UNIX group, PFM - Agent for Platform allows you to set those multiple UNIX users or the UNIX group as a workgroup and collect process information related to the workgroup.

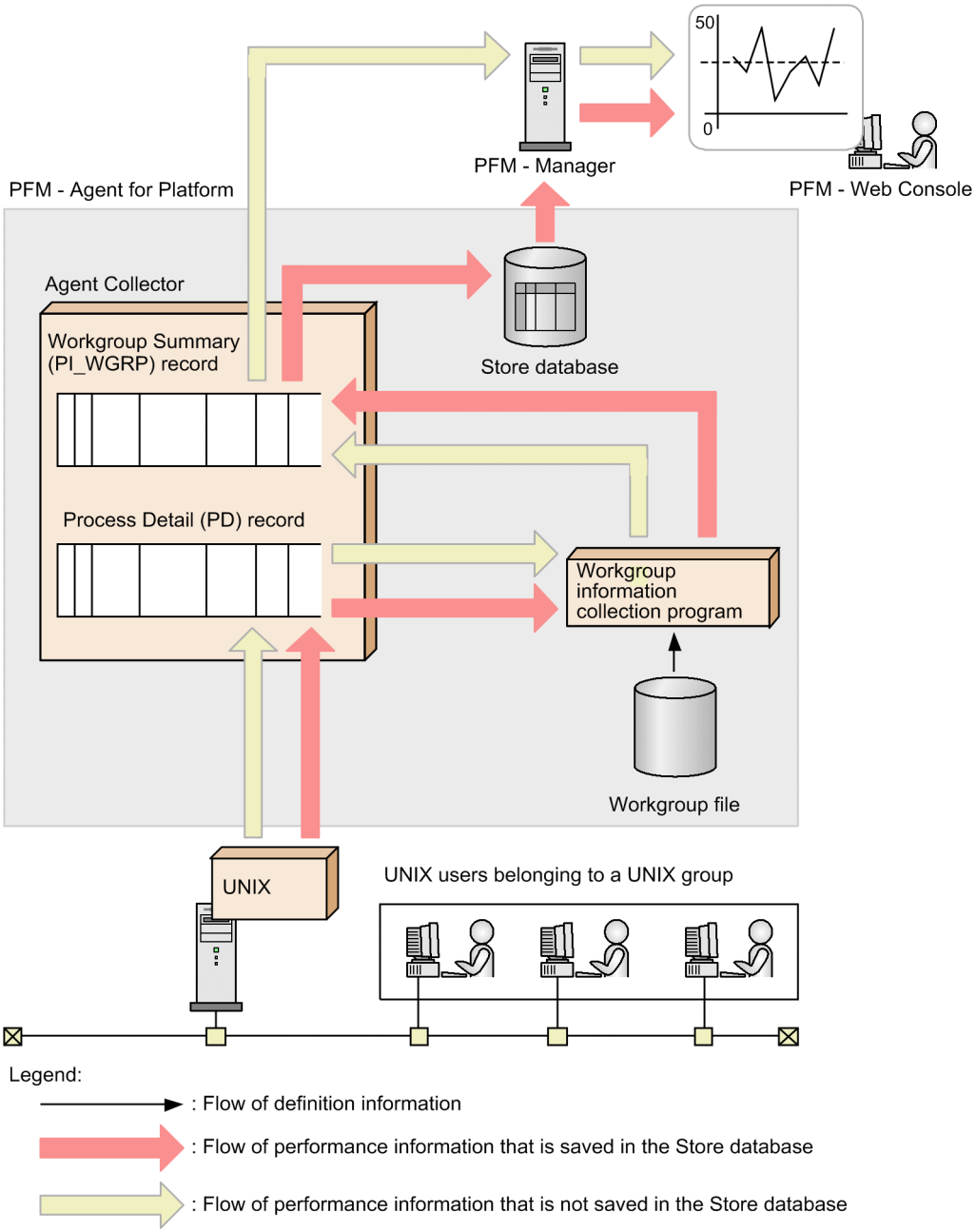
You can set a workgroup in the following units:

- UNIX user
- UNIX group
- Program executed by process

The PFM - Agent for Platform workgroup information collection program summarizes workgroup-related data of performance data of the Process Detail (PD) record based on the workgroup name set in the workgroup file. Summarized performance data is managed as the Workgroup Summary (PI_WGRP) record. As with other records, you can use the Workgroup Summary (PI_WGRP) record to display reports and monitor alarm usage.

The figure below shows the flow of workgroup information monitoring.

Figure 2–1: Workgroup information monitoring flow



2.2.2 Process operating status information

The following records are used to collect and manage information about whether processes are operating under specified conditions and whether the number of processes is the expected number or less:

- Application Summary Extension (`PD_APP2`)
- Application Process Detail (`PD_APPD`)
- Application Process Overview (`PD_APS`)
- Application Process Interval (`PD_APSI`)

You can specify the processes to be monitored in PFM - Web Console.

2.2.3 Application operating status information

PFM - Agent for Platform can collect information about whether processes are operating under specified conditions and whether the number of processes is the expected number or less. PFM - Agent for Platform can then manage the collected information as the Application Summary (`PD_APP`) record. The processes to be monitored can be specified in PFM - Web Console.

2.2.4 User-specific performance data

PFM - Agent for Platform can collect user-specific performance data such as information that is not provided by PFM - Agent for Platform and information specific to a machine or environment. To collect user-specific performance data, you need to create scripts called *user commands*.

There are seven types of PI records and two types of PD records that can store user-specific performance data.

- User Data Detail (`PD_UPD`)
- User Data Detail - Extended (`PD_UPDB`)
- User Data Interval (`PI_UPI`)
- User Data Interval - Extended (`PI_UPIB`)
- User Data Interval - Expanded 1 (`PI_XUI1`)
- User Data Interval - Expanded 2 (`PI_XUI2`)
- User Data Interval - Expanded 3 (`PI_XUI3`)
- User Data Interval - Expanded 4 (`PI_XUI4`)
- User Data Interval - Expanded 5 (`PI_XUI5`)

2.3 Overview of operations on virtualized systems

This subsection explains how to use PFM - Agent for Platform to monitor performance in virtualized systems.

2.3.1 Advantages of virtualized systems

Virtualized systems are attracting attention as a technology that makes the most of excess system resources. For example, with the current one-server, one-application paradigm, most servers consistently have excess resources. Specifically, except for times of server job congestion, usually only 5% to 7% of resources are used. Virtualized systems are a way to solve this issue.

Virtualized systems also have the following advantages:

- Operating systems and devices can be shared and centrally managed.
- Several old servers can be made into a virtual server, and run as one new server.
- The number of servers can be reduced, cutting electricity and air conditioning costs.

2.3.2 Objectives of performance monitoring on virtualized systems

As discussed above, the benefits of a virtualized system are greatest when system resources are used in the proper amounts. Virtualized systems cannot be used effectively if system resources are overconsumed, or at the other extreme, hardly used at all.

In situations like these, the status of resource consumption can be monitored to determine whether the current system is set up effectively.

At a basic level, PFM - Agent for Platform is designed to collect information about a single physical operating system. However, it can be used creatively to monitor performance information that depends on virtualized systems.

The main objectives of using PFM - Agent for Platform to monitor virtualized system performance are as follows:

- Analyzing performance data trends to better understand impact on load allocation and the affected system resources.
- Analyzing performance data to find the causes of bottlenecks.
- Monitoring whether the operating system is running properly in the virtualized system.

In virtualized systems, specific bottlenecks might adversely affect performance of the entire virtualized system. The following may cause such bottlenecks:

- Insufficient memory on a logical partition
- Load allocation between virtualized systems
- Monopolization of a specific resource by a program

The following explains methods for using PFM - Agent for Platform to monitor the performance of virtualized systems. Note that the thresholds shown here for performance monitoring are just reference values. Use the appropriate values based on the system configuration and usage.

2.3.3 System resources most in need of performance monitoring on virtualized systems

The following are the most important system resources to be monitored for performance:

- Processors
- Memory
- Disks
- Networks
- Processes

In a virtualized system, some performance data for the above resources might not be collected or might become meaningless when changed dynamically. Performance changes for each OS, or by each virtualized system.

2.3.4 Installing PFM - Agent for Platform in a virtualized system

In a virtualized system, each type of resource is basically managed by virtualized functionality (software or hardware), and configured by being allocated to different logical partitions. The managed resources include processors, memory, disks, and networks.

These resources can be obtained as performance data on the logical partition on which PFM - Agent for Platform is installed.

Some virtualized systems distinguish between guest operating systems and host operating systems. Examples of such systems include VMware/KVM, Hitachi Compute Blade logical partitioning feature, and other software-based virtualizations. The host OS manages the guest OS, and the guest OS runs on the host OS. In such cases, PFM - Agent for Platform is installed on the guest OS to monitor performance.

PFM - Agent for Platform collects performance data on the installed logical partition, or guest OS.

Note

Because the host OS for VMware is a special OS for managing virtualized functionality, operation of applications is not guaranteed. Since Hitachi Compute Blade logical partitioning feature creates an off-disk management area called an *SVP (Service Processor) frame*, data collected by PFM - Agent for Platform cannot be saved as a database in this management area.

2.3.5 Setting up PFM - Agent for Platform in a virtualized system

The procedures for collecting information for an installed instance of PFM - Agent for Platform are the same for virtualized systems. The existing settings can be used as is.

When a virtual machine is set up using a virtualized system, and PFM - Agent for Platform is installed on the machine, PFM - Agent for Platform collects performance data about the virtual machine.

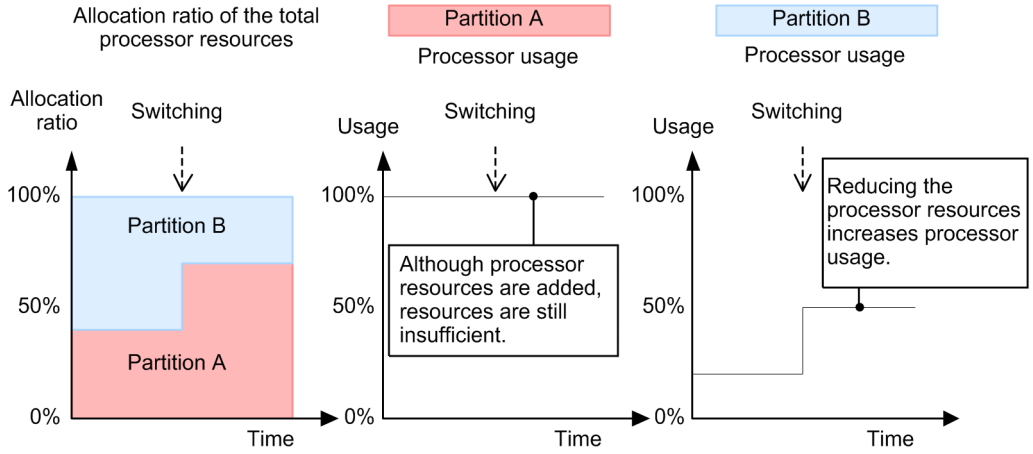
2.3.6 Examples of using Agent for Platform - PFM in a virtualized system: Determining whether to change virtualized system settings

If there are extra physical resources despite the consumption of a significant amount of resources on a specific guest OS or logical partition, we recommend that you change the virtualized system settings.

PFM - Agent for Platform can be used as a tool to determine these kinds of situations.

The following graph shows a case in which the allocation of processor resources is not sufficient.

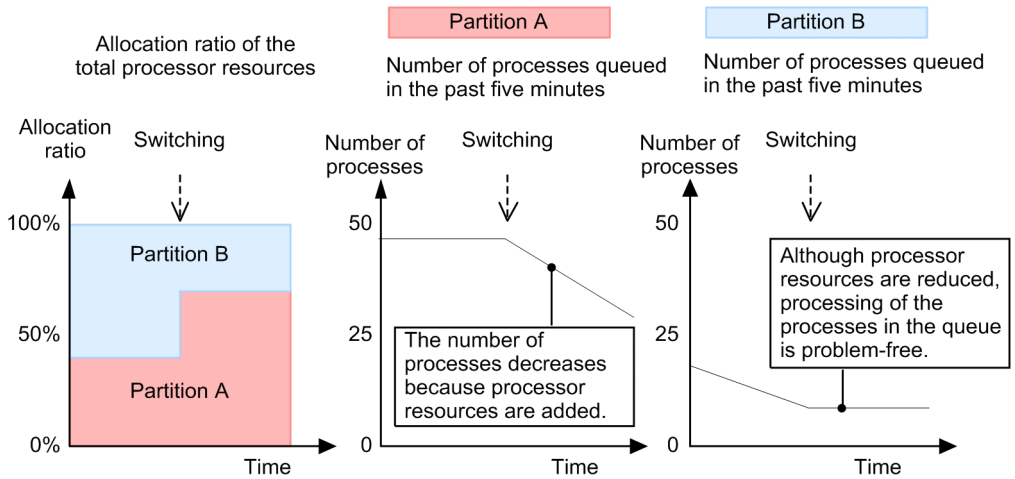
Figure 2–2: Case in which processor resource allocation is not sufficient (during processor usage monitoring)



Because the processor usage for logical partition A is 100%, the allocation ratio is being switched. However, the processor usage for logical partition A does not decrease, while the processor for logical partition B is not fully used. In cases like this, change the allocation ratio settings.

The following graph shows a case in which the allocation of processor resources is sufficient.

Figure 2–3: Case in which processor resource allocation is sufficient (during processor usage monitoring)



The processor resource ratio is being switched while there are many processes in the queue for logical partition A. As a result, the number of processes in the queue for logical partition A is decreasing.

2.3.7 Examples of using Agent for Platform - PFM in a virtualized system: Determining whether additional physical resources are needed

If physical resources always seem to be heavily loaded, consider adding another physical resource.

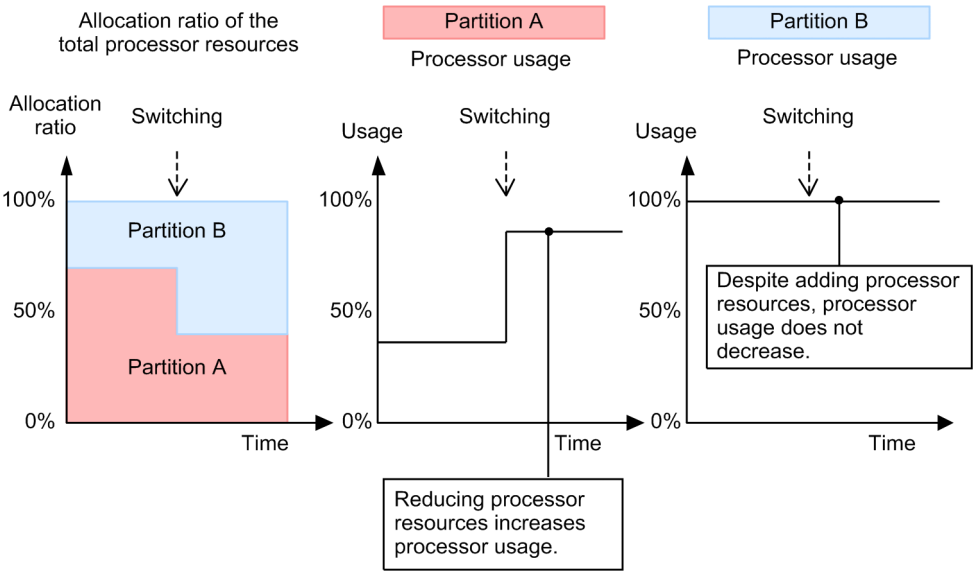
Note, however, that even when a single guest OS or logical partition is heavily loaded in a virtualized system, the system-wide load might seem light. Therefore, check whether all guest OSs or all logical partitions are heavily loaded.

When using PFM - Agent for Platform to check the load status in a virtualized system, you might require a different perspective from the one you use for an ordinary environment.

A virtualized system contains resources that can be changed dynamically, such as processors and memory. Therefore, when monitoring processor and memory usage, also monitor the items not easily affected by dynamic changes in resources, such as the current queue length and paging frequency.

The following graph gives an example of processor resources.

Figure 2–4: Case in which additional processor resources are recommended (during processor usage monitoring)



In the figure example, the allocation ratio is being switched due to the heavy load on processor usage for logical partition B, but usage is high for both logical partitions A and B. In this kind of case, physical resources are insufficient.

2.3.8 Notes regarding usage of PFM - Agent for Platform in virtualized systems

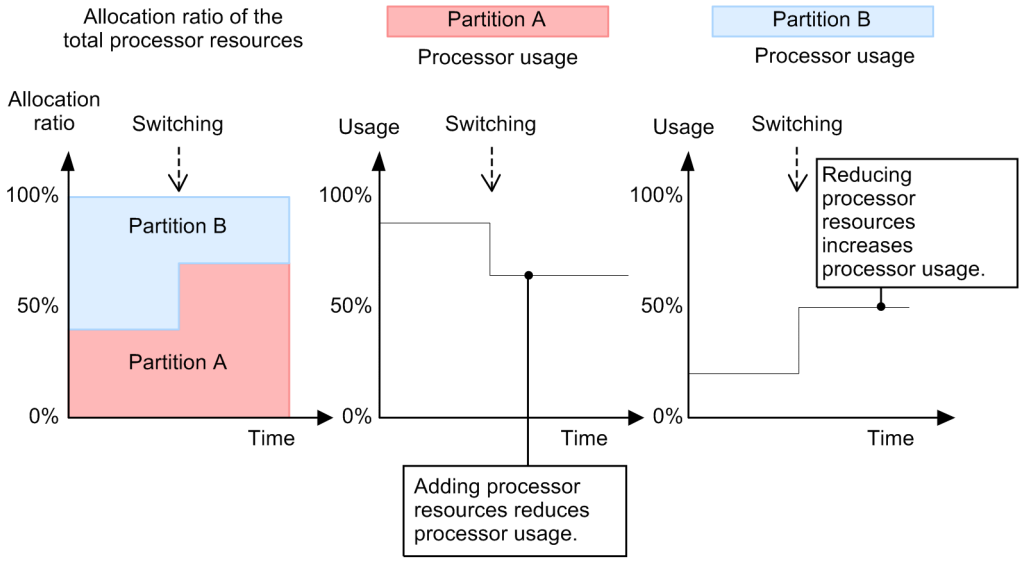
The performance data collected by PFM - Agent for Platform in a virtualized system is specific to each virtual machine. Therefore, install PFM - Agent for Platform on each virtual machine.

Performance data about each virtual machine cannot be collected from the overall virtualized system, or from the management machine.

2.3.9 Notes regarding collection of processor information

The processor usage collected in a virtualized system might not yield the intended values. This is because the virtualized system divides up processor resources and dynamically allocates them to other logical partitions. For example, if significant amounts of processor resources are being used in a specific logical partition due to a heavy load, only limited processor resources might be available for other logical partitions.

Figure 2–5: Case in which collected values differ due to dynamic resource changes



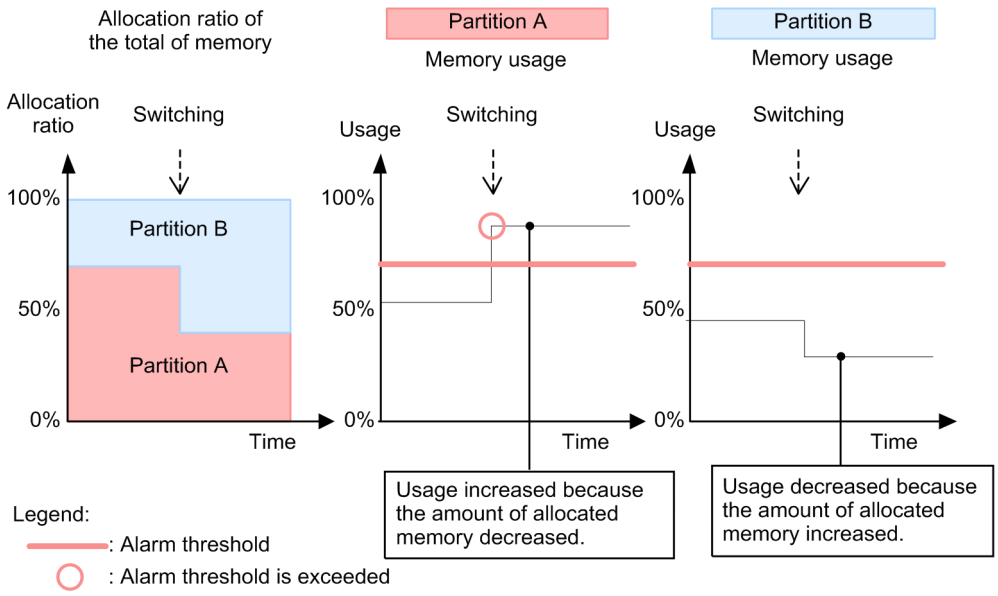
Note that if an application on a given logical partition is using a physical processor, jobs on other logical partitions might be forced to wait due to the operation. In this case, processor usage cannot be accurately ascertained.

2.3.10 Notes regarding collection of memory information

As with processors, information about memory usage in a virtualized system may also differ from the system-wide ratio, due to dynamic resource changes.

For example, when 256 MB of memory area is allocated, and only 64 MB of it is used, PFM - Agent for Platform displays a value of 25% for the memory usage allocation. In this case, when the memory allocation area is dynamically changed from 256 MB to 128 MB, PFM - Agent for Platform displays 50% for the memory usage. Keep in mind that an unexpected warning might occur when a memory resource is specified for an alarm.

Figure 2–6: Case in which the memory resource is specified for an alarm



2.3.11 Notes regarding collection of disk information

As with processor resources, sometimes disk busy rates cannot be accurately ascertained. For example, on a virtualized system, some jobs may be using physical disk resources while others are waiting due to operation.

2.3.12 Notes regarding collection of network information

PFM - Agent for Platform measures only incoming and outgoing packets for network information. Therefore, operation in a virtualized system is the same as normal usage, but in some cases virtual network information is displayed for network information. This is not an issue, because PFM - Agent for Platform collects network information recognized by the OS running on the virtual machine.

2.3.13 Notes regarding collection of process information

Process-related performance data consists of items that show the process count and items that show the amount of resources used by processes. In both cases, since data on the logical partition is obtained for virtualized systems, this process information does not reflect the entire virtualized system.

PFM - Agent for Platform collects both the number of processes on each logical partition, and the amount of resources being used by the processes.

2.3.14 Virtualized system functionality and collected performance data

In virtualized systems, the setup of provided resources differs depending on how the system is implemented. Also, resource changes might or might not require temporary stoppage of a logical partition. If a logical partition needs to be stopped, this should not affect much because PFM - Agent for Platform is also stopped. However, if a logical partition does not need to be stopped, and the system resources change dynamically, the performance data obtained by PFM -

Agent for Platform might be significantly affected. In this case, we recommend that you collect performance data not significantly affected by dynamic resource changes.

The following table shows the resources that can be collected for each virtualized system.

Table 2–2: Resources for each virtualized system

Virtualized system	Resource			
	Processor count	Memory	Disk	Network
Zone	--	--	TS	--
PowerVM	DS	DS	TS	TS
HP Integrity Virtual Machines (I-VM)	TS	TS	TS	TS
VMware/KVM	TS	TS	TS	TS
Hitachi Compute Blade logical partitioning feature	TS	TS	--	TS
WPAR	DS	DS	DS	--
Linux container#	--	--	TS	TS

Legend:

DS: Dynamic change is possible.

TS: Logical partitions or the Linux container must be suspended for changes to take effect in them or the Linux container itself.

--: Not supported

#:

PFM - Agent for Platform supports Docker and Podman as a Linux container engine.

2.3.15 Data that can be collected with virtualized systems that use zones

(1) Zone functionality

Unlike the logical resource partitioning functions implemented in other OSs (for example, LPAR in AIX and VMware), the zone functionality installed in Solaris provides an application workspace. Accordingly, the positioning of a virtualized system based on zones slightly differs from that of virtualized systems on other OSs. Despite the difference, however, zones are used like similar functions in other OSs (for example, for the implementation of multiple OSs in a single system).

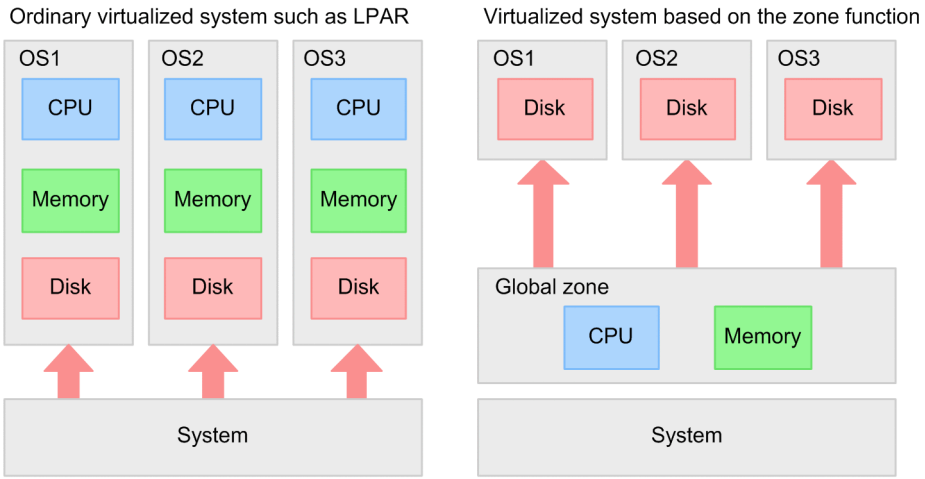
(2) Structure of the zone functionality

There are two types of zones: a global zone and a non-global zone. The global zone has almost the same meaning as the entire system, whereas a non-global zone represents an application workspace provided by the system.

In an ordinary virtualized system, resources are completely separated from the system when viewed by the user. In a virtualized system based on the zone functionality, resources other than disks are basically owned by the global zone.

PFM - Agent for Platform can collect information about the global zone and non-global zones. Some information cannot be collected using records in non-global zones. For details, see Table 2-10 When PFM - Agent for Platform is installed in the non-global zone in an environment in which zones have been created in *2.3.22(1)(a) Zone environment*.

Figure 2–7: Concept of an ordinary virtualized system and the zone functionality

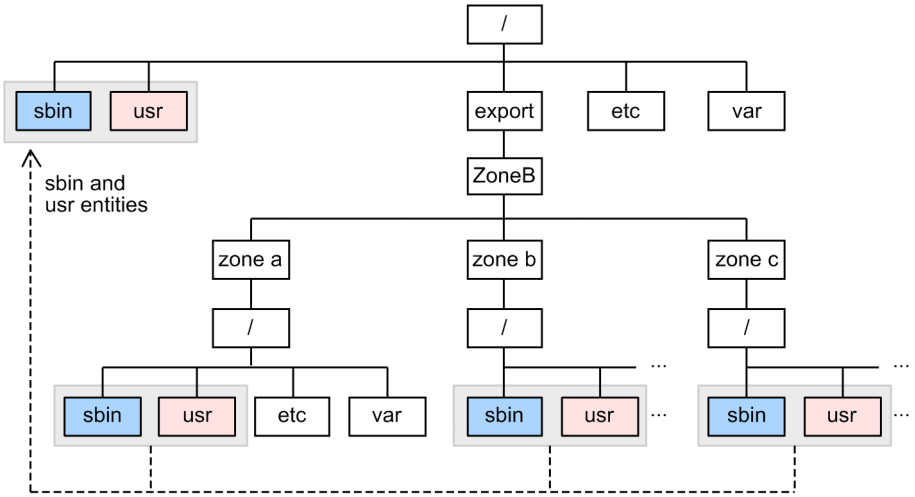


In a non-global zone, the zone functionality operates by using a single file system as seen from the global zone.

Part of the file system of the global zone is shared in the non-global zone.

In the following example, the entities of `sbin` and `usr` under `zone a`, `zone b`, and `zone c` are `sbin` and `usr` under `/`.

Figure 2–8: Example of a zone configuration



(3) Monitoring processes in the global zone

When PFM - Agent for Platform is running in the global zone, you can set the `ALL Zone Collection for Process` property to select the zone for which process information is collected. You can set either of the following values for the `ALL Zone Collection for Process` property.

`Yes`

Collect process information for the global zone and non-global zones.

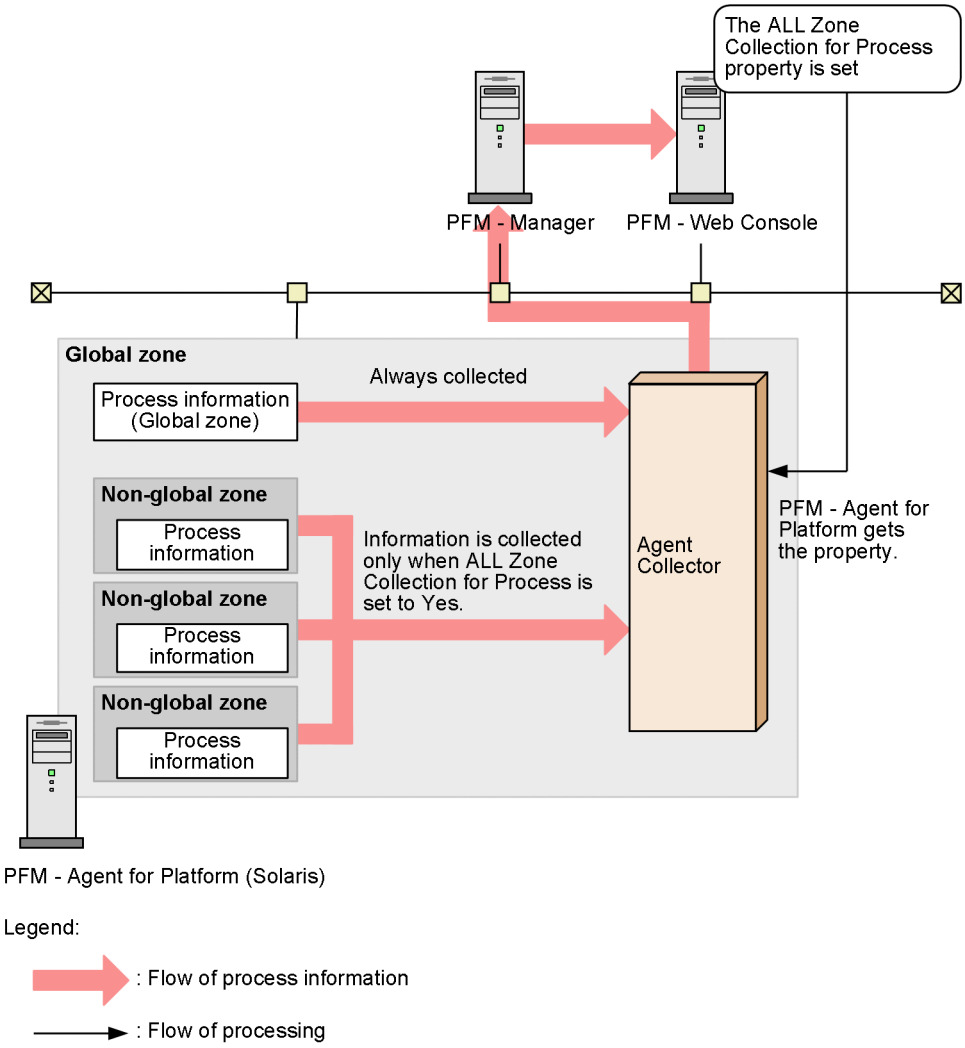
`No`

Collect process information for the global zone.

The `ALL Zone Collection for Process` property is set to `Yes` by default.

The following figure shows process information that is collected when PFM - Agent for Platform is running in the global zone.

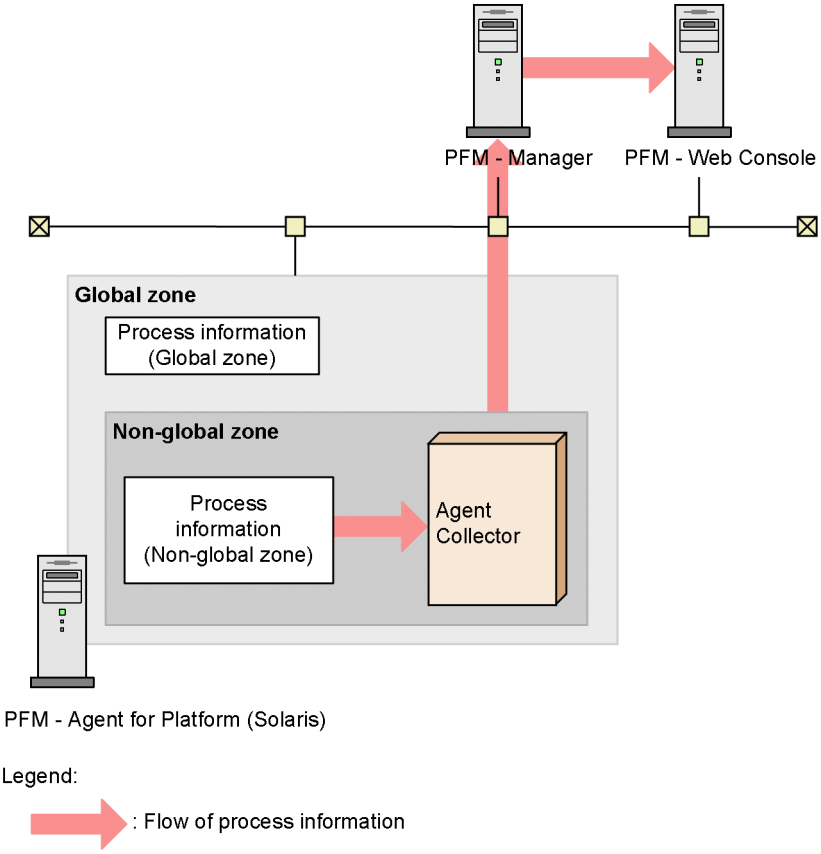
Figure 2–9: Process information collected for the global zone



(4) Monitoring processes in a non-global zone

When PFM - Agent for Platform is running in a non-global zone, information only about that non-global zone is collected regardless of the setting of the `ALL Zone Collection for Process` property. The following figure shows the process information that is collected when PFM - Agent for Platform is running in a non-global zone.

Figure 2–10: Process information collected for a non-global zone



2.3.16 Data that can be collected with virtualized systems that use PowerVM

(1) PowerVM functions

PowerVM is extended functionality that implements an integrated virtualized system. PowerVM provides the following functions:

- Micro-Partitioning
  This function allocates processor resources in 10% units to an LPAR.
- Simultaneous multithreading (SMT)
  This function simultaneously executes two threads on one physical processor.
- Virtual I/O server (VIOS)
  As one of the major components of the PowerVM functionality, this function virtualizes both I/O resources and network resources.
- Virtual SCSI adapter
  This function divides up a physical disk into logical partitions and allocates them to virtual machines.
- Shared Ethernet adapter
  This adapter is used for sharing a physical Ethernet adapter on an I/O server by the virtual Ethernet adapters of multiple virtual machines.
- Integrated Virtual Manager (IVM)

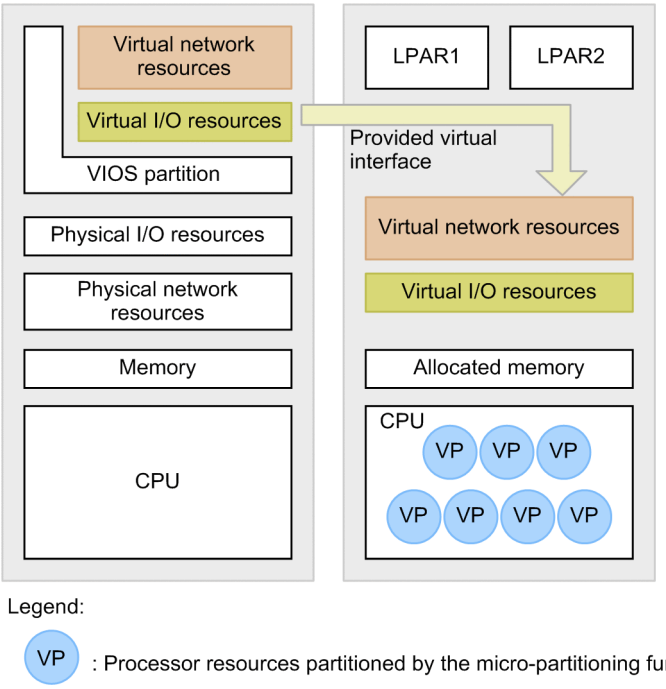
This virtualized system software enables you to use a Web browser to perform logical partitioning easily without using a special management terminal.

(2) PowerVM structure

PowerVM uses components to implement a virtualized system.

A simple explanation of the PowerVM structure is that the Micro-Partitioning function allocates one processor resource to multiple LPARs so that the LPARs can use physical device resources limited by the VIOS function. IVM (PowerVM) is the software that facilitates the use of these resources.

Figure 2–11: PowerVM structure



(3) Using PFM - Agent for Platform on PowerVM

When using PFM - Agent for Platform on PowerVM, note that information about processors and memory is changed dynamically. Also note that when you use records for handling processor information, a different type of operation might be required depending on whether your virtualized system creates the same number of instances as the processors that can be allocated.

Processor information

For a record that indicates processor information, the number of instances displayed is ten times as many as the number of processors partitioned by the Micro-Partitioning function. Note, however, that this only means that a maximum of 10 logical processors can be allocated to one processor, and a single instance will not always be 10% of the processor resources. Thus, when one physical processor is partitioned into four logical processors by using the Micro-Partitioning function, the total usage of these processors will be 100%. The remaining six logical processors that have not been allocated are displayed as `Offline`, which means that they are not in use.

When the SMT function is used, one physical processor is partitioned into two logical processors. Operation is therefore the same as when the Micro-Partitioning is used.

PFM - Agent for Platform enables a setting to be made so that processor performance data can be collected from the results output by the `sar` command of AIX.

For details see *2.3.23 Monitoring processor performance based on results output by the sar command in an AIX environment*.

Memory

Be careful when setting an alarm for memory as a resource that can be changed dynamically. Even when the amount of memory used is fixed, if the allocated memory resources are reduced, memory usage will grow, and could cause alarms to be reported unintentionally.

Devices

Disk performance data is only collected for logical partition groups on LPARs. System-wide performance data cannot be collected.

2.3.17 Data that can be collected on virtualized systems that use HP Integrity Virtual Machines (I-VM)

(1) HP Integrity Virtual Machines functions

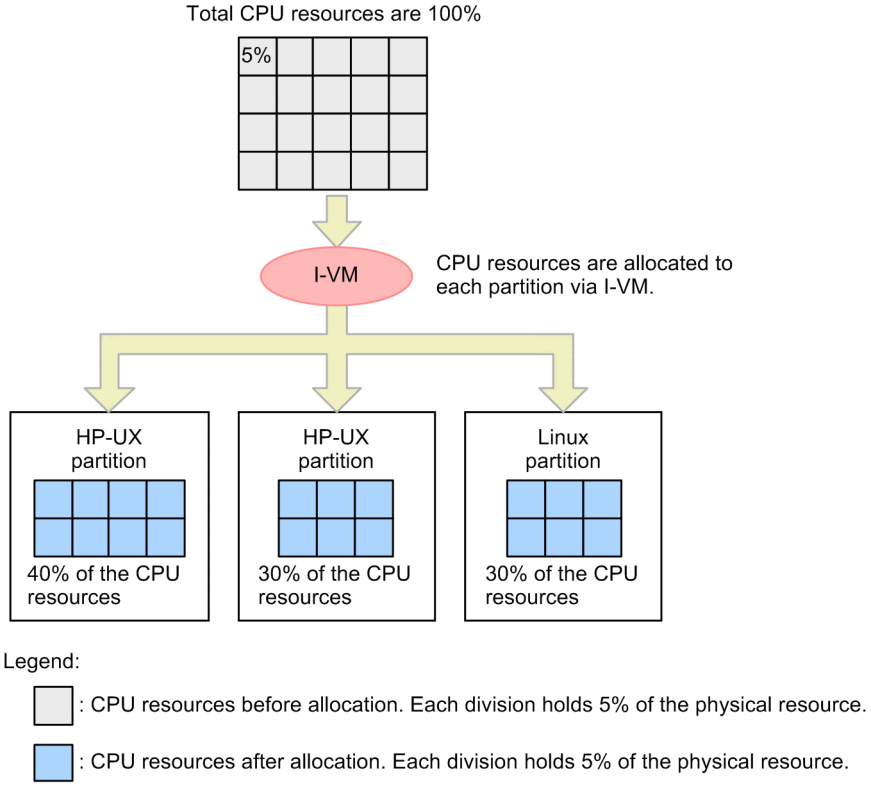
HP Integrity Virtual Machines provides a sub-CPU partitioning capability that dynamically partitions a processor resource into 5% units and then allocates the units to logical partitions. By using this function, you can set a maximum of 20 logical processors for one physical processor. When this function is used, PFM - Agent for Platform collects performance data for the logically partitioned processors.

In addition to processor resources, HP Integrity Virtual Machines supports sharing of I/O by multiple logical partitions and supports memory virtualization. Because HP Integrity Virtual Machines supports Linux as well as HP-UX, Linux compatibility is also provided.

(2) HP Integrity Virtual Machines structure

The processor resource allocation function provided by HP Integrity Virtual Machines is almost the same as the Micro-Partitioning function provided by AIX.

Figure 2–12: HP Integrity Virtual Machines structure



(3) Using PFM - Agent for Platform on HP Integrity Virtual Machines

When using PFM - Agent for Platform on HP Integrity Virtual Machines, note that the information about processors, memory, and devices is changed.

Processor information

A record that indicates processor information displays the same number of instances as the number of virtual processors allocated to the virtual machine.

Because the resources allocated to the virtual processor on HP Integrity Virtual Machines change dynamically, the *amount* of resources that is actually used differs even when the processor usage is the same. For example, 30% of the processor resources might be allocated to processor #1 whose usage is 50%. If the allocation ratio changes to 60%, the usage of processor #1 becomes 25% (in actuality, there is a margin of error due to overhead and the kernel structure). Accordingly, to determine whether the processor usage is low, you should use fields that are unlikely to be affected by processor resource allocation, such as the current queue length.

Memory

Although HP Integrity Virtual Machines supports memory virtualization, virtualized memory cannot be changed dynamically. Accordingly, operation can proceed on a virtual machine the same way as on a non-virtual machine.

Devices

As with memory, HP Integrity Virtual Machines supports I/O virtualization. Disk performance data is only collected for logical partition groups. System-wide performance data cannot be collected.

2.3.18 Data that can be collected with virtualized systems that use VMware/KVM

VMware/KVM is software that provides virtualized systems on the Intel architecture. Because it provides virtualized systems implemented in software, it is a general purpose implementation that can run flexibly on several environments.

(1) VMware/KVM functionality

Like other virtualized systems, VMware/KVM provides the following functionality:

- Virtual networks
- Virtual processors
- Virtual memory
- Virtual disks

VMware/KVM manages combinations of the above functionality to create a virtual machine (VM). The created VM is recognized as a single host from other hosts, and can be treated the same as a physical machine.

Because multiple VMs can be created and run, operating systems like Linux and Windows can run concurrently.

(2) VMware/KVM configuration

VMware/KVM implements a virtualized system by creating a distinction between a host OS and a guest OS.

To set up a virtualized system:

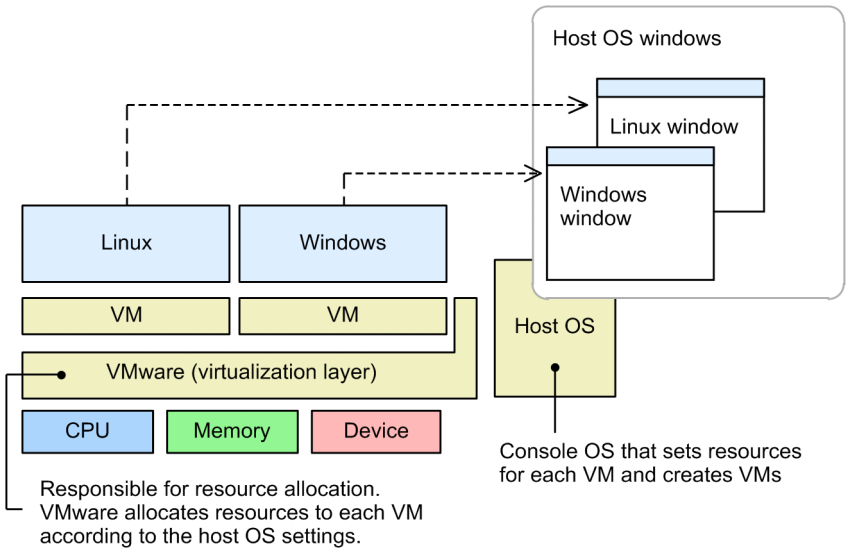
1. Install Windows, Linux, or another OS on the actual hardware.
2. Install VMware/KVM on the installed host OS.
   Here, the OS on which VMware/KVM is installed is called the *host OS*.
3. Use VMware/KVM to create a VM.
   Install other operating systems on the created VM. An operating system installed on the VM is called a *guest OS*.

As shown above, VMware/KVM uses a parent/child configuration to implement a virtualized system, in which the guest OS is executed on the host OS.

One feature of VMware is that a specialized custom OS is used as the host OS instead of Windows or Linux. In other words, VMware implements a virtualized system in which Windows or Linux is running on a specialized custom OS.

Depending on the type of VMware/KVM, the host OS can also be Windows or Linux instead of the custom OS.

Figure 2–13: VMware configuration



(3) Using PFM - Agent for Platform on VMware/KVM

Keep the following in mind when using PFM - Agent for Platform on VMware/KVM.

Installing PFM - Agent for Platform

To use PFM - Agent for Platform, install it on the guest OS. Because the host OS for VMware is a custom OS optimized to implement virtualized systems with VMware, operation of applications is not guaranteed.

Processor information

In a virtualized system, if sufficient resources are not allocated to the host OS, the impact will extend to the operation of the host OS. In particular, the impact is significant on processor resources. If overload occurs, it might be difficult to determine whether the processor resources allocated to the VM are insufficient, or whether the host OS processor resources are insufficient.

VMware/KVM also has a feature that allows upper and lower limits to be set for processor usage allocation, which can cause wide swings in usage because resource switching is automatically performed within the set range.

If this seems to be the case, monitor items that are not easily impacted by dynamic changes to processor resources, such as the current queue length.

Memory

Be careful when setting an alarm for memory as a resource that can be changed dynamically. Even when the amount of used memory is fixed, if the allocated memory resources are reduced, memory usage will grow, and could cause alarms to be reported unintentionally.

Devices

Disk performance data is collected only for logical partition groups on the guest OS. System-wide performance data including the host OS cannot be collected. For the network adapter information, information about virtual network adapters allocated to the guest OS is obtained.

2.3.19 Data that can be collected with virtualization systems that use Hitachi Compute Blade logical partitioning feature

(1) Hitachi Compute Blade logical partitioning feature functionality

Like other virtualized systems, Hitachi Compute Blade logical partitioning feature provides the following functionality:

- Virtual networks
- Virtual processors
- Virtual memory

Hitachi Compute Blade logical partitioning feature manages the above functionality to implement a virtualized system. The logical partition must be stopped in order to change the virtual network or virtual memory, or to change the number of assigned virtual processors. The logical partition does not need to be stopped to change the virtual processor assignment percentages.

(2) Hitachi Compute Blade logical partitioning feature configuration

Like VMware, Hitachi Compute Blade logical partitioning feature has a parent/child configuration.

To set up a virtualized system on Hitachi Compute Blade logical partitioning feature:

1. Start the installed instance of Hitachi Compute Blade logical partitioning feature.
2. Create and set up a virtual machine.
3. Start the virtual machine.

(3) Using PFM - Agent for Platform on Hitachi Compute Blade logical partitioning feature

Keep the following in mind when using PFM - Agent for Platform on Hitachi Compute Blade logical partitioning feature.

Installing PFM - Agent for Platform

Because Hitachi Compute Blade logical partitioning feature is configured in a special environment called an *SVP frame*, do not install PFM - Agent for Platform on an SVP frame.

Processor information

Because a virtualized system based on Hitachi Compute Blade logical partitioning feature runs after the logical partition is set up, the number of processors is not changed while PFM - Agent for Platform is running. Accordingly, operation can be performed as usual, but the number of allocated processors may not match the number of processors that physically exist.

Memory

Because a virtualized system based on Hitachi Compute Blade logical partitioning feature runs after the logical partition is set up, the amount of memory is not changed while PFM - Agent for Platform is running. Accordingly, operation can be performed as usual.

Devices

Disk performance data is collected only for logical partition groups. System-wide performance data including the host OS cannot be collected. For the network adapter information, information about virtual network adapters allocated to the logical partition is obtained.

2.3.20 Data that can be collected with virtualized systems that use WPAR

(1) WPAR function

WPAR included in AIX provides a function that runs multiple AIX instances, which are virtualized as software, in a single LPAR. There are two types of WPAR environments, system WPAR and application WPAR. The global environment required for configuring WPAR is also provided. The following describes these environments.

- Global environment
  This environment has a view that includes all processes, IPC, file systems, devices, and other user-level objects and system-level objects.
- System WPAR
  This is a virtual system environment that has a unique file system, users and groups, resource control, login, and network.
- Application WPAR
  This environment is used for separating applications from their resources.

(2) WPAR structure

The structure of WPAR is similar to that of Solaris Zone, which provides workspace for applications.

In the global environment, you can display or interact with the processes, file systems, and other system components allocated on WPAR running in the system. A new instance of WPAR can be created only in the global environment. Another instance of WPAR cannot be created in the WPAR environment. Most management tasks can be executed only in the global environment. Most commands behave differently depending on whether they are executed in the global environment or in the WPAR environment.

PFM - Agent for Platform can collect performance data in the following cases:

- PFM - Agent for Platform is running in the global environment that uses the WPAR function.
- PFM - Agent for Platform is running in the system WPAR environment.

Note

PFM - Agent for Platform installed in the global environment can monitor performance of the entire WPAR environment including the application WPAR environment.

(3) Monitoring processes in the global environment

When PFM - Agent for Platform is running in the global environment, you can set the `ALL WPAR Collection for Process` property to select the environment for which process information is collected. You can set either of the following values for the `ALL WPAR Collection for Process` property.

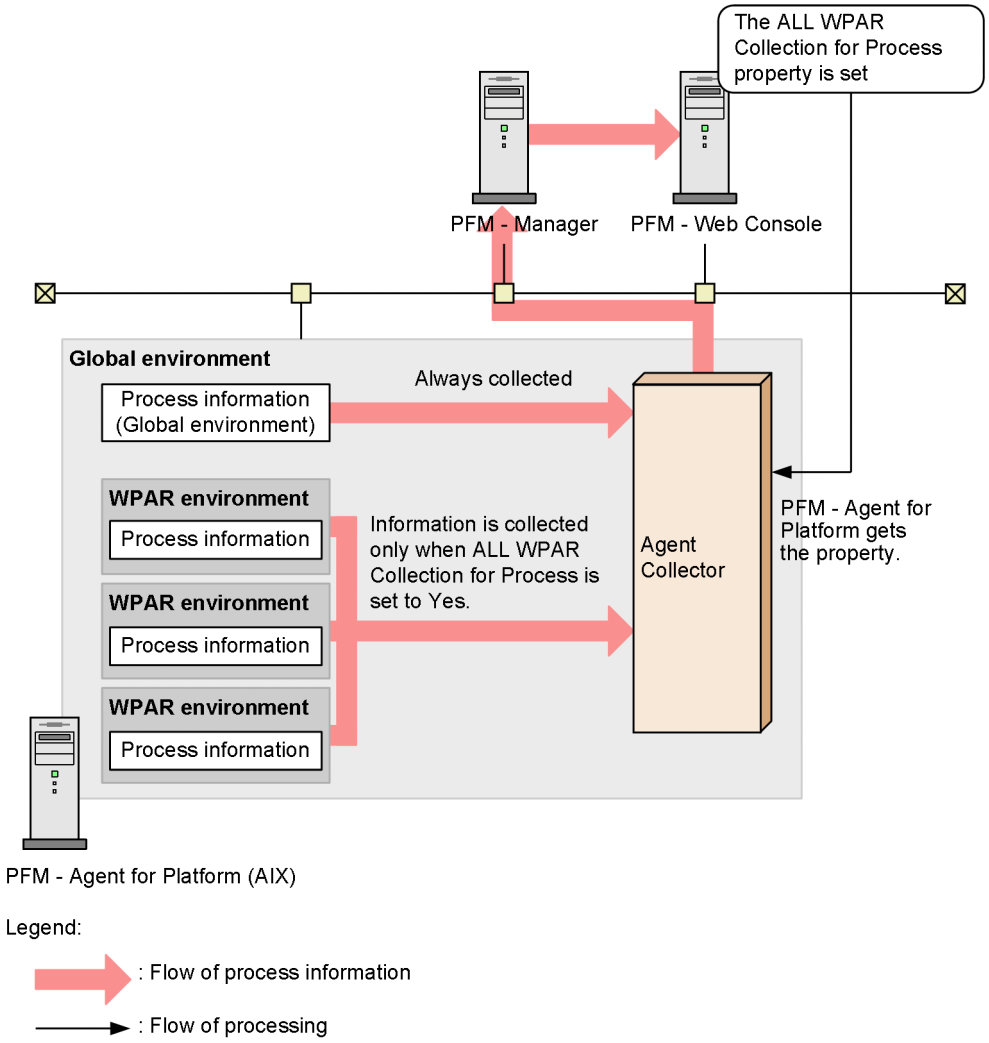
`Yes`
: Collect process information for the global environment and all WPAR environments.

`No`
: Collect process information for the global environment.

The `ALL WPAR Collection for Process` property is set to `Yes` by default.

The following figure shows the process information that is collected when PFM - Agent for Platform is running in the global environment.

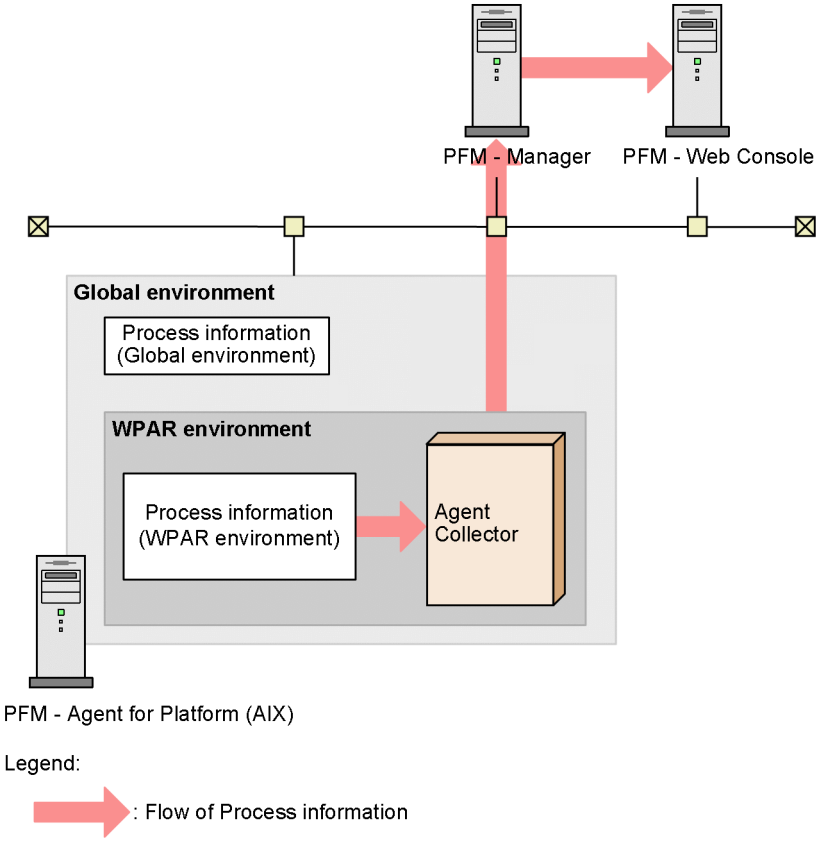
Figure 2–14: Process information collected for the global environment



(4) Monitoring processes in a system WPAR environment

When PFM - Agent for Platform is running in a system WPAR environment, information only about that system WPAR environment is collected regardless of the setting of the `ALL WPAR Collection for Process` property. The following figure shows the process information that is collected when PFM - Agent for Platform is running in the system WPAR environment.

Figure 2–15: Process information collected for the system WPAR environment



(5) Using PFM - Agent for Platform on WPAR

When using PFM - Agent for Platform on WPAR, note that the information about processors and memory is dynamically changed.

Installing PFM - Agent for Platform

PFM - Agent for Platform can be installed in the global environment or system WPAR environment. Note, however, that when you install PFM - Agent for Platform in the system WPAR environment, the following information cannot be obtained because a separate application environment is provided due to the characteristics of WPAR.

- Information about the WPAR environments other than the system WPAR environment in which PFM - Agent for Platform is installed
- Some device information, such as `/dev/mem` and `/dev/kmem`
- Some network information related to the NFS Server

When you configure a system WPAR environment after PFM - Agent for Platform is installed in the global environment, the files related to PFM - Agent for Platform are copied to the system WPAR environment. However, this does not affect PFM - Agent for Platform in the global environment. Note that the copy of the PFM - Agent for Platform instance in the system WPAR environment cannot be used as is. To use PFM - Agent for Platform in the system WPAR environment, you need to manually delete the files related to PFM - Agent for Platform (all files under the `/opt/jp1pc` directory) in the system WPAR environment. Then perform a new installation of PFM - Agent for Platform.

Processor information

Because a virtualized system based on WPAR runs according to the allocation ratio of the processor, the number of processors is not changed while PFM - Agent for Platform is running. However, because the processor allocation

ratio can be dynamically changed while the system is running, information such as the processor usage for applications might be affected.

Memory

Be careful when setting an alarm for memory as a resource that can be changed dynamically. Even when the amount of memory used is fixed, if the allocated memory resources are reduced, memory usage will grow, and could cause alarms to be reported unintentionally.

Devices

When monitoring devices on WPAR, information for only logical partition groups on WPAR can be collected. To collect information for physical partition groups, install PFM - Agent for Platform in the global environment and monitor the devices in the global environment.

2.3.21 Data that can be collected on a virtualized system based on the Linux container

(1) Overview of a Linux container

A Linux container allows you to operate multiple Linux environments within a single Linux system by virtualizing the Linux environments and treating them like software. Because a Container shares resources with the host OS, the overhead of running applications is smaller compared to that incurred with virtual machines.

(2) Structure of a Linux container

A Linux container resembles a Solaris zone (which provides a workspace for applications) in structure.

The Linux container consists of a host environment that manages it and container environments that run on a host and provide separate spaces for applications.

PFM - Agent for Platform allows you to collect information on the host environment and container environments.

PFM - Agent for Platform supports Docker and Podman as a container engine. The Docker and Podman host environments supported by PFM - Agent for Platform are as follows:

Docker host OS version

- Red Hat(R) Enterprise Linux(R) Server 7.1 or later
- CentOS 7.1 or later

Podman host OS version

- Red Hat(R) Enterprise Linux(R) Server 8.1 or later
- CentOS 8.1 or later

The rest of this manual provides the descriptions common to Docker and Podman as those of the Linux container.

(3) Process monitoring in the host environment

If you are running PFM - Agent for Platform in the host environment, you can collect information on processes running inside the containers.

By setting the `ALL Container Collection for Process` property, you can select whether to collect information about processes running inside containers. The following settings are available for the `ALL Container Collection for Process` property:

`Yes`

If you select this setting, information# about processes running inside both the host environment and container environments is collected.

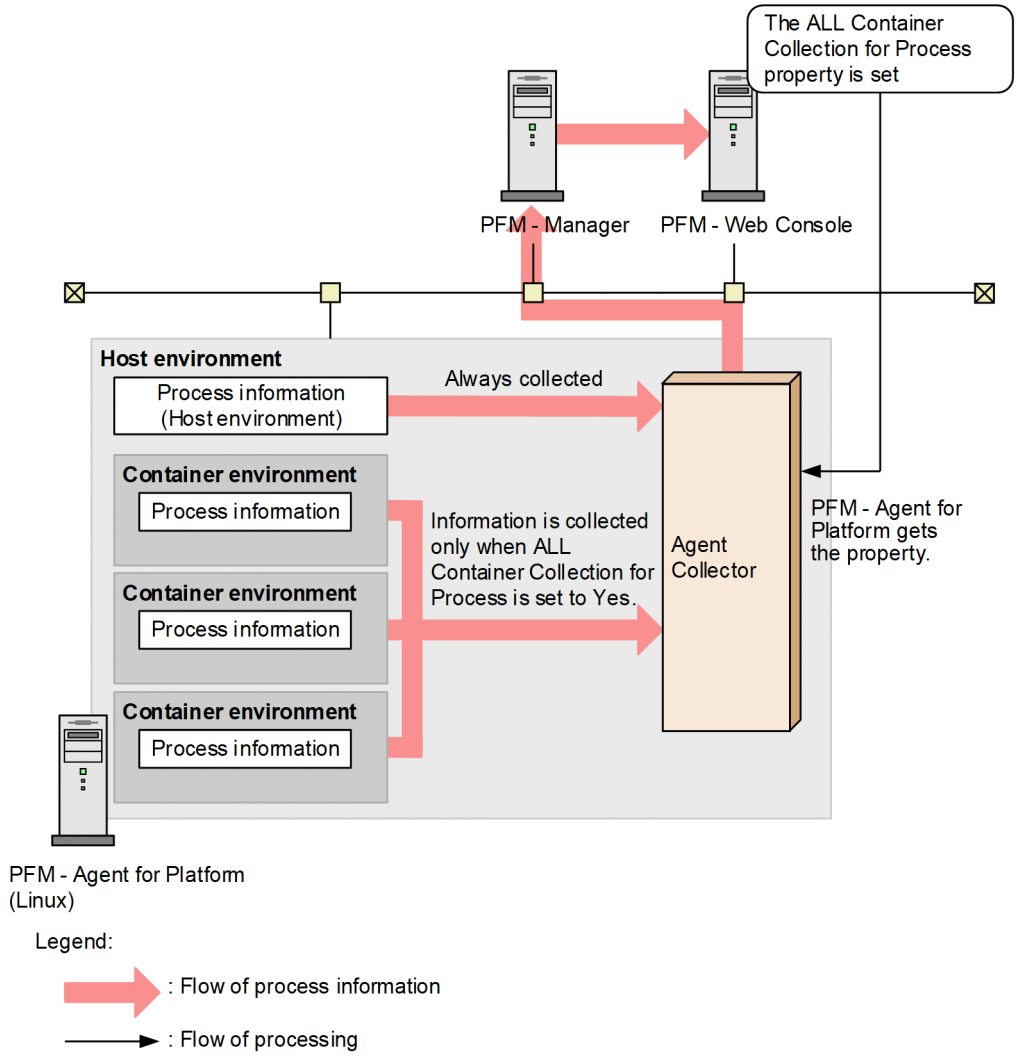
`No`

If you select this setting, only information about processes running inside the host environment is collected.

The `ALL Container Collection for Process` property is set to `Yes` by default.

The figure below shows process information collected when PFM - Agent for Platform runs on the host environment.

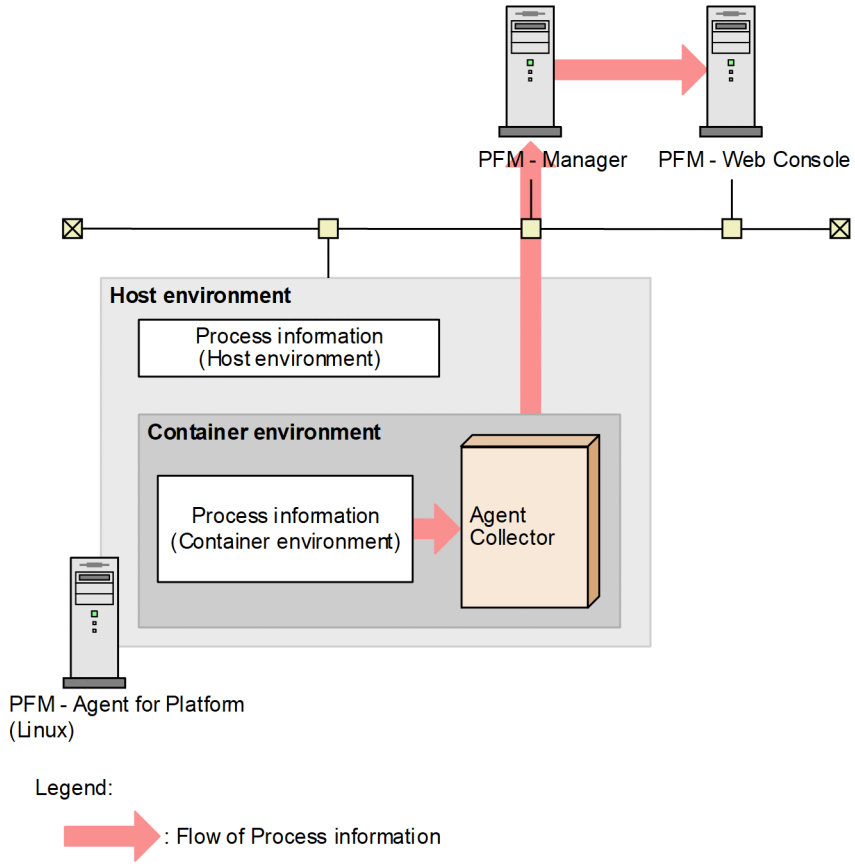
Figure 2–16: Process information collected on the host environment



(4) Process monitoring in the container environment

When PFM - Agent for Platform is running in a container environment, PFM - Agent for Platform only collects process information about the container environment in which it is running, regardless of the setting specified for the `ALL Container Collection for Process` property. The figure below shows process information collected when PFM - Agent for Platform runs on the container environment.

Figure 2–17: Process information collected on the container environment



(5) Notes on the host environment

For notes about operations in a container environment, see the document *JP1/Performance Management Notes on container environment on Linux and Windows* on the JP1 website.

(a) Notes on the Docker host environment

When you have installed PFM - Agent for Platform on the Docker host environment to collect information about processes running on it, use the Docker Engine API to collect information about processes running on the Docker container.

The notes on the Docker Engine API are provided here.

- By using the Docker Engine API, PFM - Agent for Platform communicates with the Docker Engine according to the default setting specified for the UNIX domain socket. For this reason, the UNIX domain socket settings of the Docker Engine API must not be edited. If you change the default UNIX domain socket settings, PFM - Agent for Platform can no longer communication with the Docker Engine, in which case `N/A` is stored in the `Virtual Env ID` field of all processes obtained with the `PD` record, `PD_APSI` record, and `PD_APS` record.
- If one or more errors occur while PFM - Agent for Platform is communicating with the Docker Engine to collect record information, or if a Docker container is started or shut down while PFM - Agent for Platform is communicating with the Docker Engine, PFM - Agent for Platform might not be able determine whether some of the processes are running on the Docker host or a Docker container. In this case, `N/A` is stored in the `Virtual Env ID` field of the `PD` record, `PD_APSI` record, and `PD_APS` record of these processes.

(b) Notes on the Podman host environment

When you use PFM - Agent for Platform installed in your Podman host environment to collect information about processes, use the `podman` command in PFM - Agent for Platform to collect information about processes running on the Podman container.

The notes on the `podman` command are provided here.

- If the `podman` command terminates in error or the processing is not completed during a given time period while PFM - Agent for Platform is collecting record information, PFM - Agent for Platform might not be able to determine whether the processes are running in either the Podman host environment or the Podman container environment. In this case, `N/A` is stored in the `Virtual Env ID` fields of the `PD` record, `PD_APSI` record, and `PD_APS` record of these processes.
- PFM - Agent for Platform executes the command below to get the list of containers. Only the containers with the `Up` status will be included in the list. Processes running on containers that are not in the `Up` status are handled as those in the host environment.

  ```
  podman ps --all --filter status=running
  ```

- If the `podman top` command cannot get the `HPID` item of processes that are running on the Podman container, PFM - Agent for Platform cannot determine whether they are running in either the Podman host environment or the Podman container environment. In this case, `N/A` is always stored in the `Virtual Env ID` fields of the `PD` record, `PD_APSI` record, and `PD_APS` record of these processes with the same command line.

  In addition, if the `podman top` command terminates in error and thus PFM - Agent for Platform cannot determine whether they are running in either the Podman host environment or the Podman container environment, `N/A` is stored in the `Virtual Env ID` fields of the `PD` record, `PD_APSI` record, and `PD_APS` record of these processes.
- PFM-Agent for Platform can collect information about processes running on a rootless container (which is started by a user other than the root user), but handles them as those in the host environment.
- As a general guide, PFM - Agent for Platform should monitor less than 30 Podman containers and about 10 processes on each Podman container.
- If you stop the Podman container while PFM - Agent for Platform is collecting process information, `N/A` might be stored in the `Virtual Env ID` fields of processes on the Podman host environment and the Podman container that has stopped.
- In Linux 8 or CentOS 8, if the `ALL Container Collection for Process` property is set to `No` when you upgrade PFM - Agent for Platform from version 12-00 to version 12-50 or later, process information of the Podman container environment cannot be collected as described below. If you still want to collect the process information of the Podman container environment in PFM - Agent for Platform version 12-50 or later as well, the `ALL Container Collection for Process` property must be set to `Yes`.

  What process information is collected when the `ALL Container Collection for Process` property is set to `No`

  Before upgrade installation

  Process information of the Podman host environment and all Podman container environments are collected.

  After upgrade installation

  Only process information of the Podman host environment is collected.
- In Linux 8 or CentOS 8, when you perform upgrade installation of PFM - Agent for Platform from version 12-00 to 12-50 or later, different process information will come to be collected as described below. If you want to collect that process information on PFM - Agent for Platform version 12-50 or later, you need to review your process monitoring settings.

  What process information is collected

Before upgrade installation

`N/A` is stored in the `Virtual Env ID` fields of the `PD` record, `PD_APSI` record, and `PD_APS` record.

After upgrade installation

`0` or a container ID is stored in the `Virtual Env ID` fields of the `PD` record, `PD_APSI` record, and `PD_APS` record.

(6) Notes on the container environment

Keep the following in mind when you have installed PFM - Agent for Platform on the container environment to collect performance information on it:

- For notes regarding the installation of PFM - Agent for Platform in a container environment, see the *Release Notes*.
- Owing to the characteristics of the container environment, a separate application environment is provided. Due to this, the following information cannot be obtained:
  - Information about container environments other than that in which PFM - Agent for Platform has been installed
- If you want to connect to a PFM - Manager in another host environment (that is, a host that is different from the one that manages the container environment in which PFM - Agent for Platform is installed), you need to first fix the IP address of the container environment by means of port forwarding, and then enable another host environment to access PFM - Agent for Platform.
- The ID of each process running in the container environment varies between the host environment and container environments.

(a) Note on the Podman container environment

PFM - Agent for Platform cannot be installed on a rootless container.

2.3.22 Range of data collected for each virtualized system

- Global zone, non-global zone, and PowerVM, HP Integrity Virtual Machines, VMware/KVM, and Hitachi Compute Blade logical partitioning feature environments

The following table describes the data collected by PFM - Agent for Platform running in a virtualized system in the global zone, non-global zone, PowerVM, HP Integrity Virtual Machines, VMware/KVM, and Hitachi Compute Blade logical partitioning feature environments. For example, when you collect data of the PD record in an environment where PFM - Agent for Platform is running in the shared-IP non-global zone, the Shared-IP Non-Global Zone column indicates *This zone only*. This means that the data only for the shared-IP non-global zone will be collected. When you collect data of the PI record in this environment, the Shared-IP Non-Global Zone column indicates *Entire system*, which means that the data for the global zone and non-global zones will be collected.

Table 2–3: Data collected for each virtualized system (global zone, non-global zone, PowerVM, HP Integrity Virtual Machines, VMware/KVM, and Hitachi Compute Blade logical partitioning feature environments)

Record	Global Zone[#1]	Non-Global Zone		PowerVM and HP Integrity Virtual Machines	VMware/KVM and Hitachi Compute Blade logical partitioning feature
		Shared-IP Non-Global Zone	Exclusive-IP Non-Global Zone		
`PD`	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS

Record	Global Zone#1	Non-Global Zone		PowerVM and HP Integrity Virtual Machines	VMware/KVM and Hitachi Compute Blade logical partitioning feature
		Shared-IP Non-Global Zone	Exclusive-IP Non-Global Zone		
PD_APP	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_APP2	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_APPD	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_APS	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_APSI	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_FSL	Entire system	This zone only	This zone only	Local disk information on the logical partition	Local disk information in the guest OS
PD_FSR	Information is collected only for the global zone. Information about a remote file system (NFS) installed in the non-global zone cannot be obtained.	This zone only	This zone only	Remote file system information registered on the logical partition	Remote file system information registered in the guest OS
PD_PDI	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_PDS	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_PGM	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_TERM	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PD_UPD	Specified value	Specified value	Specified value	Performance data that the user specified on the logical partition	Performance data that the user specified in the guest OS
PD_UPDB	Specified value	Specified value	Specified value	Performance data that the user specified on the logical partition	Performance data that the user specified in the guest OS
PD_USER	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PI#2	Entire system	Entire system	Entire system	System values on the logical partition	System values in the guest OS
PI_CPUP	Entire system	Entire system	Entire system	Processor information allocated to the logical partition	Processor information allocated to the guest OS
PI_DEVD	Entire system	Entire system	Entire system	Disk information allocated to the logical partition	Disk information allocated to the guest OS

Record	Global Zone#1	Non-Global Zone		PowerVM and HP Integrity Virtual Machines	VMware/KVM and Hitachi Compute Blade logical partitioning feature
		Shared-IP Non-Global Zone	Exclusive-IP Non-Global Zone		
PI_DEVS	Entire system	Entire system	Entire system	Disk information allocated to the logical partition	Disk information allocated to the guest OS
PI_NIND#3	Interface information shared by the global zone and all shared-IP non-global zones	--	This zone only	NIC information allocated to the logical partition	NIC information allocated to the guest OS
PI_NINS#3	Interface information shared by the global zone and all shared-IP non-global zones	--	This zone only	NIC information allocated to the logical partition	NIC information allocated to the guest OS
PI_UPI	Specified value	Specified value	Specified value	Performance data that the user specified on the logical partition	Performance data that the user specified in the guest OS
PI_UPIB	Specified value	Specified value	Specified value	Performance data that the user specified on the logical partition	Performance data that the user specified in the guest OS
PI_WGRP	Entire system	This zone only	This zone only	Processes on the logical partition	Processes on the guest OS
PI_XUI1 to PI_XUI5	Specified value	Specified value	Specified value	Performance data that the user specified on the logical partition	Performance data that the user specified in the guest OS
PL_MESS	Specified value	Specified value	Specified value	The file specified in evfile or the Messages File property	The file specified in evfile or the Messages File property

Legend:

--: Not collected

#1

The values of items marked with "Entire system" might differ from the actual values of the entire system.

Therefore, for the "Non - Global Zone" items in the table, use only "This zone only" records.

#2

For the following fields, interface information is collected even in the non-global zone environment:

ICMP Pkts In, ICMP Pkts Out, ICMP6 Pkts In, ICMP6 Pkts Out, IP Pkts In, IP Pkts Out, IP6 Pkts In, IP6 Pkts Out, TCP Pkts In, TCP Pkts Out, Total Pkts, Total Pkts In, Total Pkts Out, UDP Pkts In, UDP Pkts Out

For the following fields, information is not collected in the non-global zone environment:

NFS Server Lookup Ops, NFS Server Ops/sec, NFS Server Read Ops, NFS Server Read Ops/sec, NFS Server Total Bad Ops, NFS Server Total Ops, NFS Server Write Ops, NFS Server Write Ops/sec, Cache Mem %, Cache Mem Mbytes, Effective Free Mem %, Effective Free Mem Mbytes

Also, the correct value is not collected in the non-global zone environment for the following fields:

Effective Used Mem %, Effective Used Mem Mbytes

(For Effective Used Mem %, 100.0000 is collected, and for Effective Used Mem Mbytes, the same value as Total Physical Mem Mbytes is collected.)

#3

Physical interface information is collected. Because the shared-IP non-global zone is a logical zone, no information is collected.

- WPAR environment

The following table describes the data collected by PFM - Agent for Platform running in a WPAR environment. For example, when you collect data of the PD record in an environment where PFM - Agent for Platform is running in a system WPAR environment, the System WPAR column indicates *System WPAR environment*. This means that the data only for the system WPAR environment is collected. When you collect data of the PI record in this environment, the System WPAR column indicates *Global environment and system WPAR environment*, which means that the data for the global environment and system WPAR environments will be collected.

Table 2–4: Data collected for each virtualized system (WPAR environment)

Record	Field	Global environment	System WPAR
PD	All	Global environment	System WPAR environment
PD_APP	All	Global environment	System WPAR environment
PD_APP2	All	Global environment	System WPAR environment
PD_APPD	All	Global environment	System WPAR environment
PD_APS	All	Global environment	System WPAR environment
PD_APSI	All	Global environment	System WPAR environment
PD_FSL	All	Global environment	System WPAR environment
PD_FSR	All	Global environment	System WPAR environment
PI#3	1-Minute Run Queue Avg, 15-Minute Run Queue Avg, 5-Minute Run Queue Avg, Active CPUs, Alloc Mem %, Alloc Mem Mbytes, Alloc Swap %, Alloc Swap Mbytes, Block Ops, Block Reads, Block Writes, Cache Read %, Cache Write %, Context Switches, CPU %, Effective Free Mem %, Effective Free Mem Mbytes, Effective Used Mem %, Effective Used Mem Mbytes, Faults, Free Mem %, Free Mem Mbytes, Free Swap %, Free Swap Mbytes,	Global environment and all WPAR environments	Global environment and system WPAR environment

Record	Field	Global environment	System WPAR
PI#	Idle %, Kernel CPU %, Logical I/O Ops, Logical Read Mbytes, Logical Reads, Logical Write Mbytes, Logical Writes, Major Faults, Pages In, Pages Out, Physical I/O Ops, Physical Reads, Physical Writes, Run Queue, Swapped-In Pages, Swapped-Out Pages, System Calls, Total Idle Time, Total Kernel-Mode Time, Total Page Scans, Total Physical Mem Mbytes, Total Swap Mbytes, Total User-Mode Time, Total Wait Time, User CPU %, Wait %	Global environment and all WPAR environments	Global environment and system WPAR environment
	ICMP Pkts In, ICMP Pkts Out, ICMP6 Pkts In, ICMP6 Pkts Out, IP Pkts In, IP Pkts Out, IP6 Pkts In, IP6 Pkts Out, NFS Client Lookup Ops, NFS Client Ops/sec, NFS Client Read Ops, NFS Client Read Ops/sec, NFS Client Total Bad Ops, NFS Client Total Ops, NFS Client Write Ops, NFS Client Write Ops/sec, NFS Server Lookup Ops, NFS Server Ops/sec, NFS Server Read Ops, NFS Server Read Ops/sec, NFS Server Total Bad Ops, NFS Server Total Ops, NFS Server Write Ops,	Global environment and all WPAR environments	System WPAR environment

Record	Field	Global environment	System WPAR
PI#	NFS Server Write Ops/sec, Processes, Processes Ended, Processes Started, TCP Pkts In, TCP Pkts Out, Total Pkts, Total Pkts In, Total Pkts Out, UDP Pkts In, UDP Pkts Out, Users	Global environment and all WPAR environments	System WPAR environment
	Block Reads/sec, Block Writes/sec, Context Switches/sec, Logical Reads/sec, Logical Writes/sec, Major Faults/sec, Page Scans/sec, Pages In/sec, Pages Out/sec, Swapped-In Pages/sec, Swapped-Out Pages/sec, System Calls/sec, Total Faults/sec	Global environment and all WPAR environments	--
	Boot Time, Interval, Record Time, System Up Time	Global environment	System WPAR environment
	Record Type	Fixed value	Fixed value
	Interrupts, Interrupts/sec, Mem I/O Ops, Minor Faults, Minor Faults/sec, Other Pkts In, Other Pkts Out, Page Reclaims/sec, Page-In Ops, Page-In Ops/sec, Page-Out Ops, Page-Out Ops/sec, Page Ops/sec, Software Lock Faults, Software Lock Faults/sec, Swap-In Ops, Swap-Ins/sec, Swap-Out Ops,	--	--

Record	Field	Global environment	System WPAR
PI#	Swap-Outs/sec, Total Page Ops, Total Page Reclaims, Total Swaps, Total Swaps/sec, Traps, Traps/sec	--	--
PI_CPUP	CPU %, Context Switches, Idle %, Idle Time, System %, System Calls, System Time, User %, User Time, Wait %, Wait Time	Global environment and all WPAR environments	Global environment and system WPAR environment
	Context Switches/sec, Sys Calls/sec	Global environment and all WPAR environments	--
	Boot Time, Interval, Record Time, Up Time	Global environment	System WPAR environment
	Processor ID, Status, Type	Global environment	Global environment
	Record Type	Fixed value	Fixed value
	Interrupts, Interrupts/sec, Traps, Traps/sec	--	--
PI_DEVD	All	Global environment	--
PI_DEVS	All	Global environment	--
PI_NIND	All	Global environment	System WPAR environment
PL_MESS	All	Global environment	System WPAR environment

Legend:

--: Not collected

#

For the following fields, information is not collected in the system WPAR environment:

NFS Server Lookup Ops, NFS Server Ops/sec, NFS Server Read Ops, NFS Server Read Ops/sec, NFS Server Total Bad Ops, NFS Server Total Ops, NFS Server Write Ops, NFS Server Write Ops/sec

- Linux container environment

The following table describes the data collected by PFM - Agent for Platform running in a Linux container environment. For example, suppose that `PD` record data is to be collected in a container environment in which PFM - Agent for Platform is running. In this case, only data from the container environment is collected, as described in the table below (see the cell under the *container environment* column in the *PD* row). Similarly, if `PI` record data is to be collected in the same environment, data from both the host environment and all container environments is collected (see the cell under the *container environment* column in the *PI* row).

Table 2–5: Data collected for each virtualized system (Linux container environment)

Record	Field	Host environment	Container environment
PD	All	Host environment and all container environments#	Container environment
PD_APP	All	Host environment and all container environments#	Container environment
PD_APP2	All	Host environment and all container environments#	Container environment
PD_APPD	All	Host environment and all container environments#	Container environment
PD_APS	All	Host environment and all container environments#	Container environment
PD_APSI	All	Host environment and all container environments#	Container environment
PD_FSL	All	Host environment	Container environment
PD_FSR	All	Host environment	Container environment
PD_PDI	All	Host environment and all container environments#	Container environment
PD_PDS	All	Host environment and all container environments#	Container environment
PD_PGM	All	Host environment and all container environments#	Container environment
PD_TERM	All	Host environment and all container environments#	Container environment
PD_UPD	All	Specified information	Specified information
PD_UPDB	All	Specified information	Specified information
PD_USER	All	Host environment and all container environments#	Container environment
PI	1-Minute Run Queue Avg, 15-Minute Run Queue Avg, 5-Minute Run Queue Avg, Active CPUs, Alloc Mem %, Alloc Mem Mbytes, Alloc Swap %, Alloc Swap Mbytes,	Host environment and all container environments	Host environment and all container environments

Record	Field	Host environment	Container environment
PI	Boot Time, Buffers Mem %, Buffers Mem Mbytes, Cache Mem %, Cache Mem Mbytes, Context Switches, Context Switches/sec, CPU %, Effective Free Mem %, Effective Free Mem Mbytes, Effective Used Mem %, Effective Used Mem Mbytes, Free Mem %, Free Mem Mbytes, Free Swap %, Free Swap Mbytes, Idle %, Interrupts, Interrupts/sec, Kernel CPU %, Pages In, Pages In/sec, Pages Out, Pages Out/sec, Swapped-In Pages, Swapped-In Pages/sec, Swapped-Out Pages, Swapped-Out Pages/sec, System Up Time, Total Idle Time, Total Kernel-Mode Time, Total Physical Mem Mbytes, Total Swap Mbytes, Total User-Mode Time, Total Wait Time, User CPU %, Wait %	Host environment and all container environments	Host environment and all container environments
	ICMP Pkts In, ICMP Pkts Out, ICMP6 Pkts In, ICMP6 Pkts Out, IP Pkts In, IP Pkts Out, IP6 Pkts In, IP6 Pkts Out, Processes, Processes Ended, Processes Started, TCP Pkts In,	Host environment and all container environments	Container environment

Record	Field	Host environment	Container environment
PI	TCP Pkts Out, Total Pkts, Total Pkts In, Total Pkts Out, UDP Pkts In, UDP Pkts Out, Users	Host environment and all container environments	Container environment
	Interval, Record Time	Host environment	Container environment
	Record Type	Fixed value	Fixed value
PI_CPUP	All	Host environment and all container environments	Host environment and all container environments
PI_DEVD	All	Host environment	Host environment
PI_DEVS	All	Host environment	Host environment
PI_NIND	All	Host environment	Container environment
PI_NINS	All	Host environment	Container environment
PI_UPI	All	Specified information	Specified information
PI_UPIB	All	Specified information	Specified information
PI_WGRP	All	Host environment and all container environments#	Container environment
PI_XUI1 to PI_XUI5	All	Specified information	Specified information
PL_MESS	All	--	--

Legend:

--: Not collected

#

If the ALL Container Collection for Process property is set to No, PFM - Agent for Platform does not collect data from container environments.

(1) Identifying processes that have the same name in a virtualized system

In a virtualized system, a host might contain processes that have the same name. The following describes how to identify such processes in the virtualized system.

(a) Zone environment

If processes that have the same name exist in the global and non-global zones, check the value of the Virtual Env ID field to determine the zone from which the process information for the PD, PD_APP, PD_APP2, PD_APSI and PD_APS records was collected.

- PD record

For process information obtained by the PD record, the following table describes the relationship between the property setting and the obtained information.

Table 2–6: When PFM - Agent for Platform is installed in the global zone in an environment in which zones have been created

ALL ZONE Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global zone and all non-global zones	ID of each zone#
No	Global zone	0

#

When process information is collected from the global zone, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected from a non-global zone, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to that non-global zone.

Table 2–7: When PFM - Agent for Platform is installed in a non-global zone in an environment in which zones have been created

ALL ZONE Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Non-global zone where PFM - Agent for Platform is installed	Non-global zone ID
No	Non-global zone where PFM - Agent for Platform is installed	Non-global zone ID

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no zones have been created.

Table 2–8: When PFM - Agent for Platform is installed in the global zone in an environment in which no zones have been created

ALL ZONE Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global zone	0
No	Global zone	0

- PD_APP record

For process information obtained by the PD_APP record, the following table describes the relationship between the property setting and the obtained information. Note that for the PD_APP record, you can set the ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT property to restrict the environment from which process information is to be collected.

Table 2–9: When PFM - Agent for Platform is installed in the global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global zone and all non-global zones	Blank
Yes	0	Global zone	0
Yes	ID of the existing non-global zone	Specified non-global zone environment	ID of the specified non-global zone
Yes	ID of the nonexistent non-global zone	--	--
No	Blank	Global zone	Blank
No	0	Global zone	0
No	ID of the existing non-global zone	--	--
No	ID of the nonexistent non-global zone	--	--

Legend:

--: Neither collected nor displayed in a report. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

Table 2–10: When PFM - Agent for Platform is installed in the non-global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	Non-global zone where PFM - Agent for Platform is installed	Blank
Yes	0	--	--
Yes	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	ID of the non-global zone where PFM - Agent for Platform is installed
Yes	ID of the non-global zone where PFM - Agent for Platform is not installed	--	--
No	Blank	Non-global zone where PFM - Agent for Platform is installed	Blank
No	0	--	--
No	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	ID of the non-global zone where PFM - Agent for Platform is installed
No	ID of the non-global zone where PFM - Agent for Platform is not installed	--	--

Legend:

--: Not collected. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

Reference note:

The following table describes the relationship between the property settings and obtained information in an environment in which no zones have been created.

Table 2–11: When PFM - Agent for Platform is installed in the global zone in an environment in which no zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global zone	Blank
Yes	0	Global zone	0
Yes	Value other than 0	--	--
No	Blank	Global zone	Blank
No	0	Global zone	0
No	Value other than 0	--	--

Legend:

--: Not collected. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

- PD_APPD and PD_APP2 records

For process information obtained by the PD_APPD and PD_APP2 records, the following table describes the relationship between the property setting and the obtained information.

Table 2–12: When PFM - Agent for Platform is installed in the global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Global zone and all non-global zones	Blank
Yes	0	Global zone	0
Yes	ID of the existing non-global zone	Specified non-global zone environment	ID of the specified non-global zone
Yes	ID of the nonexistent non-global zone	--	--
No	Blank	Global zone	Blank
No	0	Global zone	0
No	ID of the existing non-global zone	--	--
No	ID of nonexistent non-global zone	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

Table 2–13: When PFM - Agent for Platform is installed in a non-global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Non-global zone where PFM - Agent for Platform is installed	Blank
Yes	0	--	--
Yes	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	ID of the specified non-global zone
Yes	ID of the non-global zone where PFM - Agent for Platform is not installed	--	--
No	Blank	Non-global zone where PFM - Agent for Platform is installed	Blank
No	0	--	--
No	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	ID of the specified non-global zone
No	ID of the non-global zone where PFM - Agent for Platform is not installed	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no zones have been created.

Table 2–14: When PFM - Agent for Platform is installed in the global zone in an environment in which no zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Global zone	Blank
Yes	0	Global zone	0
Yes	Value other than 0	--	--
No	Blank	Global zone	Blank
No	0	Global zone	0
No	Value other than 0	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

- PD_APSI record

For process information obtained by the PD_APSI record, the following table describes the relationship between the property setting and the obtained information.

Table 2–15: When PFM - Agent for Platform is installed in the global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global zone and all non-global zones	ID of that zone#2
Yes	0	Global zone	0
Yes	ID of the existing non-global zone	Specified non-global zone environment	ID of the specified non-global zone
Yes	ID of the nonexistent non-global zone	--#3	--
No	Blank	Global zone	0
No	0	Global zone	0
No	ID of the existing non-global zone	--#3	--

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
No	ID of the nonexistent non-global zone	--#3	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

When process information is collected from the global zone, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected from a non-global zone, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to that non-global zone.

#3

Not displayed in a report.

Table 2–16: When PFM - Agent for Platform is installed in the non-global zone in an environment in which zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	Non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed
Yes	0	--#2	--
Yes	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed
Yes	ID of the non-global zone where PFM - Agent for Platform is not installed	--#2	--
No	Blank	Non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed
No	0	--#2	--
No	ID of the non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed	Non-global zone where PFM - Agent for Platform is installed
No	ID of the non-global zone where PFM - Agent for Platform is not installed	--#2	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

Not displayed in a report.

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no zones have been created.

Table 2–17: When PFM - Agent for Platform is installed in the global zone in an environment in which no zones have been created

Property settings		Obtained information	
ALL ZONE Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global zone	0
Yes	0	Global zone	0
Yes	Value other than 0	--#2	--
No	Blank	Global zone	0
No	0	Global zone	0
No	Value other than 0	--#2	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

Not displayed in a report.

- PD_APS record

For process information obtained by the PD_APS record, the following table describes the relationship between the property setting and the obtained information.

Table 2–18: When PFM - Agent for Platform is installed in the global zone in an environment in which zones have been created

ALL ZONE Collection for Process setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global zone and all non-global zones	ID of that zone#

ALL ZONE Collection for Process setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
No	Global zone	0

#

When process information is collected from the global zone, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected from a non-global zone, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to that non-global zone.

Table 2–19: When PFM - Agent for Platform is installed in the non-global zone in an environment in which zones have been created

ALL ZONE Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Non-global zone where PFM - Agent for Platform is installed	Non-global zone ID
No	Non-global zone where PFM - Agent for Platform is installed	Non-global zone ID

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no zones have been created.

Table 2–20: When PFM - Agent for Platform is installed in the global zone in an environment in which no zones have been created

ALL ZONE Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global zone	0
No	Global zone	0

(b) WPAR environment

If processes that have the same name exist in the global environment and WPAR environment, check the value of the Virtual Env ID field to determine the environment from which the process information for the PD, PD_APP, PD_APP2, PD_APSI and PD_APS records was collected.

- PD record

For process information obtained by the PD record, the following table describes the relationship between the property setting and the obtained information.

Table 2–21: When PFM - Agent for Platform is installed in the global environment in an environment in which a system WPAR has been created

ALL WPAR Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global environment and all WPAR environments	ID of each WPAR#

ALL WPAR Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
No	Global environment	0

#

When process information is collected from the global environment, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected from a WPAR environment, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to that WPAR environment.

Table 2–22: When PFM - Agent for Platform is installed in a system WPAR environment in an environment in which a system WPAR has been created

ALL WPAR Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	System WPAR environment in which PFM - Agent for Platform is installed	0
No	System WPAR environment in which PFM - Agent for Platform is installed	0

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no system WPAR has been created.

Table 2–23: When PFM - Agent for Platform is installed in the global environment in an environment in which no system WPAR has been created

ALL WPAR Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global environment	0
No	Global environment	0

- PD_APP record

For process information obtained by the PD_APP record, the following table describes the relationship between the property setting and the obtained information. Note that for the PD_APP record, you can set the ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT property to restrict the environment from which process information is collected.

Table 2–24: When PFM - Agent for Platform is installed in the global environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global environment and all WPAR environments	Blank
Yes	0	Global environment	0
Yes	ID of the existing WPAR	Specified WPAR environment	ID of the specified WPAR

Property settings		Obtained information	
ALL WPAR Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	ID of the non-existing WPAR	--	--
No	Blank	Global environment	Blank
No	0	Global environment	0
No	ID of the existing WPAR	--	--
No	ID of the non-existing WPAR	--	--

Legend:

--: Not collected. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

Table 2–25: When PFM - Agent for Platform is installed in a system WPAR environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	System WPAR environment in which PFM - Agent for Platform is installed	Blank
Yes	0	System WPAR environment in which PFM - Agent for Platform is installed	0
Yes	Value other than 0	--	--
No	Blank	System WPAR environment in which PFM - Agent for Platform is installed	Blank
No	0	System WPAR environment in which PFM - Agent for Platform is installed	0
No	Value other than 0	--	--

Legend:

--: Not collected. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

Reference note:

The following table describes the relationship between the property settings and obtained information in an environment in which no system WPAR has been created.

Table 2–26: When PFM - Agent for Platform is installed in a global environment in an environment in which no system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global environment	Blank
Yes	0	Global environment	0
Yes	Value other than 0	--	--
No	Blank	Global environment	Blank
No	0	Global environment	0
No	Value other than 0	--	--

Legend:

--: Not collected. In the Process*XX* Count field, 0 is displayed (where *XX* is a number between 01 and 15).

- PD_APPD and PD_APP2 records

For process information obtained by the PD_APPD and PD_APP2 records, the following table describes the relationship between the property setting and the obtained information.

Table 2–27: When PFM - Agent for Platform is installed in the global environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Global environment and all WPAR environments	Blank
Yes	0	Global environment	0
Yes	ID of the existing WPAR	Specified WPAR environment	ID of the specified WPAR
Yes	ID of the non-existing WPAR	--	--
No	Blank	Global environment	Blank
No	0	Global environment	0
No	ID of the existing WPAR	--	--
No	ID of the non-existing WPAR	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings >

Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

Table 2–28: When PFM - Agent for Platform is installed in a system WPAR environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	System WPAR environment in which PFM - Agent for Platform is installed	Blank
Yes	0	System WPAR environment in which PFM - Agent for Platform is installed	0
Yes	Value other than 0	--	--
No	Blank	System WPAR environment n which PFM - Agent for Platform is installed	Blank
No	0	System WPAR environment n which PFM - Agent for Platform is installed	0
No	Value other than 0	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

Reference note:

The following table describes the relationship between the property setting and obtained information in an environment in which no system WPAR has been created.

Table 2–29: When PFM - Agent for Platform is installed in the global environment in an environment in which no system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Global environment	Blank

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	0	Global environment	0
Yes	Value other than 0	--	--
No	Blank	Global environment	Blank
No	0	Global environment	0
No	Value other than 0	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

- PD_APSI record

For process information obtained by the PD_APSI record, the following table describes the relationship between the property setting and the obtained information.

Table 2–30: When PFM - Agent for Platform is installed in the global environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global environment and all WPAR environments	ID of the specified WPAR#2
Yes	0	Global environment	0
Yes	ID of the existing WPAR	Specified WPAR environment	ID of the specified WPAR
Yes	ID of the non-existing WPAR	--#3	--
No	Blank	Global environment	0
No	0	Global environment	0
No	ID of the existing WPAR	--#3	--
No	ID of the non-existing WPAR	--#3	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

When process information is collected from the global zone, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected in a WPAR environment, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to the WPAR environment.

#3

Not displayed in a report.

Table 2–31: When PFM - Agent for Platform is installed in a system WPAR environment in an environment in which a system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	System WPAR environment where PFM - Agent for Platform is installed	0
Yes	0	System WPAR environment where PFM - Agent for Platform is installed	0
Yes	Value other than 0	--#2	--
No	Blank	System WPAR environment where PFM - Agent for Platform is installed	0
No	0	System WPAR environment where PFM - Agent for Platform is installed	0
No	Value other than 0	--#2	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

Not displayed in a report.

Reference note:

The following table describes the relationship between the property setting and the obtained information in an environment in which no system WPAR has been created.

Table 2–32: When PFM - Agent for Platform is installed in the global environment in an environment in which no system WPAR has been created

Property settings		Obtained information	
ALL WPAR Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
Yes	Blank	Global environment	0
Yes	0	Global environment	0
Yes	Value other than 0	--#2	--
No	Blank	Global environment	0
No	0	Global environment	0
No	Value other than 0	--#2	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring - Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

Not displayed in a report.

- PD_APS record

For process information obtained by the PD_APS record, the following table describes the relationship between the property setting and the obtained information.

Table 2–33: When PFM - Agent for Platform is installed in the global environment in an environment in which a System WPAR has been created

ALL WPAR Collection for Process setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global environment and all WPAR environments	ID of the specified WPAR#
No	Global environment	0

#

When process information is collected from the global zone, the virtualized environment ID (value of the Virtual Env ID field) is 0. When process information is collected in a WPAR environment, the virtualized environment ID (value of the Virtual Env ID field) is the ID assigned to the WPAR environment.

Table 2–34: When PFM - Agent for Platform is installed in a system WPAR environment in an environment in which a system WPAR has been created

ALL WPAR Collection for Process setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	System WPAR environment where PFM - Agent for Platform is installed	0
No	System WPAR environment where PFM - Agent for Platform is installed	0

Reference note:

The following table describes the relationship between the property setting and the obtained information in an environment in which no system WPAR has been created.

Table 2–35: When PFM - Agent for Platform is installed in the global environment in an environment in which no system WPAR has been created

ALL WPAR Collection for Process setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Global environment	0
No	Global environment	0

(c) Linux-container-based environment

If processes that have the same name exist in host environment and container environment, check the value of the Virtual Env ID field to determine the zone from which the process information for the PD, PD_APP2, PD_APSI and PD_APS records was collected.

- PD record

For process information obtained by the PD record, the following table describes the relationship between the property setting and the obtained information.

Table 2–36: When PFM - Agent for Platform is installed in the host environment in an environment in which container environment have been created

ALL Container Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Host environment and all container environments	The ID of each container#
No	Host environment	0

#

Either 0 (in the case of the host environment) or the ID assigned to a container (in the case of the container environment) is stored as the ID of the virtualized environment (value of the Virtual Env ID field)

Table 2–37: When PFM - Agent for Platform is installed in a container environment in an environment in which container environment have been created

ALL Container Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Container environment in which PFM - Agent for Platform is installed	N/A
No	Container environment in which PFM - Agent for Platform is installed	N/A

- PD_APP record

With the PD_APP record, you cannot collect only information about processes running in a specific environment. If you want to collect only information about processes running in a specific environment, use the PD_APP2 record.

Note that when PFM - Agent for Platform is installed on the host environment, the range of data that can be collected for the PD_APP record will depend on the setting specified for the ALL Container Collection for Process property.

For process information obtained by the PD_APP record, the following table describes the relationship between the property setting and the obtained information.

Table 2–38: When PFM - Agent for Platform is installed in the host environment in an environment in which container environment have been created

Property settings	Obtained information
ALL Container Collection for Process	Data is collected from:
Yes	Host environment and all container environments
No	Host environment

- PD_APPD and PD_APP2 records

For process information obtained by the PD_APPD and PD_APP2 records, the following table describes the relationship between the property setting and the obtained information.

Table 2–39: When PFM - Agent for Platform is installed in the host environment in an environment in which container environment have been created

Property settings		Obtained information	
ALL Container Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Host environment and all container environments	Blank
Yes	0	Host environment	0
Yes	The ID of an existing container	The specified container environment	The ID of the specified container
Yes	The ID of a non-existent container	--	--
No	Blank	Host environment	Blank

Property settings		Obtained information	
ALL Container Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
No	0	Host environment	0
No	The ID of an existing container	--	--
No	The ID of a non-existent container	--	--

Legend:

--: Not collected. For PD_APPD records, 0 is displayed in the **Monitoring Count** field.

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The PD_APPD record has no Virtual Env ID field.

Table 2–40: When PFM - Agent for Platform is installed in the container environment in an environment in which container environment have been created

Property settings		Obtained information	
ALL Container Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value#2
Yes	Blank	Container environment in which PFM - Agent for Platform is installed	N/A
Yes	0	Container environment in which PFM - Agent for Platform is installed	N/A
Yes	Value other than 0	Container environment in which PFM - Agent for Platform is installed	N/A
No	Blank	Container environment in which PFM - Agent for Platform is installed	N/A
No	0	Container environment in which PFM - Agent for Platform is installed	N/A
No	Value other than 0	Container environment in which PFM - Agent for Platform is installed	N/A

#1

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings >

Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

The `PD_APPD` record has no Virtual Env ID field.

- `PD_APSI` record

For process information obtained by the `PD_APSI` record, the following table describes the relationship between the property setting and the obtained information.

Table 2–41: When PFM - Agent for Platform is installed in the host environment in an environment in which container environment have been created

Property settings		Obtained information	
ALL Container Collection for Process	Virtual Environment ID#1	Data is collected from:	Virtual Env ID field value
`Yes`	Blank	Host environment and all container environments	The ID of each container#2
`Yes`	`0`	Host environment	`0`
`Yes`	The ID of an existing container	The specified container environment	The ID of the specified container
`Yes`	The ID of a non-existent container	--#3	--
`No`	Blank	Host environment	`0`
`No`	`0`	Host environment	`0`
`No`	The ID of an existing container	--#3	--
`No`	The ID of a non-existent container	--#3	--

Legend:

--: Not collected.

#1

This is a property displayed when *host-name* `<UNIX>` (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

#2

Either `0` (in the case of the host environment) or the ID assigned to a container (in the case of the container environment) is stored as the ID of the virtualized environment (value of the `Virtual Env ID` field)

#3

Not displayed in a report.

Table 2–42: When PFM - Agent for Platform is installed in the container environment in an environment in which container environment have been created

Property settings		Obtained information	
ALL Container Collection for Process	Virtual Environment ID#	Data is collected from:	Virtual Env ID field value
Yes	Blank	Container environment in which PFM - Agent for Platform is installed	N/A
Yes	0	Container environment in which PFM - Agent for Platform is installed	N/A
Yes	Value other than 0	Container environment in which PFM - Agent for Platform is installed	N/A
No	Blank	Container environment in which PFM - Agent for Platform is installed	N/A
No	0	Container environment in which PFM - Agent for Platform is installed	N/A
No	Value other than 0	Container environment in which PFM - Agent for Platform is installed	N/A

Legend:

--: Not collected.

#

This is a property displayed when *host-name* <UNIX> (Agent Collector service) is selected in the **Services** tree of PFM - Web Console, and then from the **Advanced application monitoring** - **Application monitoring setting** tree, an application is selected. This corresponds to the **Virtual environment ID** in the Process monitoring settings > Create a New Application Definition window or Process monitoring settings > Edit an Application Definition window of the PFM - Web Console's **Agents** tree.

- PD_APS record

For process information obtained by the PD_APS record, the following table describes the relationship between the property setting and the obtained information.

Table 2–43: When PFM - Agent for Platform is installed in the host environment in an environment in which container environment have been created

ALL Container Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
Yes	Host environment and all container environments	The ID of each container#
No	Host environment	0

#

Either 0 (in the case of the host environment) or the ID assigned to a container (in the case of the container environment) is stored as the ID of the virtualized environment (value of the Virtual Env ID field)

Table 2–44: When PFM - Agent for Platform is installed in the container environment in an environment in which container environment have been created

ALL Container Collection for Process property setting	Obtained information	
	Data is collected from:	Virtual Env ID field value
`Yes`	Container environment in which PFM - Agent for Platform is installed	N/A
`No`	Container environment in which PFM - Agent for Platform is installed	N/A

2.3.23 Monitoring processor performance based on results output by the sar command in an AIX environment

If you are using simultaneous multithreading (SMT) or micro-partitioning in an AIX environment, the amount of physical CPU allocated is included in the calculation of CPU usage displayed by the `sar` command, but is not included in the calculation of CPU usage displayed by PFM - Agent for Platform. For this reason, the CPU usage figure displayed by PFM - Agent for Platform might be lower than that output by the `sar` command.

PFM - Agent for Platform enables settings to be changed so that processor performance data can be collected from the results output by the `sar` command of AIX.

This section explains how to collect and monitor processor performance data from the results output by the `sar` command of AIX when PFM - Agent for Platform is used.

(1) Settings

This section describes how to set the system to collect processor performance data from the results output by AIX's `sar` command.

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.

2. On the navigation frame of the **Main** window, click the **Services** tab.
   The Services window appears.

3. From the navigation frame, expand the contents of the **Machines** folder.
   A folder that has the name of the host on which the Performance Management service is installed appears. Expand the folder that has the host name to display the services installed on that host.
   The names of the services are displayed, in order of the service IDs. For details about service IDs, see *C. List of Identifiers* and the sections that explain the service naming conventions in the appendixes of the *JP1/Performance Management Planning and Configuration Guide*.
   The format of the service ID depends on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter that describes the Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the monitoring agent host folder, and select *host-name*`<UNIX>` (Agent Collector service).
   A check mark is displayed next to the selected Agent Collector service.

5. Select the `Properties` method in the method frame.

The Service Properties window appears.

6. Select **Agent Configuration**.

7. In **sar Command Monitoring** at the bottom of the information frame, specify whether to collect processor performance data from the results output by the `sar` command.

   Specify one of the following:

   - `Yes`: Collect from the `sar` command's output results.
   - `No`: Do not collect from the `sar` command's output results (default).

8. In **sar Command Interval** at the bottom of the information frame, specify the interval (in seconds) to be specified as the `sar` command argument.

   Specify the numeric value that corresponds to the `INTERVAL` argument of the following AIX `sar` command:

```
/usr/sbin/sar -P ALL INTERVAL 1
```

   Specify a value between 1 and 50 (the default is 5).

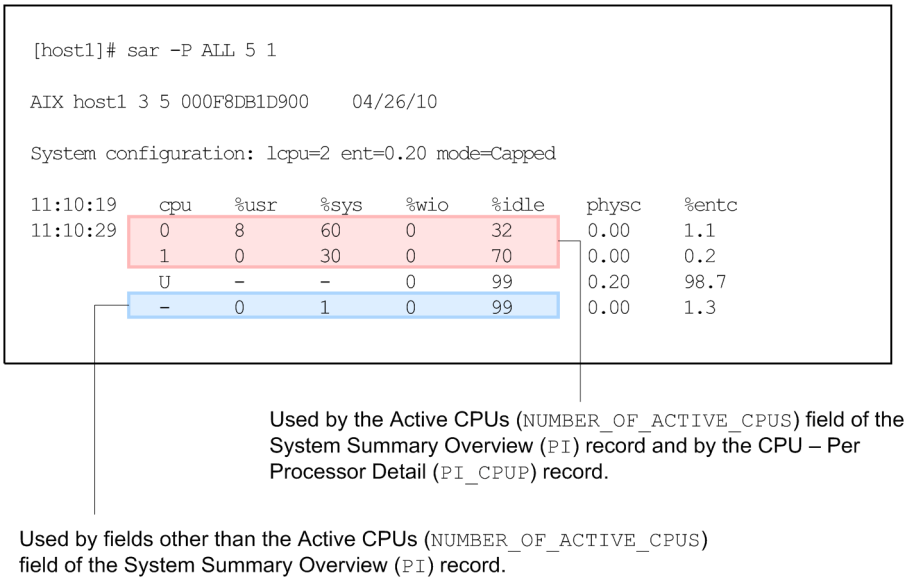
(2) Correspondence between sar command output results and PFM - Agent for Platform records

If you are using simultaneous multithreading (SMT) or micro-partitioning in an AIX environment, PFM - Agent for Platform enables you to collect the results output by the following AIX `sar` command:

```
/usr/sbin/sar -P ALL INTERVAL 1
```

An example of the results output by the AIX `sar` command is shown below.

Figure 2–18: Example of results output by AIX sar command



With PFM - Agent for Platform, the results output by the `sar` command are stored in the fields that monitor processor performance data of the System Summary Overview (`PI`) and CPU - Per Processor Detail (`PI_CPUP`) records.

The following table describes the correspondence between the fields of the System Summary Overview (`PI`) record and the results output by the `sar` command.

Table 2–45: Correspondence between System Summary Overview (PI) record fields and the results output by the sar command

Record	Field	Field description	Results output by the sar command
System Summary Overview (PI)	Active CPUs (NUMBER_OF_ACTIVE_CPUS)	Number of processors	Number of rows in which a number is displayed in the cpu column
	CPU % (KERNELMODE_USERMODE_PERCENT)	CPU usage (%). This also indicates the average percentage for each processor.	%sys column + %usr column
	Idle % (IDLE_TIME_PERCENT)	Percentage of time spent in idle state (%). This is also the average of the percentages for all processors.	%idle column
	Kernel CPU % (KERNELMODE_PERCENT)	Percentage of time running in kernel mode (%). This is also the average of the percentages for all processors.	%sys column
	User CPU % (USERMODE_PERCENT)	Percentage of time running in user mode (%). This is also the average of the percentages for all processors.	%usr column
	Wait % (WAIT_TIME_PERCENT)	Percentage of time spent in I/O standby (%). This is also the average of the percentages for each processor.	%wio column

The following table describes the correspondence between the fields of the CPU - Per Processor Detail (PI_CPUP) record and the results output by the sar command.

Table 2–46: Correspondence between CPU - Per Processor Detail (PI_CPUP) record fields and the results output by the sar command

Record	Field	Field description	Results output by the sar command
CPU - Per Processor Detail (PI_CPUP)	CPU % (KERNELMODE_USERMODE_PERCENT)	CPU usage per processor (%)	%sys column + %usr column
	Idle % (PROCESSOR_IDLE_PERCENT)	Percentage of time spent in idle state (%)	%idle
	Processor ID (LOGICAL_PROCESSOR_ID)	Processor identifier	cpu column
	System % (PROCESSOR_SYSTEM_PERCENT)	Percentage of time running in kernel mode (%)	%sys column
	User % (PROCESSOR_USER_PERCENT)	Percentage of time running in user mode (%)	%usr column

Record	Field	Field description	Results output by the sar command
CPU - Per Processor Detail (PI_CPUP)	Wait % (PROCESSOR_WAIT_PERCENT)	Percentage of time spent in I/O standby (%)	%wio column

When the setting is made to collect processor performance data from the results output by AIX's sar command, 0 is set as the value for the fields of the System Summary Overview (PI) record shown in the table below.

Table 2–47: System Summary Overview (PI) record fields whose value is 0 when collection of processor performance data from the results output by the AIX sar command is set

Record	Field	Field description	Alarm used	Report used
System Summary Overview (PI)	Total Idle Time (TOTAL_IDLE_TIME)	Total time spent in idle state by all processors (sec)	None	None
	Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME)	Total time spent running in kernel mode by all processors (sec)	None	None
	Total User-Mode Time (TOTAL_USERMODE_TIME)	Total time spent running in user mode by all processors (sec)	None	None
	Total Wait Time (TOTAL_WAIT_TIME)	Total time spent in I/O standby by all processors (sec)	None	None

When the setting is made to collect processor performance data from the results output by AIX's sar command, a blank is set as the value for the fields of the CPU - Per Processor Detail (PI_CPUP) record shown in the table below.

Table 2–48: CPU - Per Processor Detail (PI_CPUP) record fields whose value is blank when collection of processor performance data from the results output by the AIX sar command is set

Record	Field	Field description	Alarm used	Report used
CPU - Per Processor Detail (PI_CPUP)	Context Switches (PROCESSOR_CONTEXT_SWITCHES)	Number of context switch executions	None	None
	Context Switches/sec (PROCESSOR_CONTEXT_SWITCHES_PER_SECOND)	Frequency of context switch executions (executions per sec)	None	None
	Idle Time (PROCESSOR_IDLE_TIME)	Time in idle state (sec)	None	None
	Sys Calls/sec (PROCESSOR_SYSTEM_CALLS_PER_SECOND)	Frequency of system calls (calls per sec)	None	None
	System Calls (PROCESSOR_SYSTEM_CALLS)	Number of system calls	None	None

Record	Field	Field description	Alarm used	Report used
CPU - Per Processor Detail (PI_CPUP)	System Time (PROCESSOR_SYSTEM_TIME)	Time running in kernel mode (sec)	None	None
	Type (PROCESSOR_TYPE)	Description of processor	None	None
	User Time (PROCESSOR_USER_TIME)	Time running in user mode (sec)	None	None
	Wait Time (PROCESSOR_WAIT_TIME)	Time spent in I/O standby (sec)	None	None

(3) Records supported in global environment and WPAR environments

When processor performance data is collected from the results output by the `sar` command, the records supported and the range of data collected for each record will depend on whether PFM - Agent for Platform is running in the global environment or in the System WPAR environment.

The following table describes the fields of System Summary Overview (`PI`) records supported in the global environment and in the System WPAR environment.

Table 2–49: Fields of System Summary Overview (PI) records supported in the global environment and the System WPAR environment

Record	Field	Global environment		System WPAR environment	
		Collected?	Range collected	Collected?	Range collected
System Summary Overview (PI)	15-Minute Run Queue Avg (FIFTEEN_MINUTE_RUN_QUEUE_AVG)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	1-Minute Run Queue Avg (ONE_MINUTE_RUN_QUEUE_AVG)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	5-Minute Run Queue Avg (FIVE_MINUTE_RUN_QUEUE_AVG)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Active CPUs (NUMBER_OF_ACTIVE_CPUS)	Yes	Global environment and all WPAR environments	No	No
	Block Ops (BLOCKIO_IO_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Block Reads (BLOCKIO_READ_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments

Record	Field	Global environment		System WPAR environment	
		Collected?	Range collected	Collected?	Range collected
System Summary Overview (PI)	Block Reads/sec (BLOCKIO_READ_OPS_PER_SECOND)	Yes	Global environment and all WPAR environments	No	No
	Block Writes (BLOCKIO_WRITE_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Block Writes/sec (BLOCKIO_WRITE_OPS_PER_SECOND)	Yes	Global environment and all WPAR environments	No	No
	Cache Read % (CACHE_READ_PERCENT)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Cache Write % (CACHE_WRITE_PERCENT)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Context Switches (CONTEXT_SWITCHES)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Context Switches/sec (CONTEXT_SWITCHES_PER_SECOND)	Yes	Global environment and all WPAR environments	No	No
	CPU % (KERNELMODE_USERMODE_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Idle % (IDLE_TIME_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Kernel CPU % (KERNELMODE_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Logical I/O Ops (LOGICAL_IO_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Logical Read Mbytes (LOGICAL_READ_MBYTES)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Logical Reads (LOGICAL_READ_OPS)	Yes	Global environment and	Yes	Global environment and

Record	Field	Global environment		System WPAR environment	
		Collected?	Range collected	Collected?	Range collected
System Summary Overview (PI)	Logical Reads (LOGICAL_READ_OPS)	Yes	all WPAR environments	Yes	all WPAR environments
	Logical Reads/sec (LOGICAL_READ_MBYTES_PER_SECOND)	Yes	Global environment and all WPAR environments	No	No
	Logical Write Mbytes (LOGICAL_WRITE_MBYTES)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Logical Writes (LOGICAL_WRITE_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Logical Writes/sec (LOGICAL_WRITE_MBYTES_PER_SECOND)	Yes	Global environment and all WPAR environments	No	No
	Physical I/O Ops (PHYSICAL_IO_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Physical Reads (PHYSICAL_READ_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	Physical Writes (PHYSICAL_WRITE_OPS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	System Calls (SYSTEM_CALLS)	Yes	Global environment and all WPAR environments	Yes	Global environment and all WPAR environments
	User CPU % (USERMODE_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Wait % (WAIT_TIME_PERCENT)	Yes	Global environment and all WPAR environments	No	No

Legend:

Yes: Supported.

No: Not supported.

Note:

Records and fields not listed are the same as for the WPAR environment when processor performance data is not collected from results output by the `sar` command.

The following table describes the fields of CPU - Per Processor Detail (PI_CPUP) records supported in the global environment and in the System WPAR environment.

Table 2–50: Fields of CPU - Per Processor Detail (PI_CPUP) records supported in the global environment and the System WPAR environment

Record	Field	Global environment		System WPAR environment	
		Collected?	Range collected	Collected?	Range collected
CPU - Per Processor Detail (PI_CPUP)	Boot Time (SYSTEM_BOOT_TIME)	Yes	Global environment	No	No
	CPU % (KERNELMODE_USERMODE_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Idle % (PROCESSOR_IDLE_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Interval (INTERVAL)	Yes	Global environment	No	No
	Processor ID (LOGICAL_PROCESSOR_ID)	Yes	Global environment	No	No
	Record Time (RECORD_TIME)	Yes	Global environment	No	No
	Record Type (INPUT_RECORD_TYPE)	Yes	Fixed value	No	No
	Status (PROCESSOR_STATUS)	Yes	Global environment	No	No
	System % (PROCESSOR_SYSTEM_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Up Time (SYSTEM_UP_TIME)	Yes	Global environment	No	No
	User % (PROCESSOR_USER_PERCENT)	Yes	Global environment and all WPAR environments	No	No
	Wait % (PROCESSOR_WAIT_PERCENT)	Yes	Global environment and all WPAR environments	No	No

Legend:

Yes: Supported.

No: Not supported.

Note:

Records and fields not listed are the same as for the WPAR environment when processor performance data is not collected from results output by the `sar` command.

(4) Notes

This section provides additional information about collecting and monitoring processor performance data from the results output by AIX's `sar` command when PFM - Agent for Platform is used.

- More time is needed to collect processor performance data from results output by a `sar` command than when data is not collected from `sar` command results. The difference in the amount of time used is the interval set on the PFM - Web Console's Service Properties window. If a value greater than the record collection time is specified for the interval on the PFM - Web Console's Service Properties window, record collection will not be performed.
- If performance data could not be collected because execution of the `sar` command failed or for some other reason, the values in the corresponding fields in the System Summary Overview (`PI`) record will be `0`, and performance data will not be gathered in the CPU - Per Processor Detail (`PI_CPUP`) record.
- The values set in the fields of the System Summary Overview (`PI`) and CPU - Per Processor Detail (`PI_CPUP`) records depend on whether processor performance data is collected from results output by the `sar` command. The following tables describe the differences between the values stored in the various fields depending on whether processor performance data is collected from the results output by the `sar` command.
- When processor performance is monitored by using the output results of the `sar` command, do not use functions (1) and (2) below simultaneously, or functions (1) and (3) below simultaneously. If you use them simultaneously, log data collection or alarm monitoring might be skipped.

  (1) Real-time display of the `PI` record or `PI_CPUP` record.

  (2) Log data collection of the `PI` record or `PI_CPUP` record.

  (3) Alarm monitoring of the `PI` record or `PI_CPUP` record.

Table 2–51: Differences in values stored in fields depending on whether processor performance data is collected from the results output by the sar command (when Indicate delta value is selected on real-time reports)

Record	Field	Data source	
		Not from results output by the sar command	From results output by the sar command
System Summary Overview (`PI`)	Active CPUs (`NUMBER_OF_ACTIVE_CPUS`)	Results calculated based on value at time of collection	Value at time of collection
	CPU % (`KERNELMODE_USERMODE_PERCENT`)	Results calculated based on amount of change	Value at time of collection
	Idle % (`IDLE_TIME_PERCENT`)	Results calculated based on amount of change	Value at time of collection
	Kernel CPU % (`KERNELMODE_PERCENT`)	Results calculated based on amount of change	Value at time of collection
	User CPU % (`USERMODE_PERCENT`)	Results calculated based on amount of change	Value at time of collection

Record	Field	Data source	
		Not from results output by the sar command	From results output by the sar command
System Summary Overview (PI)	Wait % (WAIT_TIME_PERCENT)	Results calculated based on amount of change	Value at time of collection
CPU - Per Processor Detail (PI_CPUP)	CPU % (KERNELMODE_USERMODE_PERCENT)	Results calculated based on amount of change	Value at time of collection
	Idle % (PROCESSOR_IDLE_PERCENT)	Results calculated based on amount of change	Value at time of collection
	Processor ID (LOGICAL_PROCESSOR_ID)	Value at time of collection	Value at time of collection
	System % (PROCESSOR_SYSTEM_PERCENT)	Results calculated based on amount of change	Value at time of collection
	User % (PROCESSOR_USER_PERCENT)	Results calculated based on amount of change	Value at time of collection
	Wait % (PROCESSOR_WAIT_PERCENT)	Results calculated based on amount of change	Value at time of collection

Table 2–52: Differences in values stored in fields depending on whether processor performance data is collected from the results output by the sar command (when Indicate delta value is not selected on real-time reports)

Record	Field	Data source	
		Not from results output by the sar command	From results output by the sar command
System Summary Overview (PI)	Active CPUs (NUMBER_OF_ACTIVE_CPUS)	Results calculated based on value at time of collection	Value at time of collection
	CPU % (KERNELMODE_USERMODE_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Idle % (IDLE_TIME_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Kernel CPU % (KERNELMODE_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	User CPU % (USERMODE_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Wait % (WAIT_TIME_PERCENT)	Results calculated based on cumulative value	Value at time of collection

Record	Field	Data source	
		Not from results output by the sar command	From results output by the sar command
CPU - Per Processor Detail (PI_CPUP)	CPU % (KERNELMODE_USERMODE_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Idle % (PROCESSOR_IDLE_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Processor ID (LOGICAL_PROCESSOR_ID)	Value at time of collection	Value at time of collection
	System % (PROCESSOR_SYSTEM_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	User % (PROCESSOR_USER_PERCENT)	Results calculated based on cumulative value	Value at time of collection
	Wait % (PROCESSOR_WAIT_PERCENT)	Results calculated based on cumulative value	Value at time of collection

- Performance data is collected only for instances whose CPU - Per Processor Detail (PI_CPUP) record's Status field is On Line.
- If the setting for collecting processor performance data from the results output by the sar command is changed from not collecting performance data to collecting performance data, the performance data from before the setting was changed will not be consistent with the data collected after the change. In such a case, treat performance data before and after the change as separate sets of performance data. If necessary, take steps such as backing up the Store database prior to changing system resources, and then clearing the Store database and starting collection of new performance data. For details about backing up the Store database, see the chapter on backing up and restoring in the *JP1/Performance Management User's Guide*.
- Executing the sar command when PFM - Agent for Platform is running in an AIX System WPAR environment will result in an error. In such a case, the fields of the System Summary Overview (PI) record will have a value of 0, and no performance data will be collected in the fields of the CPU - Per Processor Detail (PI_CPUP) record.
- When you monitor processor performance according to the output results of the sar command, do not use functions 1 and 2 or functions 1 and 3 (below) at the same time. Using these functions at the same time might cause historical data collection or alarm monitoring to be skipped.
  1. Displaying a real-time report for the PI or PI_CPUP record
  2. Collecting historical data for the PI or PI_CPUP record
  3. Monitoring alarms for the PI or PI_CPUP record
- You cannot connect to the Agent Collector service from the **Service Properties** window while the Agent Collector service is collecting performance data from the sar command. Connect to the Agent Collector service after the time specified in **sar Command Interval** has elapsed.
- If you change the value of **sar Command Monitoring** in the **Service Properties** window for PFM - Web Console, performance data might become extremely large or negative at the initial collection. Therefore, if you have bound alarms that monitor performance for the PI_CPUP and PI records, unbind them before updating the properties. Normal values will be collected for performance data at the second collection and thereafter.

- If the `sar Command Monitoring` property (displayed by selecting `Agent Configuration` in PFM - Agent for Platform) is set to `Yes`, performance information might not be collected correctly if either of the following conditions are met:
  - The `iostat` kernel parameter is set to `false`.
  - The `sar` command has been executed.
  - On two or more hosts within the same machine, `Yes` is set for the `sar Command Monitoring` property (displayed by selecting `Agent Configuration` in PFM - Agent for Platform).

  In addition to the above conditions, performance information might not be collected correctly if the PFM - Remote Monitor for Platform instance settings match one of the following conditions:
  - A host that runs PFM - Agent for Platform is specified as a host to be monitored.
  - The `iostat` kernel parameter is set to `false`.
  - The `Std_Category` property or `Disk_Category` property of the instance is set to `Y`.#

    #

    - If the `Std_Category` property is set to `Y`, CPU performance information might not be collected.

    - If the `Disk_Category` property is set to `Y`, disk performance information might not be collected.

2.4 Overview of monitoring templates

In Performance Management, you can use the following methods to define alarms and reports:

- Using the alarms and reports defined in PFM - Agent without any modification
- Copying and customizing the alarms and reports defined in PFM - Agent
- Using the wizard to define new alarms and reports

A collection of alarms and reports provided by PFM - Agent is called a *monitoring template*. Because the necessary information is predefined for the alarms and reports in the monitoring templates, you can copy and use them without modification, or you can customize them according to the user environment. Therefore, you can easily carry out the preparation necessary for monitoring the operation status of the monitoring target without having to use the wizard to create new alarms and reports.

For details about the settings of alarms and reports in the monitoring templates defined in PFM - Agent for Platform, see *8. Monitoring Templates*. For details about how to use the monitoring templates, see the chapter that explains operation monitoring that uses alarms and creates reports for operation analysis in the *JP1/Performance Management User's Guide*.

Notes:

- The thresholds specified for monitoring template alarms are for reference only. To use monitoring template alarms, copy them and specify appropriate thresholds for the environment or operating system.
- If the setup files for PFM - Agent for Platform were copied to an instance of PFM - Manager earlier than version 10-00 for setup, an alarm may not be issued even if abnormal conditions are met for the following alarms on the monitoring template:

  - Application Status alarm

  To use the above alarm in an instance of PFM - Manager earlier than version 10-00 that meets these criteria, copy the alarm from the monitoring template and update it.

  To copy and update the Application Status alarm:

  1. In the navigation frame of the main window of PFM - Web Console, display the **Alarms** page.
  2. In the navigation frame, select **UNIX** and then **PFM UNIX Platform Template Alarms [APP] 09.10**.
  3. In the Methods frame, select the **Copy** method.
  4. Enter the alarm table name in the [**New alarm table name**] field in the information frame, and then click **OK**.
  5. Select the Application Status alarm in the alarm table you copied from the navigation frame.
  6. In the method frame, select the **Edit** method.
  7. In the information frame, click **Finish**. (You do not need to make any changes to the settings.)

  If you click the **Start** button under **Process monitoring alarm settings** in the **Process Monitoring Settings** window of version 10-00 or later of PFM - Web Console, the Application Status alarm in the monitoring template is bound instead. When using the PFM - Manager earlier than version 10-00, do not bind the alarm from the **Process Monitoring Settings** window.

  Note that you can resolve this issue by upgrading to version 10-00 or later of PFM - Manager.

3

Introduction to Performance Monitoring Operations by Example

Through use of examples, this chapter introduces you to PFM - Agent for Platform as a means to carry out performance monitoring operations.

3.1 Performance monitoring examples

This section explains how to monitor performance for each system resource, and provides examples of performance data collection.

- Record IDs are used as record names, and PFM - View names are used as field names. For details about the formal record names and field names, see *9. Records*.
- Fields are not explained in detail. For details about fields, see *9. Records*.
- If you want to monitor information from multiple programs together, see *7.1 Settings for collecting workgroup information*.
- For details about how to monitor various types of user-defined records, such as how to monitor information about the operating status of processes and applications, see *7. User-Defined Record Collection*.

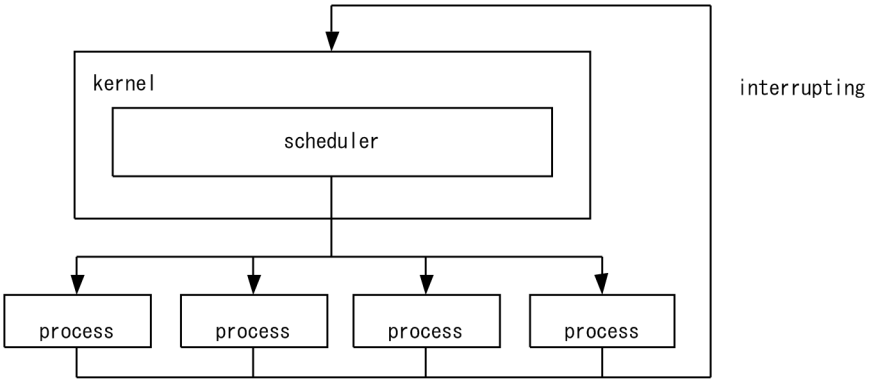
3.1.1 Processor monitoring examples

This subsection explains how to monitor processor performance.

(1) Overview

By monitoring processor performance, you can check the performance trends for the entire system. In UNIX, as illustrated in the following figure, the processor operations include processor operations for the kernel and for user processes.

Figure 3–1: Relationship between the kernel and processes



The processor usage status can be checked by monitoring the CPU usage, which is the typical way, or by monitoring the number of queued jobs.

Jobs, such as processes, are executed by the CPU according to the schedule determined by the OS. The number of queued jobs is the number of jobs that are waiting to be executed by the CPU. When overall system load is high, the number of queued jobs tends to increase.

Monitoring templates provide functionality such as Kernel CPU alarms, User CPU alarms, Run Queue alarms, CPU Status (Multi-Agent) reports.

To monitor processor performance with more detail than with the monitoring template, the processor usage per processor, processor usage per process, processor queue count, and processor interrupts from hardware can also be monitored.

The following table lists and describes the principal records and fields related to processor monitoring.

Table 3–1: Principal records and fields related to processor monitoring

Record	Field	Description (example)
PI	1-Minute Run Queue Avg	The average number of threads, exclusive of those for kernels pending within the execution queue. If value of this field is large, there might be a problem with the efficiency of processor usage.
	5-Minute Run Queue Avg	
	15-Minute Run Queue Avg	
	CPU %	The CPU usage (the percentage of time that the CPU was being used). If value of this field is large, the CPU load is high.
	Idle %	The percentage of time that the CPU was not being used. If value of this field is large, the CPU load is low.
PI_CPUP	CPU %	The CPU usage for a processor.
	System %	The CPU usage for a processor executed in kernel mode. If the value of this field is large, there might be a problem with the OS or operation.
	User %	The CPU usage for a processor executed in user mode. If the value of this field is large, there might be a problem with a specific application.
PD_PDI	CPU %	The CPU usage for a process. If the value of this field is large, there might be a problem with the OS, operation, or a specific application.
	System CPU	The time the CPU takes to execute each process in kernel mode. If the value of this field is large, there might be a problem with the OS.
	User CPU	The time the CPU takes to execute each process in user mode. If the value of this field is large, there might be a problem with a specific application.

(2) Monitoring methods

(a) Monitoring kernel CPU usage

You can use the Kernel CPU alarm provided by the monitoring template to monitor system-wide kernel CPU usage.

For details, see *3.2.1(1) Monitoring template*.

(b) Monitoring user CPU usage

You can use the User CPU alarm provided by the monitoring template to monitor system-wide user CPU usage.

For details, see *3.2.1(1) Monitoring template*.

(c) Monitoring the CPU usage for each processor

By monitoring the CPU usage for each processor, you can check for problems with OS operation, such as an unbalanced CPU load. You can use the monitoring result of the CPU usage as a guideline for taking corrective measures.

Monitoring the kernel CPU usage, user CPU usage, and processor congestion is an effective way to monitor the CPU usage for each processor.

If the user CPU usage for a processor (the `User %` field of the `PI_CPUP` record) is at or above the threshold, you might need to look for processes that excessively use the CPU and to take action.

If the kernel CPU usage for a processor (the `System %` field of the `PI_CPUP` record) is at or above the threshold, the system environment is inadequate for the processing. In this case, you might need to upgrade the processor or add processors.

For definition examples, see *3.2.1(2) Definition examples other than for monitoring templates*.

(d) Monitoring processor congestion

You can use the Run Queue alarm provided by the monitoring template to monitor processor congestion.

In addition to processor usage, you can monitor processor congestion (the number of queued requests) to monitor the processor load status. Monitoring both processor congestion and processor usage is an effective way to monitor the processor load status.

For details, see *3.2.1(1) Monitoring template*.

(e) Checking processes whose processor usage is high

By monitoring kernel CPU usage, user CPU usage, CPU usage for each processor, and processor congestion, you can determine if a bottleneck exists. If one does exist, you can use a real-time report (the `CPU %` field of the `PD_PDI` record) to find processes that are monopolizing the processor.

If no such processes exist, the system environment is inadequate for the processing. In this case, you might need to upgrade the processor or add processors.

For definition examples, see *3.2.1(2) Definition examples other than for monitoring templates*.

3.1.2 Memory monitoring examples

This subsection explains how to monitor memory performance.

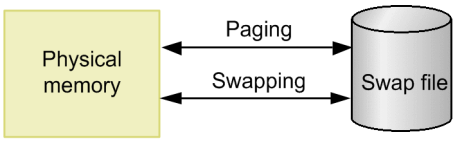
(1) Overview

You can monitor memory performance to detect physical memory shortages and incorrect process operations.

Memory consists of physical memory and a swap file, as illustrated below. However, because the causes of bottlenecks are not limited to a small amount of physical memory or a small swap file, the paging status, page faults, and other items related to efficient memory usage must be monitored as well.

The following figure illustrates the configuration of the memory space.

Figure 3–2: Conceptual diagram of the memory space



Insufficient physical memory degrades overall system performance.

Memory areas not accessed by programs for a long time are saved to the swap file, and are loaded into physical memory on demand. Physical memory is efficiently used in this manner. Note, however, that swap file access is markedly slower than physical memory access. Therefore, frequent swapping or frequent page faults will considerably delay system processing.

- Paging

  Paging is the movement of code and data between physical memory and a paging file. Loading from the paging file into physical memory is referred to by the term *page-in*, and saving from physical memory to the paging file is referred to by the term *page-out*.

- Page fault

  A page fault is an attempt to access an area that does not exist in physical memory.

Because paging often occurs even in normal processing, measure performance when the system is operating stably before attempting to determine proper thresholds.

The Swap Outs alarm is provided by the monitoring template. If you want to perform more detailed monitoring, see the following table, which lists and describes the principal records and fields related to memory monitoring.

Table 3–2: Principal fields related to memory monitoring

Record	Field	Description (example)
PI	Total Physical Mem Mbytes	The amount of physical memory.
	Free Mem Mbytes	The amount of available physical memory.
	Alloc Mem Mbytes	The amount of physical memory in use.
	Alloc Mem %	Physical memory usage.
	Total Swap Mbytes	Swap area size.
	Free Swap Mbytes	Free space in the swap area.
	Alloc Swap Mbytes	Swap area in use. If the value of this field continues to be at or above the threshold (the Total Physical Mem Mbytes field value of the PI record), a larger amount of memory might be required.
	Alloc Swap %	Swap area usage. If the value of this field continues to be at or above the threshold (determined by the system load status), the swap area might need to be expanded.
	Page Scans/sec	The number of page scans that occurred per second. If the value of this field continues to be at or above the threshold (150), memory might be a bottleneck.
	Swapped-Out Pages/sec	The number of swapped-out pages per second. If the value of this field continues to be at or above the threshold (200), memory might be a bottleneck.
	Buffers Mem %	The percentage of physical memory allocated to the file buffer. The field is unavailable in AIX, HP-UX, and Solaris.
	Buffers Mem Mbytes	The amount of physical memory allocated to the file buffer (MB). The field is unavailable in AIX, HP-UX, and Solaris.
	Cache Mem %	The percentage of physical memory allocated as cache memory. The field is unavailable in HP-UX and Solaris non-global zone environment.
	Cache Mem Mbytes	The amount of physical memory allocated as cache memory (MB). The field is unavailable in HP-UX and Solaris non-global zone environment.
	Effective Free Mem %	The percentage of physical memory available to applications. The field is unavailable in Solaris non-global zone environment.
	Effective Free Mem Mbytes	The amount of physical memory available to applications (MB). The field is unavailable in Solaris non-global zone environment.
	Effective Used Mem %	The percentage of physical memory actually allocated. The field is unavailable in Solaris non-global zone environment.

Record	Field	Description (example)
PI	Effective Used Mem Mbytes	The amount of physical memory actually allocated (MB). The field is unavailable in Solaris non-global zone environment.

The cause of a system memory shortage is not always physical memory itself. A problem with a program can also cause a shortage. By monitoring memory usage for each process, you can identify the cause of a shortage. If there is a process improperly occupying memory or if the amount of memory used by a process continues to increase steadily, the program running the process is likely to be defective.

The following table lists and describes the principal records and fields related to monitoring the memory usage of a specific process.

Table 3–3: Principal fields related to the memory monitoring for each process

Record	Field	Description (example)
PD_PDI	Real Mem Kbytes	The amount of physical memory used by a process. If the value of this field is large, a specific process might be using a large amount of physical memory.
	Virtual Mem Kbytes	The amount of virtual memory used by a process. If the value of this field is large, a specific process might be using a large amount of virtual memory.
	Swaps	The number of times swapping occurred for a process. If the value of this field is large, you might need to look for a process that is causing a bottleneck due to frequent swapping.

(2) Monitoring methods

(a) Monitoring the memory usage status

You can use the usage status of virtual memory as a guideline for determining whether to increase physical memory.

Even when memory usage is temporarily high, if the high load status does not persist, performance degradation might be permissible. Therefore, monitoring the number of page scans and the number of swapped-out pages in addition to memory usage is recommended.

If the amount of used virtual memory (the Alloc Swap Mbytes field of the PI record) is larger than the total amount of physical memory (the Total Physical Mem Mbytes field of the PI record), more memory might be required.

For definition examples, see *3.2.2(2) Definition example other than for monitoring templates*.

(b) Monitoring the number of page scans

You can use the Pagescans alarm provided by the monitoring template to monitor the number of page scans (Page Scans/sec field of the PI record).

Monitoring the swap-out status and memory usage in addition to the number of page scans is recommended.

If you find processes that are performing many page scans, you can take appropriate action. If no such processes exist, the system environment is inadequate for the processing. In this case, you might need to increase the amount of physical memory.

For details, see *3.2.2(1) Monitoring template*.

(c) Monitoring the swap-out status

You can use the Swap Outs alarm provided by the monitoring template to monitor the swap-out processes (`Swapped-Out Pages/sec` field of the PI record).

Monitoring the number of page scans and the memory usage status in addition to the swap-out status is recommended.

If you find processes that are excessively swapping pages, take appropriate action. If no such processes exist, the system environment is inadequate for the processing. In this case, you might need to increase the amount of physical memory.

For details, see *3.2.2(1) Monitoring template*.

(d) Monitoring memory usage by process

If you think there is a problem with a process after monitoring the memory usage status, the number of page scans, and the swap-out status, you must identify the process causing the problem.

If server activities have not increased, you can use a real-time report to monitor the memory usage of each process (by using the `Real Mem Kbytes` field of the `PD_PDI` record, for example) for a number of minutes. In the results displayed as a line graph, you can then check for a process whose memory usage is increasing steadily.

If you identify a process that is causing a memory leak or is excessively swapping pages, contact the vendor or take other appropriate action.

For definition examples, see *3.2.2(2) Definition example other than for monitoring templates*.

(e) Checking the amount of memory that the system can actually use

In Linux, AIX, and Solaris, information already stored in memory is retained as long as possible. For this reason, when the `Free Mem Mbytes` field of the `PI` record is used to monitor the amount of available memory, the value of the field gradually approaches 0. However, the stored information can be released from memory anytime and does not prevent applications from using memory. The amount of memory that can be freed up anytime can be monitored by using the `Buffers Mem Mbytes` field (Linux) or `Cache Mem Mbytes` field (Linux, AIX, and Solaris) of the `PI` record. PFM - Agent for Platform calculates the amount of memory that is available in reality from these fields and outputs the result to the `Effective Free Mem Mbytes` field of the `PI` record. You can use this field to monitor the amount of memory that is actually available to the system.

In HP-UX, the same value is output to the `Free Mem Mbytes` and `Effective Free Mem Mbytes` fields. This value is the amount of memory that is available in reality.

3.1.3 Disk monitoring examples

This subsection explains how to monitor disk performance.

(1) Overview

You can monitor disk performance to detect disk resource shortages and bottlenecks caused by a disk. Continuous monitoring of disk performance allows you to check for trends in increased disk space usage so that you can determine an appropriate configuration for the system or determine when the system configuration should be expanded.

A disk stores programs, the data used by the programs, and other data. If the amount of free disk space becomes insufficient, data might be lost or system response might slow down.

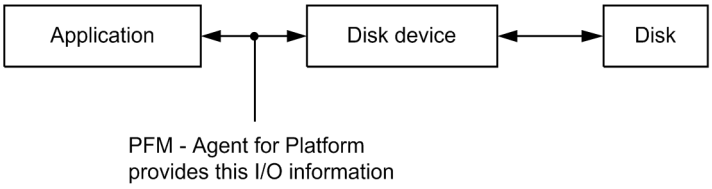
If a program that is performing a disk I/O operation must pause (that is, wait for the response), the disk is becoming a bottleneck.

A disk bottleneck can cause any of several types of performance degradation, such as slow process response. For this reason, it is important to check that disk performance is not degrading.

When you monitor the number of disk I/O operations, note the following.

I/O information for a disk that PFM - Agent for Platform acquires is the I/O information that the OS has acquired from the disk device. This information is not related to the I/O operations on the actual disk. The following figure shows I/O processing that occurs between an application and a disk.

Figure 3–3: Conceptual figure for I/O processing



The fields related to disk I/O load are `Avg Service Time` and `Busy %`.

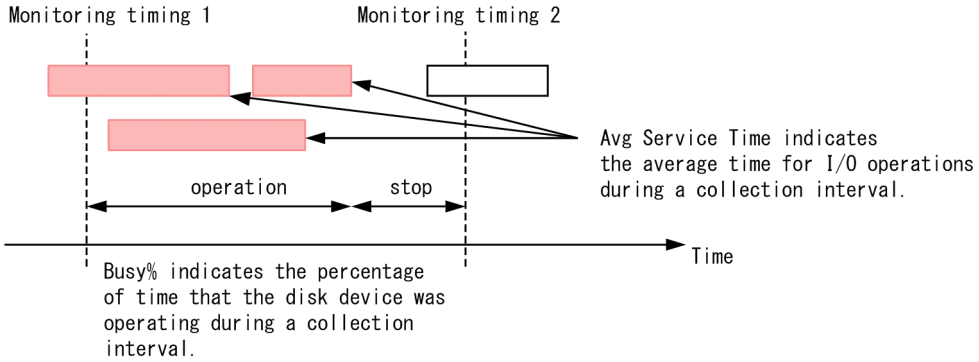
The `Avg Service Time` field indicates the average time required for one I/O operation. If a very large amount of information is input or output, or if an I/O operation is delayed, the value of this field becomes large.

The `Busy %` field indicates the percentage of time that the disk device was operating during a collection interval. The value of this field becomes large if I/O operations are concentrated in a short period of time.

As described above, the `Avg Service Time` and `Busy %` fields are related to the disk device load. You can therefore use these fields in the way that best suits your monitoring requirements.

The following figure explains what the `Avg Service Time` and `Busy %` field values indicate.

Figure 3–4: Values of the Avg Service Time and Busy % fields



The Disk Service Time alarm and Disk Busy % alarm are provided by the monitoring templates. If you want to perform more detailed monitoring, see the following table, which lists and describes the principal records and fields related to the monitoring of disk performance.

Table 3–4: Principal fields related to disk monitoring

Record	Field	Description (example)
`PI_DEVD`	`Avg Service Time`	The average time for I/O operations. If the value of this field is large, the amount of information being input or output might be very large.

Record	Field	Description (example)
PI_DEVD	Avg Wait Time	The average wait time for I/O operations. If the value of this field is large, the amount of information being input or output might be very large.
	Busy %	The disk busy rate. If the value of this field is large, I/O operations might be concentrated on a specific disk.
	I/O Mbytes	The total amount of information input or output. If the value of this field is large, the amount of information being input or output might be very large.
	Total I/O Ops	The number of I/O operations that occurred. If the value of this field is large, I/O operations might be concentrated on a specific disk.
	Queue Length	The queue length. If the value of this field continues to be at or above the threshold, the device is congested.
PD_FSL	Mbytes Free	The amount of available disk space. If the value of this field is small, the amount of free disk space is insufficient.
	Mbytes Free %	
PD_FSR	Mbytes Free	
	Mbytes Free %	

(a) Monitoring the disk free space

You can use the File System Free(L) alarm or File System Free(R) alarm provided by the monitoring template to monitor the amount of free disk space.

You can use an alarm to effectively monitor the percentage of free logical-disk space.

If the amount of free logical-disk space (the Mbytes Free or Mbytes Free % field of the PD_FSL or PD_FSR record) falls below the threshold, you might need to take appropriate action, such as deleting unnecessary files or adding a disk.

For details, see *3.2.3(1) Monitoring template*.

(b) Monitoring the disk I/O delay status

You can use the I/O Wait Time alarm provided by the monitoring template to monitor the disk I/O delay.

The alarm includes the Wait % field (of the PI record), with which you can monitor the disk I/O delay status. If the value of this field is large, you might need to take appropriate action, such as checking for a process that is performing too many I/O operations to update a database, for example.

For details, see *3.2.3(1) Monitoring template*.

(c) Monitoring the disk I/O status

You can use the Disk Service Time alarm provided by the monitoring template to monitor the disk I/O.

The alarm includes the Avg Service Time field (of the PI_DEVD record), which enables you to check for a process that is inputting or outputting a very large amount of information.

For details, see *3.2.3(1) Monitoring template*.

(d) Monitoring the disk busy rate

You can use the Disk Busy % alarm provided by the monitoring template to monitor the disk busy rate.

You can monitor the disk busy rate by using an alarm to check whether excessive paging (reading and writing of pages by processes) is occurring.

If the disk busy rate (the `Busy %` field of the `PI_DEVD` record) continues to be at or above the threshold, you might need to take action. For example, you might need to identify the processes that frequently request disk I/O operations, and then distribute the processing of these processes.

When you monitor the disk busy rate, monitoring the disk I/O delay status, disk I/O status, and disk congestion is also recommended.

For details, see *3.2.3(1) Monitoring template*.

(e) Monitoring disk congestion

You can use the Disk Queue alarm provided by the monitoring template to monitor disk congestion.

You can monitor disk congestion by using an alarm to check whether I/O requests have been excessive.

If the disk congestion level (the `Queue Length` field of the `PI_DEVD` record) continues to be at or above the threshold, you might need to take action. For example, you might need to identify those processes that frequently request disk I/O, and then distribute the processing of the processes.

When you monitor disk congestion, monitoring the disk I/O delay status, disk I/O status, and disk busy rate is also recommended.

For details, see *3.2.3(1) Monitoring template*.

3.1.4 Network monitoring examples

This subsection explains how to monitor network performance.

(1) Overview

You can monitor network information to check the response time of system functionality.

Continuous monitoring of network data traffic allows you to plan network reconfiguration or expansion.

The following table lists and describes the principal records and fields related to monitoring network performance.

Table 3–5: Principal fields related to network monitoring

Record	Field	Description (example)
`PI_NIND`	`Pkts Rcvd/sec`	The number of packets received per second. If the value of this field is large, many packets have been received successfully.
`PI_NINS`		
`PI_NIND`	`Pkts Xmitd/sec`	The number of packets sent per second. If the value of this field is large, many packets have been sent successfully.
`PI_NINS`		
`PI_NIND`	`Max Transmission Unit`	The maximum packet size. In an environment in which an MTU is automatically allocated, if the value of this field is large (1500 or more), splitting of sent or received data occurs. If the value of this field is small (below 1500), the number of control signals and blocks increases. This increase could cause a network bottleneck.

(2) Monitoring methods

(a) Monitoring for data traffic that exceeds the NIC bandwidth (the maximum amount of data that can be transferred per unit of time)

You can use the Network Rcvd/sec alarm provided by the monitoring template to monitor the bandwidth of a network interface card.

You can monitor the number of packets sent or received over the network by using an alarm to monitor the bandwidth of a network interface card (NIC).

If the number of packets continues to be at or above the threshold, you might need to upgrade the NIC or the physical network.

For details, see *3.2.4(1) Monitoring template*.

3.1.5 Process monitoring examples

This subsection explains how to monitor process performance.

(1) Overview

Because system functionality is provided by individual processes, understanding the operating status of processes is essential for stable system operation.

If one of the processes that provides system functionality terminates abnormally, the system stops with serious consequences. In order to detect such an abnormal condition early and take appropriate action, it is necessary to monitor the status of processes, including their generation and disappearance.

Note that PFM - Agent for Platform performs a process check at the same intervals that information is collected. Accordingly, the time that the disappearance of a process is detected is the time that PFM - Agent for Platform collects information, not the actual time that the process disappeared.

The following table lists and describes the principal records and fields related to the monitoring of processes.

Table 3–6: Principal fields related to the monitoring of processes

Record	Field	Description (example)
PI_WGRP	Process Count	The number of processes. If the value of this field is the threshold or less (the minimum number of processes that need to be activated), some or all of the required processes are not running.#
PD_PDI	Program	The name of a process. If this record is not collected, the process is not running.
PD_APS	Program Name	The name of a process. If this record is not collected, the process has stopped.
PD_APP, PD_APP2	Application Name	The name of an application definition
	Application Exist	A status of the applications. NORMAL indicates that the status of any one of the monitored targets is NORMAL. ABNORMAL indicates that the status of all the monitored targets is ABNORMAL.
	Application Status	A status of the applications. NORMAL indicates that the status of all the monitored targets is NORMAL. ABNORMAL indicates that the status of any one of the monitored targets is ABNORMAL.

Record	Field	Description (example)
PD_APPD	Application Name	Conditional results on the number of monitors. If the value of the Monitoring Status field is ABNORMAL, the number of running programs or command lines is not within the specified range.
	Monitoring Label	
	Monitoring Status	

#

The /opt/jp1pc/agtu/agent/wgfile file must be setup to collect this record.

(2) Monitoring methods

(a) Monitoring process disappearance

You can use the Process End alarm provided by the monitoring template to monitor process disappearance.

If a process terminates abnormally, the system stops with serious consequences. You can monitor the disappearance of processes by using an alarm, enabling prompt recovery of the system.

For details, see *3.2.5(1) Monitoring template*.

(b) Monitoring process generation

You can use the Process Alive alarm provided by the monitoring template to monitor process generation.

You can use an alarm to monitor the generation of processes for each application or the status of scheduled processes, enabling you to check the operation status of the production system.

By setting a workgroup in the wgfile file and using the PI_WGRP record, you can perform several types of monitoring. For example, you can monitor the following items: process generation, process disappearance, the number of processes that have the same name, the number of processes for each application, and the number of processes activated for each user.

For details, see *3.2.5(1) Monitoring template*.

3.1.6 Examples of collecting information about used ports

PFM - Agent for Platform provides functionality to convert user-specific performance data output by users to text files (user-created data) into a format that can be stored in records provided by PFM - Agent for Platform (user data files). For details about user-specific performance data, see *7.4 Settings for collecting user-specific performance data*.

The following shows an example for collecting used port information in PI_UPIB records as user-specific performance data. The following table shows the format in which used port information is stored.

Option	Value
tt	"TCP"
ks	The host name
lr	The total number of TCP ports for the host
lr	The number of currently active TCP ports for the host
lr	The number of listening TCP ports for the host

To collect information:

1. Create a shell script for collecting information about used ports.

   In this example, a shell script is used to collect information about used ports. The following shows an example of creating a shell script.

   Example of creating a shell script in Linux (`/homework/sample.sh`):

```
#!/bin/sh
echo "Product Name=PFM-Agent for Platform (UNIX)" > /homework/userdata.tcp
echo "FormVer=0001" >> /homework/userdata.tcp
echo "tt ks lr lr lr" >> /homework/userdata.tcp
#All TCP port
ALL_TCP=`netstat -at | wc -l`
ALL_TCP=`expr $ALL_TCP - 2`
#Active TCP port
ACTIVE_TCP=`netstat -at | grep ESTABLISHED | wc -l`
#Listen TCP port
LISTEN_TCP=`netstat -at | grep LISTEN | wc -l`
#Output
echo "TCP `uname -n` $ALL_TCP $ACTIVE_TCP $LISTEN_TCP" >> /homework/userda
ta.tcp
```

   Important

   Note that the example shell script shown here was created for Linux. This shell script might not operate correctly in other OSs, and might not always operate correctly in Linux due to differences in environments.

2. Execute the shell script created in step 1.

   The following shows the user-created data created as a result of executing the shell script.

   User-created data (`/homework/userdata.tcp`):

```
Product Name=PFM-Agent for Platform (UNIX)
FormVer=0001
tt ks lr lr lr
TCP jp1ps05 15 3 12
```

3. Convert the user-created data created in step 2 to a user data file.

   The following shows an example of executing the `jpcuser` command to convert user-created data into a user data file.

   Example of `jpcuser` command execution:

```
/opt/jp1pc/agtu/agent/jpcuser/jpcuser PI_UPIB
-file /homework/userdata.tcp
```

4. Use PFM - Agent for Platform to collect the user data file output in step 3.

   When PFM - Agent for Platform collects records, the contents of the user data file are stored in user records.

3.1.7 Examples of collecting performance data from multiple hosts on which PFM products are not installed

You can use the user-created data collection functionality provided by PFM - Agent for Platform to collect performance data specific to hosts on which PFM products are not installed. You can also monitor the status of multiple hosts at the same time by converting the performance data for the hosts into a single user data file. In this case, a script such as a shell script needs to be prepared because user-created data will be created on each host on which PFM products have not been installed. The following shows an example for collecting performance data from hosts on which PFM products are not installed, and outputting the data as PFM - Agent for Platform record information.

(1) Collected data

The following example obtains information using the user-created data created in *3.1.6 Examples of collecting information about used ports*.

(2) Prerequisites

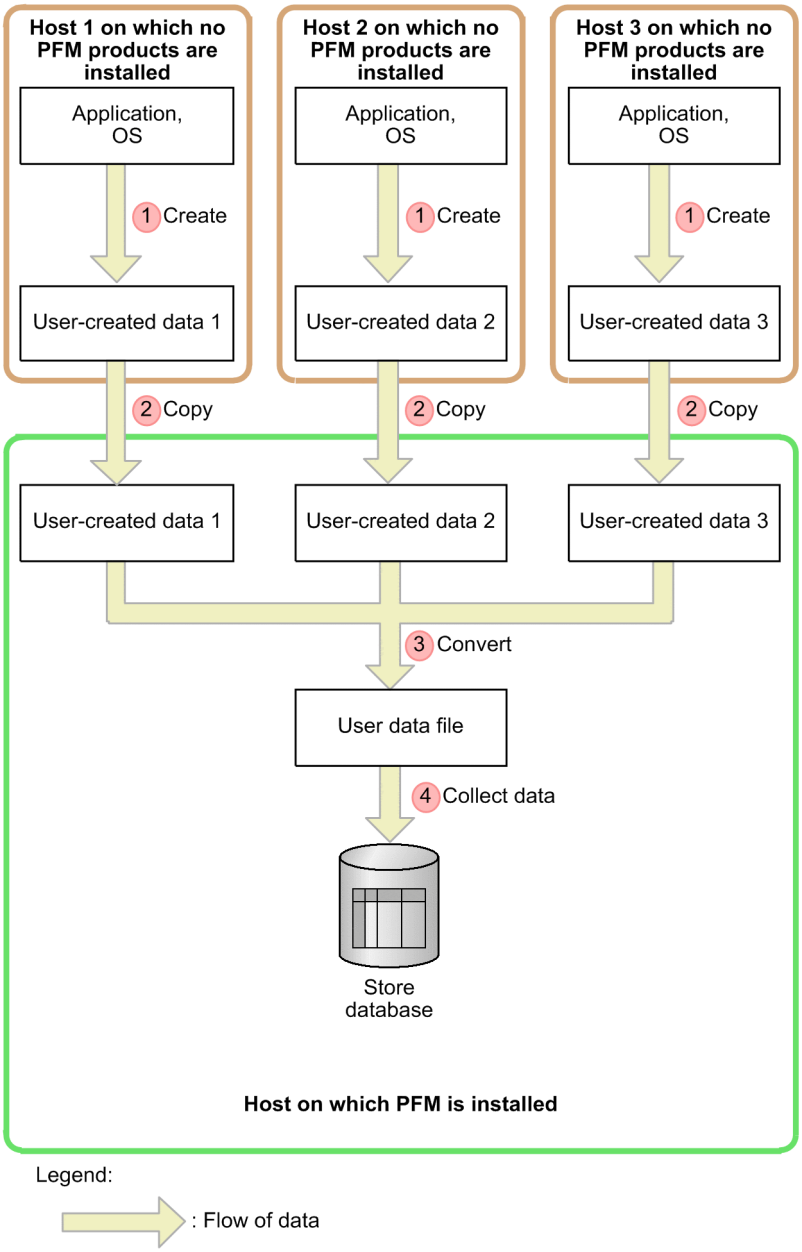
The prerequisites for collecting performance data from multiple hosts on which PFM products are not installed are as follows:

- The hosts on which PFM products are installed and the hosts on which PFM products are not installed are connected in a trusted environment to enable the exchange of files (an NFS area shared by the hosts is set).
- The version of PFM - Agent for Platform on the hosts on which PFM products are installed is 08-11 or later.

(3) Procedures for data collection

The following figure shows the flow of data collection for hosts on which PFM products are not installed.

Figure 3–5: Flow of data collection for hosts on which PFM products are not installed



The following uses the numbering in the figure to explain processing. To collect performance data from multiple hosts, perform these steps for each host.

1. Create user-created data for hosts on which PFM products are not installed.

   Execute the script to collect performance data, and generate user-created data. The user-created data generated in *3.1.6 Examples of collecting information about used ports* is used here.

2. Copy files between remote hosts.

   Copy the user-created data created in step 1 to the hosts on which PFM products are installed. Here, user-created data is copied to the `/nfshome/` area shared between hosts by using NFS. The following shows an example of the `cp` command.

   Example of the `cp` command:

```
# /homework/sample.sh
# cp /homework/userdata.tcp /nfshome/userdata.tcp
```

> **Important**
>
> When collecting user-created data from multiple hosts, make sure that the file names are unique. If file names are duplicated, files might be overwritten during file copying.

3. Execute the `jpcuser` command on hosts on which PFM products are installed.

   Execute the `jpcuser` command on hosts on which PFM products are installed to convert the user-created data copied in step 2 to user data files. The following shows an example in which the user-created data obtained through steps 1 and 2 from the hosts on which PFM products are not installed is converted into a single user data file.

   Example of the `jpcuser` command:

```
/opt/jp1pc/agtu/agent/jpcuser/jpcuser PI_UPIB
-file user-created-data-1 -file user-created-data-2 -file user-created-dat
a-3
```

4. Collect record data for hosts on which PFM products are installed.

   On the hosts on which PFM products are installed, collect the contents of the user data file output in step 3 as record data.

3.2 Performance monitoring definition examples

The following provides examples of definitions for the monitoring template and definitions for items not included in the monitoring template for each monitored resource. The following notes apply to reading the definition examples:

- In the examples, the PFM - Web Console check boxes are shown as follows:
  ☒ (selected) and ☐ (not selected)
- In the examples, the PFM - Web Console radio buttons are shown as follows:
  ◉ (selected) and ○ (not selected)
- In the examples, *xxx*, *yyy*, *zzz*, and *dummy* are variables that the user replaces with the character strings appropriate for the system environment. For other definition items, the values should be changed as required.
- In the examples, the proper values for the frequency of occurrence settings (for example, *m* **occurrence(s) during** *n* **interval(s)**) differ depending on the system environment. Accordingly, specify the appropriate values. For example, assume that the status whose threshold has been exceeded for at least two minutes in the system environment is the high-load status. Further assume that the collection interval is 60 seconds and that the maximum for the number of times that the threshold can be exceeded is twice per five intervals. Under these conditions, an unacceptable high-load condition occurs when the threshold is exceeded at least three times per five intervals. The setting in this case is `3` **occurrence(s) during** `5` **interval(s)**.

3.2.1 Processor monitoring definition examples

This subsection describes examples of definitions for processor monitoring.

(1) Monitoring template

(a) Processor-related monitoring template alarms

Processor-related alarms in the monitoring template are stored in the alarm table for PFM UNIX Template Alarms [CPU] 09.00.

Table 3–7: Processor-related monitoring template alarms

Monitoring template alarm	Record	Field	Error threshold	Warning threshold	Description
Kernel CPU	`PI`	`Kernel CPU %`	> 75	> 50	If the percentage of time during which the processor operates continues to be above the threshold, there might be a problem with the OS or the operation method.#
User CPU	`PI`	`User CPU %`	> 85	> 65	If the percentage of time during which the processor operates continues to be above the threshold, there might be a problem with a specific application.#
Run Queue	`PI`	`5-Minute Run Queue Avg`	> 8	> 4	If the average number of threads is above the threshold, there might be a problem with the OS, operation method, or a specific application.#
CPU Per Processor(K)	`PI_CPUP`	`Processor ID`	>= 0	>= 0	If the CPU usage continues to be at or above the threshold, there might be a problem with the OS or operation method.
		`System %`	> 75	> 50	

Monitoring template alarm	Record	Field	Error threshold	Warning threshold	Description
CPU Per Processor(U)	PI_CPUP	Processor ID	>= 0	>= 0	If the CPU usage continues to be at or above the threshold, there might be a problem with a specific application.
		User %	> 85	> 65	

#

Any processes that are excessively using the processor must be found, and appropriate action must be taken. If no such processes exist, operations that outstrip the kernel's scheduling capabilities, such as the generation and deletion of many processes in a short time, might be occurring. In such a case, because the system environment is not adequate for the processing, you might need to upgrade the processor or add processors.

If you want to perform more detailed monitoring of processor performance you can use your own alarms or reports in addition to the existing alarms. For details about how to create your own alarms or reports, see *3.1.1(2) Monitoring methods*.

For details about the settings for the existing alarms, see *8. Monitoring Templates*.

(b) Processor-related monitoring template reports

Table 3–8: Processor monitoring template reports

Report name	Displayed information
CPU Per Processor Status	Real-time information about the processor status
CPU Per Processor Usage	Real-time information about the CPU usage for a processor
CPU Trend	Daily CPU usage for a host for the last month
CPU Usage Summary	A history of the CPU usage on a minute-by-minute basis for the last hour
CPU Status(Multi-Agent)	An hourly history of the CPU usage by multiple hosts for the last 24 hours
CPU Status	Real-time information about the CPU usage
CPU Trend(Multi-Agent)	A daily history of the CPU usage by multiple hosts for the last month

For details about settings for existing reports, see *8. Monitoring Templates*.

(2) Definition examples other than for monitoring templates

(a) Real-time report for checking processes whose processor usage is high

Table 3–9: Definition example

Item			Explanation
Name and Type	**Report name**		PD_PDI - Memory
	Product		UNIX (6.0)
	Report type	○ **Real-time (single agent)**	◉ (Select)
		○ **Historical (single agent)**	--
		○ **Historical (multiple agents)**	--
Field	**Record**		PD_PDI

Item			Explanation

Field	**Selected fields**		`Program`
			`PID`
			`CPU %`
			`System CPU`
			`User CPU`

Filter	**Conditional expression:**		⦿ (Select **Simple** or **Complex**.) PID <> "0"
	Specify when displayed		☐ (Clear)
Indication settings	☐ **Specify when displayed**		☒ (Select)
	☐ **Indicate delta value**		☐ (Clear)
	Refresh interval	☐ **Do not refresh automatically**	☐ (Clear)
		Initial value	30
		Minimum value	30
	Display by ranking	**Field**	`CPU %`
		Display number	10#
		☐ **In descending order**	☐ (Clear)
Components	**Table**		All fields
	List		--
	Graph		`System CPU` `User CPU`
	Display key	**Field**	(None)
		In descending order	--
Graph	**Graph type**		Stacked bar graph
	Series direction		Row
	Axis labels	**X-axis**	Process name (process ID)
		Y-axis	CPU usage time
	Data label	**Data label 1**	Process name
		Data label 2	Process ID
Drilldown	**Report drilldown**		Arbitrary
	Field drilldown		Arbitrary

Legend:

--: Do not specify this item.

#

Specify a value appropriate for the circumstances.

3.2.2 Memory monitoring definition examples

This subsection describes examples of definitions for memory monitoring.

(1) Monitoring template

(a) Memory-related monitoring template alarms

Memory-related alarms in the monitoring template are stored in the alarm table for PFM UNIX Template Alarms [MEM] 09.00.

Table 3–10: Memory monitoring template alarms

Monitoring template alarm	Record	Field	Error threshold	Warning threshold	Description
Pagescans	PI	Page Scans/sec	> 150	> 100	If the number of page scans that occurred is greater than the threshold, memory might be insufficient.
Swap Outs	PI	Swapped-Out Pages/sec	> 200	> 100	If the number of pages is greater than the threshold, memory might be insufficient.
Alloc Mem Mbytes	PI	Alloc Swap Mbytes	>= 1800	>= 1024	If the amount of used memory continues to be at or above the threshold (the Total Physical Mem Mbytes field value of the PI record), physical memory might be insufficient.

For details about the settings for the existing alarms, see *8. Monitoring Templates*.

(b) Memory-related monitoring template reports

Table 3–11: Memory monitoring template reports

Report name	Displayed information
Memory Paging	A history of memory use on a minute-by-minute basis for the last hour#
Memory Paging (8.4)	
Memory Paging Status	Real-time information about memory usage and the paging status#
Memory Paging Status (8.4)	
Memory Paging Status (Multi-Agent)	An hourly history of memory usage by multiple hosts for the last 24 hours#
Memory Paging Status (Multi-Agent) (8.4)	
Paging Trend (Multi-Agent)	A daily history of page scans performed by multiple hosts for the last month#
Paging Trend (Multi-Agent) (8.4)	

#

This report cannot be used in Linux.

(c) System-related monitoring template reports (for memory)

Table 3–12: System monitoring template reports

Report name	Displayed information
I/O Overview	A history of the number of I/Os on a minute-by-minute basis for the last hour. This report cannot be used in Linux.
Process Trend	A daily history of the number of operated processes for the last month
System Overview (real-time report indicating the system operating status)	Real-time information about the system operating status
System Overview (historical report indicating the system operating status)	A history of the system operating status on a minute-by-minute basis for the last hour
System Overview (8.4) (historical report indicating the system operating status)	
System Utilization Status	Real-time information about the system operating status
Workload Status	Real-time information about the system workload
Workload Status (Multi-Agent)	An hourly history of the workload for multiple hosts for the last 24 hours

For details about the settings for the existing reports, see *8. Monitoring Templates*.

(2) Definition example other than for monitoring templates

(a) Real-time report for checking the memory usage of a process

Table 3–13: Definition example

Item			Explanation
Name and Type	**Report name**		`PD_PDI - Memory`
	Product		`UNIX (6.0)`
	Report type	○ **Real-time (single agent)**	◉ (Select)
		○ **Historical (single agent)**	--
		○ **Historical (multiple agents)**	--
Field	**Record**		`PD_PDI`
	Selected fields		Select all fields.
Filter	**Conditional expression:**		◉ (Select **Simple** or **Complex**.) `PID <> "0"`
	Specify when displayed		☐ (Clear)
Indication settings	☐ **Specify when displayed**		☒ (Select)
	☐ **Indicate delta value**		☐ (Clear)
	Refresh interval	☐ **Do not refresh automatically**	☐ (Clear)

Item			Explanation
Indication settings	**Refresh interval**	**Initial value**	30
		Minimum value	30
	Display by ranking	**Field**	`Virtual Mem Kbytes`#1
		Display number	30#2
		☐ **In descending order**	☒ (Select)

Components	**Table**		`Program`
			`PID`
			`Real Mem Kbytes`
			`Virtual Mem Kbytes`
			`Major Faults`
			`Swaps`
			`Context Switches`
			`CPU %`

	List		--
	Graph		`Virtual Mem Kbytes` `Real Mem Kbytes`
	Display name		--
	Display key	**Field**	(None)
		In descending order	--
Graph	**Graph type**		Line graph
	Series direction		Row
	Axis labels	**X-axis**	Time
		Y-axis	Memory usage
	Data label	**Data label 1**	(None)
		Data label 2	(None)
Drilldown	**Report drilldown**		Arbitrary
	Field drilldown		Arbitrary

Legend:

--: Do not specify this item.

#1

Set the fields that you want to monitor.

#2

Specify a value appropriate for the circumstances.

3.2.3 Disk monitoring definition examples

This subsection describes examples of definitions for disk monitoring.

(1) Monitoring template

(a) Disk-related monitoring template alarms

Disk-related alarms in the monitoring template are stored in the alarm table for PFM UNIX Template Alarms [DSK] 09.00.

Table 3–14: Disk monitoring template alarms

Monitoring template alarms	Record	Field	Error threshold	Warning threshold	Description
I/O Wait Time	PI	Wait %	> 80	> 60	If the percentage of time during which the processor is waiting for I/O is greater than the threshold, I/O operations, such as updating a database, might be delayed.
Disk Service Time	PI_DEVD	Avg Service Time	> 0.1	> 0.06	If the average operating time is greater than the threshold, the amount of information being input or output might be very large.
File System Free(L)	PD_FSL	File System	<> dummy	<> dummy	If there is little unused area, disk space is insufficient.
		Mbytes Free	< 5120	< 10240	
File System Free(R)	PD_FSR	File System	<> dummy	<> dummy	If there is little unused area, disk space is insufficient.
		Mbytes Free	< 5120	< 10240	
Disk Busy %	PI_DEVD	Device Name	<> dummy	<> dummy	If the disk busy rate continues to be at or above the threshold, I/O operations might be concentrated on a specific disk.
		Busy %	>= 90	>= 80	
Disk Queue	PI_DEVD	Device Name	<> dummy	<> dummy	If the queue length continues to be at or above the threshold, the device is congested.
		Queue Length	>= 5	>= 3	

For details about the settings for the existing alarms, see *8. Monitoring Templates*.

(b) Disk-related monitoring template reports

Table 3–15: Disk monitoring template reports

Report name	Displayed information
Avg Service Time - Top 10 Devices	Real-time information about the 10 devices with the longest average operating time
Avg Service Time Status - Top 10 Devices	Real-time information about the 10 devices with the longest average operating time
Device Detail	Real-time information about the selected device
Device Usage Status	Real-time information about device usage
Device Usage Status (Multi-Agent)	An hourly history of device usage by multiple hosts for the last 24 hours
Free Space Mbytes - Top 10 Local File Systems	Real-time information about the 10 local file systems with the smallest amount of free space
Local File System Detail	Real-time information about the selected local file system
Remote File System Detail	Real-time information about the selected remote file system
Space Usage - Top 10 Local File Systems	Real-time information about the 10 local file systems with the highest usage

Report name	Displayed information
Space Usage - Top 10 Remote File Systems	Real-time information about the 10 remote file systems with the highest usage
NFS Activity Overview	A history of the operating status of the NFS clients and server on a minute-by-minute basis for the last hour#
NFS Load Trend	A daily history of the operating status of the NFS clients and server for the last month#
NFS Usage Status	Real-time information about the operating status of the NFS clients and server#
NFS Usage Status (Multi-Agent)	An hourly history of the operating status of the NFS clients and server for the last 24 hours#

#

This report cannot be used in Linux.

For details about the settings for the existing reports, see *8. Monitoring Templates*.

3.2.4 Network monitoring definition examples

This subsection describes examples of definitions for network monitoring.

(1) Monitoring template

(a) Network-related monitoring template alarms

Network-related alarms in the monitoring template are stored in the alarm table for PFM UNIX Template Alarms [NET] 09.00.

Table 3–16: Network monitoring template alarms

Monitoring template alarm	Record used	Field used	Abnormal condition	Warning condition	Meaning
Network Rcvd/sec	`PI_NINS`	`Pkts Rcvd/sec`	>= 9	>= 8	If the number of packets is large, many packets have been received successfully.

For details about the settings for the existing alarms, see *8. Monitoring Templates*.

(b) Network-related monitoring template reports

Table 3–17: Network monitoring template reports

Report name	Displayed information
Network Interface Detail	Real-time information about network usage of the selected system
Network Interface Summary (real-time report indicating the network usage)	Real-time information about network usage
Network Interface Summary (historical report indicating the network usage)	A history of network usage on a minute-by-minute basis for the last hour
Network Overview	A history of network usage on a minute-by-minute basis for the last hour
Network Status (Multi-Agent)	An hourly history of the network usage by multiple hosts for the last 24 hours
Network Status	Real-time information about network usage

For details about the settings for the existing reports, see *8. Monitoring Templates*.

3.2.5 Process monitoring definition examples

This subsection describes examples of definitions for process monitoring.

(1) Monitoring template

(a) Process-related monitoring template alarms

Process-related monitoring template alarms are stored in the alarm table for PFM UNIX Template Alarms [PS] 09.10.

Table 3–18: Process monitoring template alarms

Monitoring template alarms	Record used	Field used	Abnormal condition	Warning condition	Meaning
Process End	PD_PDI	Program	= jpcsto	= jpcsto	If performance data is not collected, this indicates that the process has stopped.
Process Alive	PI_WGRP	Process Count	> 0	> 0	This indicates that the workgroup process is running.
		Workgroup	= workgroup	= workgroup	
Application Status	PD_APP2	Application Name	= *	= *	If the value of the Application Exist field is ABNORMAL, this indicates that all the monitored targets set for the application are in ABNORMAL status. If the value of the Application Status field is ABNORMAL, this indicates that one of the monitored targets set for the application is in ABNORMAL status.
		Application Exist	= ABNORMAL	= NORMAL	
		Application Status	= ABNORMAL	= ABNORMAL	
Process Existence	PD_APS	Program Name	= jpcsto	= jpcsto	If this record is not collected, this indicates that the process has stopped.

For details about the settings for the existing alarms, see *8. Monitoring Templates*.

(b) Process-related monitoring template reports

Table 3–19: Process monitoring template reports

Report name	Displayed information
CPU Usage - Top 10 Processes	Real-time information about the 10 processes with the highest CPU usage
I/O Activity - Top 10 Processes	Real-time information about the 10 processes that processed the most I/O operations. This report cannot be used in HP-UX, AIX, and Linux.
Major Page Faults - Top 10 Processes	Real-time information about the 10 processes with the most page faults causing physical I/O operations
Process Detail	Real-time information about the processes on the selected host
Process Overview	A history of the process operating status on a minute-by-minute basis for the last hour
Process Summary Status	Real-time information about the process operating status

For details about the settings for the existing reports, see *8. Monitoring Templates*.

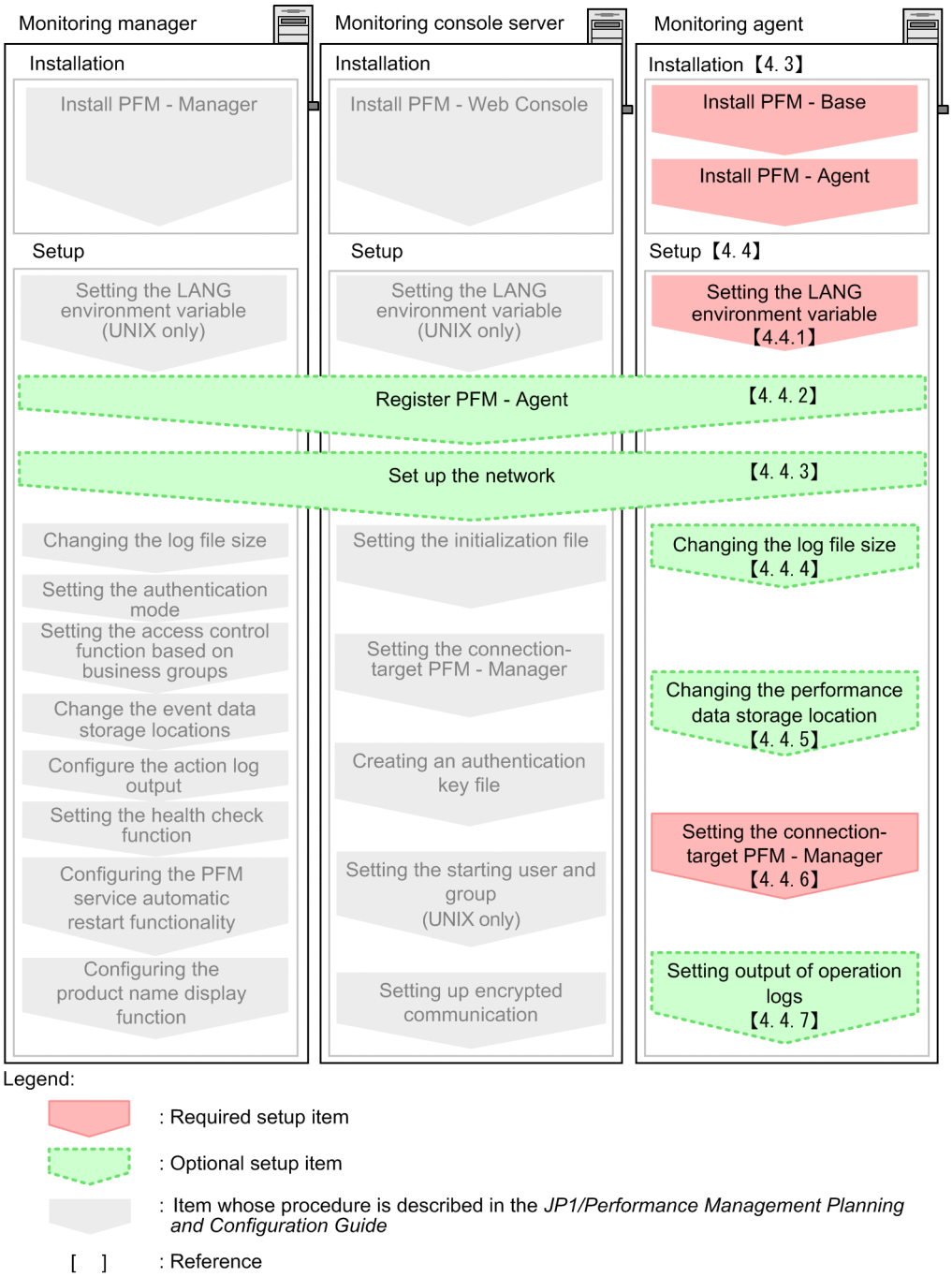
4 Installation and Setup in UNIX

This chapter explains how to install and set up PFM - Agent for Platform. For details about how to install and set up the entire Performance Management system, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

4.1 Installation and setup workflow

This section explains the flow for installing and setting up PFM - Agent for Platform.

Figure 4–1: Installation and setup flow



For details about how to install and set up PFM - Manager and PFM - Web Console, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Note that setup commands that require information to be entered by the user can be selected to run interactively or non-interactively.

When a command is run interactively, a prompt is issued to the user requesting entry of a required value.

When a command is run non-interactively, the user is not prompted. Instead, the input information required for execution of the command is provided by means of option specifications and definition files. By automating setup tasks through batch processing and remote execution, you can reduce the administrative burden and operational costs.

For details about commands, see the manual *JP1/Performance Management Reference*.

4.2 Preparations required before installation

4.2.1 Required OS

PFM - Agent for Platform can run on the following OSs:

- HP-UX
- Solaris
- AIX
- Linux

4.2.2 Network environment setup

The following explains the network environment that is necessary for Performance Management to operate.

(1) IP address setup

You must set up at the PFM - Agent host an environment in which IP address can be resolved from host names. PFM - Agent cannot start in an environment in which IP addresses cannot be resolved.

Actual host names or alias names can be used for a monitoring host name (the name used as the host name of a Performance Management system).

- When using an actual host name as a monitoring host name
  Execute the `hostname` command on a Windows system or `uname -n` command on a UNIX system to check the host name, and set up the environment so that it can resolve IP addresses. Note that on a UNIX system, the host name obtained by the `hostname` command can also be used.
- When using an alias name as a monitoring host name
  Set up the environment so that it can resolve the IP addresses for the set alias name.

For details about monitoring host name settings, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide.*

Use one of the following methods to set up the host name and IP address:

- The host information settings file for Performance Management (`jpchosts` file)
- `hosts` file
- DNS

Important

- Performance Management can be run on a DNS environment, but does not support host names in the FQDN format. Accordingly, leave out the domain name when setting monitoring host names.
- If PFM - Agent for Platform is to be used in multiple LAN environments, set up the IP address in the `jpchosts` file. For details, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

- Performance Management cannot be run on a host on which DHCP is used to assign dynamic IP addresses. Set fixed IP addresses on all hosts on which Performance Management is installed.

Performance Management supports IPv6 network configurations in addition to IPv4 environments. Performance Management also works in network configurations that are made up of a mix of IPv4 and IPv6 environments.

PFM - Agent for Platform can communicate with PFM - Manager over IPv6, but only if both are hosted on Windows or Linux.

For details about and limitations applicable to communication in the IPv4 and IPv6 environments, see *L. Communication in IPv4 and IPv6 Environments*.

To communicate over IPv6, you must enable IPv6 on the PFM - Manager host, as well as on each PFM - Agent host. Before installing PFM - Agent for Platform, you must also enable the use of IPv6 on the PFM - Agent host. To configure this setting, execute the `jpcconf ipv6 enable` command. If the use of IPv6 is already enabled, there is no need to configure this setting. To check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command.

For details about the `jpcconf ipv6 enable` and `jpcconf ipv6 display` commands, see the chapter that describes commands in the manual *JP1/Performance Management Reference*. For details about the timing and conditions for executing the `jpcconf ipv6 enable` command, see the chapter that describes network configurations that include IPv6 environments in the *JP1/Performance Management Planning and Configuration Guide*.

Also note that if you want PFM - Agent for Platform to communicate with PFM - Manager over IPv6, you must specify host names that can be resolved.

Communication between PFM - Agent for Platform and PFM - Manager is established using resolvable IP addresses. When PFM - Agent for Platform and PFM - Manager communicate in an environment in which IPv4 and IPv6 coexist and an attempt to communicate with one resolvable IP address fails, no attempt will be made to communicate using the other IP address.

For example, when an IPv4 connection fails, a retry over IPv6 will not be attempted. Similarly, when an IPv6 connection fails, a retry over IPv4 will not be attempted. We recommend to confirm connections in advance, if possible.

(2) Port number setup

The default port numbers shown in the table below are assigned to Performance Management program services. For any other services or programs, port numbers currently not in use in the system are automatically assigned when the service or program is started. When you are using Performance Management in an environment with a firewall, assign fixed port numbers. For details about how to fix port numbers, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Table 4–1: Default port numbers and Performance Management program services (for UNIX)

Service explanation	Service name	Parameter	Port number	Remarks
Service configuration information management function	Name Server	`jp1pcnsvr`	22285	Port number used by the Name Server service of PFM - Manager. This port number is set for all hosts in Performance Management.
Service status management function	Status Server	`jp1pcstatsvr`	22350	Port number used by the Status Server service of PFM - Manager and PFM - Base. This port number is set for hosts in which PFM - Manager and PFM - Base are installed.

Service explanation	Service name	Parameter	Port number	Remarks
JP1/SLM linkage facility	JP1/ITSLM	--	20905	Port number set in JP1/SLM.

Legend:

--: None

Set up the network to enable these PFM - Agents to communicate using the assigned port numbers.

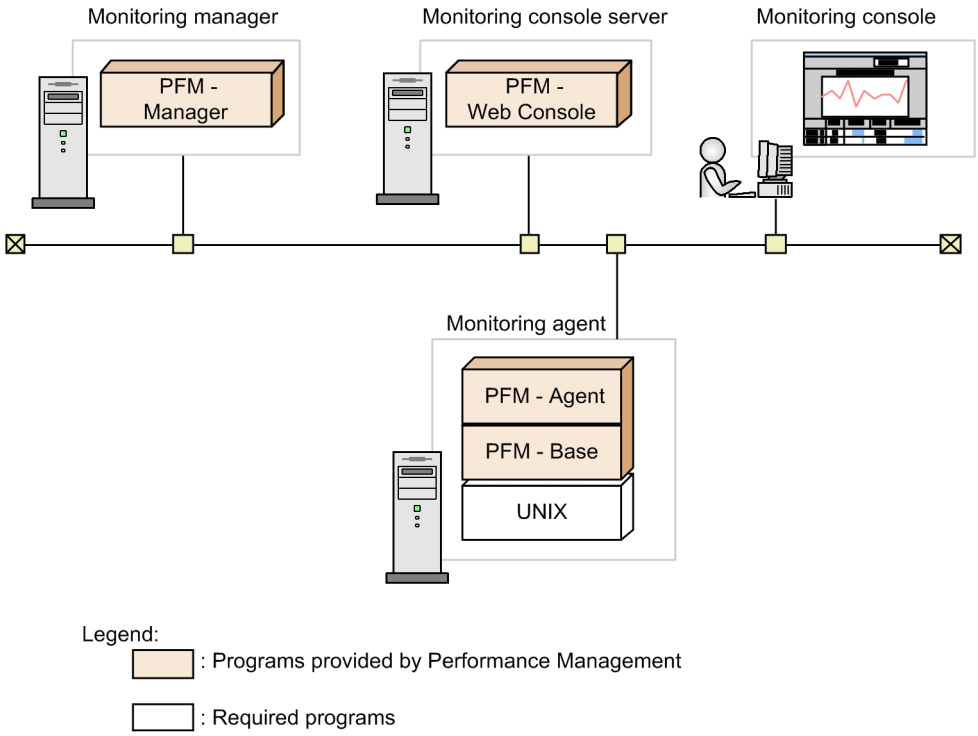
4.2.3 OS user permissions needed for installation

When installing PFM - Agent for Platform, you must execute the installation procedure from an account that has superuser permissions.

4.2.4 Prerequisite programs

This subsection explains the prerequisite programs that are necessary for installing PFM - Agent for Platform. The figure below shows the program configuration.

Figure 4–2: Program configuration diagram



(1) Monitoring target programs

PFM - Agent for Platform monitors the following programs:

- HP-UX
- Solaris
- AIX

- Linux

These monitoring target programs and PFM - Agent for Platform must be installed on the same host.

(2) Performance Management programs

Install PFM - Agent and PFM - Base on the monitoring agent. PFM - Base is a prerequisite program for PFM - Agent. Even if you have multiple copies of PFM - Agent or PFM - RM installed, you need only a single copy of PFM - Base. However, if you install PFM - Manager and PFM - Agent on the same host, PFM - Base is not required. For details about the relationships between Performance Management program versions and the hosts on which they may be installed, see the chapter that describes system configuration version compatibility in the *JP1/Performance Management Planning and Configuration Guide*.

To use PFM - Agent for Platform to monitor operation of a UNIX server, you must also have PFM - Manager and PFM - Web Console.

4.2.5 Prerequisite for specifying 4,096-byte monitoring conditions for processes

Starting with version 10-00 of PFM - Manager and PFM - Web Console, you are able to specify up to 4,096 bytes of monitoring conditions to be used for performance monitoring.

When you install PFM - Base or PFM - Manager on a PFM - Agent host, be sure to install version 10-00 or later.

4.2.6 Preparations for collecting data in the event of a failure

When a problem occurs, you might need to acquire a core dump file to determine the cause of the problem. Output of a core dump file depends on the user's environment settings, so check the following settings before a problem occurs.

Size setting for core dump files

The maximum size of a core dump file is limited by the root user's size setting for a core dump file (`ulimit -c`). Try setting the script as follows:

```
ulimit -c unlimited
```

If this setting violates a security policy for the machine being used, change this script setting into a comment line, as shown below:

```
# ulimit -c unlimited
```

Important

If this setting is specified in a comment line, it might become impossible to investigate the cause of a failure, because core dumps will not be output at times when core dump files are supposed to be output, such as when a segmentation fault or path error occurs in a process.

Kernel parameter settings pertaining to core dumps (Linux only)

If the Linux kernel parameter (`kernel.core_pattern`) is used to change a core dump file's output destination or file name from the default value, it might become impossible to collect core dumps. For this reason, we recommend that you not change the settings of the Linux kernel parameter (`kernel.core_pattern`).

4.2.7 Points to note before installing PFM - Agent for Platform

Note the following points when installing and setting up Performance Management.

(1) Notes on environment variables

Because Performance Management uses `JPC_HOSTNAME` as an environment variable, do not set it as a user-specific environment variable, as this will prevent Performance Management from operating properly.

(2) Notes on installing and setting up multiple Performance Management programs on the same host

With Performance Management, you can install PFM - Manager, PFM - Web Console, and PFM - Agent on the same host. When doing so, note the following points:

- If you are installing PFM - Manager and PFM - Agent on the same host, you do not need PFM - Base. In this case, because the prerequisite program for PFM - Agent is PFM - Manager, install PFM - Manager first and then install PFM - Agent.
- You cannot install PFM - Base and PFM - Manager on the same host. To install PFM - Manager on the host on which PFM - Base and PFM - Agent are installed, first uninstall all Performance Management programs except PFM - Web Console, and then install PFM - Manager, followed by PFM - Agent. To install PFM - Base on the host on which PFM - Manager and PFM - Agent are installed, first uninstall all Performance Management programs except PFM - Web Console, and then install PFM - Base, followed by PFM - Agent.
- When you install PFM - Agent on a host on which PFM - Manager is installed, PFM - Manager of the local host becomes PFM - Manager for the connection destination. In this case, you cannot change PFM - Manager for the connection destination to PFM - Manager of a remote host. To connect to PFM - Manager of a remote host, make sure that PFM - Manager is not installed on the host on which you want to install PFM - Agent.
- Before installing PFM - Agent on a host on which PFM - Web Console is installed, close all browser windows.
- When a Performance Management program is newly installed, the status management function is enabled as the default. However, if you upgrade from version 07-00 to 08-00 or newer, the settings for the status management function remain the same as they were in the older version. For details about how to change the settings for the status management function, see the chapter explaining error detection in Performance Management in the *JP1/Performance Management User's Guide*.

Tip

To improve system performance and reliability, we recommend running PFM - Manager, PFM - Web Console, and PFM - Agent on separate hosts.

(3) Notes on version upgrade

For notes about upgrading the Performance Management programs, see the section that presents notes about upgrading in the chapter describing installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

For notes about upgrading PFM - Agent for Platform, see *H. Upgrade Procedure and Notes on Upgrading*.

For details about upgrading, see the appendix in the *JP1/Performance Management Planning and Configuration Guide*.

(4) Other notes

- When installing a Performance Management program, check whether any of the following security-related programs are installed. If they are installed, perform the measures explained.
  - Security monitoring programs

    Stop or change the settings for security monitoring programs so that installation of the Performance Management program is not prevented.
  - Virus detection programs

    Stop any virus detection programs before you install the Performance Management program

    If a virus detection program is running during installation of the Performance Management program, the installation speed may suffer, and installation might not be executed, or might not be performed correctly.
  - Process monitoring programs

    Stop or change the settings for any process monitoring programs, so that monitoring is not performed for Performance Management services or processes, and Common Component services or processes.

    If a process monitoring program starts or stops these services or processes during installation of the Performance Management program, installation might fail.
- If you install a new Performance Management program in an environment in which no Performance Management program is installed, make sure that there is no file or directory in the installation directory.
- If `Installation failed` appears in the status bar during installation and installation fails, obtain the installation log and store it in a file. You need to back up the log file as needed because this log file is overwritten the next time you install a program. For details about the default file name for the installation log, see *12.4.3 Operation information to be collected*.
- When you install Performance Management programs by setting a link to the installation directory, some files and directories might remain in the linked directory even if you uninstall all of the Performance Management programs. Also, you must be careful when you install Performance Management programs in the linked directory, as files or directories having the same name in the linked directory are overwritten.
- When the PFM - Agent for Platform setup file exists in the `/opt/jp1pc/setup` directory, additional setup of a new PFM - Agent for Platform is executed. When additional setup of the new PFM - Agent for Platform is successful, the execution result message `KAVE05908-I New agent setup ended successfully.` is output to the common message log. Check the log.
- PFM - Agent for Platform supports the disk replication installation functionality of JP1/ServerConductor/Deployment Manager, and also supports the replication functionality of virtualization platforms through conversion to image files.

  For details on disk replication installation, see precautions regarding disk replication installation in the *JP1/Performance Management Planning and Configuration Guide*.

4.3 Installation

This section explains the order in which to install PFM - Agent programs and how to install them from the distribution media.

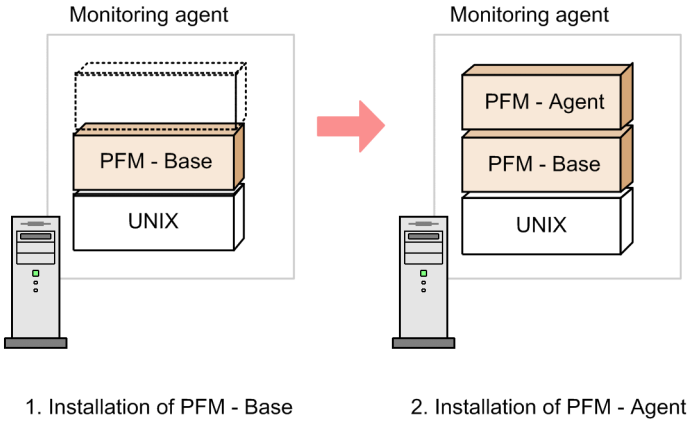
4.3.1 Order in which to install the programs

First install PFM - Base and then install PFM - Agent. You cannot install PFM - Agent on a host on which PFM - Base is not installed.

When installing PFM - Agent on the same host as PFM - Manager, install PFM - Manager first, followed by PFM - Agent. When the Store database version is updated from 1.0 to 2.0, the setup procedure differs depending on the version of PFM - Manager or PFM - Base that is already installed. For details about how to set up version 2.0 of the Store database, see *4.7.2 Updating the Store version to 2.0*.

When installing multiple copies of PFM - Agent on the same host, the order in which the copies are installed does not matter.

Figure 4–3: Order of program installation



4.3.2 Installation procedure for PFM - Agent for Platform

To install Performance Management programs on a UNIX host, you can either use the distribution media or remotely install them using JP1/Software Distribution. For details about how to use JP1/Software Distribution, see the following manuals:

- *Job Management Partner 1/Software Distribution Manager*
- *Job Management Partner 1/Software Distribution SubManager Description and Administrator's Guide*
- *Job Management Partner 1/Software Distribution Client Description and User's Guide* (for UNIX systems)

Important

Depending on the machine environment, directory and file names might differ from those indicated in the manuals. Use the `ls` command to check the correct directory and file names in your environment. When

you execute the command that starts Hitachi PP Installer, specify the directory and file names displayed by the `ls` command.

The following explains the installation procedure for each OS when you use the distribution media.

(1) For HP-UX

To install Performance Management programs using the distribution media:

1. At the host where the programs are to be installed, log in as a superuser, or use the `su` command to change the user to a superuser.

2. Stop any Performance Management services running on the local host.

   You must stop all Performance Management services on physical and logical hosts. For details about how to stop services, see the chapter that describes starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the distribution media into the appropriate drive.

4. Execute the `mount` command to mount the distribution media.

   For example, to mount the distribution media on `/cdrom`, specify and execute the following command:

   ```
   /usr/sbin/mount -F cdfs -r device-special-file-name /cdrom
   ```

   The specified command differs depending on the environment used.

5. Execute the following command to start Hitachi PP Installer:

   ```
   mount-directory/IPFHPUX/setup mount-directory
   ```

   Hitachi PP Installer starts and the initial window appears.

6. Enter `I` in the initial window.

   A list of programs that can be installed appears.

7. Select the programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space key to select programs.

8. After installation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

(2) For Solaris (SPARC)

To install Performance Management programs using the distribution media:

1. As the superuser, log on to the host on which programs are to be installed, or use the `su` command to change the user to superuser.

2. Stop any Performance Management services running on the local host.

   You must stop all Performance Management services on physical and logical hosts. For details about how to stop services, see the chapter that describes starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the distribution media into the appropriate drive.

4. Execute the following command to start Hitachi PP Installer:#

```
mount-directory/SOLARIS/setup mount-directory
```

   Hitachi PP Installer starts and the initial window appears.
   The specified command differs depending on the environment used.

5. Enter `I` in the initial window.
   A list of programs that can be installed appears.

6. Select the programs you wish to install, and enter `I`.
   The selected programs are installed. Note that you can move the cursor and use the space key to select programs.

7. After installation is correctly finished, enter `Q`.
   The initial window of Hitachi PP Installer reappears.

#

In an environment in which the auto mount function has been disabled, specify the `/usr/sbin/mount` command as below and mount the distribution media before starting Hitachi PP Installer:

```
/usr/sbin/mount -F cdfs -r device-special-file-name /cdrom/cdrom
```

The specified command differs depending on the environment used.

(3) For AIX

To install Performance Management programs using the distribution media:

1. As the superuser, log on to the host on which programs are to be installed, or use the `su` command to change the user to superuser.

2. Stop any Performance Management services that are running on the local host.
   You must stop all Performance Management services on physical and logical hosts. For details about how to stop services, see the chapter that describes starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the distribution media into the appropriate drive.

4. Execute the `mount` command to mount the distribution media.
   For example, to mount the distribution media on `/cdrom`, specify and execute the following command:

```
/usr/sbin/mount -r -v cdrfs /dev/cd0 /cdrom
```

5. Execute the following command to start Hitachi PP Installer:

```
mount-directory/AIX/setup mount-directory
```

   Hitachi PP Installer starts and the initial window appears.

6. Enter `I` in the initial window.
   A list of programs that can be installed appears.

7. Select the programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space key to select programs.

8. After installation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

(4) For Linux

To install Performance Management programs using the distribution media:

1. Make sure that the directory in which programs are to be installed is a real directory.

2. As the superuser, log on to the host on which programs are to be installed, or use the `su` command to change the user to superuser.

3. Stop any Performance Management services running on the local host.

   You must stop all Performance Management services on physical and logical hosts. For details about how to stop services, see the chapter that describes starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

4. Insert the distribution media into the appropriate drive.

5. Execute the following command to start Hitachi PP Installer:#

   ```
   mount-directory/X64LIN/setup mount-directory
   ```

   Hitachi PP Installer starts and the initial window appears.

6. Enter `I` in the initial window.

   A list of programs that can be installed appears.

7. Select the programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space key to select programs.

8. After installation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

#

In an environment in which the auto mount function has been disabled, specify the `/bin/mount` command as below and mount the distribution media before starting Hitachi PP Installer:

```
/bin/mount -r -o mode=0544 /dev/cdrom /media/cdrecorder
```

The above specified command name and the underlined parts, which indicate the device special file name and mount-point directory name, vary depending on the environment being used.

4.3.3 Notes on installation

- If installation fails due to insufficient disk space during installation, the temporary working directory is not deleted. When installation fails, delete the aforementioned working directory, check the system status, and then perform installation again.

- If Install failed is displayed in the Hitachi PP Installer interface, and installation fails, collect the `/etc/.hitachi/.hitachi.log` or .install.log file and contact the system administrator. Note that, because hitachi.log is overwritten when a Hitachi PP installer is started and .install.log is overwritten when an installation is performed, you must obtain those files promptly if an installation fails.

4.4 Setup

This section explains the set up procedure needed for using PFM - Agent for Platform.

Options indicates setup items that may be required depending on the usage environment or optional setup items that are used for changing the default settings.

4.4.1 Setting the LANG environment variable

You need to set the LANG environment variable.

Before setting LANG environment variables, you must check that the language environment to be set is correctly installed and configured. If it is not correctly installed and configured, characters may be garbled or definition data may be rewritten inappropriately.

Note:

The language of the common message log is determined by the value set in the LANG environment variable at the time of service startup or command execution. This may result in character strings in multiple language encodings, such as Japanese and English.

The following table shows the LANG environment variable that can be used on PFM - Agent for Platform. If you specify any language other than those shown in the table (such as German, French, Spanish, Korean, or Russian), the system will assume that the LANG environment variable is set to C.

Table 4–2: LANG environment variable that can be used on PFM - Agent for Platform (for AIX)

Language	Character encoding	LANG environment variable value
Japanese	EUC	• ja_JP • ja_JP.IBM-eucJP (Cannot be used with PFM - Manager)
	Shift_JIS (SJIS)	• Ja_JP • Ja_JP.IBM-932
	UTF-8	• JA_JP • JA_JP.UTF-8
English	ASCII	• C
Chinese (simplified characters)	GB18030	• Zh_CN • Zh_CN.GB18030
	UTF-8	• ZH_CN • ZH_CN.UTF-8

Table 4–3: LANG environment variable that can be used on PFM - Agent for Platform (for HP-UX)

Language	Character encoding	LANG environment variable value
Japanese	EUC	• ja_JP.eucJP • Japanese.euc
	Shift_JIS (SJIS)	• ja_JP.SJIS • japanese

Language	Character encoding	LANG environment variable value
Japanese	UTF-8	• ja_JP.utf8
English	ASCII	• C
Chinese (simplified characters)	GB18030	• zh_CN.gb18030
	UTF-8	• zh_CN.utf8

Table 4–4: LANG environment variable that can be used on PFM - Agent for Platform (for Linux)

Language	Character encoding	LANG environment variable value
Japanese	Shift_JIS (SJIS)	• ja_JP.SJIS# • ja_JP.sjis#
	UTF-8	• ja_JP.UTF-8 • ja_JP.utf8
English	ASCII	• C
Chinese (simplified characters)	GB18030	• zh_CN.gb18030
	UTF-8	• zh_CN.UTF-8 • zh_CN.utf8

#: This value can only be used with SUSE Linux.

Table 4–5: LANG environment variable that can be used on PFM - Agent for Platform (for Solaris)

Language	Character encoding	LANG environment variable value
Japanese	EUC	• ja • ja_JP.eucJP# • japanese
	Shift_JIS (SJIS)	ja_JP.PCK
	UTF-8	ja_JP.UTF-8
English	ASCII	C
Chinese (simplified characters)	GB18030	• zh_CN.GB18030 • zh_CN.GB18030@pinyin • zh_CN.GB18030@radical • zh_CN.GB18030@stroke
	UTF-8	• zh_CN.UTF-8 • zh_CN.UTF-8@pinyin • zh_CN.UTF-8@radical • zh_CN.UTF-8@stroke • zh.UTF-8

#: This value can only be used with Solaris 11 (SPARC).

4.4.2 Registering PFM - Agent for Platform in PFM - Manager and PFM - Web Console Options

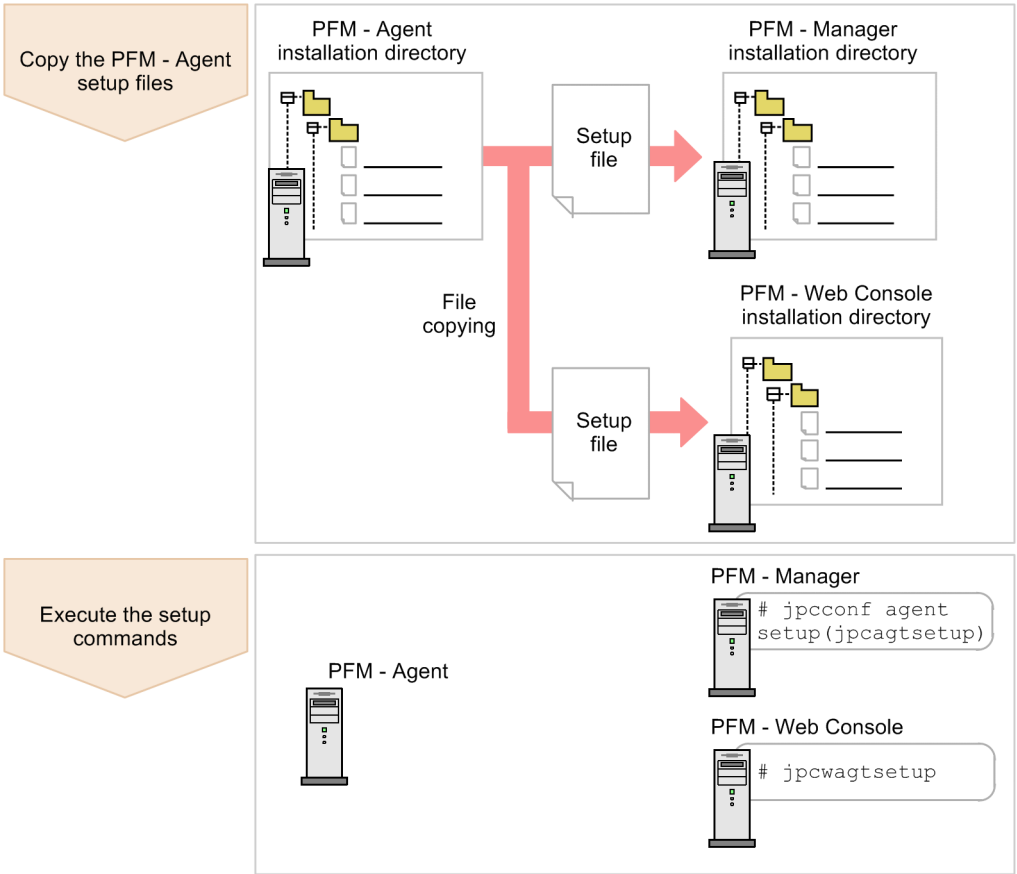
To use PFM - Manager and PFM - Web Console to centrally manage PFM - Agent, you need to register PFM - Agent for Platform in PFM - Manager and PFM - Web Console.

When the version of PFM - Manager is 09-00 or later, you do not need to perform the procedure explained here, because PFM - Agent is automatically registered.

However, you might have to manually register a version of PFM - Agent or PFM - RM that was released after the version of PFM - Manager you are using. For details on whether manual registration is required, see the *Release Notes* for PFM - Manager.

The following figure shows the flow for registering PFM - Agent.

Figure 4–4: PFM - Agent registration flow



Important

- If you add the same version of PFM - Agent for Platform to a Performance Management system on which PFM - Agent for Platform information is already registered, you need not register PFM - Agent.
- If the data model version of PFM - Agent for Platform is 3.0 or 4.0, setup for updating the data model version is not necessary.
- When installing PFM - Agent for Platform of differing versions on different hosts, set up the older version first and then the new version.

- When you install PFM - Agent on the same host as PFM - Manager, the `jpcconf agent setup` command is automatically executed. When the message `KAVE05908-I New agent setup ended successfully.` is output to the common message log, check the result. If the command was not executed correctly, re-execute it. For details about command execution, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(1) Copying the PFM - Agent for Platform setup files

Copy the setup files located at the host on which PFM - Agent for Platform was installed to the host on which PFM - Manager and PFM - Web Console are installed. To copy the setup files:

1. If PFM - Web Console is active, stop it.
2. Copy the PFM - Agent setup files in the binary mode.
   The table below shows the file storage locations and copying destinations.

   Table 4–6: Setup files to be copied

Copy destination			PFM - Agent setup files
PFM program name	OS	Copy destination directory	
PFM - Manager	Windows	*PFM-Manager-installation-folder* `\setup\`	`/opt/jp1pc/setup/jpcagtuw.EXE`
	UNIX	`/opt/jp1pc/setup/`	`/opt/jp1pc/setup/jpcagtuu.Z`
PFM - Web Console	Windows	*PFM-Web-Console-installation-folder* `\setup\`	`/opt/jp1pc/setup/jpcagtuw.EXE`
	UNIX	`/opt/jp1pcwebcon/setup/`	`/opt/jp1pc/setup/jpcagttu.Z`

(2) Executing the setup command on the PFM - Manager host

In PFM - Manager, execute the following command to set up PFM - Agent for Platform:

```
jpcconf agent setup -key UNIX
```

In this example, the `jpcconf agent setup` command is executed interactively, but it can also be run non-interactively. For details about the `jpcconf agent setup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

Important

If you execute the `jpcconf agent setup` command when the Performance Management programs and services of the local host on which the command is being executed have not completely stopped, an error may occur. In this case, make sure the Performance Management programs and services have stopped completely, and then re-execute the `jpcconf agent setup` command.

After this operation is finished, you may delete the PFM - Agent setup files located on the PFM - Manager host.

(3) Executing the setup command on the PFM - Web Console host

In PFM - Web Console, execute the following command to set up PFM - Agent for Platform:

```
jpcwagtsetup
```

After this operation is finished, you may delete the PFM - Agent for Platform setup files located on the PFM - Web Console host.

4.4.3 Network setup Options

Some settings might need to be changed depending on the configuration of the network in which Performance Management is used.

In network setup, you can specify the following two items:

- IP addresses

  Specify IP addresses when Performance Management is to be used in a network connected to multiple LANs. To specify multiple IP addresses, define host names and IP addresses in the `jpchosts` file. The same `jpchosts` file that is set up must be used throughout the entire Performance Management system.

  For details, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

- Port number

  You can specify the port number to be used by Performance Management. To avoid operational confusion, use the same port number and service name throughout the entire Performance Management system.

  For details about port number setup, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

4.4.4 Changing the log file size Options

The operational status of Performance Management is output to Performance Management's own log file. This log file is called the *common message log*. Modify this setting if you want to change the file size.

For details, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

4.4.5 Changing the performance data storage destination Options

This is necessary only if you want to change the storage destination for the database for storing the performance data managed by PFM - Agent for Platform, the backup destination, or the export destination directory.

By default, performance data is stored in the following locations:

- Storage destination: `/opt/jp1pc/agtu/store/`
- Backup destination: `/opt/jp1pc/agtu/store/backup/`
- Partial backup destination: `/opt/jp1pc/agtu/store/partial/`

- Export destination: `/opt/jp1pc/agtu/store/dump/`
- Import destination: `/opt/jp1pc/agtu/store/import/`

For details, see *4.7.1 Changing the storage location of performance data*.

4.4.6 Setting up PFM - Manager for the connection destination of PFM - Agent for Platform

On the host on which PFM - Agent is installed, set up the PFM - Manager that manages that PFM - Agent. To set up PFM - Manager for the connection destination, use the `jpcconf mgrhost define` command.

Important

- Even when multiple copies of PFM - Agent are installed on the same host, you can specify only a single PFM - Manager as the connection destination. You cannot specify a different PFM - Manager for each PFM - Agent.
- When PFM - Agent and PFM - Manager are installed on the same host, PFM - Manager for the connection destination becomes the PFM - Manager of the local host. In this case, you cannot change the PFM - Manager for the connection destination to another PFM - Manager.

To set up PFM - Manager for the connection destination:

1. Stop Performance Management programs and services.

   Before executing a setup operation, stop all Performance Management programs and services on the local host if they are active. For details about how to stop services, see the chapter explaining how to start and stop Performance Management in the *JP1/Performance Management User's Guide*.

   If Performance Management programs and services are active when you attempt to execute the `jpcconf mgrhost define` command, a message is displayed asking the user whether to stop those programs and services.

2. Specify the host name of the PFM - Manager host at the connection destination and execute the `jpcconf mgrhost define` command.

   For example, if the PFM - Managerhost at the connection destination is on `host01`, specify the following:

   ```
   jpcconf mgrhost define -host host01
   ```

   In this example, the `jpcconf mgrhost define` command is executed interactively, but it can also be run non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

4.4.7 Setting up the action log Options

You can log information in the action log when, for example, PFM services start and stop, and the status of the connection to PFM - Manager changes. The action log stores history information that is output in conjunction with the alarms for thresholds related to system load and other conditions.

For details about how to set up the action log, see *J. Outputting Action Log Data*.

4.5 Uninstallation

The following explains how to perform uninstallation and unsetup for PFM - Agent for Platform.

4.5.1 Points to note before uninstalling PFM - Agent for Platform

Note the following points when uninstalling PFM - Agent for Platform.

(1) Notes on the OS user permission required for uninstallation

To uninstall PFM - Agent for Platform, you must do so from an account that has the superuser permission.

(2) Notes on the network

Even when you uninstall Performance Management programs, the port numbers defined in the `services` file are not deleted.

(3) Notes on programs

- If you uninstall PFM - Agent for Platform while another Performance Management program or service is running, or while another program that might reference Performance Management files is running, files and directories of PFM - Agent may still remain. In this case, manually delete all files and directories under the `/opt/jp1pc/agtu` directory.
- If PFM - Base and PFM - Agent are both installed on a host, you cannot uninstall PFM - Base until you uninstall PFM - Agent. In this case, first uninstall PFM - Agent and then PFM - Base. Likewise, if PFM - Manager and PFM - Agent are both installed on a host, you must first uninstall PFM - Agent and then PFM - Manager.
- When uninstallation is performed during the following circumstances, some files or directories might remain. If any residual files remain, go to `/opt/jp1pc` and delete the `instagt7.ini`, the `agt7` directory and its contents.

  If you were using the service in a logical host environment, also go to <*environment-directory*>`/jp1pc` and delete the `agt7` directory and its contents.

  - When a JP1/PFM program or service is running
  - `/etc/hitachi_x64setup` is run while the current directory is in `/opt/jp1pc`.
  - A link is pasted to `/opt/jp1pc`, and a PFM program is installed.
  - Deletion of a logical host environment fails.

(4) Notes on services

If you uninstall only PFM - Agent, the information on the services that can be displayed using the `jpctool service list` command is not deleted. For details about how to delete service information, see the section on service deletion in the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

(5) Other notes

Before uninstalling Performance Management programs from a host on which PFM - Web Console is installed, close all browser windows.

4.5.2 Releasing the connection to PFM - Manager

To release the connection to PFM - Manager, you must delete the service information for the PFM - Agent for Platform system connected to the target PFM - Manager system.

For details about how to delete service information, see the section that describes the service information deletion procedure in the chapter describing installation and setup (for UNIX) in the *JP1/Performance Management Planning and Configuration Guide*.

To change the connection to a different PFM - Manager, see *4.4.6 Setting up PFM - Manager for the connection destination of PFM - Agent for Platform*.

4.5.3 Uninstallation procedure

To uninstall PFM - Agent for Platform:

1. As the superuser, log on to the host from which Performance Management programs are to be uninstalled or use the `su` command to change the user to superuser.

2. Stop Performance Management programs and services on the local host.

   Display service information to check whether any service is active.

   If any Performance Management programs and services are active on the local host, stop them all. For details about how to display service information and to stop services, see the chapter explaining how to start and stop Performance Management in the *JP1/Performance Management User's Guide*.

3. Execute the following command to start Hitachi PP Installer:

   Linux:

   `/etc/hitachi_x64setup`

   Other than Linux:

   `/etc/hitachi_setup`

   Hitachi PP Installer starts and the initial window appears.

4. Enter `D` in the initial window.

   A list of programs that can be uninstalled appears.

5. Select the Performance Management programs you wish to uninstall and enter `D`.

   The selected programs are uninstalled. Note that you can move the cursor and use the space key to select programs.

6. When uninstallation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

4.6 Changing the system configuration of PFM - Agent for Platform

When the network configuration or host name of the monitoring target system changes, it is sometimes necessary to change the PFM - Agent for Platform system configuration as a result. This section explains how to change the PFM - Agent for Platform system configuration.

When you change the PFM - Agent for Platform system configuration, you also need to change the settings for PFM - Manager and PFM - Web Console. For details about how to change the Performance Management system configuration, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*. Note that when a physical host name or alias name is changed, some instances of PFM - Agent will require specific addition operations, but PFM - Agent for Platform will not.

4.7 Changing the operation of PFM - Agent for Platform

A change in the way collected operation monitoring data is used may sometimes change the PFM - Agent for Platform operation method. This section explains how to change the PFM - Agent for Platform operation method. For details about how to change the operation method for the entire Performance Management system, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

4.7.1 Changing the storage location of performance data

Performance data collected by PFM - Agent for Platform is managed in the Store database of the Agent Store service of PFM - Agent for Platform. The following explains how to change the storage location for performance data.

(1) Using the jpcconf db define command to change settings

The Store database uses the following directories to manage the collected performance data. These directories can be changed by using the `jpcconf db define` command. If you need the performance data collected before the Store database storage location is changed, execute the `jpcconf db define` command with the `-move` option specified. For details about the `jpcconf db define` command, see the manual *JP1/Performance Management Reference*.

- Save destination directory
- Backup destination directory
- Partial backup destination directory#
- Export destination directory
- Import destination directory#

#: This directory can be set only when the Store version is 2.0.

The following table describes the options of the `jpcconf db define` command, including the values that can be specified.

Table 4–7: Options of the command that changes the performance data storage location

Item	Option	Specifiable value (Store 1.0)#	Specifiable value (Store 2.0)#	Default
Save destination directory	`sd`	Directory name of 1-127 bytes	Directory name of 1-214 bytes	`/opt/jp1pc/agtu/store`
Backup destination directory	`bd`	Directory name of 1-127 bytes	Directory name of 1-211 bytes	`/opt/jp1pc/agtu/store/backup`
Partial backup destination directory	`pbd`	--	Directory name of 1-214 bytes	`/opt/jp1pc/agtu/store/partial`
Maximum backup generation number	`bs`	1 to 9	1 to 9	5
Export destination directory	`dd`	Directory name of 1-127 bytes	Directory name of 1-127 bytes	`/opt/jp1pc/agtu/store/dump`
Import destination directory	`id`	--	Directory name of 1-222 bytes	`/opt/jp1pc/agtu/store/import`

Legend:

--: This item cannot be set.

#

The directory name must be an absolute path name or a relative path name from the default Store database directory (`/opt/jp1pc/agtu/store`).

(2) Editing the jpcsto.ini file to change settings (for Store version 1.0 only)

If the Store version is 1.0, you can directly edit the `jpcsto.ini` file to change the settings of the above items.

(a) Items to be set in the jpcsto.ini file

The table below shows the label names and setting ranges that can be edited in the `jpcsto.ini` file.

Table 4–8: Setting items in the performance data storage destination ([Data Section] section of jpcsto.ini)

Item	Label name	Specifiable value (Store 1.0)#1	Default
Save destination directory	`Store Dir`#2	Directory name of 1-127 bytes	`/opt/jp1pc/agtu/store`
Backup destination directory	`Backup Dir`#2	Directory name of 1-127 bytes	`/opt/jp1pc/agtu/store/backup`
Maximum backup generation number	`Backup Save`	1 to 9	5
Export destination directory	`Dump Dir`#2	Directory name of 1-127 bytes	`/opt/jp1pc/agtu/store/dump`

#1

- The directory name must be an absolute path name or a relative path name from the default Store database directory (`/opt/jp1pc/agtu/store`).
- Characters that can be specified are alphanumeric characters, symbols, and spaces, excluding the characters listed below:

  `;`, `,`, `*`, `?`, `'`, `"`, `<`, `>`, `|`
- If the specified value is invalid, the Agent Store service cannot start.

#2

You cannot specify the same directories for `Store Dir`, `Backup Dir`, and `Dump Dir`.

(b) Before editing the jpcsto.ini file

- When changing the Store database directory, make sure that the directory after the change has already been created.
- When the Store database directory is changed, performance data collected before the change can no longer be used. If the performance data collected before the change is needed, inherit the data over as follows:
  1. Use the `jpctool db backup` command to back up the performance data stored in the Store database.
  2. Change the Store database directory as described in *(c) Editing the jpcsto.ini file*.
  3. Use the `jpctool db restore` command to restore the backed up data into the new directory.

(c) Editing the jpcsto.ini file

To edit the `jpcsto.ini` file:

1. Stop the PFM - Agent service.
   If PFM - Agent programs and services are active on the local host, stop them all.

2. Use a text editor, for example, to open the `jpcsto.ini` file.

3. Change the storage destination directory for performance data, for example.
   Modify the shaded areas below as needed.

```
        ...
[Data Section]
Store Dir=.
Backup Dir=./backup
Backup Save=5
Dump Dir=./dump
        ...
```

> **Important**
>
> - Do not insert a space at the beginning of the line or before or after the equal sign (=).
> - (`.`) in each label value indicates the default storage destination directory (`/opt/jp1pc/agtu/store`) for the Store database of the Agent Store service. To change the storage destination, specify a relative path from the storage destination directory, or specify an absolute path.
> - The `jpcsto.ini` file also describes definition information in addition to the database storage destination directory. Therefore, do not change values other than those in the `[Data Section]` section. If you change values other than those in the `[Data Section]` section, Performance Management may not operate normally.

4. Save and close the `jpcsto.ini` file.

5. Start the Performance Management programs and services.

Note:

When you use this procedure to change the Store database directory, the performance data files are not deleted from the previous directory. If these files are no longer necessary, delete only the following files:

- All files with the `.DB` extension
- All files with the `.IDX` extension

4.7.2 Updating the Store version to 2.0

The Store database comes in two versions, 1.0 and 2.0. The difference between the two versions is the saving format. For details about Store version 2.0, see the chapter that describes the design of the operations monitoring system in the *JP1/Performance Management Planning and Configuration Guide*.

Store version 2.0 is provided as the default when you perform a new installation of PFM - Agent for Platform version 08-11 or later in an environment with PFM - Base or PFM - Manager version 08-11 or later. In other cases, the Store version remains 1.0. If you want to use Store 2.0, you must use a setup command to update the version to 2.0.

If you have to restore Store 1.0 from Store 2.0, perform unsetup of Store 2.0.

The following table shows the installation conditions, whether Store 2.0 is available for the given conditions, and the procedure for enabling Store 2.0.

Table 4–9: Availability of Store 2.0 and the procedure for enabling it

Installation conditions		Whether Store 2.0 is available	Procedure for enabling Store 2.0
Version of installed PFM - Base or PFM - Manager	PFM - Agent installation type		
Earlier than 08-11	Overwrite installation	No	Update the version of PFM - Base or PFM - Manager to 08-11, and then execute the `setup` command. Update the version of PFM - Base or PFM - Manager to 08-11 or later, and then execute the `setup` command.
	New installation		
08-11 or later	Overwrite installation	Available after setup	Execute the `setup` command.
	New installation	Yes	None

(1) Setup of Store 2.0

1. Estimate the system resources and determine the retention period.

   Check whether the system resources that Store 2.0 requires are suitable for the execution environment. The system resources to be considered are the following:

   - Disk capacity
   - Number of files
   - Number of files opened by one process

   These can be adjusted by the retention period settings. When you specify the retention period, take into account the resources available in the execution environment. See the *Release Notes* for details about estimating system resources.

2. Review the directory settings.

   When the Store version is updated to 2.0, the Agent Store service might fail to start with the same directory settings that were available in Store 1.0. For this reason, you must review the settings of the directories used by the Agent Store service. You can use the `jpcconf db define` command to view and change the settings of these directories.

   The maximum length of the names of the save destination directory, backup destination directory, and other directories used by the Store database in Store 2.0 differs from the maximum length in Store 1.0. In particular, be careful when the directory settings have been changed to use a relative path name in Store 1.0. If the settings have been changed, confirm that the length of the absolute path name for the relative path name is no more than 214 bytes, which is the maximum length for directory names in Store 2.0. If the absolute path name is longer than the maximum, change the settings of each directory used by the Agent Store service before proceeding to the next step.

3. Execute the setup command.

   Execute the following command to update the Store version to 2.0:

   ```
   jpcconf db vrset -ver 2.0 -key UNIX
   ```

   For details about the `jpcconf db vrset` command, see the manual *JP1/Performance Management Reference*.

4. Set the retention period.

   Specify the retention period that you determined in step 1. Start the Agent Store service, and then specify the retention period in PFM - Web Console.

(2) Unsetup of Store 2.0

Use the `jpcconf db vrset -ver 1.0` command to perform unsetup of Store 2.0. When unsetup is performed, the entire Store database is initialized and the Store version reverts to 1.0.

For details about the `jpcconf db vrset` command, see the manual *JP1/Performance Management Reference*.

(3) Notes

(a) When migrating from Store version 1.0 to Store version 2.0

When the Store database is migrated from Store version 1.0 to Store version 2.0, retention period settings for records of the PI record type are carried over. For records of the PD record type, however, the default number of retention days is set for each record regardless of the previously set value (number of retained records), and data collected before the number of retention days is deleted.

For example, in Store version 1.0, when the number of retained records is set to 1,000 for PD records for which **Collection Interval** is set to 3,600 seconds, 24 PD records are retained in a day. As a result, data for 1,000 ÷ 24 (roughly 42) days is retained. After this Store database has been migrated to Store version 2.0, if the default number of retention days is set to 10, data from 11 or more days before is deleted and can no longer be viewed.

Before migrating to Store version 2.0, check the settings for the number of retained records for records of the PD record type. If data is set to be retained for the default number of retention days or more for Store version 2.0, use the `jpctool db dump` command to output the data in the database. See the *Release Notes* for details about the default number of days saved in Store version 2.0.

(b) When returning from Store version 2.0 to Store version 1.0

When unsetup is performed for Store version 2.0, data is initialized. Therefore, before changing the version to Store version 1.0, use the `jpctool db dump` command to output Store version 2.0 information.

4.8 Backup and restoration

The following explains how to back up and restore PFM - Agent for Platform.

To protect against the system becoming corrupted as a result of a failure, we recommend that you back up the settings information for PFM - Agent for Platform. We recommend that you also perform a backup when changing the system, such as when setting up PFM - Agent for Platform.

For details about backup and restoration for an entire Performance Management system, see the chapter explaining backup and restoration in the *JP1/Performance Management User's Guide*.

4.8.1 Backup

Perform backup in any way desired, such as by copying files. When obtaining a backup, first stop all PFM - Agent for Platform services.

The following table lists the files to be backed up for PFM - Agent for Platform settings information.

For other files, see the section that presents a list of PFM - Agent files to be backed up (for UNIX) in the *JP1/Performance Management User's Guide*.

Table 4–10: Files to be backed up for PFM - Agent for Platform

File name	Description
`/opt/jp1pc/agtu/agent/*.ini`	Settings files for the Agent Collector service
`/opt/jp1pc/agtu/store/*.ini`	Settings files for the Agent Store service
`/opt/jp1pc/agtu/agent/evfile`#1	Event file
`/opt/jp1pc/agtu/agent/jpcapp`#2	Application definition file (version 09-00 or earlier)
`/opt/jp1pc/agtu/agent/jpcapp2`#2	Application definition file (version 10-00 or later)
`/opt/jp1pc/agtu/agent/wgfile`#3	Workgroup file
`/opt/jp1pc/agtu/agent/jpcuser/*.ini`	JPCUSER definition file

Table 4–11: Files to be backed up for PFM - Agent for Platform (for a logical host)

File name	Description
environment-directory#4`/jp1pc/agtu/agent/*.ini` file	Settings files for the Agent Collector service
environment-directory#4`/jp1pc/agtu/store/*.ini` file	Settings files for the Agent Store service
environment-directory#4`/jp1pc/agtu/agent/jpcapp` file#2	Application definition file (version 09-00 or earlier)
environment-directory#4`/jp1pc/agtu/agent/jpcapp2` file#2	Application definition file (version 09-00 or later)
environment-directory#4`/jp1pc/agtu/agent/jpcuser/*.ini`	JPCUSER definition file

#1

This file is backed up when log information collection is set. For Linux, however, this file is not installed.

#2

This file does not exist unless process monitoring is set.

#3

This file is backed up when workgroup information is set.

#4

environment-directory is the directory that is created on the shared disk when a logical host is created. If no logical host has been created, backup is not performed.

Important

When creating a backup for PFM - Agent for Platform, make sure that the product version number of the environment in which the backup was created is managed. For details about product version numbers, see the *Release Notes*.

4.8.2 Restoration

To restore settings information for PFM - Agent for Platform, check the following prerequisites, and then copy the backed up files to their original locations. The backed up settings information files will overwrite the settings information files on the host.

Prerequisites

- PFM - Agent for Platform is installed.
- PFM - Agent for Platform services are stopped.
- For a logical host environment, the following conditions are also required:
- A logical host environment has already been created, and the logical host name and environment directory have the same configuration as when they were backed up.
- The logical host definition file has already been imported to the standby node.

Important

When settings information for PFM - Agent for Platform is restored, the product version number of the backed up environment and restored environment must be identical. For details about product version numbers, see the *Release Notes*. The following gives examples for whether restoration is possible.

Cases for which restoration can be performed:

Settings information backed up from PFM - Agent for Platform 09-00 is restored to PFM - Agent for Platform 09-00.

Cases for which restoration cannot be performed:

- Settings information backed up from PFM - Agent for Platform 08-00 is restored to PFM - Agent for Platform 09-00.

- Settings information backed up from PFM - Agent for Platform 09-00 is restored to PFM - Agent for Platform 09-00-04.

4.9 Settings for browsing manuals in a Web browser

The manuals for Performance Management can be viewed in a Web browser by copying the manuals from the distribution media shipped with the program product to the host on which PFM - Web Console is installed. When running PFM - Web Console on a cluster, copy the manuals to each physical host on the active node and standby node.

4.9.1 Settings for browsing manuals

(1) When browsing the manual from the help section of PFM - Web Console

1. Follow the setup procedures for PFM - Web Console to register PFM - Agent with PFM - Web Console (perform additional setup for PFM - Agent).

2. On the host on which PFM - Web Console is installed, create the directory into which the manual will be copied.
   - For Windows: *Web-Console-installation-directory*`\doc\`*language-code*`\`*xxxx*
   - For UNIX: `/opt/jp1pcwebcon/doc/`*language-code*`/`*xxxx*

     For *PFM-Agent-help-id*, specify the help ID of the PFM - Agent. For details about Help IDs, see *C. List of Identifiers*.

3. Copy the following files and directories from the manual distribution media to the directory created above.

   For HTML manuals

   For Windows: All HTML files, CSS files, and the `FIGURE` folders in *applicable-drive*`\MAN\3021\`*manual-number* (such as `03004A0D`)

   For UNIX: All HTML files, CSS files, and the `FIGURE` directory in `/` *distribution-media-mount-point*`/MAN/3021/`*manual-number* (such as `03004A0D`)

   For PDF manuals

   For Windows: PDF files in *applicable-drive*`\MAN\3021\`*manual-number* (such as `03004A0D`)

   For UNIX: PDF files in `/` *distribution-media-mount-point*`/MAN/3021/`*manual-number* (such as `03004A0D`)

   When copying, make sure that the `INDEX.HTM` file for the HTML manual, or the PDF files for the PDF manual, is placed directly within the created directory.

4. Restart PFM - Web Console.

(2) When browsing the manual from the hard disk drive on the local machine

Copy the HTML, CSS, PDF, and GIF files directly from the distribution media to a desired directory. For the HTML manual, use the following directory configuration:

`html` (`.HTML` files and CSS files)

└ `FIGURE` (GIF files)

4.9.2 Browsing a manual

To browse a manual:

1. From the menu bar frame of the Main window for PFM - Web Console, click the **Help** menu to display a help selection window.

2. Click the manual name, or **PDF** next to the manual name.

   When the manual name is clicked, the HTML manual is displayed. When **PDF** is clicked, the PDF manual is displayed.

Note on character display in Web browsers

In Windows, when the online manual is displayed from the **Start** menu, the HTML manual may be displayed in a Web browser window that is already open.

4.10 Notes regarding operation

- For notes on executing the `jpctool service list` command for the Agent Collector service or Agent Store service, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.
- For notes on executing the `jpcspm start` command or the `jpcspm stop` command for PFM - Agent for Platform, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.
- Hitachi Tuning Manager cannot monitor Performance Management.
- Due to the effects of a virus detection program, file access permission for the files and folders used by Performance Management might be locked by the exclusion control function. As a result, the following symptoms might occur:
  - Performance Management cannot be started.
  - Performance information cannot be collected, or collection is delayed.
  - A Performance Management command terminates abnormally.
  - Logs cannot be output, or a problem cannot be investigated.

  If you want to check viruses while Performance Management is running, deselect the files (including the files in the new storage location if you changed the storage location of the Store database) under the installation folder of Performance Management.

  If you want to check viruses when Performance Management stops and then to restart Performance Management, confirm that the virus check for the files under the installation folder of Performance Management (including the files in the new storage location if you changed the storage location of the Store database) is complete.
- If automatic startup of services is set in a version earlier than 06-70-/I or 07-00-/G, the following problem may occur:

  If PFM service terminates abnormally, for example due to a power failure, rebuilding the Store database index file when starting the Agent Store service takes a long time At that time, Agent Collector service may fail to start because it cannot communicate with the Agent Store service.

  In order to prevent this problem, it is recommended to install PFM - Agent Product of version 06-70-/I or 07-00-/G or later, because the contents of the automatic startup script model file has changed. Please make a backup copy of the automatic startup script file which is currently used, and change its contents by using the new model file. For details on how to do this, see the chapter that explains how to enable or disable the automatic service start feature for monitoring managers and monitoring agents (UNIX) in the *JP1/Performance Management User's Guide*.

  In addition, after resetting the automatic startup script file, service startup operations will change as follows:
  - Because the PFM service startup processes are executed in the background, these services might still be starting up even after the OS startup processing has completed.
  - Since each Performance Management service is synchronized before starting up, startup time might increase.
  - Before reset, the following message is no longer output to the console. Please see the common message log to confirm that the PFM service have started.

    KAVE06007-I The service will now start. (service=*service-name*, lhost=*logical-host-name*, inst=*instance-name*)
- In HP-UX, even when the Node and Host Name Expansion function is enabled, the length of host names that can be used in Performance Management is limited to 8 bytes from the start of the string. If you need to run Performance Management on a host whose name is more than 8 bytes long, see the chapter that presents notes regarding operation in the *JP1/Performance Management User's Guide*.
- In HP-UX and Solaris, the automatic start function is valid when the PFM service is started on an OS in Runlevel 2. The automatic stop function is valid when the PFM service is started on an OS in Runlevel 1.

  In Linux, the automatic start function is valid when the PFM service is started on an OS in Runlevel 3 or 5. The automatic stop function is valid when the PFM service is started on an OS in Runlevel 0 or 6.

When the PFM service is started in other Runlevels, neither the automatic start nor the automatic stop function operates.

- In HP-UX, the operation of PFM - Agent for Platform in high reliability mode is not supported.
- When PD_FSL and PD_FSR are collected, system calls are issued at collection-time intervals and information is collected. Because file systems are accessed in order to collect information, if you attempt to use the umount command to unmount the file system while information is being collected, the execution of the umount command might fail. In such instances, execute the umount command again at a time when information is not being collected.
- In Linux, PFM - Agent for Platform does not support environments where the number of CPUs changes dynamically. If the number of CPUs changes, the Agent Collector service must be restarted.
- When the NFS File System property is not set to No, operate PFM - Agent for Platform in the status in which information for the mounted remote file system can be referenced (status in which df command can be executed normally). If PD_FSL, PD_FSR records are collected when the mounted remote file system has not responded, the Agent Collector service hangs up, making it impossible to continue collection of all performance data.

  When data collection stopped while PD_FSL or PD_FSR records were being collected:

  1. Use the kill command to stop the jpcagtu process.

     # kill -TERM (or KILL) <*jpcagtu-process-ID*>
  2. Recover the status in which the remote file system is normally mounted. For example, restart the NFS daemon.
  3. Use the jpcspm start command to start the Agent Collector service.

     ```
     # /opt/jp1pc/tools/jpcspm start -key agtu
     ```
- In AIX, in version PFM - Agent for Platform 08-00 and later, since the information of non-operational CPU can be allocated to the system, and is added as an instance, hence there can be cases where the instance count of PI_CPUP record would have increased, when compared with versions prior to PFM - Agent for Platform 08-00.
- If the Emulex Fibre Channel card virtualization function and N_Port ID Virtualization (NPIV) function that are provided in HP-UX are used to dynamically change a disk device, the following problems sometimes occurs in the Physical Disk Overview (PI_PDSK) record:

  - Performance data for a dynamically added disk device is collected only when I/O operations are being performed and not collected during other periods.
  - When delta values are to be indicated in real-time reports, an invalid value is displayed in the performance data that was collected during the first data collection operation for a dynamically added disk device.

  If such problems occur, consider operating the system by using the following methods:

  - To evaluate an alarm, remove any dynamically added disk devices from the monitoring target.
  - Make sure Indicate delta value is not selected in the window for configuring the report display settings. Alternatively, to check the delta value of a dynamically added disk device, refer to the performance data collected during the second or a later data collection operation.

5 Collecting Log Information

This chapter explains methods for using PFM - Agent for Platform to collect log information and PFM - Web Console to monitor log information.

5.1 Overview of collecting log information

PFM - Agent for Platform can collect the following log information[#]:

- UNIX log information
- Log information for applications that run on UNIX
- Log information for databases that run on UNIX

#

PFM - Agent for Platform can collect log information from a monotone increasing log file in text format. However, it can only collect single-byte characters.

Note:

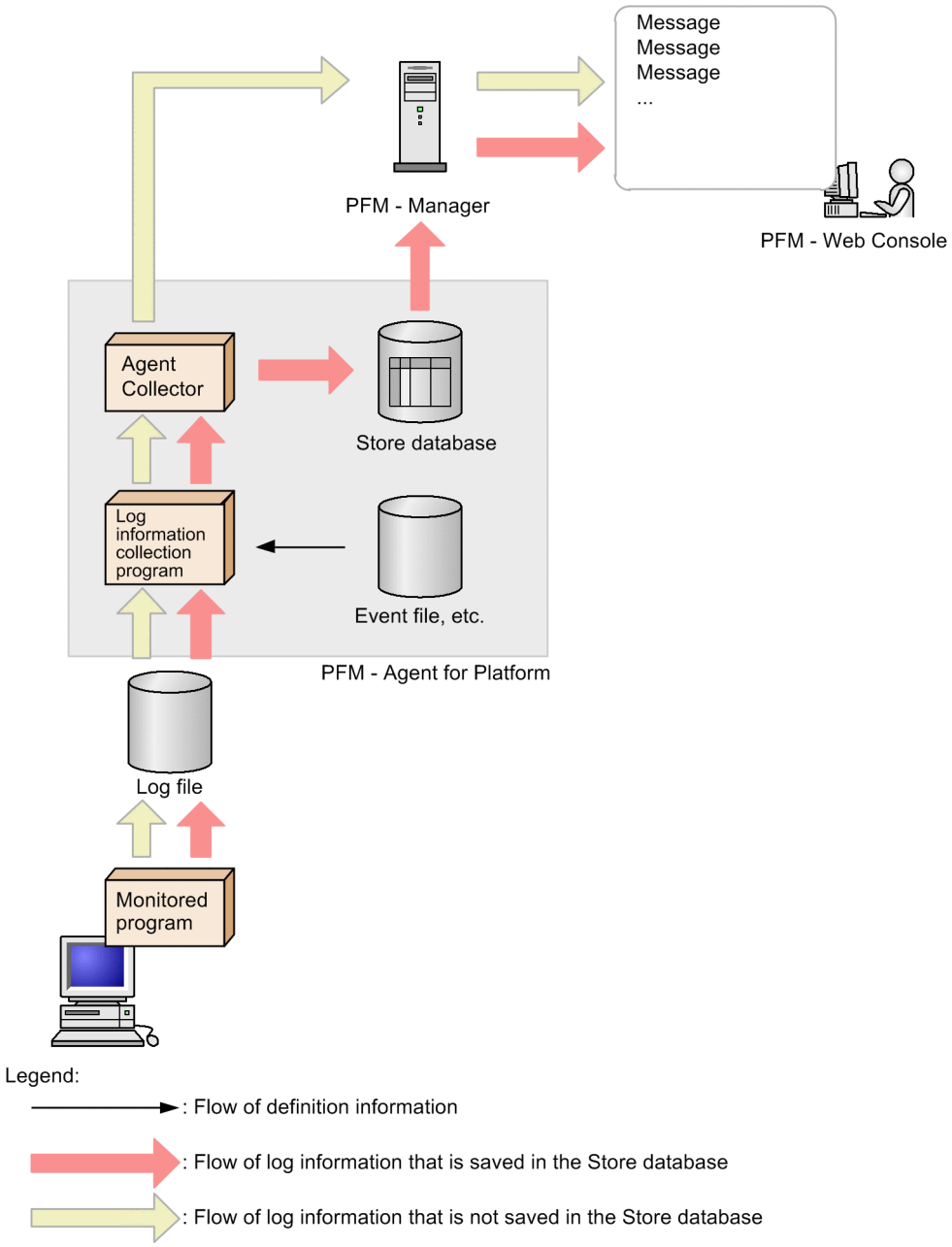
In Linux, log information cannot be collected because the Logged Messages (`PL_MESS`) record is unavailable.

In PFM - Web Console, if you set specific log information such as an error message as a threshold value in an alarm, the user can be notified when the set message is output.

The PFM - Agent for Platform log information collection program collects log information from log files based on log file names or filter conditions set in the event file. The Agent Collector service collects the log information and manages it using the Logged Messages (`PL_MESS`) record, which is a PL record. Like other records, the Logged Messages (`PL_MESS`) record can be used for report display and alarm monitoring.

The figure below shows the flow of log information monitoring.

Figure 5–1: Log information monitoring flow



5.2 Setup for collecting log information

To use PFM - Agent for Platform to collect log information and PFM - Web Console to monitor log information:

1. Specify the event file.
2. On PFM - Web Console, specify that the performance data of the Logged Messages (`PL_MESS`) record is stored in the Store database.
   You need this setting to display historical reports on PFM - Web Console.
3. Re-start PFM - Agent for Platform.

For details about each procedure, see the following section.

5.2.1 Setting up the event file

To collect log information, you need to set up the event file. The event file is used to specify information such as the name of log files, where collected log information is stored, and filter conditions.

You can use only one event file.

The event file name is:

```
/opt/jp1pc/agtu/agent/evfile
```

This event file includes only a comment line (a hash mark (#) is placed at the beginning of the line). To specify information, directly edit this event file or copy an event file to the directory and then edit it.

(1) Setting up the event file

To specify the event file:

1. Use a text editor to open the event file.
2. Add the following parameters to the event file:

   ```
   logfile=file-name
   [id=identifier]
   [regexp=filter-condition]
   ```

   The following explains each parameter.

   `logfile=`*file-name*

   Specify the name of the log file in which the collected log information will be stored as an absolute path. You can use alphanumeric characters to specify the log file name. For the number of bytes that can be specified, see the manual of each OS.

   `id=`*identifier*

   Specify text displayed as the identifier for log information. Identifiers can contain a maximum of 1,023 bytes of alphanumeric characters and symbols, except asterisks (`*`). The value specified by this parameter is the character string after the character string `jpcagtu` in the Message Text (`MESSAGE_TEXT`) field of the Logged

Messages (PL_MESS) record. If you omit this parameter, the log file name without the directory name is displayed.

regexp=*filter-condition*

Specify the filter condition for log information collected in the Logged Messages (PL_MESS) record. Filter conditions can contain a maximum of 2,040 bytes of alphanumeric characters and symbols, including the linefeed character. To define condition expressions, use extended normal expressions. For details of extended normal expressions, see the manual of each OS. If you specify multiple expressions, it is assumed to be an OR statement (logical sum).

You can use Portable Operating System Interface for UNIX (POSIX) to specify the filter conditions. Using suffix /i stores log information in the Logged Messages (PL_MESS) record without differentiating between upper and lowercase characters.

Notes:

- The system does not differentiate between upper and lowercase characters for each parameter.
- When you add parameters, do not insert a blank character or tab character before or after the equal sign (=).
- To insert a comment line, place a hash mark (#) at the beginning of the line.

3. To collect information from multiple log files, specify the parameter for each log file.

4. Save the event file.

   Always save the event file under the default event file name evfile.

To return the setting content of the evfile file to the status that was effective at the time of installation, copy evfile.model, which is the model file of the evfile file, to evfile.

(2) Example of event file specification

When you collect the log information for Sample Application stored in /opt/sampleapp/log, to store only the *warning*, *error*, and *fatal* status log information in the Logged Messages (PL_MESS) record without differentiating between upper and lowercase characters, specify as follows:

```
logfile=/opt/sampleapp/log
id=SAMPLE
regexp=warning/i
regexp=error/i
regexp=fatal/i
```

5.2.2 Setting up PFM - Web Console

To display a historical report, you use PFM - Web Console to specify the performance data of the Logged Messages (PL_MESS) record stored in the Store database.

For details about the setup method, see the chapter explaining operation monitoring data management in the *JP1/Performance Management User's Guide*.

5.2.3 Notes on collecting log information

Note the following points when monitoring messages using the Logged Messages (PL_MESS) record:

- When a message occurs that satisfies the condition expression set when the alarm is defined, the alarm status changes to error. After that, when a message occurs that does not satisfy the condition expression, the alarm status returns to normal. If you set the alarm monitoring using messages, check the messages on the Event Monitor or Event History window of PFM - Web Console.
- Character strings that can be monitored by the condition expression of the alarm definition consist of a maximum of 511 bytes that can be stored in the Logged Messages (`PL_MESS`) record. Because character strings stored in the Logged Messages (`PL_MESS`) record include header information such as the identifier (`id`), the length of messages that can be monitored is shorter than the 511 bytes for the header information portion.
- To monitor a character string of 512 bytes or more, use the event file setting of PFM - Agent for Platform to set the character string you wish to monitor as a filter condition. At this time, set an identifier (`id`) for the message as well. By setting this identifier (`id`) in the condition expression of the alarm definition, you can monitor messages that include the character string set as the filter condition.

  For example, if you want to monitor messages that include character string ABC, use the event file of PFM - Agent for Platform to set `Console` for the identifier (`id`) and set ABC for the filter condition:

  ```
  logfile=/tmp/console_log
  id=Console
  regexp=ABC
  ```

  After that, in the New Alarm Table > Alarm Conditions window of PFM - Web Console, set `Console` as the error value or warning threshold value.

  This setting sets the identifier (`id`) `Console` as the header information for messages that include the character string ABC. PFM - Web Console issues an alarm when a message occurs that includes the `Console` character string.

6 Operation in a Cluster System

This chapter describes the installation and setup of PFM - Agent for Platform in a cluster system, and the procedures for operating PFM - Agent for Platform in a cluster system.

6.1 Configuration of PFM - Agent for Platform in a cluster system

This section describes the configuration in which PFM - Agent for Platform is applied to an HA cluster system. For an overview of cluster systems and details about the system configuration for operating Performance Management in a cluster system, see the chapter that describes construction and operations in a cluster system in the *JP1/Performance Management User's Guide*.

In this chapter, the term *cluster system* refers to an HA cluster system.

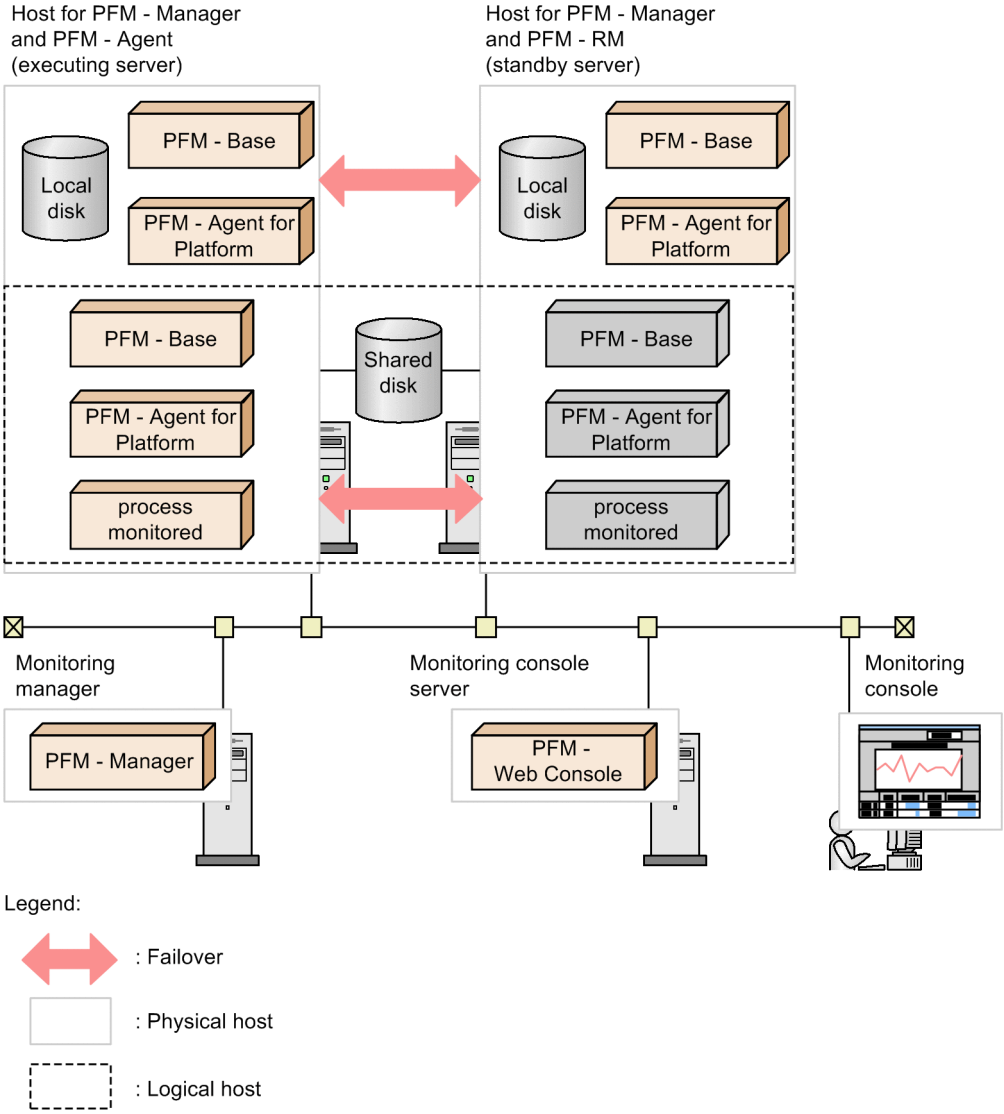
If you operate PFM - Agent for Platform in a cluster system, you can perform a failover if an error occurs, thus improving availability.

To operate PFM - Agent for Platform in a cluster system, you need to create an environment in which the same PFM - Agent for Platform can be executed on both the executing node and standby node. You also need to store a set of data, including data files, configuration files, and log files, on the shared disk.

Provided that you are monitoring processes that are running in cluster configuration, we recommend that you operate PFM -Agent for Platform by using the logical host.

To operate PFM - Agent for Platform in a cluster system, use the configuration in the following figure.

Figure 6–1: Example configuration of PFM - Agent for Platform in a cluster system



Necessary data such as definition information and performance information is stored on the shared disk, and is inherited in the event of a failover. If a single logical host contains multiple Performance Management programs, they use the same shared directory.

Multiple instances of PFM - Agent for Platform can be executed on a single node. For a configuration that contains multiple cluster configurations (an active-active configuration), execute PFM - Agent for Platform in each logical host environment. Each instance of PFM - Agent for Platform runs independently and can perform a failover separately.

6.2 Processing at failover

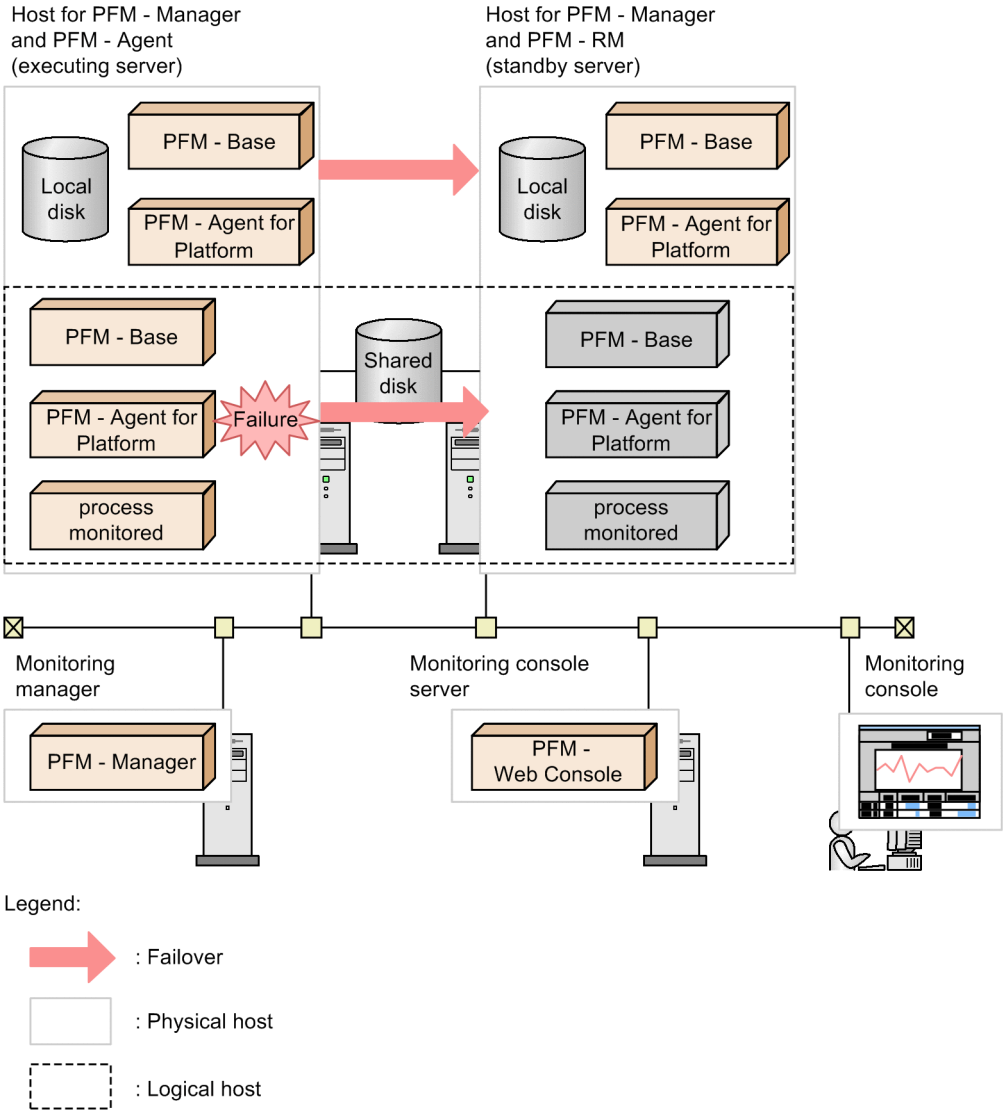
If a failure occurs on the executing host, control transfers to the standby host.

This section describes failover processing in the event of a failure in PFM - Agent for Platform. This section also describes how a failure of PFM - Manager affects PFM - Agent for Platform.

6.2.1 Failover when an error occurs on a PFM - Agent host

The following figure shows the processing when failover occurs on a host that is running PFM - Agent for Platform.

Figure 6–2: Processing when failover occurs on the PFM - Agent host



If PFM - Web Console is used while PFM - Agent for Platform is engaged in failover processing, the message `There was no answer(-6)` is displayed. When this message is displayed, wait until the failover processing is completed.

After PFM - Agent for Platform has failed over, an attempt to use PFM - Web Console connects you to the PFM - Agent for Platform that has started at the target node.

6.2.2 Effects of PFM - Manager stopping, and corrective action

The stopping of PFM - Manager affects the entire Performance Management system.

PFM - Manager provides centralized management of agent information for the instances of PFM - Agent for Platform running on all the nodes. It also controls both alarm event notifications when thresholds are exceeded during performance monitoring by PFM - Agent for Platform, and the execution of actions based on alarm events.

The following table describes how PFM - Agent for Platform is affected when PFM - Manager is stopped, and the corrective action to be taken.

Table 6–1: Effects of PFM - Manager stopping

Program name	Effects	Action
PFM - Agent for Platform	If PFM - Manager is stopped while PFM - Agent for Platform is running, the following occurs: • Collection of performance data continues. • Because alarm events are not reported to PFM - Manager, the alarm events for each alarm definition are saved. PFM - Agent for Platform retries notification until PFM - Manager starts. When the number of saved alarm events exceeds 3, the oldest alarm event is overwritten. If PFM - Agent for Platform stops, the saved alarm events are deleted. • Notifications of alarm status already sent to PFM - Manager are reset when PFM - Manager restarts. After the reset, PFM - Manager checks the status of PFM - Agent for Platform and then refreshes the alarm statuses. • An attempt to stop PFM - Agent for Platform takes time because notification of this event cannot be sent to PFM - Manager.	Start PFM - Manager. Active instances of PFM - Agent for Platform can continue to operate. Because alarms might not be notified as expected, after PFM - Manager has been recovered, check the `KAVE00024-I` messages that have been output to the common message log.

Evaluate your operating method based on how the stopping of PFM - Manager affects the system. Apart from system crashes, events such as configuration changes or maintenance might require you to stop PFM - Manager. We recommend that you perform maintenance at a time that least affects operations.

6.3 Installation and setup in a cluster system

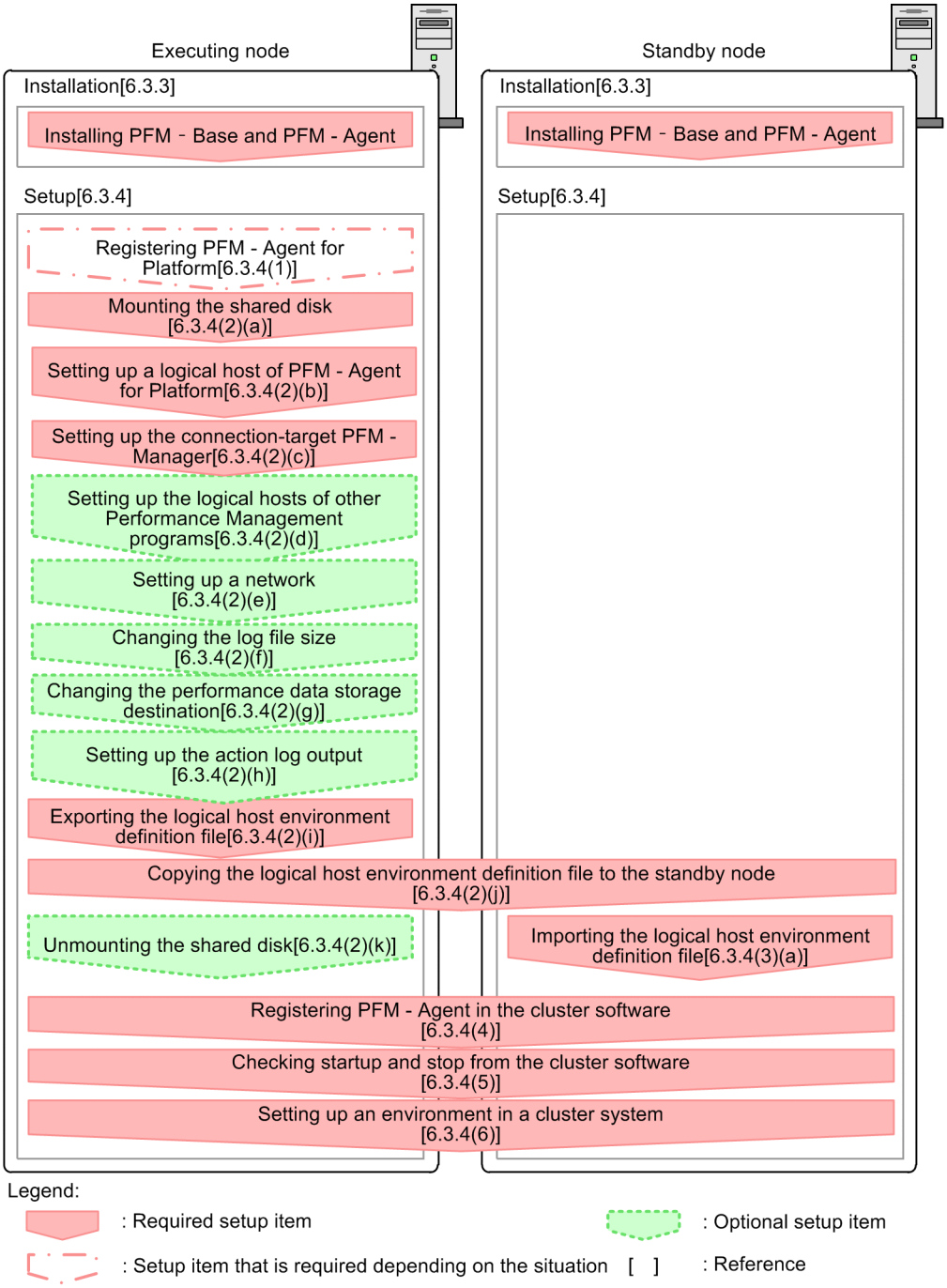
This section describes the procedures for installing and setting up PFM - Agent for Platform in a cluster system.

For details about the procedures for installing and setting up PFM - Manager, see the chapter that describes construction and operation in a cluster system in the *JP1/Performance Management User's Guide*.

6.3.1 Flow for installation and setup in a cluster system

The following figure shows the flow of installing and setting up PFM - Agent for Platform for logical host use in a cluster system.

Figure 6–3: Flow of installing and setting up PFM - Agent for Platform in a cluster system



Note that for setup commands that require user input, you can select whether to execute such commands interactively or non-interactively.

If you execute a setup command interactively, you need to enter values according to command directives.

If you execute a setup command non-interactively, user input during command execution is not required because such input can be provided instead through values in option specifications or in definition files. Also, batch processing or remote execution can automate setup operations to reduce administrator workload and operating costs.

For details about commands, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

6.3.2 Items to be checked before installation in a cluster system

This subsection describes the prerequisites, required information, and notes you must confirm before you start installation and setup.

(1) Prerequisites

To use PFM - Agent for Platform in a cluster system, the following prerequisites must be satisfied.

(a) Performance Management version

To use PFM - Agent for Platform in a cluster system, JP1/Performance Management - Base 11-01 or later or JP1/Performance Management - Manager 11-01 or later is required in devices where PFM - Agent for Platform is installed.

(b) Cluster system

Confirm that the following conditions are met:

- The cluster system is controlled by the cluster software.
- The cluster software is configured to control the startup and termination of PFM - Agent for Platform running on logical hosts.

(c) Shared disk

Make sure that the following conditions are satisfied:

- A shared disk is available for each logical host, and information can be inherited from the executing node to the standby node.
- The shared disk is physically connected to each node via Fibre Channel or SCSI.

  Performance Management does not support configurations in which a disk replicated by a network drive or via a network is used as a shared disk.
- The shared disk can be forcibly unmounted (for example, by the cluster software) to achieve failover even if some processes using the shared disk continue due to a problem.
- Instances of Performance Management programs on the same logical host use the same directory name on the shared disk. Note that you can change the location of the Store database to another directory on the shared disk.

(d) Logical host names and logical IP addresses

Make sure that the following conditions are satisfied:

- Each logical host has a logical host name and a corresponding logical IP address, and this information can be inherited from the executing node to the standby node.
- The logical host names and logical IP addresses are set in the `hosts` file and name server.
- If DNS is used, the host name without the domain name is used as the logical host name, not the FQDN name.
- All physical and logical host names are unique within the system.

Important

- Do not specify a physical host name (that is, a host name displayed by the `uname -n` command) as a logical host name. If one is specified, normal communication processing might not be achieved.

- The logical host name must consist of 1 to 32 characters.
- The logical host name must consist of alphanumeric characters, hyphens (-), and underscores (_) only.
- For the logical host name, you cannot specify `localhost`, an IP address, or a host name beginning with a hyphen (-).

(e) Settings when IPv6 is used

Because Performance Management supports IPv6 network configurations in addition to IPv4 environments, you can use Performance Management in a network configuration that contains both IPv4 and IPv6 environments. PFM - Agent for Platform can communicate with PFM - Manager by using IPv6.

Note, however, that this applies only when Windows or Linux runs on a host with PFM - Manager installed.

For details about the scope of application for communication in IPv4 and IPv6 environments, see *L. Communication in IPv4 and IPv6 Environments*.

To use IPv6 for communication, you need to enable the use of IPv6 on the PFM - Agent host before installing PFM - Agent for Platform. If you want to check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command. After confirming that IPv6 is enabled on the PFM - Agent host, you need to enable the use of IPv6 on both the PFM - Manager host and the PFM - Agent host by using the `jpcconf ipv6 enable` command. The following describes when you need to execute this command.

Cases when the `jpcconf ipv6 enable` command needs to be executed:

- When switching from an IPv4 environment to an IPv6 environment on each host
- When switching the PFM - Manager environment from IPv4 to IPv6 in an environment in which both IPv4 are used

Cases when the `jpcconf ipv6 enable` command does not need to be executed:

- When each host already operates in an IPv6 environment
- When both IPv4 and IPv6 environments are used and PFM - Manager operates in an IPv6 environment

The following shows an example of executing the `jpcconf ipv6 enable` command:

```
jpcconf ipv6 enable
```

Execute the `jpcconf ipv6 enable` command on both the executing node and the standby node.

For details about the `jpcconf ipv6 enable` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*. For details about the conditions or occasions for executing the `jpcconf ipv6 enable` command, see the chapter that describes network configuration examples in an environment that includes IPv6 in the *JP1/Performance Management Planning and Configuration Guide*.

(2) Information required to set up PFM - Agent for Platform for logical host use

To operate PFM - Agent for Platform on a logical host, you must provide the information in the following table, in addition to the environment information that is needed for normal setup of PFM - Agent for Platform.

Table 6–2: Information required to set up PFM - Agent for Platform for logical host use

No.	Item	Example
1	Logical host name	`jp1-halaop`
2	Logical IP address	`172.16.92.100`
3	Shared disk	`/jp1`

When multiple Performance Management programs are installed on a single logical host, the programs share the same directory on the shared disk.

For details about the capacity required on the shared disk, see the *Release Notes*.

(3) Note on making the logical host subject to failover by PFM - Agent for Platform

If you employ a system configuration in which PFM - Agent for Platform operates on a logical host, you must evaluate whether the entire logical host is subject to failover if a PFM - Agent for Platform failure occurs.

If failover of the entire logical host occurs, other job applications running on the same logical host also fail over, which might affect the job.

We recommend that you use one of the following cluster software settings so that errors on PFM - Agent for Platform do not affect the operations of other job applications:

- Do not monitor operation of PFM - Agent for Platform
- Do not allow detection of errors in PFM - Agent for Platform to result in failover

(4) Note on version upgrading in a logical host environment

To upgrade the version of PFM - Agent for Platform in a logical host environment, you need to mount the shared disk on either the executing node or the standby node.

6.3.3 Installation procedure in a cluster system

Install PFM - Base and PFM - Agent for Platform on both executing and standby nodes.

The installation target is the local disk. Do not install PFM - Base or PFM - Agent for Platform on a shared disk.

The installation procedure is the same as for a non-cluster system. For details about the installation procedure, see *Chapter 4. Installation and Setup in UNIX*.

6.3.4 Setup procedure in a cluster system

This subsection describes the setup required to operate Performance Management in a cluster system.

There are setup procedures for the executing node and for the standby node. Set up the executing node first, and then set up the standby node.

Note that the icon Executing system indicates an item that must be performed for the executing node, and the icon Standby system indicates an item that must be performed for the standby node. The icon Optional indicates either of the following setup items:

- A setup item that is required depending on the environment that is used
- A setup item that is used for changing the default setting

(1) Registering PFM - Agent for Platform Executing system Optional

The procedure for registering PFM - Agent for Platform is the same as for a non-cluster system. For details about this procedure, see *Chapter 4. Installation and Setup in UNIX*.

(2) Setting up a logical host of PFM - Agent

You must set up a logical host environment for PFM - Agent for Platform on the executing node.

Note:

Before starting this setup, stop all services of Performance Management programs in the entire Performance Management system. For details about how to stop services, see the chapter that describes startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.

(a) Mounting the shared disk Executing system

Make sure that the shared disk is mounted. If it is not mounted, mount it by using a cluster software operation or volume manager operation.

(b) Setting up a logical host of PFM - Agent for Platform Executing system

Execute the `jpcconf ha setup` command to create a logical host environment. This command creates a logical host environment by copying necessary data to the shared disk and setting definitions for a logical host.

To set up a logical host of PFM - Agent for Platform:

1. Execute the `jpcconf ha setup` command to create a logical host environment for PFM - Agent for Platform

   Execute the command as follows:

   ```
   jpcconf ha setup -key UNIX -lhost jp1-halaop -d /jp1
   ```

   Use the `-lhost` option to specify the logical host name. This example specifies `jp1-halaop` as the logical host name. If you use DNS, specify the logical host name without the domain name.

   Specify the directory name on the shared disk in the environment directory name of the `-d` option. For example, if `-d/jp1` is specified, `/jp1/jp1pc` is created and then files for the logical host environment file are created.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Make sure that the created logical host environment is valid.

(c) Setting up the connection-target PFM - Manager Executing system

Execute the `jpcconf mgrhost define` command to set up the PFM - Manager that manages PFM - Agent for Platform.

To set up the connection-target PFM - Manager:

1. Execute the `jpcconf mgrhost define` command to set the connection-target PFM - Manager.

   Execute the command as follows:

   ```
   jpcconf mgrhost define -host jp1-hal -lhost jp1-halaop
   ```

   Specify the host name of the connection-target PFM - Manager in the `-host` option. If the connection-target PFM - Manager runs on a logical host, specify the logical host name of the connection-target PFM - Manager in the `-host` option. This example specifies `jp1-hal` as the PFM - Manager's logical host name.

   Use the `-lhost` option to specify the logical host name of PFM - Agent for Platform. This example specifies `jp1-halaop` as the logical host name of PFM - Agent for Platform.

Although the above example shows interactive execution, you can also execute the `jpcconf mgrhost define` command non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

(d) Setting up the logical hosts of other Performance Management programs Executing system Optional

If you have other PFM - Manager or PFM - Agent programs to be set up on the same logical host in addition to PFM - Agent for Platform, set them up at this stage.

For details about the setup procedure, see the chapter that describes operations in a cluster system in the *JP1/Performance Management User's Guide* or in the applicable PFM - Agent manual.

(e) Setting up a network Executing system Optional

You must set up the network when Performance Management programs run in a network environment that uses a firewall.

The two items below are specified in the network environment settings. Change the settings if necessary.

- IP addresses

  You must specify IP addresses if you use Performance Management in a network connected to multiple LANs. To specify the IP addresses, directly edit the contents of the `jpchosts` file.

  After editing the `jpchosts` file, copy it from the executing node to the standby node.

  For details about how to set IP addresses, see the chapter that describes installation and setup in UNIX in the *JP1/Performance Management Planning and Configuration Guide*.

- Port numbers

  If you establish communication between Performance Management programs via a firewall, use the `jpcconf port define` command to set the port numbers.

  For details about how to set port numbers, see the chapter that describes installation and setup in UNIX, and the chapter that describes construction and operations in a cluster system, in the *JP1/Performance Management Planning and Configuration Guide*.

(f) Changing the log file size Executing system Optional

The operating status of Performance Management is output to a log file unique to Performance Management. This log file is called the *common message log*. This setting is required only when you wish to change this file size.

For details, see the chapter that describes installation and setup in UNIX in the *JP1/Performance Management Planning and Configuration Guide*.

(g) Changing the performance data storage destination Executing system Optional

This setting is required only when you want to change the directory where the database of performance data managed by PFM - Agent is saved, backed up, exported, or imported. For details about how to specify the setting, see *Chapter 4. Installation and Setup in UNIX*.

(h) Setting up the action log output Executing system Optional

This setting is required to output action log data when an alarm occurs. Action log data is history information that is output in cooperation with the alarm function related to exceeded thresholds, such as for system load.

For details about how to set up the action log, see *J. Outputting Action Log Data*.

(i) Exporting the logical host environment definition file Executing system

After you have created a logical host environment for PFM - Agent for Platform, you must export the environment definition to a file. This export processing involves output of the definition information for the Performance Management program set up on the logical host to a file in the batch mode. If you are setting up other Performance Management programs on the same logical host, export the environment definitions after all setup processes have been completed.

To export the logical host environment definition:

1. Execute the `jpcconf ha export` command to export the logical host environment definition.

   Output to an export file the definition information that has been created so far for the logical host environment. You can assign any name to the export file.

   For example, to export the logical host environment definitions to the `lhostexp.txt` file, execute the following command:

   ```
   jpcconf ha export -f lhostexp.txt
   ```

   Although the above example shows interactive execution, you can also execute the `jpcconf ha export` command non-interactively. For details about the `jpcconf ha export` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

(j) Copying the logical host environment definition file to the standby node Executing system Standby system

Copy the logical host environment definition file exported in *(i) Exporting the logical host environment definition file* from the executing node to the standby node.

(k) Unmounting the shared disk Executing system Optional

Unmount the shared disk by using a cluster software operation or volume manager operation, to complete the task. If you plan to continue using the shared disk you unmounted, there is no need to perform the unmount operation again.

Note:

Even if the shared disk is unmounted, you need to perform setup without mounting the shared disk if the specified environment directory contains the `jp1pc` directory under which files are stored.

In this case, use the following procedure:

1. Use the `tar` command to archive the `jp1pc` directory in the specified environment directory on the local disk.
2. Mount the shared disk.
3. If the specified environment directory does not exist on the shared disk, create the environment directory.
4. Expand the `tar` file to the environment directory on the shared disk.
5. Unmount the shared disk.
6. Delete the files under the `jp1pc` directory in the specified environment directory on the local disk.

(3) Setting up a logical host environment for the standby node Standby system

Set up a logical host environment for PFM - Agent for Platform on the standby node.

(a) Importing the logical host environment definition file Standby system

Import the export file that was copied from the executing node to the standby node.

To specify settings for executing (on the standby node) the Performance Management program of the logical host created on the executing node, use the `jpcconf ha import` command. If multiple Performance Management programs have been set up for a single logical host, the settings for all the programs are imported in batch mode.

When executing this command, you do not need to mount the shared disk in advance.

To import the logical host environment definition file:

1. Execute the `jpcconf ha import` command to import the logical host environment definition.

   Execute the command as follows:

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although the above example shows interactive execution, you can also execute the `jpcconf ha import` command non-interactively. For details about the `jpcconf ha import` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

   This command changes settings to make the environment for the standby node the same as in the export file. As a result, setup is performed for starting PFM - Agent for Platform on the logical host.

   If fixed port numbers were set by the `jpcconf port define` command during setup, they are also set in the same manner.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Make sure that the displayed information is the same as when the `jpcconf ha list` command is executed on the executing node.

(4) Registering PFM - Agent in the cluster software Executing system Standby system

To run Performance Management programs in a logical host environment, you must register them in the cluster software, and set up the environment so that the Performance Management programs start and terminate from the cluster software.

This subsection describes the settings for registering PFM - Agent for Platform in the cluster software.

Generally, when you register applications in the UNIX cluster software, you need to specify the four items: Start, stop, monitoring of action, and forced stop.

The following table describes how to specify the required settings in PFM - Agent for Platform.

Table 6–3: How to control PFM - Agent Option for Platform (UNIX) to be registered in the cluster software

No.	Item	Description
1	Start	Execute the following commands in this sequence to start PFM - Agent for Platform: 1. `jpcspm start -key AH -lhost` *logical-host-name* 2. `jpcspm start -key UNIX -lhost` *logical-host-name* PFM - Agent for Platform starts after the shared disk and the logical IP address are ready for use.
2	Stop	Execute the following commands in this sequence to stop PFM - Agent for Platform: 1. `jpcspm stop -key UNIX -lhost` *logical-host-name* 2. `jpcspm stop -key AH -lhost` *logical-host-name* PFM - Agent for Platform stops before the shared disk and the logical IP address are disabled. Note that if the service has stopped due to, for example, an error, the return value of the `jpcspm stop` command is `3`. In this case, normal termination is assumed because the service has already stopped. If the cluster software determines the execution results based on the return value, take appropriate action such as setting the return value to `0`.

3	Monitoring of action	To monitor actions of PFM - Agent for Platform, execute the `ps` command to make sure that the following process is running:
		`ps -ef | grep` *process-name logical-host-name* `| grep -v  grep` *monitored-process*
		Note:
		The `ps` command may output information other than that for the monitored processes, as in the example below. Take measures such as changing the `grep` statement so that unnecessary information is not output.
		(Example) Output result when the `ps` command is executed in an environment where the color display of the `bash` shell is enabled
		`jpcagtu -d /opt/jp1pc/agtu/agent`
		`grep -- color=auto jpcagtu`
		For *monitored-process*, you can specify the following:
		• `jpcagtu`
		• `agtu/jpcsto`
		• `jpcah`
		For details about process names, see the chapter that describes construction and operations in a cluster system in the *JP1/Performance Management User's Guide*, and *D. List of Processes*.
		Note that, during operation of Performance Management, processes might temporarily stop due to maintenance or for other reasons. In anticipation of such stopping of processes, we recommend that you provide a means for suppressing the monitoring of actions (for example, monitoring is not performed if a file indicating maintenance exists).

4	Forced stop	If you need to perform a forced stop, execute the following command: • `jpcspm stop -key all -lhost` *logical-host-name* `-kill immediate`

No.	Item	Description
4	Forced stop	You can only specify `all` for the service key in the `-key` option. *Note:* When this command is executed, all Performance Management processes in the specified logical host environment are stopped forcibly by the SIGKILL signal. At this time, forced stop of Performance Management is performed for each logical host, not for each service. Specify the setting so that forced stop is performed only when normal stop operation does not work.

Notes:

- Performance Management programs registered in the cluster are started and stopped from the cluster. Therefore, specify the setting so that the programs will not start automatically during OS startup.
- Before executing a Performance Management command, make sure that the `LANG` environment variable is specified in the script to be registered in the cluster software.
- If the cluster software determines the execution results based on the return value of the command, specify the setting so that return values of Performance Management commands are converted to the value expected by the cluster software. For details about return values of Performance Management commands, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.
- If you want to use the `ps` command to monitor actions, execute the `ps` command in advance to make sure that the entire logical host name is displayed up to the end. Note that if you use `()` or `[]` to specify a process name and logical host name in the `ps` command, acquisition of the process name and logical host name might fail. In such a case, check the reference for the `ps` command for the OS, and then re-execute the command.

(5) Checking startup and stop from the cluster software Executing system Standby system

Make sure that Performance Management programs function normally by starting and terminating the programs from the cluster software on each node.

(6) Setting up an environment in a cluster system Executing system Standby system

When you have finished setting up Performance Management programs, set up their environment so that PFM - Web Console can display operating status reports for monitored programs as appropriate to the operations. You must also set up the environment so that notifications can be sent to the user in the event of a problem in a monitored program.

For details about how to set up an environment for the Performance Management programs, see the chapter that describes construction and operation in a cluster system in the *JP1/Performance Management User's Guide*.

To transfer an environment used in a physical environment to a logical environment, change the settings to prevent unnecessary records from being collected in each of the physical and logical environments.

6.4 Uninstallation and unsetup in a cluster system

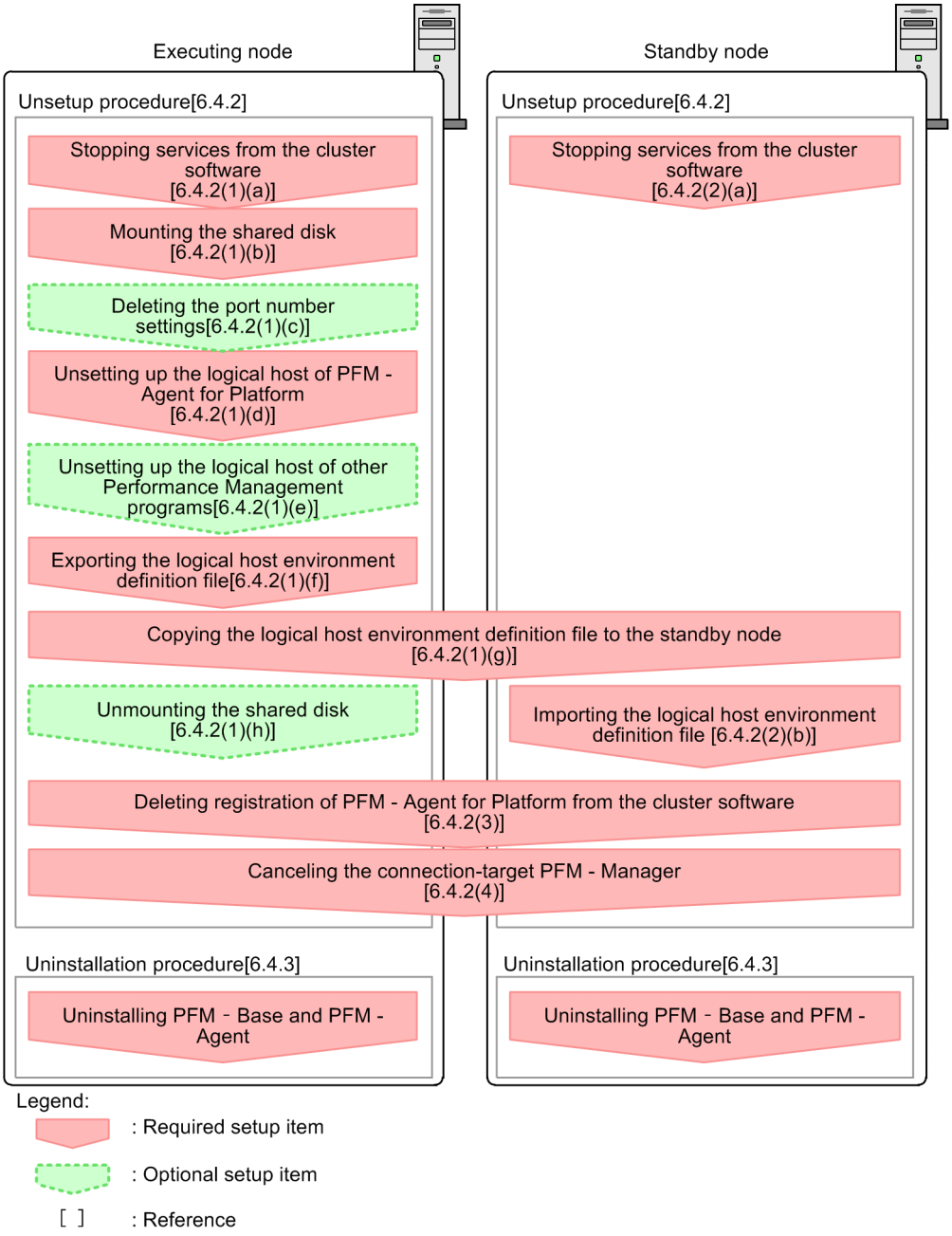
This section describes how to uninstall and unset up PFM - Agent for Platform that is running in a cluster system.

For details about how to uninstall and unset up PFM - Manager, see the chapter that describes construction and operation in a cluster system in the *JP1/Performance Management User's Guide*.

6.4.1 Flow of uninstallation and unsetup in a cluster system

The following figure shows the flow of uninstalling and unsetting up PFM - Agent for Platform that is running in a cluster system.

Figure 6–4: Flow of uninstalling and unsetting up PFM - Agent for Platform in a cluster system



6.4.2 Unsetup procedure in a cluster system

Unset up the logical host environment. There are unsetup procedures for the executing node and for the standby node. Unset up the executing node first, and then the standby node.

Note that the icon Executing system indicates an item that must be performed for the executing node, and the icon Standby system indicates an item that must be performed for the standby node. The icon Optional indicates either of the following setup items:

- A setup item that is required depending on the environment that is used
- A setup item that is used for changing the default setting

The following describes the unsetup procedure for PFM - Agent for Platform.

(1) Unsetting up the logical host environment on the executing node

(a) Stopping services from the cluster software Executing system

Stop the Performance Management programs and services that are running on the executing nodes from the cluster software. For details about how to stop the programs and services, see the cluster software documentation.

(b) Mounting the shared disk Executing system

Make sure that the shared disk is mounted. If it is not mounted, mount it by using a cluster software operation or volume manager operation.

Note:

Even if the shared disk is unmounted, if the environment directory for the logical host to be unset up contains the `jp1pc` directory under which files are stored, setup has been performed without mounting the shared disk. In this case, use the following procedure:

1. Use the `tar` command to archive the `jp1pc` directory in the specified environment directory for the logical host to be unset up on the local disk.
2. Mount the shared disk.
3. If the environment directory for the logical host to be unset up does not exist on the shared disk, create the environment directory.
4. Expand the `tar` file to the environment directory for the logical host to be unset up on the shared disk.
5. Unmount the shared disk.
6. Delete the files under the `jp1pc` directory in the environment directory for the logical host to be unset up on the local disk.

(c) Deleting the port number settings Executing system Optional

This procedure is required only when the `jpcconf port define` command was used to set port numbers during setup in an environment that uses a firewall.

For details about how to delete port numbers, see the chapter that describes installation and setup in UNIX and the chapter that describes operations in a cluster system in the *JP1/Performance Management Planning and Configuration Guide*.

(d) Unsetting up the logical host of PFM - Agent for Platform Executing system

To unset up the logical host of PFM - Agent for Platform:

Note:

If a logical host environment is deleted when the shared disk is not mounted, the logical host settings are deleted from the physical host, but the directories and files are not deleted from the shared disk. In such cases, you must mount the shared disk, and then manually delete the `jp1pc` directory under the environment directory.

1. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all -lhost jp1-halaop
   ```

   You must check the current settings before you unset up the logical host environment. Check information such as the logical host name and the shared disk path.

2. Execute the `jpcconf ha unsetup` command to delete the logical host information for PFM - Agent for Platform.

   Execute the command as follows:

   ```
   jpcconf ha unsetup -key UNIX -lhost jp1-halaop
   ```

   For details about the `jpcconf ha unsetup` command, see the chapter that explains commands in the manual *JP1/Performance Management Reference*.

   The `jpcconf ha unsetup` command deletes the settings for starting PFM - Agent for Platform on the logical host. This command also deletes files for the logical host from the shared disk.

3. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Make sure that PFM - Agent for Platform has been deleted from the logical host environment.

(e) Unsetting up the logical host of other Performance Management programs Executing system Optional

If you have other PFM - Agent programs to be unset up from the same logical host (in addition to PFM - Agent for Platform), they must be unset up at this time.

For details about the unsetup procedure, see the chapter that describes operations in a cluster system in the *JP1/Performance Management User's Guide* or in the applicable PFM - Agent manual.

(f) Exporting the logical host environment definition file Executing system

After you have deleted PFM - Agent for Platform from the logical host, you must export the environment definition to a file.

Performance Management uses a method that matches the environment of the standby system to that of the executing system by exporting and importing the environment definitions.

The Performance Management definitions from before the deletion still exist on the standby node. When environment definitions exported from the executing node are imported to the standby node, Performance Management compares the definitions, and then automatically deletes the differences from the standby node. Thus, you do not have to delete the environment definitions from the standby node.

To export the logical host environment definition file:

1. Execute the `jpcconf ha export` command to import the logical host environment definition.
   Execute the command as follows:

   ```
   jpcconf ha export -f lhostexp.txt
   ```

   Although the above example shows interactive execution, you can also execute the `jpcconf ha export` command non-interactively. For details about the `jpcconf ha export` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

(g) Copying the logical host environment definition file to the standby node Executing system Standby system

Copy the logical host environment definition file exported in *(f) Exporting the logical host environment definition file* from the executing node to the standby node.

(h) Unmounting the shared disk Executing system Optional

Using a cluster software operation or volume manager operation, unmount the shared disk to complete the task. If you intend to continue using the shared disk you unmounted, there is no need to perform the unmount operation again.

(2) Unsetting up the logical host environment on the standby node Standby system

Copy the file you exported from the executing node to the standby node, and then unset up the logical host environment on the standby node.

The following describes the procedure.

(a) Stopping services from the cluster software Standby system

Stop the Performance Management programs and services that are running on the standby nodes from the cluster software. For details about how to stop the programs and services, see the cluster software documentation.

(b) Importing the logical host environment definition file Standby system

Import the export file copied from the executing node, onto the standby node to apply the file to the standby node. Note that there is no need to mount the shared disk on the standby node during the import operation.

To import the logical host environment definition file:

1. Execute the `jpcconf ha import` command to import the logical host environment definition.
   Execute the command as follows:

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although the above example shows interactive execution, you can also execute the `jpcconf ha import` command non-interactively. For details about the `jpcconf ha import` command, see the chapter that explains the commands in the manual *JP1/Performance Management Reference*.

   This command changes settings to make the environment for the standby node the same as in the export file. As a result, the settings for starting PFM - Agent for Platform on the logical host are deleted. If you have unset up other Performance Management programs on the logical host, these settings are also deleted. If fixed port numbers were set by using the `jpcconf port define` command during setup, they are also deleted.

2. Execute the `jpcconf ha list` command to check the logical host settings.
   Execute the command as follows:

```
jpcconf ha list -key all
```

Make sure that the displayed information is the same as when the `jpcconf ha list` command is executed on the executing node.

(3) Deleting registration of PFM - Agent for Platform from the cluster software Executing system Standby system

Delete the settings related to PFM - Agent for Platform on the logical host from the cluster software.

For details about how to delete the settings, see the cluster software documentation.

(4) Canceling the connection-target PFM - Manager Executing system Standby system

To cancel the connection-target PFM - Manager, you need to delete the service information of the PFM - Agent for Platform that is connected to that PFM - Manager.

For details about how to delete service information, see the section describing how to delete service information in the chapter that explains installation and setup (in UNIX) in the *JP1/Performance Management Planning and Configuration Guide*.

To change the connection target to another PFM - Manager, see *6.3.4(2)(c) Setting up the connection-target PFM - Manager*.

6.4.3 Uninstallation procedure in a cluster system

Uninstall PFM - Base and PFM - Agent for Platform.

The uninstallation procedure is the same as for a non-cluster system. For details, see *Chapter 4. Installation and Setup in UNIX*.

Notes:

- When you uninstall PFM - Agent for Platform, stop all the Performance Management programs and services on the node where PFM - Agent for Platform is to be uninstalled.
- If you uninstall PFM - Agent for Platform without deleting the logical host environment, the environment directory might remain. In this case, delete the environment directory manually.

6.5 Changing the PFM - Agent for Platform operation method in a cluster system

This section describes how to change the operation method of PFM - Agent for Platform in a cluster system. For details about how to change the operation method of Performance Management as a whole, see the chapter that explains construction and operation in a cluster system in the *JP1/Performance Management Planning and Configuration Guide*.

6.5.1 Exporting and importing the logical host environment definition file in a cluster system

You must export and import the logical host environment definition file only if you perform any one of the following operations:

- When you set up the logical host, you change the node configuration on the logical host.

  For details about how to set up the logical host of PFM - Agent, see *6.3.4(2) Setting up a logical host of PFM - Agent*.
- When you set up the logical host of other Performance Management programs, you perform an operation that requires exporting the logical host environment definition file.

  For details about how to set up the logical host of other Performance Management programs, see *6.3.4(2)(d) Setting up the logical hosts of other Performance Management programs*.
- When you set up the network, you set port numbers.

  For details about how to set up the network, see *6.3.4(2)(e) Setting up a network*.

For details about how to export and import the logical host environment definition file, see the explanations from *6.3.4(2)(i) Exporting the logical host environment definition file* to *6.3.4(3) Setting up a logical host environment for the standby node*.

6.6 Notes on operating PFM - Agent for Platform in a cluster system

This section describes the notes on operating PFM - Agent for Platform in a cluster system.

The following functions are not supported in a logical host environment:

- Collecting log information
- Collecting work group information

6.6.1 Notes on specifying the settings for collecting information about the operating status of applications and processes

Settings for collecting information about the application operating status and settings for collecting information about the operating status of processes are not inherited from a physical host environment to a logical host environment. If you want to collect information about the operating status of applications and processes in a logical host environment, specify the necessary settings after setting up the logical host environment. Also note that the `jpcappcvt` command is not supported in a logical host environment.

6.6.2 Notes on collecting user-specific performance data

The following describes the notes on collecting user-specific performance data in a logical host environment:

- Locations of user-created data and user commands

  Make sure that the user-created data and user commands are located on the shared disk so that data collection continues in the event of a failover. When you use the functionality for periodically executing user commands, the path to access the user commands must be the same for the executing node and standby node.

- Current directory

  The current directory used by the function for periodically executing user commands is a PFM - Agent for Platform directory (*environment-directory*`/jp1pc/agtu/agent`).

- `jpcuser` command
  - If you execute the `jpcuser` command in a logical host environment, you must specify a logical host name in the `-lhost` option.
  - If the line ending in a user-created data file is CR+LF when the `jpcuser` command is being executed, the command fails and the KAVF10985-E error message is output. In this case, change the line ending to LF and then execute the command.

- Collected information

  If the collected information includes the name of the host on which user commands are executed, you can easily tell which node the command is executed on.

  For details about how to include such host names in the collected information, see *7.4.2 Settings for collecting user-specific performance data*.

6.6.3 Note on continuity of performance data before and after failover

There is no continuity of performance data before and after failover because the physical hosts whose performance data is to be collected are different before and after failover.

6.6.4 Note on collecting the same records from multiple hosts in the same machine (only in AIX)

You can collect the same records in both a physical host environment and logical host environment within the same machine, or from multiple logical hosts in the same machine. However, if the following conditions exist for multiple hosts in the same machine, performance information might not be collected correctly:

- `PI` records or `PI_CPUP` records are collected.
- The **sar Command Monitoring** property selected from **Agent Configuration** for PFM - Agent for Platform is set to `Yes`.

Therefore, when you collect `PI` records or `PI_CPUP` records from multiple hosts in the same machine, specify `Yes` for the **sar Command Monitoring** property (selected from **Agent Configuration** for JP1/PFM - Agent for Platform) for only one host.

7 User-Defined Record Collection

This chapter explains how to specify settings for using PFM - Agent for Platform to collect user-defined records.

7.1 Settings for collecting workgroup information

To use PFM - Agent for Platform to collect workgroup information and PFM - Web console to monitor workgroup information:

1. Specify a workgroup file.
2. On PFM - Web Console, specify that the performance data of the Workgroup Summary (`PI_WGRP`) record be stored in the Store database.
   You need this setting to display historical reports on PFM - Web Console.
3. Restart PFM - Agent for Platform.
   To validate the corrected definition, you need to stop and then restart PFM - Agent for Platform.

7.1.1 Setting a workgroup file for collecting workgroup information

To collect workgroup information, you need to specify a workgroup file. The workgroup file is used to specify information such as the workgroup name.

You can use only one workgroup file. Changing the file name invalidates the file.

The workgroup file name is:

```
/opt/jp1pc/agtu/agent/wgfile
```

This workgroup file includes only a comment line (a hash mark (`#`) is placed at the beginning of the line). To specify information, directly edit this event file or copy this file and then edit it.

(1) Setting up the workgroup file

To specify the workgroup file:

1. Use a text editor to open the workgroup file.
2. Add the parameters given in the following subsection to the workgroup file.
   For details about how to specify parameters, see *(2) Specifying the workgroup file*.
3. To collect information from multiple workgroups, specify the parameters for each workgroup.
4. Save the workgroup file.
   Always save the file under the default workgroup file name `wgfile`.
   To return the setting content of the `wgfile` file to the status that was effective at the time of installation, copy `wgfile.model`, which is the model file of the `wgfile` file, to `wgfile`.

Note:

When you specify multiple parameters, if at least one of the specified parameter values matches the value of a field of the Process Detail (`PD`) record, performance data is stored in the Workgroup Summary (`PI_WGRP`) record.

(2) Specifying the workgroup file

(a) Parameter specification format

The following describes the format used to specify parameters:

```
workgroup=workgroup-name
[users=UNIX-user-name] or [users_02=UNIX-user-name]
[groups=UNIX-group-name] or [groups_02=UNIX-group-name]
[programs=program-name] or [programs_02=program-name]
[arguments_02=argument-of-monitoring-program]
[regexp=monitoring-conditions]
```

Notes:

- The end of each file parameter is the linefeed.
- Always specify the workgroup parameter at the beginning. The order of other parameters is as you like.
- You can omit parameters enclosed by square brackets (`[]`) . However, in addition to the `workgroup` parameter, you must specify at least one parameter.
- The system does not differentiate between upper and lowercase characters in each parameter.
- Do not insert a space character or tab character before or after the equal sign (=).
- When one parameter is extended to multiple lines, insert a comma (`,`) at the end of the line.
- To insert a comment line, place a hash mark (`#`) at the beginning of the line.
- You can use a regular expression to specify parameters. For details about regular expressions, see the manual for each OS. Also, for examples using regular expressions, see *(3) Workgroup file specification example*.
- You can specify multiple values for parameters other than the workgroup parameter. Use a delimiter character to separate specification values. Usually, you use a comma or space character as the delimiter; however, when you specify a double quotation mark (`"`) at the beginning of the specification value in the `arguments_02`, `groups_02`, `programs_02` or `users_02` parameter to specify the monitoring target, use a double quotation mark (`"`) as the delimiter and a linefeed (`"\n`) as the termination character. For details, see Parameter details.
- When you specify a double quotation mark (`"`) at the beginning of a specification value in the `arguments_02`, `groups_02`, `programs_02` or `users_02` parameter, if there is no corresponding delimiter character, the value is not recognized as the monitoring target because the range of the character string is not specified.
- When you specify a double quotation mark (`"`) at the beginning of the specification value in the `arguments_02`, `groups_02`, `programs_02` or `users_02` parameter, a double quotation mark (`"`) that exists alone before the corresponding delimiter character is ignored. If you want to include a double quotation mark (`"`) in the monitoring target, enter two double quotation marks (`""`).
- For the `arguments_02`, `groups_02`, `programs_02` and `users_02` parameters, you can mix specification values that do and do not include a double quotation mark (`"`) at the beginning.
- To monitor a command line whose suffix is a space character, use the `regexp` parameter.

Specification examples

Command line to be monitored	How a workgroup is specified
"*command1*Δ*arg1*Δ*arg2*Δ*arg3*"	*arguments_02*="*command1*Δ*arg1*Δ*arg2*Δ*arg3*"
"*command1*Δ*arg1*Δ*arg2*Δ*arg3*Δ"	*regexp*={*arguments_02*="*command1*Δ*arg1*Δ*arg2*Δ*arg3*Δ"}
"*command1*Δ*arg1*Δ*arg2*Δ*arg3* ΔΔ ..."	*regexp*={*arguments_02*="*command1*Δ*arg1*Δ*arg2*Δ*arg3**"}

Note:

Δ: One single-byte space

(b) Parameter details

The following explains each parameter. For parameter setting examples, see *(3) Workgroup file specification example*.

`workgroup=`*workgroup-name*

Use alphanumeric characters to specify the name of the workgroup to be monitored. The character string for the name can contain a maximum of 2,037 bytes, including linefeeds; however, the Store database stores a maximum of 29 bytes. If you specify more than one workgroup name using the same name, the last specified one (that is, the workgroup having the largest row number) is applied.

`users=`*UNIX-user-name*

Specify the names of UNIX users from which to collect information as a workgroup. UNIX user names can contain a maximum of 2,041 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

To specify more than one UNIX user name, use at least one comma or space character to separate the names. All specified users become the target of monitoring.

The values specified in this parameter are displayed in the Users (`USERS`) field of the Workgroup Summary (`PI_WGRP`) record.

You specify the value to be stored in the Real User (`REAL_USER_NAME`) field of the Process Detail (`PD`) record.

`users_02=`*UNIX-user-name*

Specify the names of UNIX users from which to collect information as a workgroup. This parameter is used to extend the `users` parameter to specify user names.

If you add a double quotation mark (`"`) at the beginning of the specification value, characters up to the next delimiter character (`"` or `"\n`) are recognized as the specification value. For example, the specification value can include space characters and commas, which are the delimiter characters in the `users` parameter. If you do not add a double quotation mark (`"`), the same operation as with the `users` parameter takes place. If you use the extended specification method, the double quotation mark (`"`) is treated as the parameter delimiter character and `"\n` is treated as the termination character. To use the extended specification method to specify a double quotation mark (`"`) for the specification value, enter two double quotation marks (`""`). UNIX user names can contain a maximum of 2,038 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

The values specified in this parameter are displayed in the Users (`USERS`) field of the Workgroup Summary (`PI_WGRP`) record.

You specify the value to be stored in the Real User (`REAL_USER_NAME`) field of the Process Detail (`PD`) record.

`groups=`*UNIX-group-name*

Specify the names of UNIX groups from which to collect information as a workgroup. UNIX group names can contain a maximum of 2,040 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

To specify more than one UNIX group name, use at least one comma or space character to separate the names. All specified users become the target of monitoring.

The values specified in this parameter are displayed in the Groups (`GROUPS`) field of the Workgroup Summary (`PI_WGRP`) record.

You specify the value to be stored in the Real Group (`REAL_GROUP_NAME`) field of the Process Detail (`PD`) record.

`groups_02=`*UNIX-group-name*

Specify the names of UNIX groups from which to collect information as a workgroup. This parameter is used to extend the `groups` parameter to specify group names.

If you add a double quotation mark (") at the beginning of the specification value, characters up to the next delimiter character (" or "\n) are recognized as the specification value. For example, the specification value can include space characters and commas, which are the delimiter characters in the groups parameter. If you do not add a double quotation mark ("), the same operation as with the groups parameter takes place. If you use the extended specification method, the double quotation mark (") is treated as the parameter delimiter character and "\n is treated as the termination character. To use the extended specification method to specify a double quotation mark (") for the specification value, enter two double quotation marks (""). UNIX group names can contain a maximum of 2,037 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

The values specified in this parameter are displayed in the Groups (GROUPS) field of the Workgroup Summary (PI_WGRP) record.

You specify the value to be stored in the Real Group (REAL_GROUP_NAME) field of the Process Detail (PD) record.

programs=*program-name*

Specify the name of programs to be executed by processes that collect information for a workgroup. The program name can contain a maximum of 2,038 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

To specify more than one program name, use at least one comma or space character to separate the names. All specified programs become the target of monitoring.

The values specified in this parameter are displayed in the Programs (PROGRAMS) field of the Workgroup Summary (PI_WGRP) record.

You specify the value to be stored in the Program (PROGRAM_NAME) field of the Process Detail (PD) record.

programs_02=*program-name*

Specify the name of programs executed by processes that collect information for a workgroup. This parameter is used to extend the programs parameter to specify program names.

If you add a double quotation mark (") at the beginning of the specification value, characters up to the next delimiter character (" or "\n) are recognized as the specification value. For example, the specification value can include space characters and commas, which are the delimiter characters in the programs parameter. If you do not add a double quotation mark ("), the same operation as with the programs parameter takes place. If you use the extended specification method, the double quotation mark (") is treated as the parameter delimiter character and "\n is treated as the termination character. To use the extended specification method to specify a double quotation mark (") for the specification value, enter two double quotation marks (""). Program names can contain a maximum of 2,035 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >.

The values specified in this parameter are displayed in the Programs (PROGRAMS) field of the Workgroup Summary (PI_WGRP) record.

You specify the value to be stored in the Program (PROGRAM_NAME) field of the Process Detail (PD) record.

arguments_02=*argument-of-monitoring-program*

Specify the arguments of programs being monitored as part of this workgroup. Arguments can contain a maximum of 2,034 bytes of alphanumeric characters, including linefeeds. However, the Store database stores a maximum of 29 bytes, and the last character becomes >. To specify more than one argument, use at least one comma or space character to separate the arguments. All specified arguments become the target of monitoring.

You can expand this parameter to specify arguments. If you add a double quotation mark (") at the beginning of a specification value, characters up to the next delimiter character (" or "\n) are recognized as the specification value. For example, a specification value can include a space character or comma, which is the delimiter character.

If you do not add a double quotation mark (") at the beginning of a specification value, the same operation as with usual parameters takes place. If you use the extended specification method, the double quotation mark (") is treated

as the parameter delimiter character and "\n is treated as the termination character. To use the extended specification method to specify a double quotation mark (") for the specification value, enter two double quotation marks (").

The values specified in this parameter are displayed in the Argument Lists (PI_ARGUMENT_LISTS) field of the Workgroup Summary (PI_WGRP) record. You specify the value to be stored in the Argument List (ARGUMENT_LIST) field of the Process Detail (PD) record.

regexp=*monitoring-conditions*

Use the arguments_02, groups, groups_02, programs, programs_02, users, and users_02 conditions to specify the process monitored as part of this workgroup. You can use a regular expression to specify the process. Processes that partially match this condition are monitored.

If you specify multiple expressions, it is assumed to be an OR statement (logical sum).

Expressions can contain a maximum of 2,040 bytes of alphanumeric characters, including linefeeds. If you specify 30 bytes or more of characters for the condition expression of arguments_02, groups, groups_02, programs, programs_02, users, or users_02, the Store database stores a maximum of 29 bytes, and the last character becomes >.

For details about extended regular expressions, see the manual for each OS.

You can use Portable Operating System Interface for UNIX (POSIX) to specify monitoring conditions. If you use the suffix /i, definition information is stored in the Workgroup Summary (PI_WGRP) record without differentiating between upper and lowercase characters.

For specification examples, see *(3) Workgroup file specification example*.

(3) Workgroup file specification example

The following examples show workgroup specification files.

Example 1:

This example specifies the following information:

- Workgroup name: sysadmin
- Program names: netscape, turkey

Setting example

```
workgroup=sysadmin
programs=netscape,turkey
```

Result

Process information whose program names is netscape or turkey is collected.

Example 2:

This example specifies the following information:

- Workgroup name: argument
- Argument: data.ini

Setting example

```
workgroup=argument
argument_02=data.ini
```

Result

Process information for programs executed with the command line argument data.ini is collected.

Example 3

This example specifies the following information:

- Workgroup name: `programs`
- Program name: `spaceΔkey`, `emacs` (Δ is a space character)

Setting example

```
workgroup=programs
programs_02="space key"
```

Result

Collects process information for a program named `spaceΔkey`.

Example 4

This example defines a process that completely matches the following argument:

- Workgroup name: `development`
- Argument: `jpcagtuΔ-dΔ/opt/jp1pc/agtu/agent` (Δ is a space character)

Setting example

```
workgroup=development
arguments_02="jpcagtu -d /opt/jp1pc/agtu/agent"
```

Result

Process information for programs executed with the command line argument `jpcagtu -d /opt/jp1pc/agtu/agent` is collected.

Example 5

This example specifies the following information:

- Workgroup name: `development`
- Argument: `quota_"_middle`

Setting example

```
workgroup=development
arguments_02="quota_""_middle"
```

Result

Process information for programs executed with the command line argument `quota_"_middle` is collected.

Example 6

This example uses regular expressions to specify the user name, group name and program name.

In the regular expression, you can enclose each parameter and its value using `{` and `}`. Also, you can specify multiple regular expressions using commas (`,`) to separate them.

This example uses regular expressions to specify the following information:

- Workgroup name: `Regexp`
- UNIX user name: `.*adm.?`
- UNIX group name: `.*adm.*`
- Program name: `jpcagt.*`, `.*grd`
- Argument: `exΔprocess` (Δ is a space character)

Setting example

```
workgroup=Regexp
regexp={users=.*adm.?},{groups=.*adm.*},{programs=jpcagt.*/i,.*grd},{ar
guments_02="ex process"}
```

Result

Process information for the executed process that matches the following condition is collected.

Each condition is assumed to be a logical OR statement.

- Process information whose UNIX user name contains `adm` and consists of four or more characters.
- Process information whose UNIX group name contains `adm`
- Process information whose program name is not case sensitive and begins with `jpcagt` or end with `grd`.
- Process information for programs executed with the command line argument `ex process`

7.1.2 Examples of alarm operation when collecting workgroup information

The following gives examples of alarm operation using functionality to collect workgroup information. Here, the contents are set to issue an alarm when multiple running processes with the same process name are monitored, and the number of process startup statuses falls below *n*.

The following shows the specification of the work group file and alarm conditions.

Specification of the work group file

`workgroup=`*workgroup-name*

`programs=`*name-of-program-to-be-monitored* (specify the value stored in the Program (`PROGRAM_NAME`) field of the Process Detail (`PD`) record)

Conditions set for the alarm

Define an alarm that reports an abnormality for the following conditions, for the Workgroup Summary (`PI_WGRP`) Record:

`workgroup=`*workgroup-name*

`AND Process Count<=`*n*

Note

n indicates the number of processes.

7.1.3 Setting up PFM - Web Console

To display historical reports, you use PFM - Web Console to specify collection of Workgroup Summary (`PI_WGRP`) record information.

For details about the set up method, see the chapter explaining operation monitoring data management in the *JP1/Performance Management User's Guide*.

7.2 Settings for collecting information about the operating status of processes

This section explains how to specify user-defined records for collecting information about the operating status of processes. It also provides an example of responding when an alarm occurs.

7.2.1 Overview of collecting information about the operating status of processes

Information about the operating status of processes can be collected by PFM - Agent for Platform and then monitored using PFM - Web Console. By monitoring information about the operating status of processes, you can ensure that the operating system is working properly.

The functionality for collecting information about the operating status of processes, which is new to PFM - Agent for Platform version 10-00, is an extension of the functionality from versions 09-00 and earlier for collecting information about the application operating status.

There are two methods for using PFM - Agent for Platform to collect information about the operating status of processes:

- Collecting information on a per-process basis
- Collecting information on a per-application basis by grouping multiple processes

Use the **Agents** tree of PFM - Web Console to configure collection of information about the operating status of processes.

If you prefer to configure collection of information about the operating status of processes in the same way it was done in versions 09-00 and earlier of PFM - Agent for Platform, you can use the **Services** tree of PFM - Web Console.

7.2.2 Differences in functionality from previous versions

The following table shows the differences in functionality between how past versions collected information about the application operating status and how version 10-00 collects information about the operating status of processes.

Table 7–1: Differences from previous versions in the collection of information about the operating status of processes

Functionality	Collection of information about the application operating status in versions 09-00 and earlier	Collection of information about the operating status of processes beginning with version 10-00
Acquisition of command line	Yes	Yes
Maximum value for a command line that can be acquired	119 bytes	4,096 bytes
Collection of performance information for specific processes	No#1	Yes#2
Maximum value for monitoring conditions that can be specified	127 bytes	4,096 bytes
Maximum value for monitoring conditions that can be displayed in PFM - Web Console	31 bytes	4,096 bytes

Functionality	Collection of information about the application operating status in versions 09-00 and earlier	Collection of information about the operating status of processes beginning with version 10-00
Case sensitivity of monitoring targets	No	Yes#3
Identifying monitoring conditions by label	No	Yes#4
Grouping targets when setting process monitoring	• Programs (`Name`) • Services (`Argument List`)	• Programs (`Program Name`) • Command line (`Command Line`)

Legend:

Yes: Can be executed.

No: Cannot be executed.

#1

Always collects performance information for all processes.

#2

Collects performance information only for the processes for which process monitoring has been set.

#3

By default, letter case is distinguished. For details, see *7.2.7 Distinguishing the letter case of monitoring targets*.

#4

Use the Monitoring*XX* Label field to identify a process.

You can use the `jpcappcvt` command to upgrade application definitions that were set in version 09-00 or earlier to application definitions for version 10-00. For details about the `jpcappcvt` command, see *10. Commands*.

Note

There is no need to upgrade application definitions that were set in version 09-00 or earlier if they are not being extended into version 10-00.

7.2.3 Records used to collect information about the operating status of processes

The following tables show the records that are used to collect information about the operating status of processes in PFM - Agent for Platform.

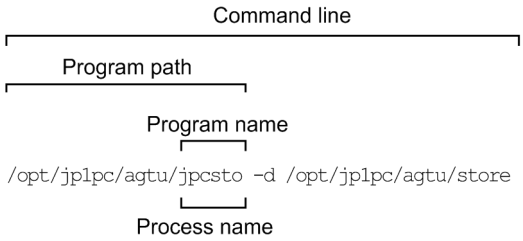
Table 7–2: Records used to collect information about the operating status of processes (on a per-process basis)

Record	Monitoring target	Information stored	Collection method
Application Process Overview (`PD_APS`)	Process	Performance data that shows the status of a process at a particular point in time	Real time
Application Process Interval (`PD_APSI`)		Performance data that shows the status at a particular point in time of a process that has been set for process monitoring	• Real time • Historical

Table 7–3: Records used to collect information about the operating status of processes (on a per-application basis)

Record	Monitoring target	Information stored	Collection method
Application Summary Extension (PD_APP2)	Process	Performance data that summarizes by application the records stored in the Application Process Overview (PD_APS) records at a particular point in time	• Real time • Historical
Application Process Detail (PD_APPD)		Performance data that summarizes by process at a particular point in time the records that are being monitored on a per-application basis and that are stored in the Application Process Overview (PD_APS) records	

The following example illustrates the relationship between the process name, program name, program path, and command line, which are used in the settings for collecting information about the operating status of processes.



7.2.4 Settings in the Agents tree for collecting information about the operating status of processes

This section describes the settings that are specified in the **Agents** tree of PFM - Web Console for collecting information about the operating status of processes, as well as how to change and delete the settings.

It also explains how to use application definition templates to simplify operations on the settings.

The settings in the **Agents** tree can be configured using PFM - Web Console version 10-00.

Screenshots from PFM - Web Console 11-50 are provided to illustrate the descriptions in this section.

(1) Setting user-defined records (monitoring targets)

To set a monitoring target:

1. Create an application.
2. Bind an alarm table to the monitoring agent.

To collect information about operating status on a per-process basis, set only one process for the application when you create the application. To collect information about operating status on a per-application basis, you can set more than one process for the application when you create the application.

The following subsections provide a more detailed description of how to set monitoring targets.

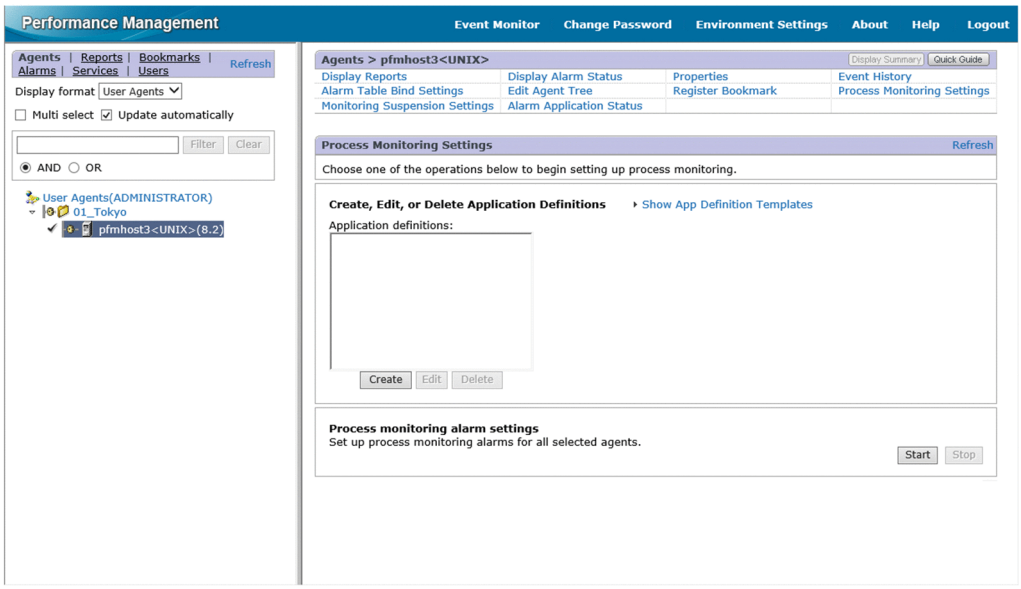
(a) Creating the application

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.

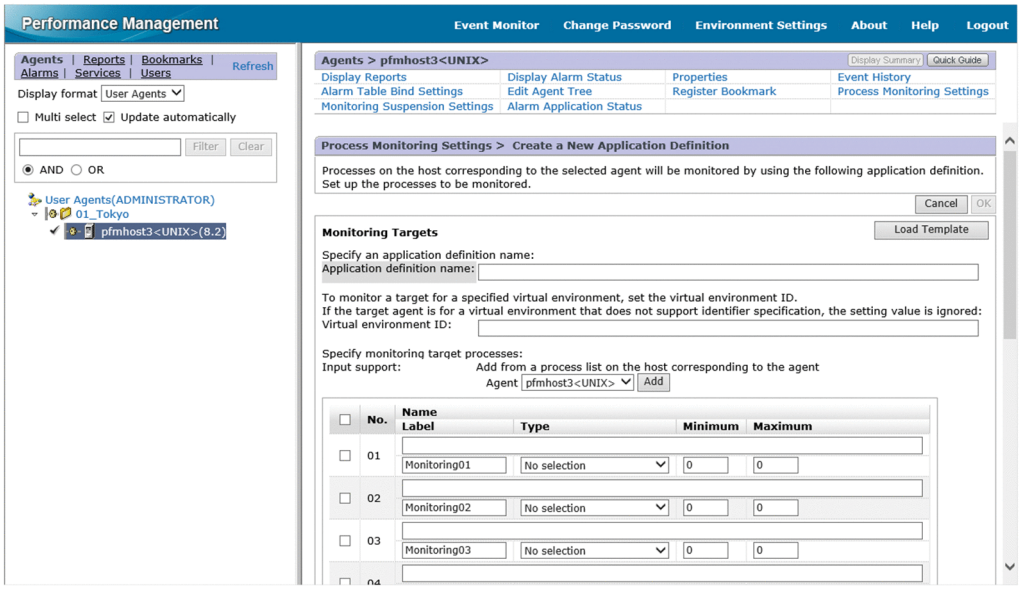
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.

3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.

4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.



5. To create a new setting, click the **Create** button. To change a setting, select in Application Definitions the application definition you want to change, and then click the **Edit** button.
   You cannot select multiple application definitions.
   The Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window appears.



6. If you are creating a new application definition, specify an application definition name in **Application definition name**.

   The application definition name, and the character string you specify for it, must conform to the following rules.

   - You can choose any name for the application definition name. The specified application definition name is stored in the Application Name field of the `PD_APP2`, `PD_APPD`, and `PD_APSI` records and will be used as an identifier for identifying the application. The application definition name you specify must be unique.
   - The name must consist of 1 to 63 bytes of single-byte alphanumeric and special characters, except for the following characters:

     Tab (`\t`) `\` `:` `;` `,` `*` `?` `"` `'` `<` `>` `|`
   - A maximum of 64 applications can be set.

7. If necessary when a virtualized system (such as a container) is used for operation, specify the identifier of the virtualized environment in **Virtual environment ID**.

   When a virtualized system is used for operation, enter an identifier to identify the virtualized environment. When an environment is specified, the monitoring target is limited to processes of the specified environment.

   For details, see *2.3.22(1) Identifying processes that have the same name in a virtualized system*.

   You can specify one of the following values as an identifier of the virtualized environment:

   - For a zone environment:

     Specify a value in the range from `0` to `2147483647`.
   - For a WPAR environment:

     Specify a value in the range from `0` to `2147483647`.
   - For a Linux container:

     Specify a 64-character hexadecimal value (a single-byte character string consisting of numbers in the range from 0 to 9 and alphabet letters in the range from a to f). If you want to monitor only the processes running in the host environment, specify `0`.

   Note:

   If the version of PFM - Web Console is earlier than 11-50, you cannot specify the container ID.

> **Note**
>
> For details about how to identify the applicable Docker container's and Podman container's IDs, see the *Release Notes*.

8. Set the application details.

   The following table shows the details that can be set in the Process Monitoring Settings > Create a New Application Definition window or the Process Monitoring Settings > Edit an Application Definition window.

   Table 7–4: Application details that can be set

Item	Description	Name of field in corresponding record
Name[#1]	Enter the conditions for identifying the monitoring target. All single-byte alphanumeric and special characters can be used except for the tab character (`\t`). You can specify a maximum of 4,096 bytes.	Monitoring Condition field of the `PD_APPD` record
Label	Specify a label for identifying the monitoring conditions. All single-byte alphanumeric and special characters can be used except for the tab character (`\t`). You can specify a maximum of 31 bytes. The default is `Monitoring`*XX*,[#2] which will be set if nothing is entered. Each monitoring label must be unique.	Monitoring Label fields of the `PD_APPD` and `PD_APSI` records
Type	Select one of **Program**, **Command Line**, or **No selection**. • **Program** Evaluate using the value in the Program Name field of the `PD_APS` record. • **Command Line** Evaluate using the value in the Command Line field of the `PD_APS` record. • **No selection** Do not evaluate.	Monitoring Field field of the `PD_APPD` and `PD_APSI` records
Minimum	Enter the minimum threshold value for the number of monitoring targets. You can specify a value from `0` to `65535`. The default is `0`.	Monitoring Min field of the `PD_APPD` record
Maximum	Enter the maximum threshold value for the number of monitoring targets. You can specify a value from `0` to `65535`. However, you must specify a value that is greater than or equal to the value specified for **Minimum**. The default is `0`.	Monitoring Max field of the `PD_APPD` record

#1

- The wildcards `*` and `?` can be used in process monitoring. An asterisk (`*`) represents zero or more instances of any character, and a question mark (`?`) represents any single character.

- If you enter more than 127 bytes for the monitoring conditions (`Monitoring`*XX*`Condition`) in the settings for collecting information about the operating status of processes, only the first 127 bytes set in the Monitoring Condition field of the `PD_APPD` record will be displayed. However, monitoring will be performed using the entire monitoring conditions that were set.

- When the monitoring targets are identified from the monitoring conditions specified in **Name**, letter case is distinguished by default. For details about changing the setting to distinguish the letter case when the monitoring targets are identified from the monitoring conditions, see *7.2.7 Distinguishing the letter case of monitoring targets*.

- Make sure that the character string you enter in **Name** matches what is set in the Program Name and Command Line fields of the `PD_APS` record.

Any character in the source information that is not in the ASCII character set range of `0x20` to `0x7E` will be converted to a hash mark (`#`: `0x23`) before it is stored in the Program Name or Command Line fields of the `PD_APS` record. Note that multi-byte characters are processed in single-byte units during conversion. For example, the multi-byte (full-width) letter `A` is converted as follows:

Source information		Post-conversion information	
Character encoding	Binary	Binary	Character string
Shift-JIS	`8260`	`2360`	`#`
EUC	`A3C1`	`2323`	`##`
UTF-8	`EFBCA1`	`232323`	`###`

#2

XX represents a two-digit numeric value in the range `01` through `15` that is set to the numeric value corresponding to the Monitoring Number field in the `PD_APPD` and `PD_APSI` records.

Note

If you select the Agent Collector service for the monitoring target from the **Agents** pull-down menu in **Input support**, and then click the **Add** button, the Process Monitoring Settings > Create a New Application Definition > Add from Running Processes window,# or the Process Monitoring Settings > Edit an Application Definition > Add from Running Processes window# is displayed, where you can select a process and set its properties.

#

Running Processes or **Running Command Lines** appears under **Monitoring target types**, depending on what is specified for **Type** in the Process Monitoring Settings > Create a New Application Definition window or the Process Monitoring Settings > Edit an Application Definition window.

If **Program** is specified for **Type** in the Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window, **Running Processes** appears. If **Command Line** is specified, **Running Command Lines** appears. The default is **Running Processes**.

9. Click the **OK** button.

   The settings are enabled.

   Note that steps 8 and later in *Creating the application* are for setting properties from the Process Monitoring Settings > Create a New Application Definition > Add from Running Processes window# or the Process Monitoring Settings > Edit an Application Definition > Add from Running Processes window;# the procedure is shown below:

   #

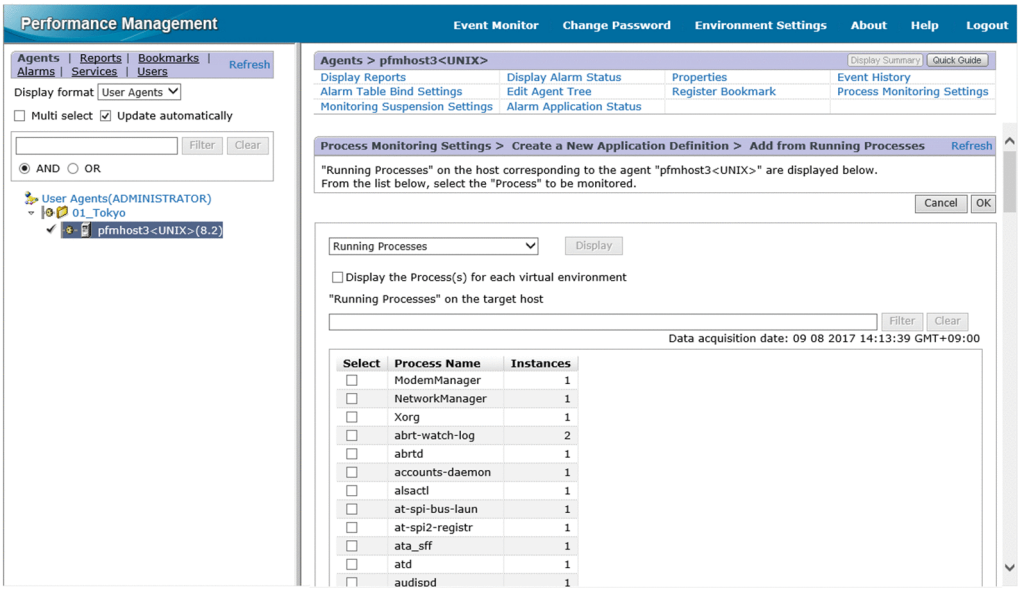
   Running Processes or **Running Command Lines** appears under **Monitoring target types**, depending on what is specified for **Types** in the Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window.

   If **Program** is specified for **Type** in the Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window, **Running Processes** appears. If **Command Line** is specified, **Running Command Lines** appears. The default is **Running Processes**.

10. Perform steps 1 to 7 of *Creating the application*, above.

11. On the **Agents** pull-down menu in **Input support**, select the Agent Collector service for the monitoring target, and then click the **Add** button.

    The Process Monitoring Settings > Create a New Application Definition > Add from Running Processes window, or the Process Monitoring Settings > Edit an Application Definition > Add from Running Processes window appears.



12. Select a monitoring target type (**Running Processes** or **Running Command Lines**) from the pull-down menu to the left of the **Display** button, and then click the **Display** button.

    A list of processes appears at the bottom of the Process Monitoring Settings > Create a New Application Definition > Add from Running Processes window or the Process Monitoring Settings > Edit an Application Definition > Add from Running Processes window.

    If you specify a keyword in **Filter** in the list of processes and then click the **Search** button, only those processes are displayed that contain the keyword in their process name. Click the **Clear** button to return to the original list of processes.

    If you select **Display the Process(s) for each virtual environment** and then select a virtual environment ID to be monitored from the **Virtual environment ID** pull-down menu, only the selected process(es) running on each virtual environment is displayed.

13. From the list of processes, select the process you want to monitor, and click the **OK** button.

    The Process Monitoring Settings > Create a New Application Definition > Add from Running Processes window or the Process Monitoring Settings > Edit an Application Definition > Add from Running Processes window closes, and the display returns to the Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window.

14. Change the necessary settings in the Process Monitoring Settings > Create a New Application Definition window or Process Monitoring Settings > Edit an Application Definition window.

    For details about the settings, see Table *7-4 Application details that can be set*.

15. Click the **OK** button.

    The settings are enabled.

(b) Binding an alarm table to the monitoring agent

Next, bind an alarm table to the monitoring agent in order to monitor information about operating status. The alarm for monitoring information about operating status is the Application Status alarm. Edit it as necessary. For details about the Application Status alarm, see *Application Status* in *8. Monitoring Templates*.

Binding an alarm table can be done in the following ways:

- Bind the alarm table in PFM UNIX Template Alarms [APP] 09.10
- Bind an alarm table created by the user

To bind the alarm table in PFM UNIX Template Alarms [APP] 09.10:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.
5. Click the **Start** button in **Process monitoring alarm settings**.
   The alarm table is bound to the monitoring agent.

To bind an alarm table created by the user:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Bind Alarm Table** method in the method frame.
   The Bind Alarm Table [Select Alarm Table] window appears.
5. Select an alarm table displayed at the bottom of the **UNIX** folder and click the **OK** button.
   The alarm table is bound to the monitoring agent.

To monitor the status of a particular process only, you can create an alarm for the following condition expressions.

Table 7–5: Condition expressions for monitoring the status of a particular process only

Item set	Condition expressions
Record	`Application Process Detail (PD_APPD)`

Field	`Application Name`
	`Monitoring Label`
	`Monitoring Status`

Abnormal condition and warning condition[#1]	`Application Name` = *Name*[#2] `AND Monitoring Label` = *Label*[#2] `AND Monitoring Status = ABNORMAL`

#1

The same conditions are specified for abnormal conditions and warning conditions.

#2

Specify the application name and monitoring label for the application you want to monitor.

(2) Deleting a user-defined record (monitoring target)

To delete a monitoring target:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, select the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.
5. Select the application definition you want to delete from Application Definitions, and then click the **Delete** button.
   The Process Monitoring Settings > Delete Application Definition window appears.
6. Click the **OK** button.
   The setting is deleted.

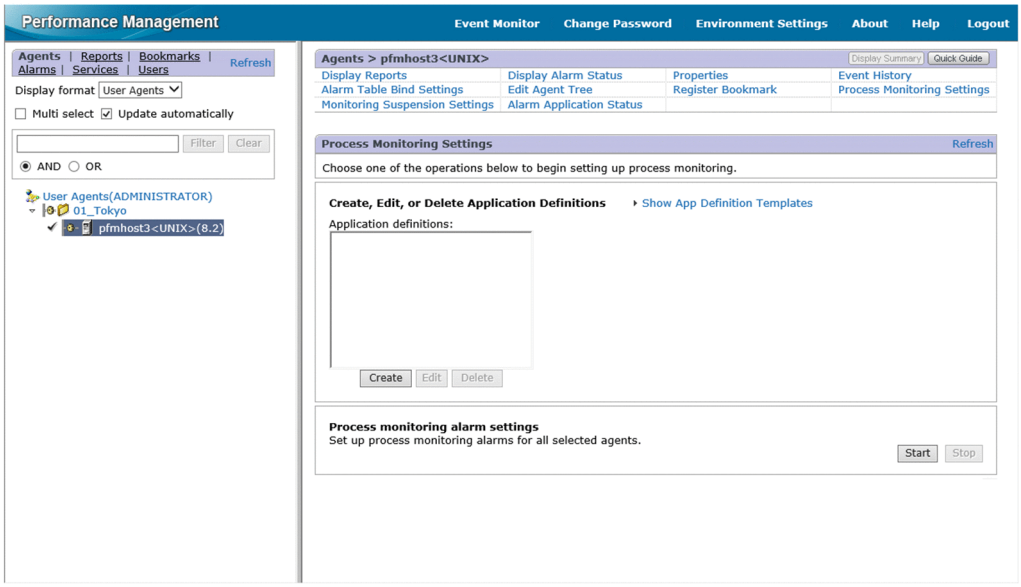
(3) Using application definition templates

You can use a template to save the settings for collecting information about the operating status of processes (application definitions) that you have made in the **Agents** tree of PFM - Web Console and then use those settings on other machines.

This section explains how to create, delete, and load application definition templates.

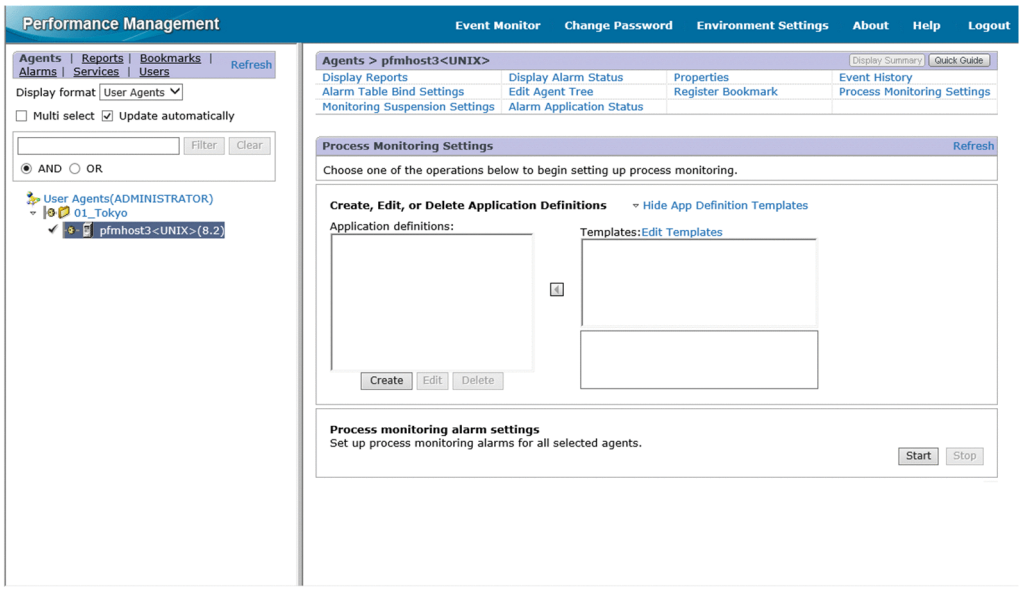
(a) Creating an application definition template

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.



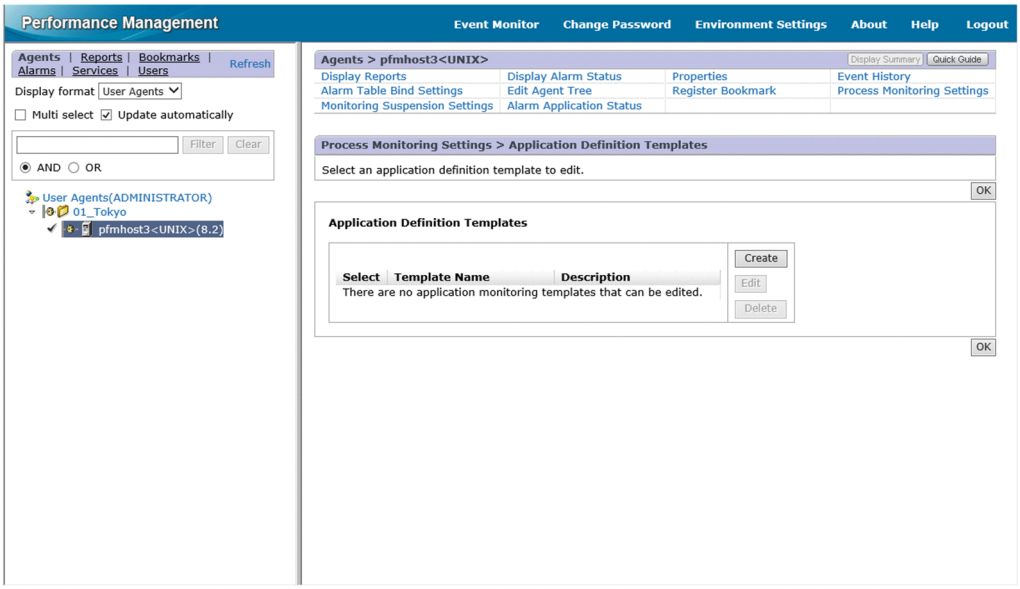
5. Select the **Show App Definition Templates** menu.

   The **Edit Templates** menu appears.



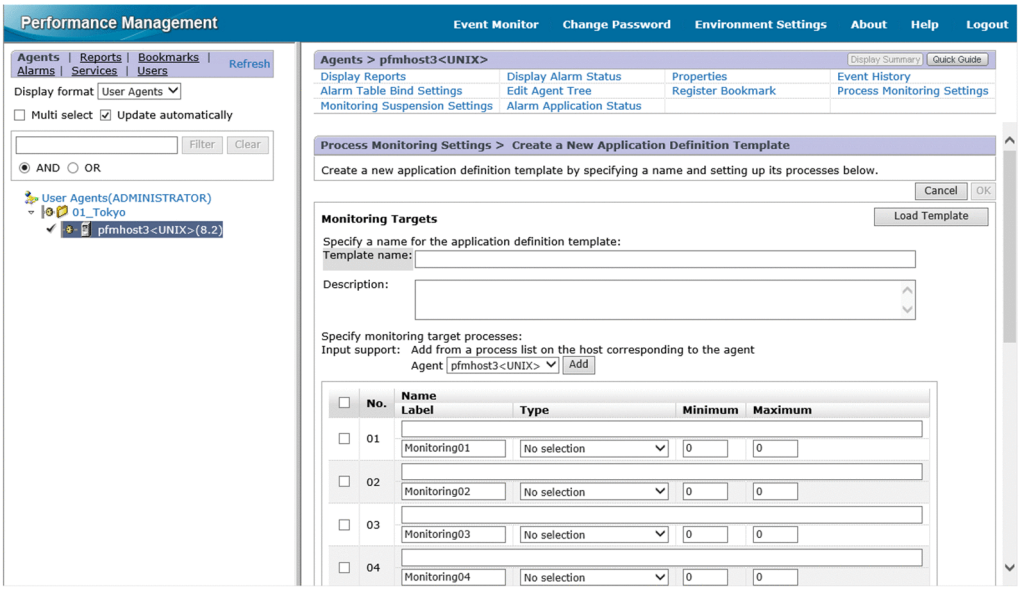
6. Select the **Edit Templates** menu.

   The Process Monitoring Settings > Application Definition Templates window appears.



7. To create a new template, click the **Create** button. To change settings, select the template you want to change from **Application Definition Templates**, and then click the **Edit** button.

   The Process Monitoring Settings > Create a New Application Definition Template window or Process Monitoring Settings > Edit an Application Definition Template window appears.



8. In **Template name**, enter a name for the template.

9. In **Description**, enter a description of the template.

10. Set the application details.

    For details about the settings, see Table *7-4 Application details that can be set* in *(1) Setting user-defined records (monitoring targets)*.

11. Click the **OK** button.

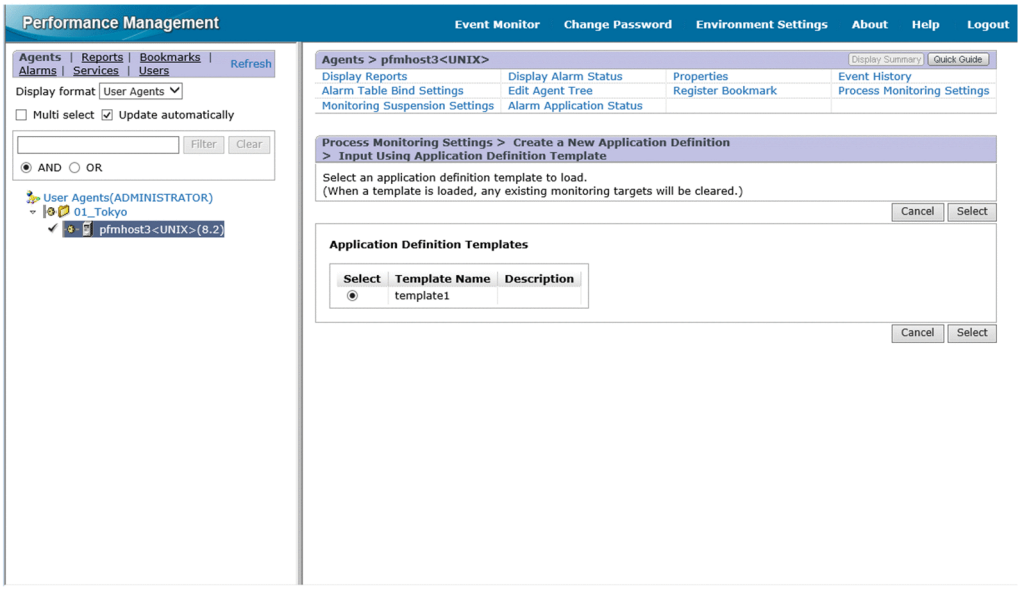
    The application definition template is created.

(b) Deleting an application definition template

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.
5. Select the **Show App Definition Templates** menu.
   The **Edit Templates** menu appears.
6. Select the **Edit Templates** menu.
   The Process Monitoring Settings > Application Definition Templates window appears.
7. From **Application Definition Templates**, select the template you want to delete, and then click the **Delete** button.
   The Process Monitoring Settings > Delete Application Definition Template window appears.
8. Click the **OK** button.
   The application definition template is deleted.

(c) Loading an application definition template

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Process Monitoring Settings** method in the method frame.
   The Process Monitoring Settings window appears.
5. Click the **Create** button.
   The Process Monitoring Settings > Create a New Application Definition window appears.
6. Click the **Load Template** button.
   The Process Monitoring Settings > Create a New Application Definition > Input Using Application Definition Template window appears.



7. From **Application Definition Templates**, select the template you want to load, and then click the **Select** button.
   The application definition template is loaded.

7.2.5 Settings in the Services tree for collecting information about the operating status of processes

This section describes the settings that are specified in the **Services** tree of PFM - Web Console for collecting information about the operating status of processes, as well as how to check, modify, and delete the settings.

(1) Setting user-defined records (monitoring targets)

To set a monitoring target:

1. Create an application.
2. Set the application's properties (such as the application name and threshold values to be monitored).
3. Bind an alarm table to the monitoring agent#

#
Alarm binding is performed in the **Agents** tree of PFM - Web Console.

To collect information about operating status on a per-process basis, you set only one process for the application when you create the application. To collect information about operating status on a per-application basis, you can set more than one process for the application when you create the application.

The following subsections provide a more detailed description of how to set monitoring targets.

(a) Creating an application

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, select the **Services** tab.

The **Services** tree appears.

3. From the navigation frame, expand the contents of the **Machines** directory.

   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on that host.

   The name of each service is displayed by its service ID. For details about the service IDs, see *C. List of Identifiers*, and the section that explains the service naming conventions in the appendix of the *JP1/Performance Management Planning and Configuration Guide*.

   The format of the service ID depends on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter that explains the Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the monitoring agent host directory, and select *host-name*<**UNIX**> (Agent Collector service).

   A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.

   The Service Properties window appears.

6. Select the **Advanced Application Monitoring** > **ADDITION OR DELETION A SETTING** tree.

7. At the bottom of the information frame, specify an application name for **ADD AN APPLICATION MONITORING SETTING**.

   The application name, and the character string you specify for it, must conform to the following rules:

   - You can choose any name for the application name. The specified application name is stored in the Application Name field of the `PD_APP2` and `PD_APPD` records and will be used as an identifier for identifying the application. The application name you specify must be unique.
   - The name must consist of 1 to 63 bytes of single-byte alphanumeric and special characters, except for the following characters:

     Tab (`\t`) `\ : ; , * ? " ' < > |`
   - A maximum of 64 applications can be set.

8. Click the **OK** button.

   The application name is generated at the bottom of the **Advanced application monitoring** > **Application monitoring setting** tree in the Service Properties window.

(b) Setting the application properties

1. After completing the steps in *Creating the application*, display the Service Properties window again and select the application name tree generated in the **Advanced application monitoring** > **Application monitoring setting** tree.

   The property information entry window appears at the bottom of the information frame.

2. Set the application properties.

   Set the virtualized environment identifier, monitoring label, monitoring field, monitoring condition, and minimum and maximum thresholds for the number of processes. You can set information for multiple processes. The following table shows the settings you can specify for the application properties.

Table 7–6: Settings for application properties

Item	Property name	Description	Name of field in corresponding record
Virtualized environment identifier	`Virtual Environment ID`[#1]	If necessary when a virtualized system (such as a container) is used for operation, enter an identifier to identify the virtualized environment. When an environment is specified, monitoring can be limited to processes of the specified environment. For details, see *2.3.22(1) Identifying processes that have the same name in a virtualized system*. You can specify one of the following values as an identifier of the virtualized environment: • For a zone environment: Specify a value in the range from `0` to `2147483647`. • For a WPAR environment: Specify a value in the range from `0` to `2147483647`. • For a Linux container: Specify a 64-character hexadecimal value (a single-byte character string consisting of numbers in the range from 0 to 9 and alphabet letters in the range from a to f). If you want to monitor only the processes running in the host environment, specify `0`.	Virtual Env ID field of the `PD_APP2` record
Monitoring label	`Monitoring`*XX* `Label`	Specify a label for identifying the monitoring conditions. All single-byte alphanumeric and special characters can be used except for the tab character (`\t`). You can specify a maximum of 31 bytes. The default is `Monitoring`*XX*,[#2] which will be set if nothing is entered. Each monitoring label must be unique.	Monitoring Label field of the `PD_APPD` record
Monitoring field	`Monitoring`*XX* `Field`	Select one of **Program Name**, **Command Line**, or **None**. • **Program Name** Evaluate using the value in the Program Name field of the `PD_APS` record. • **Command Line** Evaluate using the value in the Command Line field of the `PD_APS` record. • **None** Do not evaluate. The default is **None**.	Monitoring Field field of the `PD_APPD` record
Monitoring conditions[#3]	`Monitoring`*XX* `Condition`	Enter the conditions for identifying the monitoring target. All single-byte alphanumeric and special characters can be used except for the tab	Monitoring Condition field of the `PD_APPD` record

Item	Property name	Description	Name of field in corresponding record
Monitoring conditions#3	MonitoringXX Condition	character (\t). You can specify a maximum of 4,096 bytes. The default is one space.	Monitoring Condition field of the PD_APPD record
Minimum and maximum thresholds for the number of processes	MonitoringXX Range	Enter minimum and maximum threshold values for the number of processes, separated by a hyphen (for example, 1-2). You can specify values from 0 to 65535. The default is 0-0.	• Minimum Monitoring Min field of the PD_APPD record • Maximum Monitoring Max field of the PD_APPD record

#1

This property is displayed only when the OS on which PFM - Agent for Platform is running is Solaris, AIX, or Linux.

#2

XX represents a two-digit numeric value in the range 01 through 15 that is set to the numeric value corresponding to the Monitoring Number field in the PD_APPD record.

#3

• The wildcards * and ? can be used in process monitoring. An asterisk (*) represents zero or more instances of any character, and a question mark (?) represents any single character.

• If you enter more than 127 bytes for the monitoring conditions (MonitoringXX Condition) in the settings for collecting information about the operating status of processes, only the first 127 bytes set in the Monitoring Condition field of the PD_APPD record will be displayed. However, monitoring will be performed using the entire monitoring conditions that were set.

• When the monitoring targets are identified from the monitoring conditions specified in **Name**, letter case is distinguished by default. For details about changing the setting to distinguish the letter case when the monitoring targets are identified from the monitoring conditions, see *7.2.7 Distinguishing the letter case of monitoring targets*.

• Make sure that the character string you enter in the MonitoringXX Condition property matches what is set in Program Name field of the PD_APS or PD_APSI record, and the Command Line field of the PD_APS record.

Any character in the source information that is not in the ASCII character set range of 0x20 to 0x7E will be converted to a hash mark (#: 0x23) before it is stored in the Program Name field of the PD_APS or PD_APSI record, and the Command Line field of the PD_APS record. Note that multi-byte characters are processed in single-byte units during conversion. For example, the multi-byte (full-width) letter A is converted as follows:

Source information		Post-conversion information	
Character encoding	Binary	Binary	Character string
Shift-JIS	8260	2360	#`
EUC	A3C1	2323	##
UTF-8	EFBCA1	232323	###

3. Click the **OK** button.

   The settings are enabled.

(c) Binding an alarm table to the monitoring agent

Finally, bind an alarm table to the monitoring agent in order to monitor information about operating status. The alarm for monitoring information about operating status is the Application Status alarm. Edit it as necessary. For details about the Application Status alarm, see *Application Status* in *8. Monitoring Templates*.

To bind an alarm table:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, click the **Agents** tab.
   The **Agents** tree appears.
3. In the navigation frame of the **Agents** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.
4. Select the **Bind Alarm Table** method in the method frame.
   The Bind Alarm Table [Select Alarm Table] window appears.
5. Select an alarm table displayed at the bottom of the **UNIX** directory, and then click the **OK** button.
   The alarm table is bound to the monitoring agent.

To monitor the status of a particular process only, you can create an alarm for the following condition expressions.

Table 7–7: Condition expressions for monitoring the status of a particular process only

Item	Condition expressions
Record	Application Process Count (`PD_APPD`)

Field	`Application Name`
	`Monitoring Label`
	`Monitoring Status`

Abnormal condition and warning condition[#1]	`Application Name` = *Name*[#2] `AND Monitoring Label` = *Label*[#2] `AND Monitoring Status = ABNORMAL`

#1
The same conditions are specified for abnormal conditions and warning conditions.

#2
Specify the application name and monitoring label for the application you want to monitor.

(2) Checking or changing the settings in a user-defined record (monitoring target)

To check or change the settings for a monitoring target for collecting information about the operating status of processes:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.
2. In the navigation frame of the Main window, select the **Services** tab.
   The **Services** tree appears.

3. From the navigation frame, expand the contents of the **Machines** directory.

   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on that host.

   The name of each service is displayed by its service ID. For details about the service IDs, see *C. List of Identifiers*, and the section that explains the service naming conventions in the appendix of the *JP1/Performance Management Planning and Configuration Guide*.

   The format of the service ID depends on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter that explains the Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the monitoring agent host directory, and then select *host-name*<**UNIX**> (Agent Collector service).

   A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.

   The Service Properties window appears.

6. Expand the **Advanced application monitoring** > **Application monitoring setting** tree, and then select the tree of the application name you want to check.

7. Check the settings.

8. To change a setting, perform step 2 under *Setting the application properties* in *(1) Setting user-defined records (monitoring targets)*.

9. Click the **OK** button.

   If you updated a setting in step 8, that change becomes effective.

(3) Deleting a user-defined record (monitoring target)

To delete a monitoring target:

1. From the monitoring console browser, log in to PFM - Web Console.

   The Main window appears.

2. In the navigation frame of the Main window, select the **Services** tab.

   The **Services** tree appears.

3. From the navigation frame, expand the contents of the **Machines** directory.

   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on that host.

   The name of each service is displayed by its service ID. For details about the service IDs, see *C. List of Identifiers*, and the section that explains the service naming conventions in the appendix of the *JP1/Performance Management Planning and Configuration Guide*.

   The format of the service ID depends on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter that explains the Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the monitoring agent host directory, and select *host-name*<**UNIX**> (Agent Collector service).

   A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.

The Service Properties window appears.

6. Select the **Advanced Application Monitoring** > **ADDITION OR DELETION A SETTING** tree.

7. At the bottom of the information frame, select the application name of the monitoring target to be deleted from **DELETE AN APPLICATION MONITORING SETTING**, and then click the **OK** button.
   The setting is deleted.

7.2.6 Command-based settings for collecting information about the operating status of processes

This section describes how to execute commands in order to specify settings for collecting information about the operating status of processes.

Settings for collecting information about the operating status of processes can also be specified through the **Agents** tree of PFM - Web Console, but using commands enables you to use batch processing to automate maintenance and configuration.

(1) Setting user-defined records (monitoring targets)

To set a monitoring target:

1. Create an application definition file.
2. Create an application.
3. Bind an alarm table to the monitoring agent.

To collect information about operating status on a per-process basis, you set only one process for the application when you create the application definition file. To collect information about operating status on a per-application basis, you can set more than one process for the application when you create the application definition file.

The subsections below provide a more detailed description of how to set monitoring targets.

For details about the `jpcmkkey`, `jpcprocdef create`, `jpcprocdef output`, and `jpctool alarm bind` commands, see the chapter describing commands in the manual *JP1/Performance Management Reference*.

The service ID is based on the PFM - Agent host name. For example, if the PFM - Agent host name is `pfmhost3`, the service ID will be `UA1pfmhost3`. For details about the service IDs, see the chapter that explains the Performance Management functions in the *JP1/Performance Management Planning and Configuration Guide*.

(a) Creating a key file for authentication

Create a key file for authentication by executing the `jpcmkkey` command on the host where PFM - Web Console is installed. This step is not necessary if you have already created a key file.

The following example executes the command to create a key file for authentication, using `ADMINISTRATOR` as the user name and `xxxxx` as the password for login authentication:

```
jpcmkkey -user "ADMINISTRATOR" -password "xxxxx"
```

(b) Creating an application definition file

The application definition file describes (in XML format) the applicable conditions for collecting information about the operating status of processes. When you create an application, the application definition file is used for the parameters for the `jpcprocdef create` command.

Log in to the host where PFM - Web Console is installed to execute the `jpcprocdef create` command. Use one of the following sample files as a starting point for creating a new application definition file.

If PFM - Web Console is on Windows

PFM-Web-Console-installation-folder`\sample\processmonitoringcommand\jpcprocdef-parameters-windows.xml`

If PFM - Web Console is on UNIX

`/opt/jp1pcwebcon/sample/processmonitoringcommand/jpcprocdef-parameters-unix.xml`

To create a new application definition by editing an existing application definition, output it using the `jpcprocdef output` command.

Log into the host on which PFM - Web Console is installed to execute the `jpcprocdef output` command. The following are examples of specifying the `jpcprocdef output` command.

If PFM - Web Console is on Windows (output to `c:\sample.xml`)

`jpcprocdef output -agent` *service-ID* `-name application1 -f c:\sample.xml`

If PFM - Web Console is on UNIX (output to `/tmp/sample.xml`)

`jpcprocdef output -agent` *service-ID* `-name application1 -f /tmp/sample.xml`

(c) Creating an application

Execute the `jpcprocdef create` command with the application definition file specified in the `-f` option to create the application.

Log in to the host where PFM - Web Console is installed to execute the `jpcprocdef create` command. The `jpcprocdef create` command enables you to create an application for a single agent. To create applications for multiple agents, execute the command repeatedly in a batch process. The following are examples of specifying the `jpcprocdef create` command.

PFM - Web Console on Windows (application definition settings information file: `c:\sample.xml`)

`jpcprocdef create -agent` *service-ID* `-f c:\sample.xml`

PFM - Web Console on UNIX (application definition settings information file: `/tmp/sample.xml`)

`jpcprocdef create -agent` *service-ID* `-f /tmp/sample.xml`

(d) Binding an alarm table to the monitoring agent

Execute the `jpctool alarm bind` command to bind an alarm table to the monitoring agent in order to monitor information about operating status.

Log in to the host where PFM - Manager is installed to execute the `jpctool alarm bind` command.

The alarm for monitoring information about operating status is the Application Status alarm. Edit it as necessary. For details about the Application Status alarm, see *Application Status* in *8. Monitoring Templates*.

If you have already bound the alarm table, you do not need to bind it every time you set a monitoring target.

Binding an alarm table can be done in the following ways:

- Bind the alarm table in PFM UNIX Template Alarms [APP] 09.10.
- Bind an alarm table created by the user.

The following examples execute the `jpctool alarm bind` command to bind the alarm table in PFM UNIX Template Alarms [APP] 09.10.

PFM - Manager is on Windows

`jpctool alarm bind -key UNIX -table "PFM UNIX Template Alarms [APP] 09.10" -id` *service-ID* `-add`

PFM - Manager is on UNIX

`jpctool alarm bind -key UNIX -table "PFM UNIX Template Alarms [APP] 09.10" -id` *service-ID* `-add`

The following examples execute the `jpctool alarm bind` command to bind an alarm table created by the user:

PFM - Manager is on Windows

`jpctool alarm bind -key UNIX -table` *user-defined-alarm-table-name*# `-id` *service-ID* `-add`

PFM - Manager is on UNIX

`jpctool alarm bind -key UNIX -table` *user-defined-alarm-table-name*# `-id` *service-ID* `-add`

#

Specify for *user-defined-alarm-table-name* the name of the user-created alarm table.

To monitor the status of a particular process only, you can create an alarm with the condition expressions listed in Table *7-5 Condition expressions for monitoring the status of a particular process only*.

(2) Deleting a monitoring target by using commands

To delete a monitoring target:

1. Check the definition name of the application definition you want to delete.
2. Unbind the alarm table
3. Delete the application definition

For details about the commands `jpcmkkey`, `jpcprocdef list`, `jpctool alarm unbind`, and `jpcprocdef delete`, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

The service ID is based on the PFM - Agent host name. For example, if the PFM - Agent host name is `pfmhost3`, the service ID will be `UA1pfmhost3`. For details about service IDs, see the chapter that explains the Performance Management functions in the *JP1/Performance Management Planning and Configuration Guide*.

(a) Creating a key file for authentication

Create a key file for authentication by executing the `jpcmkkey` command on the host where PFM - Web Console is installed. This step is not necessary if you have already created a key file.

The following example executes the command to create a key file for authentication, using `ADMINISTRATOR` as the user name and `xxxxx` as the password for login authentication:

```
jpcmkkey -user "ADMINISTRATOR" -password "xxxxx"
```

(b) Checking the name of the application definition you want to delete

Execute the `jpcprocdef list` command to check the name of the application definition you want to delete from the list of application definitions defined in the monitoring agent.

Log in to the host where PFM - Web Console is installed to execute the `jpcprocdef list` command. The following is an example of specifying the `jpcprocdef list` command:

```
jpcprocdef list -agent service-ID
```

(c) Unbinding the alarm table

Execute the `jpctool alarm unbind` command to unbind the alarm table where the monitoring agent is bound and stop monitoring.

Log in to the host where PFM - Manager is installed to execute the `jpctool alarm unbind` command. For details about the `jpctool alarm unbind` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

If the alarm table is already unbound, you do not need to unbind it every time you delete a monitoring target.

The following is an example of specifying the `jpctool alarm unbind` command to unbind the alarm table in PFM UNIX Template Alarms [APP] 09.10:

```
jpctool alarm unbind -key UNIX -table "PFM UNIX Template Alarms [APP] 09.10" -id
service-ID
```

(d) Deleting the application definition

Execute the `jpcprocdef delete` command to delete an application definition.

Log in to the host where PFM - Web Console is installed to execute the `jpcprocdef delete` command.

The `jpcprocdef delete` command enables you to delete the application definition for a single agent. To delete the application definition for multiple agents, execute the command repeatedly in a batch process.

The following example uses the `jpcprocdef delete` command to delete the application definition `application5`:

```
jpcprocdef delete -agent service-ID -name "application5"
```

7.2.7 Distinguishing the letter case of monitoring targets

You can set whether to distinguish between uppercase and lowercase letters in monitoring target process names. By default, uppercase and lowercase letters are distinguished.

To change whether uppercase and lowercase letters are distinguished in monitoring target names

1. From the monitoring console browser, log in to PFM - Web Console.

The Main window appears.

2. In the navigation frame of the Main window, select the **Services** tab.
   The **Services** tree appears.

3. From the **Services** tree, select *host-name*<**UNIX**> (Agent Collector service).
   A check mark is displayed next to the Agent Collector service.

4. Select the **Properties** method in the method frame.
   The Service Properties window appears.

5. Select the **Advanced application monitoring** > **Application monitoring setting** tree.
   The property information entry window appears at the bottom of the information frame.

6. Change the value of the `Case Sensitive` property.
   Select one of the following.
   - **Yes**: Distinguish between uppercase and lowercase letters.
   - **No**: Do not distinguish between uppercase and lowercase letters.

7. Click the **OK** button.
   The setting takes effect.

The following table shows the effects of the `Case Sensitive` property values.

Table 7–8: Effects of the Case Sensitive property values

Names of the running processes	Value set in the MonitoringXX Condition property	Value set in the Case Sensitive property	Number of processes
• `ProcessA` • `PROCESSA`	`ProcessA`	`Yes`	1
		`No`	2
	`PROCESSA`	`Yes`	1
		`No`	2
	`processa`	`Yes`	0
		`No`	2

Legend:

`Yes`: Uppercase and lowercase letters are distinguished.

`No`: Uppercase and lowercase letters are not distinguished.

The table shows the difference in the number of processes that are determined to be running, depending on the values set in the `Monitoring`*XX* `Condition` property and the `Case Sensitive` property. In this example, the two processes `ProcessA` and `PROCESSA` are running on the PFM - Agent host.

Note:

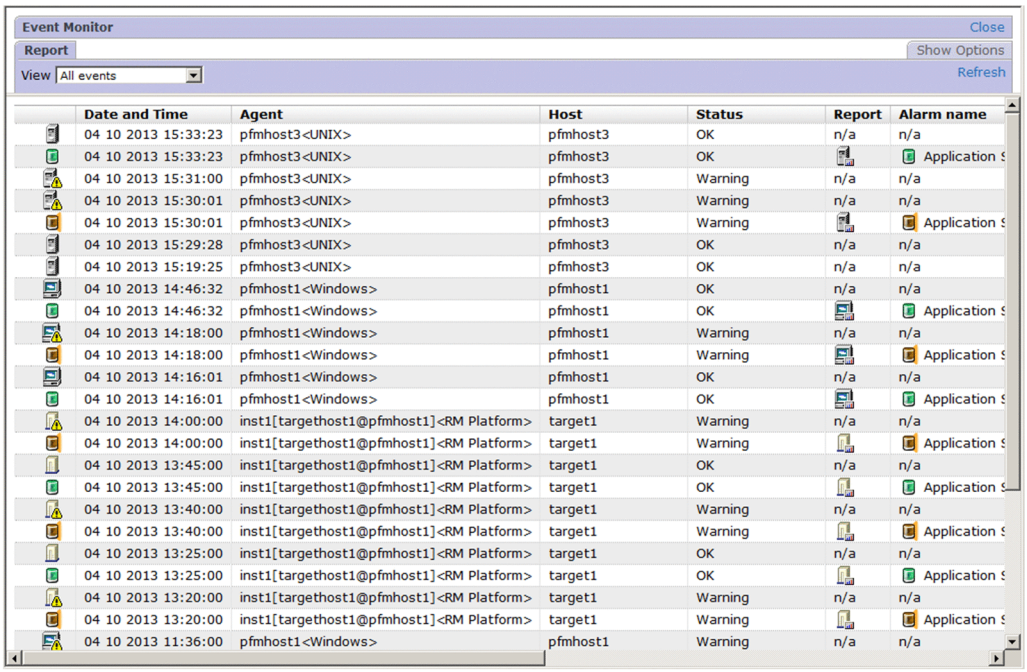
The setting of the `Case Sensitive` property affects all application definitions. If you change the setting of the `Case Sensitive` property, check and, if necessary, revise existing application definitions.

7.2.8 Example of responding to an alarm while collecting information about the operating status of processes

The example in this section illustrates how to identify the process of concern in the event an alarm occurs while the operating status of processes is being monitored.

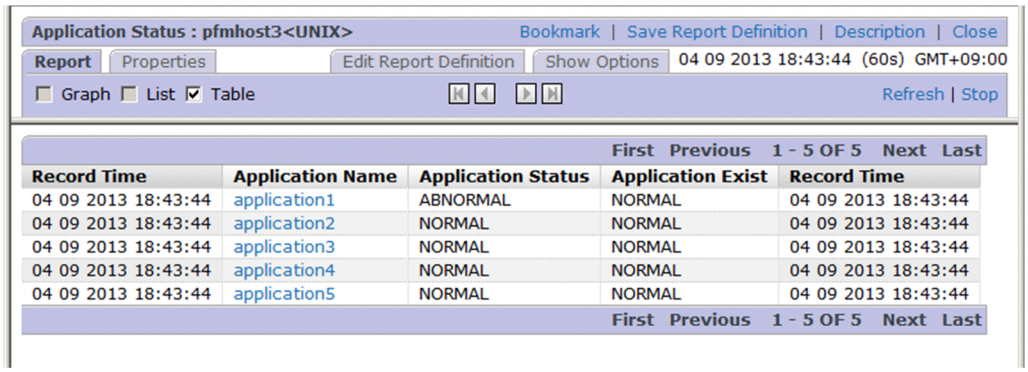
For details about alarms and reports, see *8. Monitoring Templates*.

1. From the menu bar in the Main window, select **Event Monitor**.
   The Event Monitor window appears.



	Date and Time	Agent	Host	Status	Report	Alarm name
	04 10 2013 15:33:23	pfmhost3<UNIX>	pfmhost3	OK	n/a	n/a
	04 10 2013 15:33:23	pfmhost3<UNIX>	pfmhost3	OK		Application S
	04 10 2013 15:31:00	pfmhost3<UNIX>	pfmhost3	Warning	n/a	n/a
	04 10 2013 15:30:01	pfmhost3<UNIX>	pfmhost3	Warning	n/a	n/a
	04 10 2013 15:30:01	pfmhost3<UNIX>	pfmhost3	Warning		Application S
	04 10 2013 15:29:28	pfmhost3<UNIX>	pfmhost3	OK	n/a	n/a
	04 10 2013 15:19:25	pfmhost3<UNIX>	pfmhost3	OK	n/a	n/a
	04 10 2013 14:46:32	pfmhost1<Windows>	pfmhost1	OK	n/a	n/a
	04 10 2013 14:46:32	pfmhost1<Windows>	pfmhost1	OK		Application S
	04 10 2013 14:18:00	pfmhost1<Windows>	pfmhost1	Warning	n/a	n/a
	04 10 2013 14:18:00	pfmhost1<Windows>	pfmhost1	Warning		Application S
	04 10 2013 14:16:01	pfmhost1<Windows>	pfmhost1	OK	n/a	n/a
	04 10 2013 14:16:01	pfmhost1<Windows>	pfmhost1	OK		Application S
	04 10 2013 14:00:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning	n/a	n/a
	04 10 2013 14:00:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning		Application S
	04 10 2013 13:45:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	OK	n/a	n/a
	04 10 2013 13:45:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	OK		Application S
	04 10 2013 13:40:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning	n/a	n/a
	04 10 2013 13:40:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning		Application S
	04 10 2013 13:25:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	OK	n/a	n/a
	04 10 2013 13:25:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	OK		Application S
	04 10 2013 13:20:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning	n/a	n/a
	04 10 2013 13:20:00	inst1[targethost1@pfmhost1]<RM Platform>	target1	Warning		Application S
	04 10 2013 11:36:00	pfmhost1<Windows>	pfmhost1	Warning	n/a	n/a

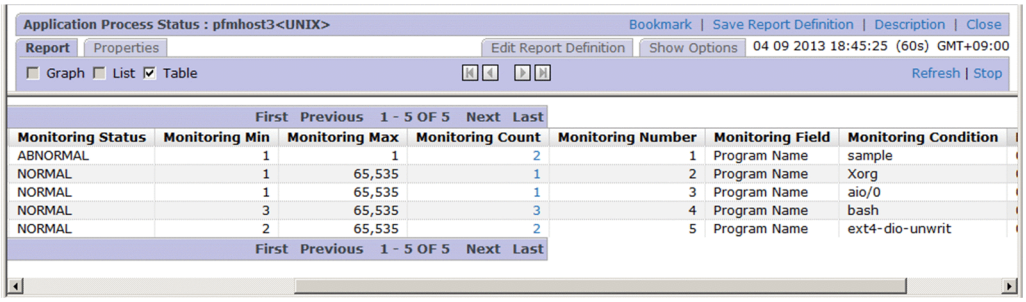
2. Click the report icon for the alarm that occurred.
   The Application Status report appears.



Record Time	Application Name	Application Status	Application Exist	Record Time
04 09 2013 18:43:44	application1	ABNORMAL	NORMAL	04 09 2013 18:43:44
04 09 2013 18:43:44	application2	NORMAL	NORMAL	04 09 2013 18:43:44
04 09 2013 18:43:44	application3	NORMAL	NORMAL	04 09 2013 18:43:44
04 09 2013 18:43:44	application4	NORMAL	NORMAL	04 09 2013 18:43:44
04 09 2013 18:43:44	application5	NORMAL	NORMAL	04 09 2013 18:43:44

3. Check for locations where the value of **Application Status** or **Application Exist** is **ABNORMAL** to identify the application for which a warning has occurred.

4. Under **Application Name**, click the application for which the warning occurred.
   In this case, click **application1**.
   An Application Process Status report appears.



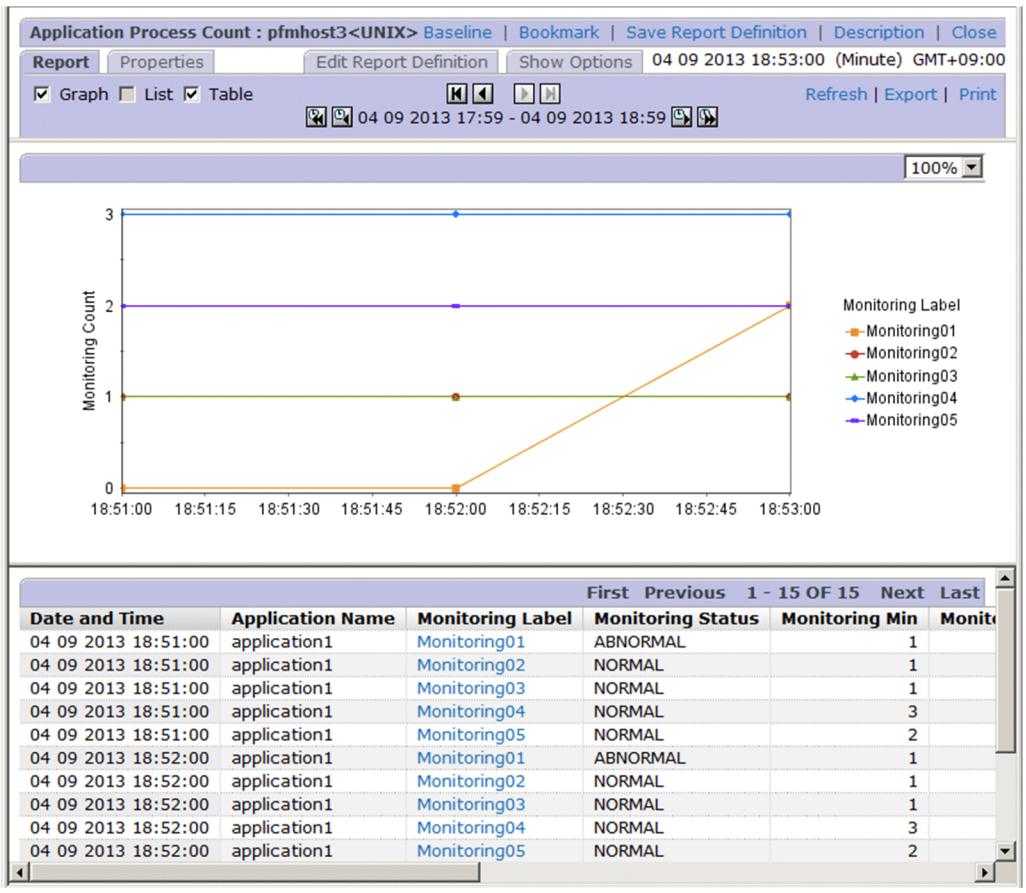
Monitoring Status	Monitoring Min	Monitoring Max	Monitoring Count	Monitoring Number	Monitoring Field	Monitoring Condition
ABNORMAL	1	1	2	1	Program Name	sample
NORMAL	1	65,535	1	2	Program Name	Xorg
NORMAL	1	65,535	1	3	Program Name	aio/0
NORMAL	3	65,535	3	4	Program Name	bash
NORMAL	2	65,535	2	5	Program Name	ext4-dio-unwrit

5. Check for locations where the value of **Monitoring Status** is **ABNORMAL** to identify the process for which a warning has occurred.

   In this case, the warning has occurred in **sample**.

6. If historical data for the `PD_APPD` record is being collected, you can click on the value under **Monitoring Count**, if desired.

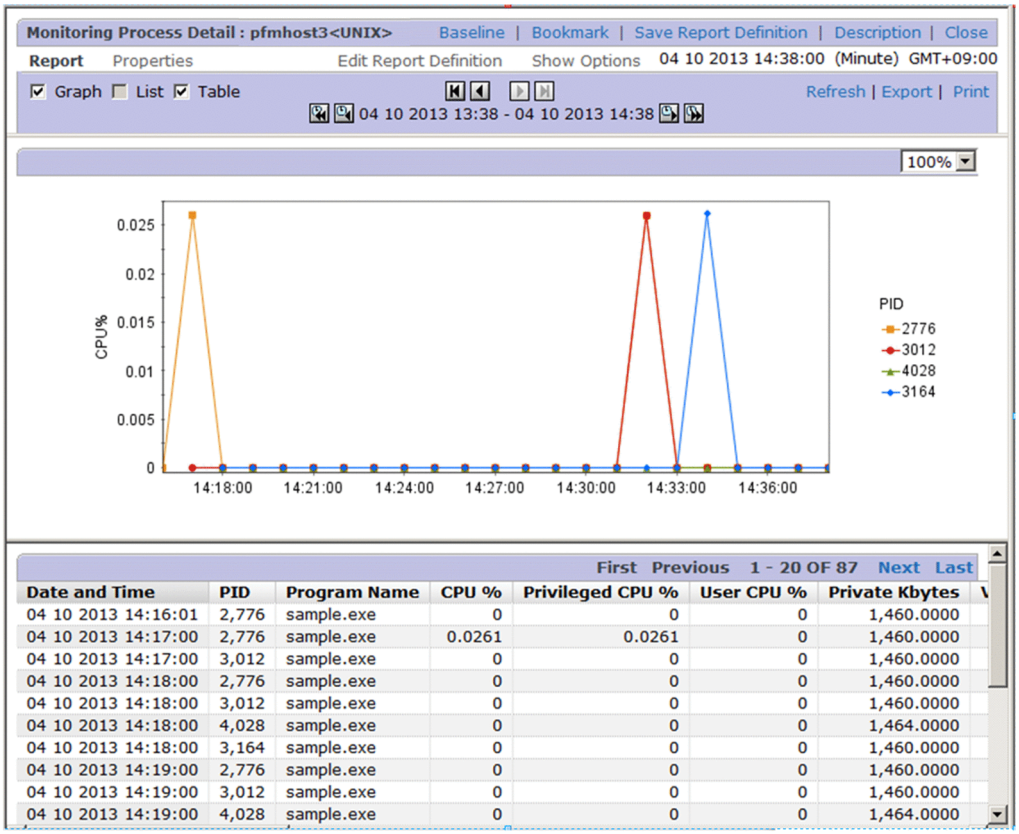
   The Application Process Count report is displayed. You use this report to check the history of changes in the number of processes and the status of each process.



Date and Time	Application Name	Monitoring Label	Monitoring Status	Monitoring Min	Monit
04 09 2013 18:51:00	application1	Monitoring01	ABNORMAL	1	
04 09 2013 18:51:00	application1	Monitoring02	NORMAL	1	
04 09 2013 18:51:00	application1	Monitoring03	NORMAL	1	
04 09 2013 18:51:00	application1	Monitoring04	NORMAL	3	
04 09 2013 18:51:00	application1	Monitoring05	NORMAL	2	
04 09 2013 18:52:00	application1	Monitoring01	ABNORMAL	1	
04 09 2013 18:52:00	application1	Monitoring02	NORMAL	1	
04 09 2013 18:52:00	application1	Monitoring03	NORMAL	1	
04 09 2013 18:52:00	application1	Monitoring04	NORMAL	3	
04 09 2013 18:52:00	application1	Monitoring05	NORMAL	2	

7. If historical data for the `PD_APSI` record is being collected, you can click on the value under **Monitoring Label**, if desired.

   The Monitoring Process Detail report is displayed. You use this report to examine performance information for specific processes for which process monitoring has been set.



Date and Time	PID	Program Name	CPU %	Privileged CPU %	User CPU %	Private Kbytes
04 10 2013 14:16:01	2,776	sample.exe	0	0	0	1,460.0000
04 10 2013 14:17:00	2,776	sample.exe	0.0261	0.0261	0	1,460.0000
04 10 2013 14:17:00	3,012	sample.exe	0	0	0	1,460.0000
04 10 2013 14:18:00	2,776	sample.exe	0	0	0	1,460.0000
04 10 2013 14:18:00	3,012	sample.exe	0	0	0	1,460.0000
04 10 2013 14:18:00	4,028	sample.exe	0	0	0	1,464.0000
04 10 2013 14:18:00	3,164	sample.exe	0	0	0	1,460.0000
04 10 2013 14:19:00	2,776	sample.exe	0	0	0	1,460.0000
04 10 2013 14:19:00	3,012	sample.exe	0	0	0	1,460.0000
04 10 2013 14:19:00	4,028	sample.exe	0	0	0	1,464.0000

Note:

Depending on the timing of the display of the Application Status report from the Event Monitor, or the display of the Application Process Status report from the Application Status report, once the alarm status has returned to normal you may not be able to identify the process of concern in these real-time reports. In such a case, browse the Event Monitor or Application Process Count report (historical reports) to check for changes in status following the occurrence of the alarm.

7.2.9 Notes on collecting information about the operating status of processes

- If an asterisk (*) is specified as a monitoring condition for the process operation and non-operation information collection functionality, the functionality sometimes improperly detects command lines in the system that contain an asterisk. This is because the asterisk in the monitoring condition acts like a wildcard.

7.3 Settings for collecting information about the application operating status

The following explains how to set, confirm, change, and delete user records for collecting information about the application operating status, and provides an example of alarm operation.

Note that the functionality for collecting information about the application operating status is associated with versions 09-00 and earlier. Collection of information about the operating status of processes is an extension in version 10-00 of this original functionality.

For details about the differences between the earlier functionality for collecting information about the application operating status and collection of information about the operating status of processes, see *7.2.2 Differences in functionality from previous versions*.

7.3.1 User-defined record settings for collecting application operating status information settings

The following explains how to specify the user-defined record settings for collecting information about the application operating status.

To perform setup, first create an instance, and then set the instance properties (such as the monitored application name and threshold).

(1) To create an instance:

1. From the monitoring console browser, log in to PFM - Web Console.

   The Main window appears.

2. In the navigation frame of the Main window, click the **Services** tab.

   The **Services** page appears.

3. From the navigation frame, expand the contents of the `Machines` directory.

   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on the host.

   The name of each service is displayed by service ID. For details about service IDs, see *C. List of Identifiers*, and the section explaining the service naming conventions in the appendix in the *JP1/Performance Management Planning and Configuration Guide*.

   The format of the service ID differs depending on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter explaining Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the PFM - Agent host directory, and then select Agent Collector service.

   Select *host-name***<UNIX>** (Agent Collector service). A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.

   The **Service Properties** window appears.

6. Select the **ADDITION OR DELETION A SETTING** tree.

7. Enter an instance name for **ADD AN APPLICATION MONITORING SETTING** at the bottom of the information frame, and click the **OK** button.

   An instance name tree is generated in the **Application monitoring setting** tree of the **Service Properties** window.

> **Important**
>
> - The instance name entered for **ADD AN APPLICATION MONITORING SETTING** can be specified by the user. The instance name specified here is stored in the `Application Name` field of the `PD_APP` record, and is used to identify the application.
> - In **ADD AN APPLICATION MONITORING SETTING**, you can specify a character string of 1-63 bytes. The character string can consist of alphanumeric characters and symbols except the following characters:
> Tab (`\t`) `\ : ; , * ? " ' < > |`
> - You can set a maximum of 64 instances.

(2) To set instance properties:

1. After finishing the above procedure, display the **Service Properties** window again and select the instance name tree generated in the **Application monitoring setting** tree.

   The property information entry window appears at the bottom of the information frame.

2. Set properties.

   Set the process type, process name, and minimum and maximum values for the process count. You can set information for multiple processes. The following table lists the properties you can set.

Table 7–9: Monitoring field properties

Item	Parameter name	Explanation	Corresponding field in the PD_APP record
Process type	`Process`*XX* `Kind`	Select `Execute` or `Command Line`. Note that evaluation is not performed when `None` is selected. • When `Execute` is selected, the value of the `Program` field in the `PD` record is used for evaluation. • When `Command Line` is selected, the value of the `Argument List` field in the `PD` record is used for evaluation.	`Process`*XX* `Kind`
Process name	`Process`*XX* `Name`	Specify a process name of no more than 127 bytes.	`Process`*XX* `Name`
Minimum and maximum thresholds for the number of processes	`Process`*XX* `Range`	Specify the minimum and maximum thresholds for the number of processes in the format *m*−*n* (for example, `1-2`). Only one value can be specified without a hyphen to set the minimum value and maximum value to the same value. For example, when `10` is specified, `10-10` is set. You can specify values in the range from 0 to 65535.	`Process`*XX* `Range`

Legend:

XX: A two-digit numeric value in the range from 01 to 15

Important

- The value specified for the `Process`*XX*`Name` property is used to evaluate whether the application is running. Note that when the `Process`*XX*`Name` field in the `PD_APP` record is displayed in a report in PFM - Web Console, only the first 31 bytes are displayed.
- If there are multiple processes for which the first 31 bytes of each name are the same, set up an instance for each process so that the process can be monitored, and set the evaluation results (the value of the `Process`*XX*`Name` field in the `PD_APP` record) to be reported by alarm. Note that because the value of the `Process`*XX*`Name` field in the `PD_APP` record is the same for all instances, alarms will not be reported properly. Accordingly, instead of using the `Process`*XX*`Name` field in the `PD_APP` record, use the value of the `Process`*XX*`Status` field for setting alarm notification. For example settings, see *7.3.4(4) Monitoring whether a given process is running when there are multiple processes for which the first 31 bytes of the names are the same*.
- Alphanumeric characters and symbols can be specified for the `Process`*XX*`Name` property.
- Wildcard characters `*` and `?` can also be specified for the `Process`*XX*`Name` property. An asterisk (`*`) represents zero or more instances of any character, and a question mark (`?`) represents any single character.
- When you select `Execute` for the `Process`*XX*`Kind` property, a value equivalent to the `Program` field of the `PD` record must be specified for the `Process`*XX*`Name` property. The value of the `Program` field of the `PD` record is almost equal to the value displayed when the `ps -e` command is executed. (This value might be different from the command execution result depending on the type and version of the OS.)
- When you select `Command Line` for the `Process`*XX*`Kind` property, a value equivalent to the `Argument List` field of the `PD` record must be specified for the `Process`*XX*`Name` property. The value of the `Argument List` field of the `PD` record is almost equal to the value displayed when the `ps -ef` command is executed. (This value might be different from the command execution result depending on the type and version of the OS.)

3. Click the **OK** button.
   The specified settings are applied.

(3) Setting the range of data for collecting information about the application operating status

When PFM - Agent for Platform is running on a virtualized system, you can set the `ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT` property to restrict the environment from which process information is to be collected.

1. After creating the instance, display the **Service Properties** window again and select the **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT** tree in the **ADDITION OR DELETION A SETTING** tree.
   The property information entry window appears at the bottom of the information frame.

> **Note**
>
> When creating an instance, you can concurrently set **ADD AN APPLICATION MONITORING SETTING** and **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT**.

2. Set properties.

   For details about the properties, see *2.3.22(1) Identifying processes that have the same name in a virtualized system*.

> **Important**
>
> - The virtual environment ID entered for **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT** can be specified by the user. The ID specified here is stored in the Virtual Env ID field of the `PD_APP` record.
> - In **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT**, you can specify a value from 0 to 2147483647. You cannot specify characters.
> - As in the case of **ADD AN APPLICATION MONITORING SETTING**, a maximum of 64 instances can be created with **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT**.

7.3.2 Checking or changing user-defined records for collecting application operating status information

To check or change the user record settings for collecting information about the application operating status:

1. From the monitoring console browser, log in to PFM - Web Console.

   The Main window appears.

2. In the navigation frame of the Main window, click the **Services** tab.

   The **Services** page appears.

3. From the navigation frame, expand the contents of the `Machines` directory.

   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on the host.

   The name of each service is displayed by service ID. For details about service IDs, see *C. List of Identifiers* and the section explaining the service naming conventions in the appendix in the *JP1/Performance Management Planning and Configuration Guide*.

   The format of the service ID differs depending on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter explaining Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the PFM - Agent host directory, and then select Agent Collector service.

   Select *host-name***<UNIX>** (Agent Collector service). A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.

   The **Service Properties** window appears.

6. Expand the **Application monitoring setting** tree, and select the tree for the instance name to be checked.

The properties are displayed.

7. Check the property settings, and click the **OK** button.

7.3.3 Deleting user-defined records for collecting application operating status information

To delete the user-defined record settings for collecting information about the application operating status:

1. From the monitoring console browser, log in to PFM - Web Console.
   The Main window appears.

2. In the navigation frame of the Main window, click the **Services** tab.
   The **Services** page appears.

3. From the navigation frame, expand the contents of the **Machines** directory.
   A directory that has the name of the host on which the Performance Management service is installed appears. Expand this directory to display the services installed on the host.
   The name of each service is displayed by service ID. For details about service IDs, see *C. List of Identifiers* and the section explaining the service naming conventions in the appendix in the *JP1/Performance Management Planning and Configuration Guide*.
   The format of the service ID differs depending on whether the product name display functionality is enabled. For details about the product name display functionality, see the chapter explaining Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

4. Expand the contents of the PFM - Agent host directory, and then select Agent Collector service.
   Select *host-name***<UNIX>** (Agent Collector service). A check mark is displayed next to the Agent Collector service.

5. Select the **Properties** method in the method frame.
   The **Service Properties** window appears.

6. Select the **ADDITION OR DELETION A SETTING** tree.

7. Select the name of the instance to be deleted from **DELETE AN APPLICATION MONITORING SETTING** at the bottom of the information frame, and click the **OK** button.
   The settings are deleted.

7.3.4 Example of using an alarm when collecting information about the application operating status

This subsection provides an example of using an alarm with the function that collects information about the application operating status.

(1) Monitoring the startup status of a specific process

The following explains settings for monitoring whether a specific process is running, from several monitoring targets.

For environments in which only one of the following processes is normally running, the following procedure sets an abnormal alarm to be reported when multiple processes (two or more) are started or stopped.

Table 7–10: Example of specific process start

Process name	Command line
GyoumuProcess	GyoumuProcess -a 1

1. Set the instance name for **ADD AN APPLICATION MONITORING SETTING** in the **ADDITION OR DELETION A SETTING** tree as follows:

   GyoumuProcess Monitor

2. Set the GyoumuProcess Monitor property generated in the **Application monitoring setting** tree as follows:

   ```
   Process01 Kind: Select Execute.
   Process01 Name: Enter GyoumuProcess. (Enter the process name)
   Process01 Range: Specify 1-1.
   ```

   As a result of the above settings, when the process is running, the values of the Process01 Count, Process01 Status, and Application Status fields in the PD_APP record are displayed as follows in the report.

   Table 7–11: Results for the values of each field in the PD_APP record

Field name	Value
Process01 Count	1#1
Process01 Status	NORMAL#2
Application Status	NORMAL#2

   #1

   This indicates the number of corresponding processes running.

   #2

   This indicates that there is no issue.

3. Specify alarm settings as follows:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Application Name.
   Condition: Select =.
   Abnormal value: Enter GyoumuProcess Monitor.
   Warning value: Enter GyoumuProcess Monitor.
   ```

   After specifying the above settings and clicking the **Add** button, specify these additional settings:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Application Status.
   Condition: Select <>.
   Abnormal value: Enter NORMAL.
   Warning value: Enter NORMAL.
   ```

Alarm conditions are evaluated using AND. To monitor only the results for **Application Status** without specifying an instance, specify only **Application Status**, <>, and NORMAL.

(2) Monitoring the startup status of a process containing arguments

The following explains settings for monitoring whether a process containing arguments is running.

The following procedure sets an abnormal alarm to be reported when the following process containing arguments stops.

Table 7–12: Example of starting the process with arguments

Process name	Command line
GyoumuProcess	GyoumuProcess -a 1

1. Set the instance name for **ADD AN APPLICATION MONITORING SETTING** in the **ADDITION OR DELETION A SETTING** tree as follows:
   GyoumuProcess Monitor

2. Set the GyoumuProcess Monitor property generated in the **Application monitoring setting** tree as follows:

   ```
   Process01 Kind: Select Command Line.
   Process01 Name: Enter GyoumuProcess. (Enter the command line)
   Process01 Range: Enter 1-1.
   ```

   As a result of the above settings, when the process is running, the values of the Process01 Count, Process01 Status, and Application Status fields in the PD_APP record are displayed as follows in the report.

   Table 7–13: Results for the values of each field in the PD_APP record

Field name	Value
Process01 Count	1#1
Process01 Status	NORMAL#2
Application Status	NORMAL#2

   #1
   This indicates the number of corresponding processes running.

   #2
   This indicates that there is no issue.

3. Specify alarm settings as follows:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Application Name.
   Condition: Select =.
   Abnormal value: Enter GyoumuProcess Monitor (the instance name set in step
    1).
   Warning value: Enter GyoumuProcess Monitor (the instance name set in step
   1).
   ```

   After specifying the above settings and clicking the **Add** button, specify these additional settings:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Application Status.
   Condition: Select <>.
   Abnormal value: Enter NORMAL.
   Warning value: Enter NORMAL.
   ```

Alarm conditions are evaluated using AND. To monitor only the results for **Application Status** without specifying an instance, specify only **Application Status**, <>, and NORMAL.

(3) Monitoring whether multiple processes are all running

The following explains settings for monitoring whether monitored processes are all running.

The following procedure sets an abnormal alarm to be reported when at least one of the following five processes has stopped, but not to be reported when they are all running.

Table 7–14: Example of startup for five processes

Process name	Command line
GyoumuProcess1	GyoumuProcess1 -a 1
GyoumuProcess2	GyoumuProcess2 -a 2
GyoumuProcess3	GyoumuProcess3 -a 3
GyoumuProcess4	GyoumuProcess4 -a 4
GyoumuProcess5	GyoumuProcess5 -a 5

1. Set the instance name for **ADD AN APPLICATION MONITORING SETTING** in the **ADDITION OR DELETION A SETTING** tree as follows:
   GyoumuProcess Monitor

2. Set the GyoumuProcess Monitor property generated in the **Application monitoring setting** tree as follows:

   ```
   Process01 Kind: Select Execute.
   Process01 Name: Enter GyoumuProcess*.#
   Process01 Range: Enter 1-5.
   ```

   #
   The wildcard character * is used for the process number. The wildcard character ? can also be used, in place of a single character.

   As a result of the above settings, when the five processes are running, the values of the Process01 Count, Process01 Status, and Application Status fields in the PD_APP record are displayed as follows in the report.

   Table 7–15: Results for the values of each field in the PD_APP record

Field name	Value
Process01 Count	5#1
Process01 Status	NORMAL#2
Application Status	NORMAL#2

   #1
   This indicates the number of corresponding processes running.

   #2
   This indicates that there is no issue.

3. Specify alarm settings as follows:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Process01 Count.
   Condition: Select <.
   ```

```
Abnormal value: Enter 5.
Warning value: Enter 5.
```

If all five processes are running, no alarm is reported. An alarm is reported when at least one process stops. Since alarm conditions are evaluated using AND, an abnormal alarm cannot be set to report for a running process count other than from 1 to 5.

(4) Monitoring whether a given process is running when there are multiple processes for which the first 31 bytes of the names are the same

The following explains settings for monitoring whether a given process is running when there are multiple processes for which the first 31 bytes of the names are the same.

The following procedure sets an abnormal alarm to be reported when either of the following two processes is running and multiple `1234567890123456789012345678901A` processes are started or stopped.

Table 7–16: Example of processes running for which the first 31 bytes of the names are the same

Process name	Command line
`1234567890123460000000000000000A`	`1234567890123456789012345678901A -A`
`1234567890123456789012345678901B`	`1234567890123456789012345678901B -B`

1. Set the instance name for **ADD AN APPLICATION MONITORING SETTING** in the **ADDITION OR DELETION A SETTING** tree as follows:

   `Long Name Process Monitor`

2. Set the `Long Name Process Monitor` property generated in the **Application monitoring setting** tree as follows:

   ```
   Process01 Kind: Select Execute.
   Process01 Name: Enter 1234567890123456789012345678901A.
   Process01 Range: Enter 1-1.
   ```

   As a result of the above settings, when only one `1234567890123456789012345678901A` process is running, the values of the `Process01 Count`, `Process01 Status`, and `Application Status` fields in the `PD_APP` record are displayed as follows in the report.

   Table 7–17: Results for the values of each field in the PD_APP record

Field name	Value
`Process01 Count`	`1`#1
`Process01 Status`	`NORMAL`#2
`Application Status`	`NORMAL`#2

   #1

   This indicates the number of corresponding processes running.

   #2

   This indicates that there is no issue.

3. Specify alarm settings as follows:

   ```
   Record: Select Application Summary (PD_APP).
   Monitored field: Select Application Name.
   ```

```
Condition: Select =.
Abnormal value: Enter Long Name Process Monitor (the instance name set in
step 1).
Warning value: Enter Long Name Process Monitor (the instance name set in s
tep 1).
```

After specifying the above settings and clicking the **Add** button, specify these additional settings:

```
Record: Select Application Summary (PD_APP).
Monitored field: Select Application Status.
Condition: Select <>.
Abnormal value: Enter NORMAL.
Warning value: Enter NORMAL.
```

In addition to the above monitoring, to monitor the `1234567890123456789012345678901B` process using a monitoring method other than what has already been used, use an instance name different than the one above, and add the instance name to the condition in the `Application Name` field of the alarm setting.

(5) Monitoring whether a process in the global zone environment is running when PFM - Agent for Platform is installed in a Solaris global zone environment

The following explains settings for monitoring whether a process is running in a monitored global zone.

When the following process is running in both of the global zone environment and a non-global zone environment, the following procedure sets an abnormal alarm to be reported when multiple processes are running or stopped in the global zone.

Table 7–18: Example of the process running in the global zone environment

Process name	Command line
`GyoumuProcess.exe`	`GyoumuProcess`

1. Set the following for **ALL Zone Collection for Process** in the **Agent Configuration** tree.

   ```
   Yes
   ```

2. Set the instance name for **ADD AN APPLICATION MONITORING SETTING** in the **ADDITION OR DELETION A SETTING** tree. Then, specify as follows for **ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT**:

   ```
   ADD AN APPLICATION MONITORING SETTING:Global Zone Process Monitor
   ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT:0
   ```

3. Click the **OK** button.

4. Set the `Global Zone Process Monitor, Virt ID = 0` property generated in the **Application monitoring setting** tree as follows:

   ```
   Process01 Kind: Select Execute.
   Process01 Name: Enter GyoumuProcess.
   Process01 Range: Enter 1-1.
   ```

As a result of the above settings, when only one GyoumuProcess process is running, the values of the Process01 Count, Process01 Status, and Application Status fields in the PD_APP record are displayed as follows.

Table 7–19: Results for the values of each field in the PD_APP record

Field name	Value
Process01 Count	1[#1]
Process01 Status	NORMAL[#2]
Application Status	NORMAL[#2]

#1

This indicates the number of corresponding processes running.

#2

This indicates that there is no issue.

5. In the navigation frame of the Main window, click the **Alarms** tab.
   The Alarm Tree window appears.

6. Expand the contents of the UNIX directory to display the existing alarm tables.

7. Select **New Alarm Table** in the method frame, and create an alarm table. Set the following basic information in the new alarm table:

```
Product: Select UNIX(7.6).
Alarm table name: Enter AlarmTable1.
Alarm name: Enter GyoumuProcess Alarm.
```

8. Click the **Next** button.

9. Specify alarm settings as follows:

```
Record: Select Application Summary (PD_APP).
Monitored field: Select Application Name.
Condition: Select =.
Abnormal value: Enter Global Zone Process Monitor (the instance name set i
n step 2).
Warning value: Enter Global Zone Process Monitor (the instance name set in
 step 2).
```

After specifying the above settings and clicking the **Add** button, specify these additional settings:

```
Record: Select Application Summary (PD_APP).
Monitored field: Select Application Status.
Condition: Select <>.
Abnormal value: Enter NORMAL.
Warning value: Enter NORMAL.
```

7.4 Settings for collecting user-specific performance data

This section provides an overview of the function that collects user-specific performance data, and explains how to set up the function.

7.4.1 Overview of the functionality for collecting user-specific performance data

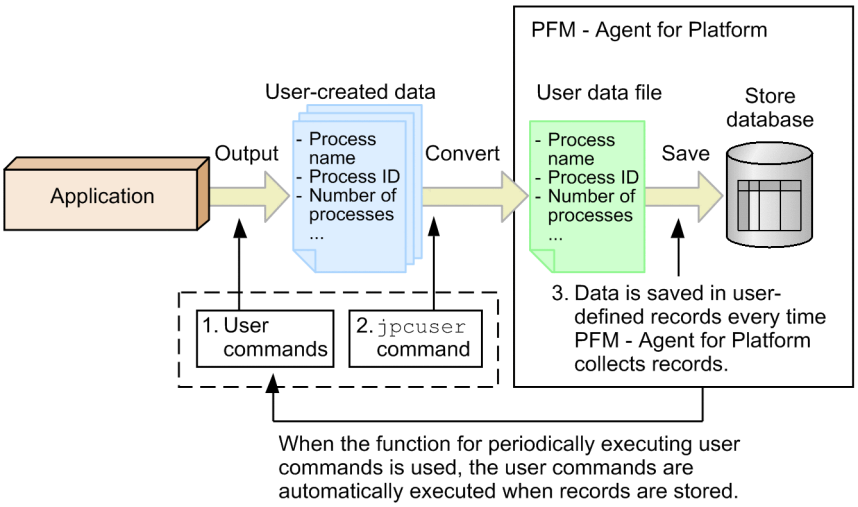
The following explains functionality for collecting user-specific performance data and functionality for periodically executing user commands.

(1) Functionality for collecting user-specific performance data

This functionality uses the `jpcuser` command to convert custom performance data output by users to a text file, into a format that can be stored in records provided by PFM - Agent for Platform (`PD_UPD`, `PD_UPDB`, `PI_UPI`, `PI_UPIB`, `PI_XUI1` to `PI_XUI5`). To use this functionality for collecting user-specific performance data, a command must be created to output performance data to text files ahead of time.

The following figure shows how user-specific performance data is collected.

Figure 7–1: Mechanism for collecting user-specific performance data



The following describes the processing corresponding to the numbers in the figure.

1. User commands are executed to create user-defined data.

   The *user commands* collect performance data, such as process name, process ID, and number of processes, and output the collected data to a text file. The data in the text file is called *user-created data*.

   These user commands must be created as scripts beforehand.

2. The `jpcuser` command is executed to convert the user-created data.

   The `jpcuser` command converts the user-created data into a file in a format that can be managed by PFM - Agent for Platform. The file resulting from the conversion is called a *user data file*.

3. The contents of the user data file are saved in user-defined records every time PFM - Agent for Platform performs record collection.

PFM - Web Console must be set beforehand so that PFM - Agent for Platform collects the records from the user data file.

When no user data file has been created due to a failure to execute a user command or for other reasons, the KAVF10707-W message# is output, and collection of the applicable records is skipped.

#

- Suppose that PFM - Agent for Platform is configured to execute user commands prior to record collection by using the functionality for periodically executing user commands. In this case, if PFM - Agent for Platform fails to activate a user command or if a user command execution times out, this message is not output.
- When PFM - Agent for Platform repeatedly fails to execute a user command, this message is output only once an hour for the same record.

To collect performance data periodically, use the functionality for periodically executing user commands to set a user command, and the `jpcuser` command, to execute automatically.

Important

When outputting a file specified for the `jpcuser` command argument, or file in a batch file or script that executes the `jpcuser` command, specify a directory other than the installation directory.

(2) Functionality for periodically executing user commands

This functionality executes a user command from PFM - Agent for Platform at a fixed interval without using a schedule functionality such as the `cron` command. The method for creating user data files from user commands is the same as described in *(1) Functionality for collecting user-specific performance data*.

The functionality for periodically executing user commands runs in accordance with the **Collection Interval** setting specified for user records. User commands are therefore executed during historical data collection and alarm monitoring data collection but not during real-time data collection.

As for the timing of executing user commands, you can select one of the following options: after record collection (after user data files are loaded) and before record collection (before user data files are loaded). By default, "before data collection" is set.

The following figure shows the flow of processing for functionality for periodically executing user commands.

Figure 7–2: Flow of processing for functionality for periodically executing user commands (when user commands are to be executed after record collection)

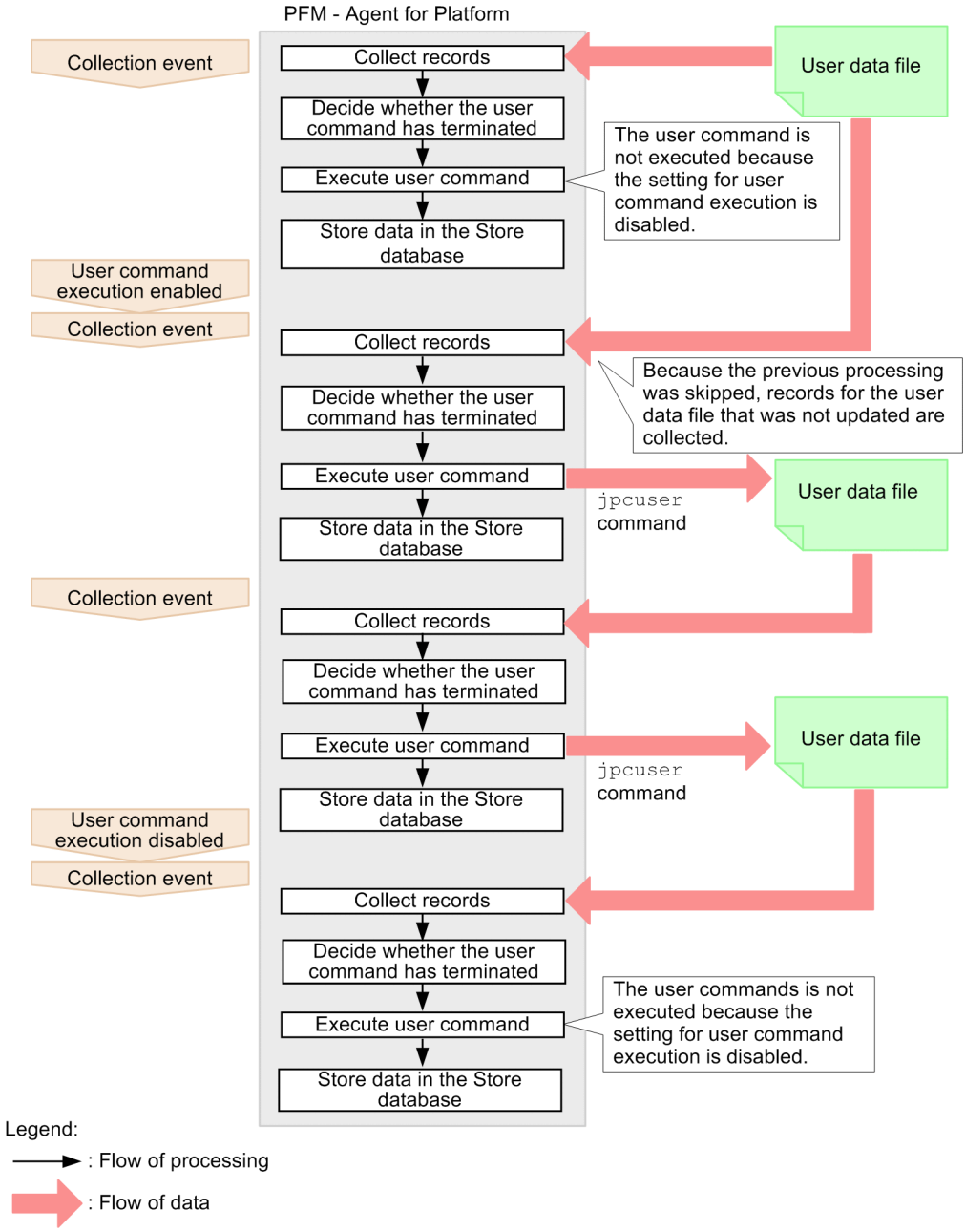
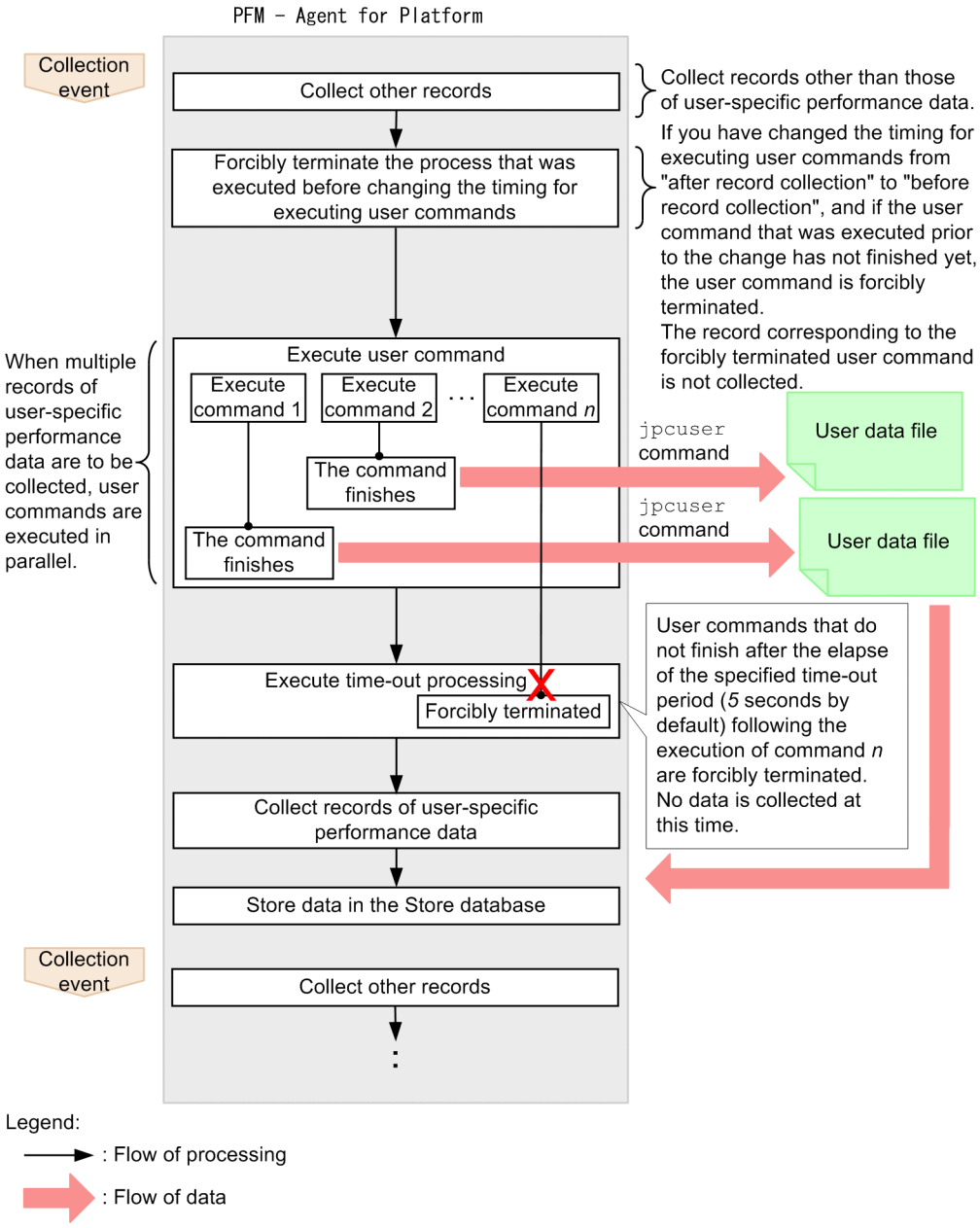


Figure 7–3: Flow of processing for functionality for periodically executing user commands (when user commands are to be executed before record collection)



Functionality for periodically executing user commands determines whether the previously started user command has terminated, and skips user command processing if it is executing.

> **Note**
>
> For versions of PFM - Web Console earlier than 09-00, functionality for periodically executing user commands cannot batch distribute properties to more than one PFM - Agent.

(3) Notes on functionality for periodically executing user commands

Executable files

The file formats that can be executed by functionality for periodically executing user commands are as follows:

- Executable file#
- Shell script file#

#

The execution attribute must be added.

Accounts

Use the root user permission account to execute functionality for periodically executing user commands. Make sure that the root user permission account can access the following files and resources:

- Files (user commands) specified for functionality for periodically executing user commands
- Resources referenced or updated from those files (user commands)

To execute a file in the NFS mount directory, make sure that the root user permission for the host account can access the file.

Environment variables

The environment variables valid for executing functionality for periodically executing user commands are the root-user permission environment variables defined when the Performance Management program service starts up. Profile information is not loaded when functionality for periodically executing user commands is executed.

Current directory

The following PFM - Agent for Platform service directories are used as the current directory for executing functionality for periodically executing user commands:

- Physical host environment: `/opt/jp1pc/agtu/agent`
- Logical host environment: *environment-directory*`/jp1pc/agtu/agent`

File `umask`

The `umask` value for using functionality for periodically executing user commands is set to `000` (file permission is set to `777`). To change the `umask` value, set `umask` again in the script file or program to be executed.

Notes on executing user commands before record collection

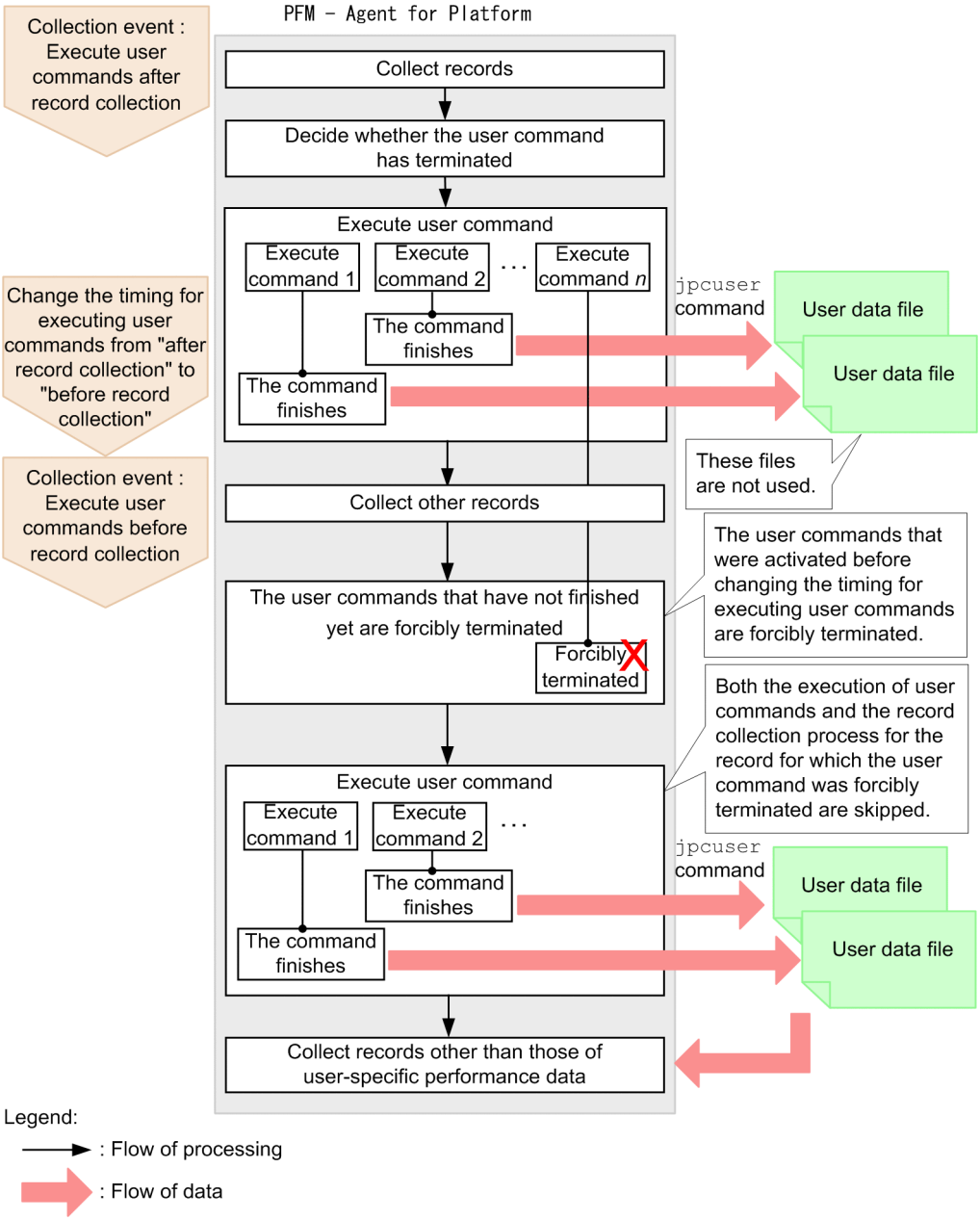
- When a user command times out, it is forcibly terminated. Processes activated from inside the user command, however, are not forcibly terminated. If necessary, add a new user command that allows you to end processes activated from the previously activated user command when the processes have timed out.
- When no user command is specified, when an attempt to activate a user command fails, or when a user command times out, the record collection process corresponding to the user command is skipped.
- User commands are executed after other records are collected. For this reason, there is a time lag between the timing of a collection event and the execution of user commands.
- Even when a user command is being executed, the Agent Collector service status is "Busy".

Notes on changing the timing of executing user commands

- If you change the timing of executing user commands from after record collection to before record collection, the results of the user commands executed prior to this change will not be collected. If the user commands that were executed prior to the change have not finished by the time the record collection process immediately after the change begins, these user commands are forcibly terminated.

  Note that both the execution of user commands and the record collection process for the record for which the user command was forcibly terminated are skipped. (The execution of user commands and the record collection process are resumed the next time a record is collected.)

Figure 7–4: Notes on changing the timing of executing user commands from after record collection to before record collection



Other notes

- Programs that require interactivity cannot be executed.
- Programs containing the `stty`, `tty`, `tset`, or `script` command that requires an interactive environment cannot be executed.
- Resident programs (programs that do not terminate) cannot be executed.
- Programs without the execution attribute added cannot be executed.
- Do not set up programs on removable disks or other disks that cannot be set up.
- The contents of the standard output and standard error output cannot be obtained for executed programs.
- When performing file output from an executed program, use an absolute path to specify the output destination file. If an absolute path is not specified, the directory for the PFM - Agent for Platform service is used as the current directory.

7.4.2 Settings for collecting user-specific performance data

To collect user-specific performance data:

1. Determine the information to be stored in fields.
2. Create user commands.
3. Set the scheduler to collect user-specific performance data periodically.
4. Specify the settings for collecting information from the user data file.

The following subsections describe the steps in this procedure.

(1) Determining the information to be stored in fields

The fields of a user-defined record store two types of information, key information and data information. You will need to consider what to store as key information and what to store as data information.

(a) Key information

A user-defined record for storing user-specific performance data is a multi-instance record in which one or more rows can be stored by one collection run. To identify each record instance in one user-defined record, key information must be set. If you specify multiple user-created data files in the `jpcuser` command, you must set key information that uniquely identifies each record instance across all of the specified files. The following table shows the types of key information.

Table 7–20: Types of key information

Type	Field name	Explanation
Transaction type	`Trans Type`	Identifies the instance type.
Transaction key	`Trans Data Key` (numeric type)	Identifies each of the instances that have the same transaction type.
	`Trans String Key` (string type)	

The transaction type is used to identify the type of the performance data. For example, assume that information about a database is stored in one record and information about a Web server is stored in another record. In this case, you can use `DATABASE` and `WEB` as transaction types to indicate which type of information (information about a database or information about a Web server) is stored.

When there are multiple instances that have the same transaction type, the transaction key is used to identify each instance. If neither the `Trans Data Key` field nor the `Trans String Key` field is set, or the same value is set for multiple transaction keys, the record instances cannot be identified uniquely. As a result, the first instance of each record is used.

(b) Data information

As data information, user-defined records can store three types of numeric data (`double`, `long`, and `ulong` types), three lengths of string data, and time data. The number of data items that can be stored differs depending on the user-defined record. For numeric data of the `PI` record type, either *average* or *cumulative* can be selected as the consolidation rule.

Select the user-defined record to be used based on the performance data to be collected. Note that a user-defined record that can store a larger amount of information consumes a larger amount of memory and other resources. We recommend that you select the user-defined record whose size is the minimum necessary.

The following table shows the number of fields for each type of user-defined record.

Table 7–21: Number of fields for each type of user-defined record

Record type	User-defined record type	Number of fields		
		Numeric data	String data	Time data
PD record type	User Data Detail (PD_UPD)	2 × 3 = 6	1 + 2 + 4 = 7	1
	User Data Detail - Extended (PD_UPDB)	5 × 3 = 15	5 + 5 + 5 = 15	1
PI record type	User Data Interval (PI_UPI)	4 × 3 = 12	1 + 2 + 4 = 7	1
	User Data Interval - Extended (PI_UPIB)	10 × 3 = 30	5 + 5 + 5 = 15	1
	User Data Interval - Expanded *n*#1 (PI_XUI*n*#1)	60 × 1 = 60#2	1#3 + 2#4 = 3	1

#1
n denotes a number in the range from 1 to 5.

#2
These are all double type.

#3
This is the number of 128-byte character strings.

#4
This is the number of 64-byte character strings.

The following table shows the criteria for selecting the recommended user-defined record.

Table 7–22: Criteria for selecting the recommended user-defined record

Will cumulative data be stored as the performance data?	Will many types of performance data be stored?	Recommended user-defined record
Yes	No	PI_UPI
Yes	Yes	PI_UPIB, PI_XUI1 to PI_XUI5
No	No	PD_UPD
No	Yes	PD_UPDB

(2) Creating user commands

User commands are scripts that are used to collect performance data to generate user-created data. You must code the scripts so that performance data is output in the format used for user-created data files.

For details about the format of user-created data files, see *7.4.6 Format of user-created data files*.

Store the user-created data at a path that can be specified by the -file option when the jpcuser command is executed. For details, see the description of the arguments in *7.4.5 Format of the jpcuser command*.

To verify the user-created data output by the user commands, execute the jpcuser command in the following format:

- Physical host environment:
  /opt/jp1pc/agtu/agent/jpcuser/jpcuser PI_UPI -file *user-created-data* -debug 1

- Logical host environment:
  `/opt/jp1pc/agtu/agent/jpcuser/jpcuser PI_UPI -file` *user-created-data* `-debug 1 -lhost` *logical-host-name*

When the command is executed, the following debug log file is generated:

- Physical host environment:
  `/opt/jp1pc/agtu/agent/jpcuser/debug/jpcuser_dbg_01.log`
- Logical host environment:
  environment-directory`/jp1pc/agtu/agent/jpcuser/debug/jpcuser_dbg_01.log`

Use the debug log file to check for errors.

For details about the `jpcuser` command, see *7.4.5 Format of the jpcuser command*.

(3) Setting a scheduler to collect user-defined performance data periodically

The following explains how to set up the functionality for periodically executing user commands, to periodically collect user-specific performance data.

To periodically collect user-specific performance data.

1. Set up user record collection in PFM - Web Console.
   The execution interval for functionality for periodically executing user commands depends on the `Collection Interval` setting for each user record.

2. Set the properties for functionality for periodically executing user commands in PFM - Web Console.
   In PFM - Web Console, set the following properties for each user record to run functionality for periodically executing user commands. The method for setting these properties is the same for `PD_UPD`, `PD_UPDB`, `PI_UPI`, `PI_UPIB`, and `PI_XUI1` to `PI_XUI5` records.

Note:

Note that, when you use PFM - Web Console to specify the settings for periodically collecting user-specific performance data, the service to be selected differs depending on whether a physical host or a logical host is used:

- Physical host environment: `UNIX`*physical-host-name* or `UA1`*physical-host-name*
- Logical host environment: `UNIX`*logical-host-name* or `UA1`*logical-host-name*

Figure 7–5: Properties for functionality for periodically executing user commands

Service properties

```
host-name<UNIX>
 |- General
 |- System
 |- Network Services
      :
 |- User Command Setting
    |- PD_UPD
    |- PD_UPDB
    |- PI_UPI
    |- PI_UPIB
    |- PI_XUI1
    |- PI_XUI2
    |- PI_XUI3
    |- PI_XUI4
    |- PI_XUI5
```

Table 7–23: Setting properties for user records

Property	Value	Description	Default value
User Command Setting - `User Command Execution Timing`	`After`/`Before`	By using the functionality for periodically executing user commands, specify when to execute user commands. • After: Execute user commands after record collection. • Before: Execute user commands before record collection.	After
User Command Setting - `User Command Timeout`	Integer in the range from 1 to 86,400	If you select `Before` as the `User Command Execution Timing` property of the functionality for periodically executing user commands, specify the time (seconds)# after which the execution of user commands is discontinued. #: The specified time when the execution of user commands is discontinued must not affect the timing of other record collection processing.	5
User Command Setting - *record-name* - `Execute`	`Yes`/`No`	Specify whether to use functionality for periodically executing user commands. • `Yes`: Use the functionality • `No`: Do not use the functionality	`No`
User Command Setting - *record-name* - `UserCommand`	Absolute path	Specify the absolute path for user commands. The maximum length of the string that can be specified for an absolute path is 255 bytes. Alphanumeric characters and symbols except for the following characters can be specified: `|` `<` `>`	Blank

Notes:

1. When the `Execute` property is set to `Yes` and the `UserCommand` property is blank, the KAVF10203-W message is output, and the user command is not executed.

2. If the specified user command does not exist, or the user command does not have execution permissions, the KAVF10013-W message is output.

3. When a logical host is used, you can use the `UserCommand` property to specify the path to the user commands located on the shared disk. If user commands are not located on the shared disk, you need to locate them at the same path for both the executing node and standby node.

4. If you select `After` as the `User Command Execution Timing` property, and if the executed user commands do not finish by the time the next record collection processing begins, no user command is executed.

Note

The UNIX `cron` command can be used to periodically collect user-specific performance data. UNIX provides the `cron` command, which can automatically execute a batch file or program at the specified time and interval. After creating a shell script that executes the user commands and then the `jpcuser` command, set the `cron` command so that the shell script is executed periodically.

(4) Specifying the settings for collecting information from the user data file

The user data file contains data that the `jpcuser` command has converted from user-created data into a record format that can be managed by PFM - Agent for Platform. The data in the user data file is stored in user-defined records every time PFM - Agent for Platform collects records. Make sure that PFM - Web Console is set so that PFM - Agent for Platform will collect user-defined records.

For details about how to collect records, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

7.4.3 Example of collecting user-specific performance data (for a physical host environment)

This subsection provides an example of collecting process information under the conditions shown in the following table.

Table 7–24: Conditions for collecting performance data in the example

Option	Explanation	Corresponding field	Value
`tt`	Transaction type	`Trans Type`	`TotalMemory`
`ki`	Transaction key (numeric type)	`Trans Data Key`	Process ID
`ss`	16-byte character string	`User String 1`	Process name
`u`	Unsigned long type	`User Unsigned Long 1`	Amount of memory in use

(1) Examples of user commands (for a physical host environment)

The following are examples of user commands (`userapplication_script` and `userdata_script`) that acquire process information from UNIX and output user-created data.

`userapplication_script`:

```
#!/bin/ksh
echo "Product Name=PFM-Agent for Platform (UNIX)"
```

```
echo "FormVer=0001"
echo "tt ki ss u"
ps -eo pid,comm,vsz,rss | grep jpcagtu | awk '{ printf("TotalMemory %d %s %d
\n",$1,$2,$3+$4);}'
```

`userdata_script`:

```
#!/bin/ksh
echo "Product Name=PFM-Agent for Platform (UNIX)"
echo "FormVer=0001
"echo "tt ks u u"
df -k |grep pshp02 |awk '{ printf("RemoteDisk %s %d %d\n",$6,$3,$4);}'
```

The following are examples of user-created data output by the user commands above.

Example of user-created data output by `userapplication_script`:

```
Product Name=PFM-Agent for Platform (UNIX)
FormVer=0001
tt ki ss u
TotalMemory 0 COMMAND 0
TotalMemory 0 sched 0
TotalMemory 1 /etc/init 1584
TotalMemory 2 pageout 0
TotalMemory 3 fsflush 0
TotalMemory 442 /usr/lib/saf/sac 2808
  ...
```

Example of user-created data output by `userdata_script`:

```
Product Name=PFM-Agent for Platform (UNIX)
FormVer=0001
tt ks u u
RemoteDisk /home1/jp1ps 67318256 4473056
RemoteDisk /home2/jp1ps 43867064 27738672
  ...
```

Note

- For an example of collecting information about used ports, see *3.1.6 Examples of collecting information about used ports*.
- For an example of collecting performance data from multiple hosts for which PFM product is not installed, see *3.1.7 Examples of collecting performance data from multiple hosts on which PFM products are not installed*.

(2) Example of a shell script used to perform periodic collection (for a physical host environment)

The following gives an example of a shell script (`UserDefinedData.sh`) for using functionality for periodically executing user commands.

```
# Delete old user-created data.
rm -Rf  /opt/perfdata.txt /opt/userdata.txt
# Generate user-created data.
/opt/userapplication_script >> /opt/perfdata.txt
/opt/userdata_script >> /opt/userdata.txt
# Use the jpcuser command to convert the data.
/opt/jp1pc/agtu/agent/jpcuser/jpcuser PD_UPD -file /opt/perfdata.txt -file /
opt/userdata.txt#
```

#: If `-debug 2` is specified, debug log information is output to the following directory:

```
/opt/jp1pc/agtu/agent/jpcuser/debug/
```

Note

The following example sets the shell script to be executed at every one minute using the UNIX `cron` command.

```
* * * * * /opt/UserDefinedData.sh
```

7.4.4 Example of collecting user-specific performance data (for a logical host environment)

The table below describes an example of collecting process information into the `PI_UPI` record. In a logical host environment, if you include name information about the host on which user commands are executed, you can easily tell which node a command is executed on.

Table 7–25: Conditions for collecting performance data in the example

Option	Explanation	Corresponding field	Value
`tt`	Transaction type	`Trans Type`	`TotalMemory`
`ki`	Transaction key (numeric type)	`Trans Data Key`	Process ID
`ss`	16-byte character string	`User String 1`	Process name
`u`	Unsigned long type	`User Unsigned Long 1`	Memory usage
`sm`	32-byte character string	`User String 5`	Host name

(1) Examples of user commands (for a logical host environment)

The following is an example of user commands (`userapplication_script`) that acquire process information from UNIX and output user-created data.

`userapplication_script`:

```
#!/bin/ksh
echo "Product Name=PFM-Agent for Platform (UNIX)"
echo "FormVer=0001"
echo "tt ki ss u sm"
```

```
ps -eo pid,comm,vsz,rss | grep jpcagtu | awk '{ printf("TotalMemory %d %s %d
 '`uname -n`'\n",$1,$2,$3+$4);}'
```

The following is the user-created data when user commands (`userapplication_script`) are executed.

Example of user-created data output by `userapplication_script`:

```
Product Name=PFM-Agent for Platform (UNIX)
FormVer=0001
tt ki ss u sm
TotalMemory 0 COMMAND 0 jp1-aop
TotalMemory 0 sched 0 jp1-aop
TotalMemory 1 /etc/init 1584 jp1-aop
TotalMemory 2 pageout 0 jp1-aop
TotalMemory 3 fsflush 0 jp1-aop
TotalMemory 442 /usr/lib/saf/sac 2808 jp1-aop
:
```

Note

- For an example of collecting information about used ports, see *3.1.6 Examples of collecting information about used ports*.
- For an example of collecting performance data from multiple hosts on which PFM products are not installed, see *3.1.7 Examples of collecting performance data from multiple hosts on which PFM products are not installed*.

(2) Example of a shell script used to perform periodic collection (for a logical host environment)

The following gives an example of a shell script (`UserDefinedData.sh`) for using functionality for periodically executing user commands.

```
# Delete old user-created data.
rm -Rf /jp1/userdata/perfdata.txt
# Generate user-created data.
/jp1/userdata/userapplication_script >> /jp1/userdata/perfdata.txt
# Use the jpcuser command to convert the data.
/opt/jp1pc/agtu/agent/jpcuser/jpcuser PD_UPD -file /jp1/userdata/perfdata.tx
t -lhost jp1-halaop#
```

#

If `-debug 2` is specified, debug log information is output to the following directory:

environment-directory`/jp1pc/agtu/agent/jpcuser/debug/`

Note

The following example sets the shell script to be executed every minute, by using the UNIX `cron` command.

```
* * * * * /opt/UserDefinedData.sh
```

7.4.5 Format of the jpcuser command

The following describes the format of the `jpcuser` command.

Format:

```
jpcuser record-name
        -file user-created-data-file-name
        [-file user-created-data-file-name]...
        [-debug [0|1|2]]
        [-lhost logical-host-name]
```

Note: Square brackets (`[]`) indicate optional items. A vertical bar (`|`) has the same meaning as *OR*. Accordingly, only one of the options separated by a vertical bar can be used at a time.

Description:

The `jpcuser` command converts user-created data (user-specific performance data output by user commands) into data in a format that PFM - Agent for Platform can use (user data file).

The command can also output debug log information that can be used for checking whether the user-created data is correct. For details about the debug log, see *7.4.7 Checking the debug log to determine whether user-created data is correct*.

To execute this command in a logical host environment, execute it on the executing node.

If an error occurs during execution of this command, an error message is output to the following folders:

- Physical host environment: `/opt/jp1pc/agtu/agent/jpcuser/log/public/`
- Logical host environment: *environment-directory*`/jp1pc/agtu/agent/jpcuser/log/public/`

Users who can execute the command:

Users with root user privileges

Location of the command:

`/opt/jp1pc/agtu/agent/jpcuser/`

Arguments:

The first argument in the command line must be *record-name*. The `-file` option, `-debug` option, and `-lhost` option can be specified in any order. The arguments that are always required on the command line are *record-name* and a `-file` option. The `-debug` option and `-lhost` option can be omitted.

record-name

Specify the name of the user-defined record in which performance data is to be stored. You can specify only one of the following names:

- `PD_UPD`
- `PD_UPDB`
- `PI_UPI`
- `PI_UPIB`
- `PI_XUI1`
- `PI_XUI2`
- `PI_XUI3`
- `PI_XUI4`
- `PI_XUI5`

`-file` *user-created-data-file-name*

Specify the name of a user-created data file whose length is no more than 1023 bytes. Use multiple `-file` options to specify multiple user-created data files. When multiple user-created data files are specified, the command creates one user data file from the user-created data files.

Wildcard characters cannot be used in the file name specified in the `-file` option.

You can specify a file name by using a relative path name from the current directory (which is the directory in which the `jpcuser` command is executed).

If a warning occurs in one or more files when multiple user-created data files are specified, the command returns a value that indicates normal termination with a warning. If an error occurs in at least one file, the command returns a value that indicates abnormal termination or the occurrence of an error.

`-debug [0|1|2]`

Use this option to specify whether to output only the user data file, only the output debug log information, or both. You can use this option to check whether the user-created data was correct. You can specify only one `-debug` option in the command.

When `-debug 1` is specified, the command performs only debugging. If you want to create user-defined records, specify the `-debug` option with a value other than `1`.

If an error message is output to the debug log, an error might exist in the user command scripts.

If this option is not specified, the command does not output debug log information.

The following table explains the values that can be specified in the `-debug` option.

Table 7–26: Values specified for the debug option

Value	User data file output?	Debug log file output?
`0`	Yes	No
`1`	No	Yes
`2`	Yes	Yes
Other values	Yes	No
No value specified	Yes	No

Legend:

Yes: The file is output.

No: The file is not output.

The user data file is `jpcuser_`*XXX*. The *XXX* part represents the record type (`UPD`, `UPDB`, `UPI`, `UPIB`, or `XUI`*n* (where *n* is a number in the range from 1 to 5).

The storage folders are as follows:

- Physical host environment: `/opt/jp1pc/agtu/agent/jpcuser/userdata`
- Logical host environment: *environment-directory*`/jp1pc/agtu/agent/jpcuser/userdata`

The debug log file is `jpcuser_dbg_`*XX*`.log`. The *XX* part is a two-digit number that indicates how new the log file is. The storage folders are as follows:

- Physical host environment: `/opt/jp1pc/agtu/agent/jpcuser/debug`
- Logical host environment: *environment-directory*`/jp1pc/agtu/agent/jpcuser/debug`

The following table explains the naming rule for debug log files.

Table 7–27: Example of debug log output

Debug log file name	Explanation
`jpcuser_dbg_01.log`	The latest debug log file
`jpcuser_dbg_02.log`	The second latest debug log file
`jpcuser_dbg_03.log`	The third latest debug log file
...	...

`-lhost`

Specify the logical host name of the host on which this command is executed. If you omit specifying this option, a physical host is assumed. For details about the format of a logical host name, see the description of the `-lhost` option of the `jpcconf ha setup` command in the manual *JP1/Performance Management Reference*.

Note that you can set this option only once.

Return value:

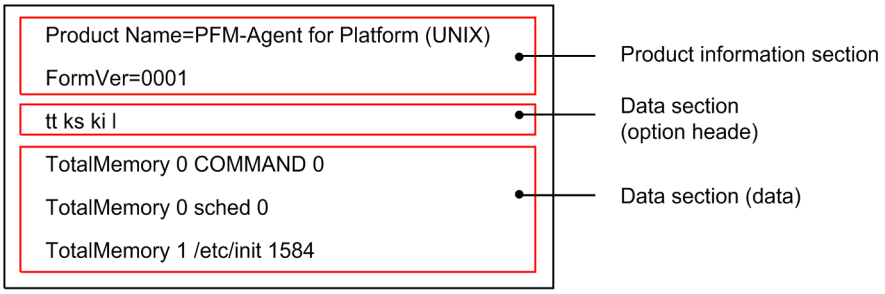
0	Normal termination
1 to 100	Normal termination with a warning
101 to 255	Abnormal termination or the occurrence of an error

7.4.6 Format of user-created data files

This subsection describes the format of user-created data files. For user commands collecting performance data, output text according to this format.

Information output to a user-created data file consists of a product information section and a data section. Both of these sections must be present in each user-created data file. The following figure gives an example configuration of user-created data.

Figure 7–6: Example configuration of user-created data



(1) Product information section

The product information section contains the constants that indicate the product name and the version of the user-created data file. The information set in this section is used only by internal functions and is not stored in records.

The following is an example of information set in this section:

```
Product Name=PFM-Agent for Platform (UNIX)
FormVer=0001
```

Note:

In the above specification method, an error will occur if there is a space before or after the equal sign (=). Note that the specified characters are case sensitive.

(2) Data section

This section sets performance data information. This information is specified below the product information section. The data section consists of the option header and a data part.

(a) Option header

The first line is the option header line, which contains the specified field options. Each option must be separated by using one or more space characters or tabs. The field options correspond to user record fields.

```
tt ks ki l ...
```

The following table lists option names and corresponding record field names. Each of the columns for number of fields indicates the maximum number of options that can be specified. For example, if `ss` is specified multiple times for the field option in a `PI_UPI` record, note that the `PI_UPI` column (No. 10) indicates 4. Accordingly, you can specify `ss` a maximum of four times, such as `ss ss ss ss`, for the `PI_UPI` record.

Table 7–28: Options that can be specified in the data section and the corresponding fields

No.	Option name	Field name	Value explanation	Number of fields (total)				
				PD_UPD (17)	PD_UPDB (34)	PI_UPI (23)	PI_UPIB (49)	PI_XUI1 to PI_XUI5 (67)
1	`tt`	`Trans Type`	Transaction type. This option is a required item.[#1] Size: 1 to 19 bytes	1	1	1	1	1
2	`ki`	`Trans Data Key`	Numeric-type transaction key. Either `ki` or `ks`, or both, must be specified. Type: `ulong` Specifiable characters: Numeric values and a plus sign (+)	1	1	1	1	1
3	`ks`	`Trans String Key`	String-type transaction key. Either `ki` or `ks`, or both, must be specified.[#1] Size: 1 to 19 bytes	1	1	1	1	1
4	`f`	`User Float`	Floating point number.[#2] Type: `double`	2	5	2	5	30
5	`fr`[#3]	`User Float Roll`	Floating point number for a cumulative value[#2] Type: `double`	--	--	2	5	30
6	`l`	`User Long`	Signed long data. Type: `long` Specifiable characters: Numeric values and signs (+, −)	2	5	2	5	--

No.	Option name	Field name	Value explanation	Number of fields (total)				
				PD_UPD (17)	PD_UPDB (34)	PI_UPI (23)	PI_UPIB (49)	PI_XUI1 to PI_XUI5 (67)
7	lr[#3]	User Long Roll	Signed long data for a cumulative value. Type: long Specifiable characters: Numeric values and signs (+, -)	--	--	2	5	--
8	sl	User String(64)	64-byte character string[#1] Size: 1 to 63 bytes + NULL	1	5	1	5	2
9	sm	User String(32)	32-byte character string[#1] Size: 1 to 31 bytes + NULL	2	5	2	5	--
10	ss	User String(16)	16-byte character string[#1] Size: 1 to 15 bytes + NULL	4	5	4	5	--
11	sv	User String(128)	128-byte character string[#1] Size: 1 to 127 bytes + NULL	--	--	--	--	1
12	t	User Time	Time data (time_t type) in the following format: *YYYY/MM/DD,hh:mm:ss* The time must be the local time of the machine on which the jpcuser command is executed.	1	1	1	1	1
13	u	User Unsigned Long	Unsigned long data. Type: ulong Specifiable characters: Numeric values and a plus sign (+)	2	5	2	5	--
14	ur[#3]	User Unsigned Long Roll	Unsigned long data for a cumulative value. Type: ulong Specifiable characters: Numeric values and a plus sign (+)	--	--	2	5	--

Legend:

--: Not specifiable.

#1

The characters that can be specified are uppercase and lowercase alphabetic characters, numeric characters, space characters, and the following symbols:

` ~ ! @ # $ % ^ & * ( ) _ + - = { } : ; < > , . ? / | \ [ ]

#2

The characters that can be specified are numeric values and the following symbols:

- + .

#3

When history data collected by specifying the `fr`, `lr`, or `ur` option is displayed in consolidation mode, the cumulative value is displayed. When a numeric-value option other than these options or the `ki` option is specified, the average value is displayed.

Note that if multiple field options are specified for user-created data, they are allocated sequentially to the target field of the user record.

For example, if `lr` is specified 3 times for the field option of the `PI_UPIB` record, this is specified as `lr lr lr`. In this case, each `lr` is allocated to its field as follows:

- First `lr`: User Long Roll 1
- Second `lr`: User Long Roll 2
- Third `lr`: User Long Roll 3

If `sl lr sl lr lr` is specified, it is allocated to the following fields:

- First `sl`: User String 11
- Second `lr`: User Long Roll 1
- Third `sl`: User String 12
- Fourth `lr`: User Long Roll 2
- Fifth `lr`: User Long Roll 3

(b) Data

The second and subsequent lines are for data. Data lines specify the performance data, corresponding to the field options specified in the optional header. Each column is separated by 1 or more bytes of spaces or tabs.

Make sure that the order of the data matches the field option type.

For example, if `tt ks lr lr ss ss` is specified for the field option, an error will occur for all lines for the following data order:

```
TCP jp1host "ESTABLISHD COUNT=" 5 "LISTENING COUNT=" 2
TCP jp1host "ESTABLISHD COUNT=" 3 "LISTENING COUNT=" 1
TCP jp1host "ESTABLISHD COUNT=" 3 "LISTENING COUNT=" 2
```

The reason this error occurs is because the field options for the third and sixth columns do not match the data types.

- Third column
  The string `"ESTABLISHD COUNT="` is specified for the `lr` field option, even though a cumulative `long` integer value should be specified.
- Sixth column
  The integer values `2, 1, 2` are specified for the `ss` field option, even though a string of size 16 should be specified.

(3) Notes

- Create user-created data files in ASCII.
- Each data line in this file must be terminated with a line feed character (LF).

- Comments cannot be specified in user-created data files.
- An error will occur if any of the first three lines of a user-created data file are empty or contain a space character.
- Any empty lines or lines that contain a space character are disregarded for the fourth and subsequent lines of a user-created data file.
- When entering a string with a space character, enclose the string in double quotation marks (`"`).
- For the string type, set a single-byte string of printable alphanumeric characters. Special single-byte characters such as `"` cannot be set.
- One field option can be specified for definition in the optional header line for each user-created data file. To define a different field option, create another user-created data file.
- Create user-created data files while making sure their contents are not prepended with a BOM[#]. If the contents of a user-created data file are prepended with a BOM, running the `jpcuser` command results in an error.

  #

  Abbreviation of "Byte Order Mark", an identifier added to the first line of the file to indicate the type of Unicode encoding (UTF-16 Little Endian, UTF-16 Big Endian, UTF-8, and so on). A 2-byte identifier is added for UTF-16, and a 3-byte identifier is added for UTF-8.

7.4.7 Checking the debug log to determine whether user-created data is correct

The debug log is used to check whether the user-created data is correct. If you want to output debug log information, specify the `-debug 1` or `-debug 2` option in the `jpcuser` command.

In the debug log, a check result code, such as `OK` (success), `NG` (failure), or `WG` (warning), is output for each data line in the user-created data. If `NG` or `WG` is output in the debug log, the user-created data might be invalid. If `NG` or `WG` has been output, check the user commands by which the user-created data was created, and make any necessary corrections for outputting the data in the correct format. For the correct format, see *7.4.6 Format of user-created data files*.

The following shows the location and name of a debug log file:

- Physical host environment:
  `/opt/jp1pc/agtu/agent/jpcuser/debug/jpcuser_dbg_{01|02|03|04|05}.log`
- Logical host environment:
  environment-directory`/jp1pc/agtu/agent/jpcuser/debug/jpcuser_dbg_{01|02|03|04|05}.log`

The following describes the debug log file format.

(1) Debug log file format

A debug log file consists of four sections:

- Product information
- The execution time and process ID of the `jpcuser` command
- Header line
- Check results

A check result is output for each data line in the user-created data. A comma is used to separate items.

The following table explains the items that are output to a debug log file.

Table 7–29: Items that are output to a debug log file

No.	Section	Item	Value	Explanation
1	Product information	Product name	`Product Name=PFM-Agent for Platform (UNIX)`	The PFM-Agent product name.
2		Format version	`FormVer=0001`	The version of the user-created data format.
3	Execution time and process ID of the `jpcuser` command	Execution time	*YYYY/MM/DD hh:mm:ss*	*YYYY*: Year *MM*: Month *DD*: Day *hh*: Hour *mm*: Minute *ss*: Second
4		Process ID	`PID=`*xxxx*	The process ID of the `jpcuser` command.
5	Header line	Header	Example (for the `PD_UPD` record): `LineNumber, Result, APITime, Recordtype, Transactiontype, t, ks, ki, L1, L2, UL1, UL2, F1, F2, SS1, SS2, SS3, SS4, SM1, SM2, SL1`	The debug log header. The header names correspond to the field options and field names specified on the option header line in a user-created data file. For the correspondence, see Table *6-29 Header line items in a debug log file and their corresponding field options and field names*. Note that the header items differ depending on the user-defined records that are to be stored.
6	Check result	User-created data file name	Example: `File=/opt/jp1pc/agtu/agent/jpcuser/sample_01.txt`	The user-specified path name of a user-created data file that is read is output.
7		Error or warning message	`KAVF`*xxxxx-x*	If an error or other problem that the user should be made aware of occurs on a line in the user-created data, the applicable error and warning messages are output at the beginning of the line.
8		Line number	Numeric value	The number of a line in the user-created data.
9		Result code	`OK`	Success. The line in the user-created data was free of problems and was converted successfully.
10			`WG`	Warning. The line in the user-created data contained a problem but was converted nevertheless. When `WG` is output, a warning message is also output.
11			`NG`	Failure.

No.	Section	Item	Value	Explanation
11	Check result	Result code	NG	The line in the user-created data contained a problem and was not converted. When NG is output, a warning message or error message, depending on the cause of the problem, is also output. If a warning message is output, processing continues. If an error message is output, processing stops.
12			BL	Null line. The line in the user-created data is empty and is ignored.
13		Data	Data	The contents of the line in the user-created data. For an empty numeric field, 0 is output. For an empty string field, two quotation marks (""") are output.

The following table lists the items output on the header line in a debug log file and their corresponding field options and field names specified on the option header line in a user-created data file.

Table 7–30: Header line items in a debug log file and their corresponding field options and field names

No.	Item on the header line in a debug log file	Field option specified on the option header line in a user-created data file	Field name (PFM - View name)	Explanation
1	Line Number	--	--	Number of the line on which the relevant data exists
2	Result	--	--	Check result of the relevant data
3	API Time	--	Collect Time	Time that the relevant data was converted
4	Record type	--	Record Type	Record type
5	Transaction type	tt	Trans Type	Transaction type
6	t	t	User Time 1	Time value
7	ks	ks	Trans String Key	String-type transaction key
8	ki	ki	Trans Data Key	Numeric-type transaction key
9	L1	l	User Long 1	Integer value of type long
10	L2	l	User Long 2	Integer value of type long
11	L3	l	User Long 3	Integer value of type long
12	L4	l	User Long 4	Integer value of type long
13	L5	l	User Long 5	Integer value of type long
14	L1R	lr	User Long Roll 1	Cumulative integer value of type long

No.	Item on the header line in a debug log file	Field option specified on the option header line in a user-created data file	Field name (PFM - View name)	Explanation
15	L2R	lr	User Long Roll 2	Cumulative integer value of type long
16	L3R	lr	User Long Roll 3	Cumulative integer value of type long
17	L4R	lr	User Long Roll 4	Cumulative integer value of type long
18	L5R	lr	User Long Roll 5	Cumulative integer value of type long
19	UL1	u	User Unsigned Long 1	Integer value of type unsigned long
20	UL2	u	User Unsigned Long 2	Integer value of type unsigned long
21	UL3	u	User Unsigned Long 3	Integer value of type unsigned long
22	UL4	u	User Unsigned Long 4	Integer value of type unsigned long
23	UL5	u	User Unsigned Long 5	Integer value of type unsigned long
24	UL1R	ur	User Unsigned Long Roll 1	Cumulative integer value of type unsigned long
25	UL2R	ur	User Unsigned Long Roll 2	Cumulative integer value of type unsigned long
26	UL3R	ur	User Unsigned Long Roll 3	Cumulative integer value of type unsigned long
27	UL4R	ur	User Unsigned Long Roll 4	Cumulative integer value of type unsigned long
28	UL5R	ur	User Unsigned Long Roll 5	Cumulative integer value of type unsigned long
29	F1	f	User Float 1	Floating point number value
30	F2	f	User Float 2	Floating point number value
31	F3	f	User Float 3	Floating point number value
32	F4	f	User Float 4	Floating point number value
33	F5	f	User Float 5	Floating point number value
34	F01 to F30	f	User Float 01 to User Float 30	Floating point number value
35	F1R	fr	User Float Roll 1	Cumulative floating point number value
36	F2R	fr	User Float Roll 2	Cumulative floating point number value
37	F3R	fr	User Float Roll 3	Cumulative floating point number value

No.	Item on the header line in a debug log file	Field option specified on the option header line in a user-created data file	Field name (PFM - View name)	Explanation
38	F4R	fr	User Float Roll 4	Cumulative floating point number value
39	F5R	fr	User Float Roll 5	Cumulative floating point number value
40	F01A to F30A	fr	User Float Add 01 to User Float Add 30	Cumulative floating point number value
41	SS1	ss	User String 1	16-byte character string
42	SS2	ss	User String 2	16-byte character string
43	SS3	ss	User String 3	16-byte character string
44	SS4	ss	User String 4	16-byte character string
45	SS5	ss	User String 5	16-byte character string
46	SM1	sm	User String 5 (for the PD_UPD or PI_UPI record) User String 6 (for the PD_UPDB or PI_UPIB record)	32-byte character string
47	SM2	sm	User String 6 (for the PD_UPD or PI_UPI record) User String 7 for the (PD_UPDB or PI_UPIB record)	32-byte character string
48	SM3	sm	User String 8	32-byte character string
49	SM4	sm	User String 9	32-byte character string
50	SM5	sm	User String 10	32-byte character string
51	SL1	sl	User String 7 (for the PD_UPD or PI_UPI record) User String 11 (for the PD_UPDB or PI_UPIB record) User String 1 (PI_XUI1 to PI_XUI5 records)	64-byte character string
52	SL2	sl	User String 12 (for the PD_UPDB or PI_UPIB record) User String 2 (PI_XUI1 to PI_XUI5 records)	64-byte character string
53	SL3	sl	User String 13	64-byte character string
54	SL4	sl	User String 14	64-byte character string
55	SL5	sl	User String 15	64-byte character string
56	SV1	sv	User String 3	128-byte character string

Legend:

--: Not applicable

(2) Example of information output to a debug log file

The following figure shows an example of information output to a debug log file.

Figure 7–7: Example of information output to a debug log file

```
##########################
#Product Name=PFM-Agent for Platform (UNIX)
#FormVer=0001
#Date=Sat Feb 24 16:00:41 2007
#PID=29714
##########################
```

```
Line Number, Result, API Time, Record type, Transaction type,
t, ks, ki, L1, L2, L3, L4, L5, L1R, L2R, L3R, L4R, L5R, UL1,
UL2, UL3, UL4, UL5, UL1R, UL2R, UL3R, UL4R, UL5R, F1, F2, F3,
F4, F5, F1R, F2R, F3R, F4R, F5R, SS1, SS2, SS3, SS4, SS5, SM1,
SM2, SM3, SM4, SM5, SL1, SL2, SL3, SL4, SL5
```

(1) Header line

```
File=/opt/jp1pc/agtu/agent/jpcuser/sample_01.txt
```

(2) File name1

Check result

Line No.

```
4,OK,Sat Feb 24 16:00:41 2007,PI_UPIB,TT,Sat Feb 24 10:09:56
2007,a,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1.000000,0.00
0000,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,0.0
00000,0.000000,ss1,,,,,,,,,,,,,
```

(3) Check result for line 4 in file name1 (OK)

```
File=./sample_02.txt
```

(4) File name2

```
KAVF10908-W Warning: The specified value (2007/02/24,10:10:010)
for the option (t) is invalid. (filename = ./sample_02.txt)
```

(5) Warning 1 for line 4 in file name 2

```
KAVF10907-W Warning:
The specified value for option is too
long. (ss = abcdefghijklmnop, filename = ./sample_02.txt)
```

(6) Warning 2 for line 4 in file name 2

```
4,WG,Sat Feb 24 16:00:41 2007,PI_UPIB,TT,n/a,
ab,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0.000000,0.000000
,0.000000,0.000000,0.000000,0.000000,0.000000,0.000000,0.00000
0,0.000000,abcdefghijklmno,,,,,,,,,,,,,
```

(7) Check result for line 4 in file name 2 (WG)

```
5,BL
```

```
6,BL
```

(8) Check result for lines 5 and 6 (BL)

```
KAVF10902-W Warning: The specified value (abcdefghijklmnopqrst)
for the option (ks) is too long. (filename = ./sample_02.txt)
7,NG
...
```

(9) Warning for line 7

(10) Check result for lines 7 (NG)

Legend:

: Location generating a warning

The following explanations are keyed to the numbers in parentheses in the figure.

1. This line is the header line.
2. The user-specified path name of a user-created data file loaded into the command is output.
3. Output of the check result for the user-created data file begins with this line. The number (`4`) at the beginning indicates the number of the line in the user-created data file. In a user-created data file, the first line contains product information, the second line contains version information, and the third line is the option header line. Therefore, checking normally begins with line 4. If the line contains no problems, `OK` is output for `Result`.
4. The user-specified path name of another user-created data file that is read is output.
5. This line warns the user of a problem on line 4 in the `sample_02.txt` file. Because the `t` value (`2007/02/24,10:10:010`) did not have the expected format, `n/a` has been output for the element corresponding to `t` (see (7) in the figure).

6. This line also warns the user of a problem on line 4 in the `sample_02.txt` file. Because the specified `ss` value (`abcdefghijklmnop`) exceeded the predefined maximum of 15 bytes, a warning message has been output. and a truncated value (`abcdefghijklmno`) has been output for the element corresponding to `SS1` (see (7) in the figure).
7. Because the warnings indicated by (5) and (6) have been issued, the check result code `WG` has been output for `Result` for line 4.
8. The check result code `BL` indicates that the line is a null line.
9. This line warns the user of a problem on line 7. A warning message has been output because the specified `ks` value exceeded the predefined maximum of 19 bytes.
10. Because the value of the `ks` unique key on line 7 in the user-created data file was incorrect, the value could not be used. Accordingly, `NG` has been output for `Result`. If the value of `Transaction type`, `ks`, or `ki`, which is a unique key, is incorrect, the line is not processed.

8 Monitoring Templates

This chapter explains the monitoring templates for PFM - Agent for Platform.

Format of alarm explanations

This section explains alarm formats. Alarms are listed in alphabetical order.

Alarm name

Indicates the alarm name in the monitoring template.

Overview

Provides an overview of the target that can be monitored with this alarm.

Primary settings

Explains the primary settings of this alarm using a table. This table shows the correspondence between alarm settings and the setting items in the Properties window, which is displayed by clicking the alarm icon in the Alarms window in PFM - Web Console and then clicking the **Properties** method. To check the details of each alarm setting, use the Properties window of the alarm in PFM - Web Console.

If − is set, it means that the setting is always ignored.

If an error condition and a warning condition are the same in a conditional expression, an alarm event is issued for the error condition only.

Alarm tables

Indicates the alarm tables in which this alarm is stored.

Related report

Indicates the monitoring template reports that are related to this alarm. To display this report, in the Agents window in PFM - Web Console, click the agent icon and then click the icon, which is displayed in the **Display Alarm Status** method.

List of alarms

A table containing one or more alarms is called an *alarm table*. The alarms defined in the monitoring template of PFM - Agent for Platform are stored in the alarm tables in the `UNIX` directory, which is displayed on the **Alarms** tab in PFM - Web Console.

The following lists the alarm table names.

- PFM UNIX Template Alarms 09.00
- PFM UNIX Template Alarms [APP] 09.10
- PFM UNIX Template Alarms [CPU] 09.00
- PFM UNIX Template Alarms [DSK] 09.00
- PFM UNIX Template Alarms [MEM] 09.00
- PFM UNIX Template Alarms [NET] 09.00
- PFM UNIX Template Alarms [PS] 09.10

Square brackets ([ ]) in an alarm table name:

The monitoring item applicable to the alarm table is enclosed in square brackets ([ ]). The alarm table without a square-bracket suffix consists of basic alarms.

Number at the end of an alarm table name:

This value indicates the alarm table version.

For PFM - Agent for Platform, the Alarms tree might display an alarm table of a version that is not available in the Performance Management system being used. When using an alarm table defined in the monitoring template, check the alarm table version being used by the Performance Management system and version compatibility. For details about the alarm table version and version compatibility, see *I. Version Compatibility*.

The table below shows the alarms defined in the monitoring template of PFM - Agent for Platform.

Table 8–1: Alarm list

Alarm table name	Alarm name	Monitoring target
PFM UNIX Template Alarms 09.00	Disk Service Time	Average I/O operation time for a device (seconds).
	I/O Wait Time	Percentage of time during which the processor for the overall host is waiting for I/O (%).
	Kernel CPU	Percentage of time during which the processor for the overall host operates in kernel mode (%).
	Pagescans	Number of page scans that occurred per second. This alarm cannot be used on Linux.
	Run Queue	Average number of threads waiting in the run queue in the past five minutes.
	Swap Outs	Number of pages taken out per second by swap-out processing.
	User CPU	Percentage of time during which the processor for the overall host operates in user mode (%).
PFM UNIX Template Alarms [CPU] 09.00	Kernel CPU	Percentage of time during which the processor for the overall host operates in kernel mode (%).
	User CPU	Percentage of time during which the processor for the overall host operates in user mode (%).

Alarm table name	Alarm name	Monitoring target
PFM UNIX Template Alarms [CPU] 09.00	Run Queue	Average number of threads waiting in the run queue in the past five minutes.
	CPU Per Processor(K)	CPU usage for a processor executed in kernel mode (%).
	CPU Per Processor(U)	CPU usage for a processor executed in user mode (%).
PFM UNIX Template Alarms [MEM] 09.00	Pagescans	Number of page scans that occurred per second. This alarm cannot be used on Linux.
	Swap Outs	Number of pages taken out per second by swap-out processing.
	Alloc Mem Mbytes	Amount of used virtual memory (MB).
PFM UNIX Template Alarms [DSK] 09.00	I/O Wait Time	Percentage of time during which the processor for the overall host is waiting for I/O (average value for each processor).
	Disk Service Time	Average I/O operation time for a device (seconds).
	File System Free(L)	Unused area (MB) in the local file system.
	File System Free(R)	Unused area (MB) in the remote file system.
	Disk Busy %	Disk busy rate (%).
	Disk Queue	Queue length of devices for which the amount of I/O processing per second is 1.
PFM UNIX Template Alarms [NET] 09.00	Network Rcvd/sec	Number of packets received per second.
PFM UNIX Template Alarms [PS] 09.10	Process End	Process name.
	Process Alive	Number of processes being executed by the workgroup.
	Process Existence	Program name.
PFM UNIX Template Alarms [APP] 09.10	Application Status	Status of the application for which information about process operating status is being collected.

Disk Service Time

Overview

The Disk Service Time alarm monitors the average I/O operation time for a device. If the average operation time is greater than the threshold, the amount of information being input or output might be very large.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	Average disk service time is %CVS secs
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	2
	interval(s)	3
Alarm Conditions	**Record**	Device Detail (PI_DEVD)
	Field	Avg Service Time
	Abnormal condition	Avg Service Time > 0.1
	Warning condition	Avg Service Time > 0.06
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [DSK] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Avg Service Time - Top 10 Devices

I/O Wait Time

Overview

The I/O Wait Time alarm monitors the percentage of time during which the processor was waiting for I/O (average value for each processor). If the percentage of time during which the processor is waiting for I/O is greater than the threshold, I/O operations, such as updating a database, might be delayed.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	I/O wait time is %CVS%
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	2
	interval(s)	3
Alarm Conditions	**Record**	System Summary Overview (PI)
	Field	Wait %
	Abnormal condition	Wait % > 80
	Warning condition	Wait % > 60
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [DSK] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/I/O Activity - Top 10 Processes

Kernel CPU

Overview

The Kernel CPU alarm monitors the percentage of time during which the processor for the overall host was running in kernel mode (average value for each processor). If this percentage of time continues to be greater than the threshold, there might be a problem with the OS or the operation method.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Kernel mode CPU usage is %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	`System Summary Overview (PI)`
	Field	`Kernel CPU %`
	Abnormal condition	`Kernel CPU % > 75`
	Warning condition	`Kernel CPU % > 50`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [CPU] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/System Overview

Pagescans

Overview

The Pagescans alarm monitors the number of page scans that occurred per second. If the number of page scans is greater than the threshold, memory might be insufficient.

Note:

This alarm cannot be used on Linux.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Pagescan rate is %CVS/sec`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	`System Summary Overview (PI)`
	Field	`Page Scans/sec`
	Abnormal condition	`Page Scans/sec > 150`
	Warning condition	`Page Scans/sec > 100`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [MEM] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Major Page Faults - Top 10 Processes

Run Queue

Overview

The Run Queue alarm monitors the average number of threads waiting in the run queue in the past five minutes. If the average number of threads is greater than the threshold, there might be a problem with the OS, operation method, or a specific application.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Five minute run queue avg is %CVS`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	`0`
	interval(s)	`0`
Alarm Conditions	**Record**	`System Summary Overview (PI)`
	Field	`5-Minute Run Queue Avg`
	Abnormal condition	`5-Minute Run Queue Avg > 8`
	Warning condition	`5-Minute Run Queue Avg > 4`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [CPU] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/CPU Usage - Top 10 Processes

Swap Outs

Overview

The Swap Outs alarm monitors the number of pages taken out per second by swap-out processing. If the number of pages is greater than the threshold, memory might be insufficient.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Swapout rate is %CVS/sec`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	`0`
	interval(s)	`0`
Alarm Conditions	**Record**	`System Summary Overview (PI)`
	Field	`Swapped-Out Pages/sec`
	Abnormal condition	`Swapped-Out Pages/sec > 200`
	Warning condition	`Swapped-Out Pages/sec > 100`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [MEM] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Major Page Faults - Top 10 Processes

User CPU

Overview

The User CPU alarm monitors the percentage of time during which the processor for the overall host was running in user mode (average value for each processor). If this percentage of time continues to be greater than the threshold, there might be a problem with a specific application.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	User mode CPU usage is %CVS%
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	2
	interval(s)	3
Alarm Conditions	**Record**	System Summary Overview (PI)
	Field	User CPU %
	Abnormal condition	User CPU % > 85
	Warning condition	User CPU % > 65
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms 09.00

PFM UNIX Template Alarms [CPU] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/System Overview

CPU Per Processor(K)

Overview

The CPU Per Processor(K) alarm monitors the CPU usage (%) for a processor executed in kernel mode. If the CPU usage continues to be at or above the threshold, there might be a problem with the OS or operation method.

> **Note**
>
> To monitor a specific processor, copy this alarm and create an alarm by changing the condition expression in the `Processor ID` field to `Processor ID` = *identifier-of-monitored-processor*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Kernel mode CPU usage is %CVS2(%CVS1)`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions 1	**Record**	`CPU - Per Processor Detail (PI_CPUP)`
	Field	`Processor ID`
	Abnormal condition	`Processor ID >= 0`#
	Warning condition	`Processor ID >= 0`#
Alarm Conditions 2	**Record**	`CPU - Per Processor Detail (PI_CPUP)`
	Field	`System %`
	Abnormal condition	`System % > 75`
	Warning condition	`System % > 50`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#

Set the identifier of the processor to be monitored. The value used in the table shows an example of setting a condition value that satisfies the `Processor ID` field.

Alarm tables

PFM UNIX Template Alarms [CPU] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/CPU Per Processor Usage

CPU Per Processor(U)

Overview

The CPU Per Processor(U) alarm monitors the CPU usage (%) for a processor executed in user mode. If the CPU usage continues to be at or above the threshold, there might be a problem with a specific application.

> **Note**
>
> To monitor a specific processor, copy this alarm and create an alarm by changing the condition expression in the `Processor ID` field to `Processor ID` = *identifier-of-monitored-processor*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`User mode CPU usage is %CVS2(%CVS1)`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions 1	**Record**	`CPU - Per Processor Detail (PI_CPUP)`
	Field	`Processor ID`
	Abnormal condition	`Processor ID >= 0`#
	Warning condition	`Processor ID >= 0`#
Alarm Conditions 2	**Record**	`CPU - Per Processor Detail (PI_CPUP)`
	Field	`User %`
	Abnormal condition	`User % > 85`
	Warning condition	`User % > 65`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#

Set the identifier of the processor to be monitored. The value used in the table shows an example of setting a condition value that satisfies the `Processor ID` field.

Alarm tables

PFM UNIX Template Alarms [CPU] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/CPU Per Processor Usage

Alloc Mem Mbytes

Overview

The Alloc Mem Mbytes alarm monitors the amount of used virtual memory area (MB). If the amount of used memory continues to be at or above the threshold (the `Total Physical Mem Mbytes` field value of the `PI` record), physical memory might be insufficient.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Alloc Mem Mbytes = %CVS1`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	`System Summary Overview (PI)`
	Field	`Alloc Swap Mbytes`
	Abnormal condition	`Alloc Swap Mbytes >= 1800`#1
	Warning condition	`Alloc Swap Mbytes >= 1024`#2
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#1

Depending on the environment, set a value that is about 90% of the value of the `Total Swap Mbytes` field of the `PI` record (except when the Solaris zone functionality is being used).

#2

Depending on the environment, set the value of the `Total Physical Mem Mbytes` field of the `PI` record (except when the Solaris zone functionality is being used).

Alarm tables

PFM UNIX Template Alarms [MEM] 09.00

Related reports

None

File System Free(L)

Overview

The File System Free(L) alarm monitors the unused area (MB) in the local file system. If there is little unused area, disk space is insufficient.

> **Note**
>
> To monitor a specific file system, copy this alarm to create an alarm by changing the condition expression in the `File System` field to `File System` = *mount-point-of-monitored-file-system*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Disk Space(%CVS1) = %CVS2 MB`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	`0`
	interval(s)	`0`
Alarm Conditions 1	**Record**	`File System Detail - Local (PD_FSL)`
	Field	`File System`
	Abnormal condition	`File System <> dummy`[#1]
	Warning condition	`File System <> dummy`[#1]
Alarm Conditions 2	**Record**	`File System Detail - Local (PD_FSL)`
	Field	`Mbytes Free`
	Abnormal condition	`Mbytes Free < 5120`[#2]
	Warning condition	`Mbytes Free < 10240`[#2]
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#1

Sets the mount point of a file system that will not be monitored. No file system with the name set here in the table exists, so all file systems will become monitoring targets.

#2

Change the value as necessary for the environment.

Alarm tables

PFM UNIX Template Alarms [DSK] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/Local File System Detail

File System Free(R)

Overview

The File System Free(R) alarm monitors the unused area (MB) in the remote file system. If there is little unused area, disk space is insufficient.

> **Note**
>
> To monitor a specific file system, copy this alarm to create an alarm by changing the condition expression in the `File System` field to `File System` = *mount-point-of-monitored-file-system*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Disk Space(%CVS1) = %CVS2 MB`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	`0`
	interval(s)	`0`
Alarm Conditions 1	**Record**	`File System Detail - Remote (PD_FSR)`
	Field	`File System`
	Abnormal condition	`File System <> dummy`[#1]
	Warning condition	`File System <> dummy`[#1]
Alarm Conditions 2	**Record**	`File System Detail - Remote (PD_FSR)`
	Field	`Mbytes Free`
	Abnormal condition	`Mbytes Free < 5120`[#2]
	Warning condition	`Mbytes Free < 10240`[#2]
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#1

Sets the mount point of a file system that will not be monitored. No file system with the name set here in the table exists, so all file systems will become monitoring targets.

#2

Change the value as necessary for the environment.

Alarm tables

PFM UNIX Template Alarms [DSK] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/Remote File System Detail

Disk Busy %

Overview

The Disk Busy % alarm monitors the disk busy rate (%). If the disk busy rate continues to be at or above the threshold, I/O operations might be concentrated on a specific disk.

> **Note**
>
> To monitor a specific device, copy this alarm to create an alarm by changing the condition expression in the `Device Name` field to `Device Name` = *name-of-monitored-device*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Disk Busy % (%CVS1) = %CVS2`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`4`
	interval(s)	`5`
Alarm Conditions 1	**Record**	`Device Detail (PI_DEVD)`
	Field	`Device Name`
	Abnormal condition	`Device Name <> dummy#`
	Warning condition	`Device Name <> dummy#`
Alarm Conditions 2	**Record**	`Device Detail (PI_DEVD)`
	Field	`Busy %`
	Abnormal condition	`Busy % >= 90`
	Warning condition	`Busy % >= 80`
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#

Sets the name of a device that will not be monitored. No device with the name set here in the table exists, so all devices will become monitoring targets.

Alarm tables

PFM UNIX Template Alarms [DSK] 09.00

Related reports

None

Disk Queue

Overview

The Disk Queue alarm monitors the queue length for devices for which the amount of I/O processing per second is 1. If the queue length continues to be at or above the threshold, the device is congested.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Disk Queue(%CVS1) = %CVS2`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`4`
	interval(s)	`5`
Alarm Conditions 1	**Record**	`Device Detail (PI_DEVD)`
	Field	`Device Name`
	Abnormal condition	`Device Name <> dummy`#1
	Warning condition	`Device Name <> dummy`#1
Alarm Conditions 2	**Record**	`Device Detail (PI_DEVD)`
	Field	`Queue Length`
	Abnormal condition	`Queue Length >= 5`#2
	Warning condition	`Queue Length >= 3`#2
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#1

Sets the name of a device that will not be monitored. No device with the name set here in the table exists, so all devices will become monitoring targets.

#2

Change the value as necessary for the environment.

Alarm tables

PFM UNIX Template Alarms [DSK] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/Device Detail

Network Rcvd/sec

Overview

The Network Rcvd/sec alarm monitors the number of packets received per second. If the number of packets is large, many packets have been received successfully.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Pkt Rcvd/sec = %CVS1`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`4`
	interval(s)	`5`
Alarm Conditions	**Record**	`Network Interface Summary (PI_NINS)`
	Field	`Pkts Rcvd/sec`
	Abnormal condition	`Pkts Rcvd/sec >= 9`#1
	Warning condition	`Pkts Rcvd/sec >= 8`#2
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#1

Depending on the environment, set a value that is around 90% of the NIC bandwidth.

#2

Depending on the environment, set a value that is around 80% of the NIC bandwidth.

Alarm tables

PFM UNIX Template Alarms [NET] 09.00

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Network Interface Summary

Process End

Overview

The Process End alarm monitors process disappearance. If performance data is not collected, the process has stopped.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	Process status has changed
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	0
	interval(s)	0
Alarm Conditions	**Record**	Process Detail Interval (PD_PDI)
	Field	Program
	Abnormal condition	Program = jpcsto#
	Warning condition	Program = jpcsto#
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Normal**

Legend:

--: Setting is always ignored.

#

Set the name of the process to be monitored. The value used in the table shows an example of setting the process name for the Agent Store service. Note that there is no abnormal condition or warning condition because this alarm only monitors a value.

Alarm tables

PFM UNIX Template Alarms [PS] 09.10

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/Process Detail

Process Alive

Overview

The Process Alive alarm monitors the generation of processes that belong to the workgroup. Set `/opt/jp1pc/agtu/agent/wgfile`# as follows for monitoring.

Setting example

```
workgroup=workgroup
programs=jpcsto
```

#

Set the `wgfile` file in an environment in which PFM - Agent for Platform is installed. For details about how to set the `wgfile` file, see *7.1 Settings for collecting workgroup information*.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	`UNIX`
	Alarm message	`Workgroup Application Process Count = %CVS1`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	`0`
	interval(s)	`0`
Alarm Conditions 1	**Record**	`Workgroup Summary (PI_WGRP)`
	Field	`Process Count`
	Abnormal condition	`Process Count > 0`
	Warning condition	`Process Count > 0`
Alarm Conditions 2	**Record**	`Workgroup Summary (PI_WGRP)`
	Field	`Workgroup`
	Abnormal condition	`Workgroup = workgroup`#
	Warning condition	`Workgroup = workgroup`#
Actions	**Email**	--
	Command	--

PFM - Web Console alarm property		Settings
Item	Detailed item	
Actions	**SNMP**	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

#

Set the name of the workgroup to be monitored. The `/opt/jp1pc/agtu/agent/wgfile` file must be defined.

Alarm tables

PFM UNIX Template Alarms [PS] 09. 10

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/Process Detail

Process Existence

Overview

The Process Existence alarm monitors for the disappearance of a process. If a process's existence cannot be confirmed, the process has stopped.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	Process status has changed
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	0
	interval(s)	0
Alarm Conditions	**Record**	Application Process Overview (PD_APS)
	Field	Program Name
	Abnormal condition	Program Name = jpcsto#
	Warning condition	Program Name = jpcsto#
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Normal**

Legend:

--: Setting is always ignored.

#

Set the name of the program to be monitored. Make sure that the character string you enter here matches what is entered in the Program Name field of the PD_APS record.

Any character in the source information that is not in the ASCII character set range of 0x20 to 0x7E will be converted to a hash mark (#: 0x23) before it is stored in the Program Name field of the PD_APS record. Note that multi-byte characters are processed in single-byte units during conversion. For example, the multi-byte (full-width) letter Ａ is converted as follows:

Source information		Post-conversion information	
Character encoding	Binary	Binary	Character string
Shift-JIS	8260	2360	#`
EUC	A3C1	2323	##
UTF-8	EFBCA1	232323	###

Alarm tables

PFM UNIX Template Alarms [PS] 09.10

Related reports

None

Application Status

Overview

The Application Status alarm monitors the status of an application that is being monitored by the Application Summary Extension (PD_APP2) record.

Primary settings

PFM - Web Console alarm property		Settings
Item	Detailed item	
Main Information	**Product**	UNIX
	Alarm message	Status of application(%CVS1) has changed
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Do not specify.
	Monitoring time range	**Always monitor**
	Report alarm when the following damping condition is reached	Do not specify.
	occurrence(s) during	0
	interval(s)	0
Alarm Conditions 1	**Record**	Application Summary Extension (PD_APP2)
	Field	Application Name
	Abnormal condition	Application Name = *
	Warning condition	Application Name = *
Alarm Conditions 2	**Record**	Application Summary Extension (PD_APP2)
	Field	Application Exist
	Abnormal condition	Application Exist = ABNORMAL
	Warning condition	Application Exist = NORMAL
Alarm Conditions 3	**Record**	Application Summary Extension (PD_APP2)
	Field	Application Status
	Abnormal condition	Application Status = ABNORMAL
	Warning condition	Application Status = ABNORMAL
Actions	**Email**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: Setting is always ignored.

Alarm tables

PFM UNIX Template Alarms [APP] 09.10

Related reports

Reports/UNIX/Troubleshooting/Real-Time/Application Status

Format of report explanations

This section explains report formats. Reports are listed in alphabetical order.

Report name

Indicates the report name of the monitoring template.

A report whose name contains `(Multi-Agent)` is a report that shows information on multiple instances.

A report whose name does not contain `(Multi-Agent)` is a report that shows information on a single instance.

Overview

Provides an overview of the information that can be displayed in this report.

Storage destination

Indicates the storage destination of this report.

Record

Indicates the record that stores the performance data used by this report. To display a historical report, you must specify in advance that the record shown in this column be collected. Before displaying a report, use the Agents window in PFM - Web Console to display the agent properties, and make sure that `Log = Yes` is set for this record. This setting is not required for a real-time report.

Fields

The record fields used by this report are explained using a table.

Drilldown report (report level)

The monitoring template report associated with this report is explained with the use of a table. To display this drilldown report, from the Drilldown Report pull-down menu in PFM - Web Console's Report window, select the applicable drilldown report name and click **Display Reports**. Note that some reports do not have a drilldown report.

Drilldown report (field level)

The monitoring template report associated with the fields of this report is explained with the use of a table. To display this drilldown report, click a graph, list, or table in PFM - Web Console's Report window. For a historical report, you can display a report with finer time intervals by displaying a drilldown report from a time item. Note that some reports do not have a drilldown report.

For details about a drilldown report, see the chapter explaining creation of reports for operation analysis in the *JP1/Performance Management User's Guide*.

Organization of report directories

The organization of PFM - Agent for Platform's report directories is described below. < > indicates a directory name.

```
<UNIX>
 +-- <Monthly Trend>
 |    +-- CPU Trend
 |    +-- CPU Trend (Multi-Agent)
 |    +-- NFS Load Trend#1
 |    +-- Paging Trend (Multi-Agent)#1
 |    +-- Paging Trend (Multi-Agent) (8.4)#1
 |    +-- Process Trend
 +-- <Status Reporting>
 |    +-- <Daily Trend>
 |    |    +-- CPU Status (Multi-Agent)
 |    |    +-- Memory Paging Status (Multi-Agent)#1
 |    |    +-- Memory Paging Status (Multi-Agent) (8.4)#1
 |    |    +-- Network Status (Multi-Agent)
 |    |    +-- NFS Usage Status (Multi-Agent)#1
 |    |    +-- Workload Status (Multi-Agent)
 |    |    +-- <Advanced>
 |    |         +-- Device Usage Status (Multi-Agent)
 |    +-- <Real-Time>
 |         +-- Avg Service Time Status - Top 10 Devices
 |         +-- CPU Per Processor Status
 |         +-- CPU Status
 |         +-- Device Usage Status
 |         +-- Free Space Mbytes - Top 10 Local File Systems
 |         +-- Memory Paging Status#1
 |         +-- Memory Paging Status (8.4)#1
 |         +-- Network Status
 |         +-- NFS Usage Status#1
 |         +-- Process Summary Status
 |         +-- System Utilization Status
 |         +-- Workload Status
 +-- <Troubleshooting>
      +-- <Real-Time>
      |    +-- Avg Service Time - Top 10 Devices
      |    +-- CPU Per Processor Usage
      |    +-- CPU Usage - Top 10 Processes
      |    +-- I/O Activity - Top 10 Processes#2
      |    +-- Major Page Faults - Top 10 Processes
      |    +-- Network Interface Summary
      |    +-- Space Usage - Top 10 Local File Systems
      |    +-- Space Usage - Top 10 Remote File Systems
      |    +-- System Overview
      |    +-- Application Status
      |    +-- <Drilldown Only>
      |         +-- Device Detail
      |         +-- Local File System Detail
      |         +-- Network Interface Detail
      |         +-- NFS Client Detail#3
      |         +-- NFS Server Detail#3
      |         +-- Process Detail
```

```
|          +-- Remote File System Detail
|          +-- Application Process Status
+-- <Recent Past>
     +-- CPU Usage Summary
     +-- I/O Overview#1
     +-- Memory Paging#1
     +-- Memory Paging (8.4)#1
     +-- Network Overview
     +-- NFS Activity Overview#1
     +-- System Overview
     +-- System Overview (8.4)
     +-- <Advanced>
          +-- Console Messages#1
          +-- Network Interface Summary
          +-- Process Overview
          +-- Application Process Count
          +-- Monitoring Process Detail
```

#1

Cannot be used on Linux.

#2

Cannot be used on HP-UX, AIX, and Linux.

#3

Reserved report that cannot be used.

The individual directories are explained below.

- `Monthly Trend` directory

  This directory stores a historical report that shows information that is summarized on a daily basis for the last month. It is used for analyzing system trends for the month.

- `Status Reporting` directory

  This directory stores reports that show information that is summarized on a daily or weekly basis. It is used for checking the system's overall status. In addition to displaying historical reports, it can also be used to display real-time reports.

  - `Daily Trend` directory

    This directory stores a historical report that shows information that is summarized on an hourly basis over the last 24 hours. It is used for checking the daily system status.

  - `Real-Time` directory

    This directory stores the real-time report for checking the system status.

- `Troubleshooting` directory

  This directory stores reports that show information that is useful for troubleshooting. It is used for investigating a problem cause when a problem occurs in the system.

  - `Real-Time` directory

    This directory stores the real-time report for checking the current system status.

  - `Recent Past` directory

    This directory stores the historical report that shows information that is summarized on a minute-by-minute basis for the last hour.

Additionally, the directories described below are located below the directories previously described. Which of the following directories is provided depends on the higher-order directory:

- `Advanced` directory

  This directory stores reports that use records that are set to `Log = No` by default. To display the reports in this directory, you must use PFM - Web Console to specify the setting for the record being used to `Log = Yes`.

- `Drilldown Only` directory

  This directory stores reports that are displayed as drilldown reports (field level). It is used for displaying detailed information related to the fields of the report.

List of reports

The table below lists the reports defined in the monitoring template by category.

Table 8–2: List of reports

Category	Report name	Information displayed
NFS	*NFS Activity Overview*	Historical information about NFS client and NFS server operation status over the last hour (on a minute-by-minute basis). This report cannot be used on Linux.
	NFS Load Trend	Historical information about NFS client and NFS server operation status over the last month (on a daily basis). This report cannot be used on Linux.
	NFS Usage Status	Real-time information about NFS client and NFS server operation status. This report cannot be used on Linux.
	NFS Usage Status (Multi-Agent)	Historical information about NFS client and NFS server multiple host operation status over the last 24 hours (on an hourly basis). This report cannot be used on Linux.
System	*I/O Overview*	Historical information about the number of I/O operations over the last hour (on a minute-by-minute basis). This report cannot be used on Linux.
	Process Trend	Historical information about the number of processes over the last month (on a daily basis).
	System Overview (real-time report indicating system operation status)	Real-time information about system operation status.
	System Overview (historical report indicating system operation status)	Historical information about system operation status over the last hour (on a minute-by-minute basis).
	System Overview (8.4) (historical report indicating system operation status)	
	System Utilization Status	Real-time information about system utilization status.
	Workload Status	Real-time information about system workload.
	Workload Status (Multi-Agent)	Historical information about multiple host system workload over the last 24 hours (on an hourly basis).
Disk	*Avg Service Time - Top 10 Devices*	Real-time information about the top 10 devices with the longest average operating hours.
	Avg Service Time Status - Top 10 Devices	Real-time information about the 10 devices with the longest average operating hours.
	Device Detail	Real-time information about the selected device.
	Device Usage Status	Real-time information about device usage status.
	Device Usage Status (Multi-Agent)	Historical information about the usage status of devices used by multiple hosts over the last 24 hours (on an hourly basis).
	Free Space Mbytes - Top 10 Local File Systems	Real-time information about the top 10 local file systems with the largest amount of free space.
	Local File System Detail	Real-time information about the selected local file system.
	Remote File System Detail	Real-time information about the selected remote file system.
	Space Usage - Top 10 Local File Systems	Real-time information about the top 10 local file systems with the highest usage rates.

Category	Report name	Information displayed
Disk	*Space Usage - Top 10 Remote File Systems*	Real-time information about the top 10 remote file systems with the highest usage rates.
Network	*Network Interface Detail*	Real-time information about the network usage status of the selected system.
	Network Interface Summary (real-time report indicating network usage status)	Real-time information about the network usage status.
	Network Interface Summary (historical report indicating network usage status)	Historical information about the network usage status over the last hour (on a minute-by-minute basis).
Process	*Application Status*	Operating status of the applications.
	Application Process Count	Operating status of each process of an application over the last hour (on a minute-by-minute basis).
	Application Process Status	Operating status of each process of an application.
	CPU Usage - Top 10 Processes	Real-time information about the top 10 processes in terms of CPU usage.
	I/O Activity - Top 10 Processes	Real-time information about the top 10 processes in terms of the number of I/O processing operations. This report cannot be used on HP-UX, AIX, and Linux.
	Major Page Faults - Top 10 Processes	Real-time information about the top 10 page fault processes that invoke physical I/O.
	Monitoring Process Detail	Performance information for a particular process over the last hour (on a minute-by-minute basis).
	Process Detail	Real-time information about processes on the selected host.
	Process Overview	Historical information about the operation status of processes over the last hour (on a minute-by-minute basis).
	Process Summary Status	Real-time information about the operation status of a process.
Processor	*CPU Per Processor Status*	Real-time information about processor status.
	CPU Per Processor Usage	Real-time information about CPU usage for each processor.
	CPU Status	Real-time information about CPU usage.
	CPU Status (Multi-Agent)	Historical information about CPU usage of multiple hosts over the last 24 hours (on an hourly basis).
	CPU Trend	Historical information about CPU usage of a specific host over the last month (on a daily basis).
	CPU Trend (Multi-Agent)	Historical information about CPU usage of multiple hosts over the last month (on a daily basis).
	CPU Usage Summary	Historical information about CPU usage over the last hour (on a minute-by-minute basis).
Protocol	*Network Overview*	Historical information about the network usage status over the last hour (on a minute-by-minute basis).
	Network Status (Multi-Agent)	Historical information about the network usage status of multiple hosts over the last 24 hours (on an hourly basis).
	Network Status	Real-time information about network usage status.
Message	*Console Messages*	Historical information about console messages over the last hour (on a minute-by-minute basis). This report cannot be used on Linux.

Category	Report name	Information displayed
Memory	*Memory Paging*	Historical information about the memory usage status over the last hour (on a minute-by-minute basis). This report cannot be used on Linux.
	Memory Paging (8.4)	
	Memory Paging Status	Real-time information about memory and paging. This report cannot be used on Linux.
	Memory Paging Status (8.4)	
	Memory Paging Status (Multi-Agent)	Historical information about the memory usage status of multiple hosts over the last 24 hours (on an hourly basis). This report cannot be used on Linux.
	Memory Paging Status (Multi-Agent) (8.4)	
	Paging Trend (Multi-Agent)	Historical information about page scans of multiple hosts over the last month (on a daily basis). This report cannot be used on Linux.
	Paging Trend (Multi-Agent) (8.4)	
Reserved report	*NFS Client Detail*	Reserved reports that cannot be used.
	NFS Server Detail	

Application Status

Overview

The Application Status report displays the operating status of the application on a real-time basis. It is displayed as a table.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Application Summary Extension (`PD_APP2`)

Fields

Field name	Explanation
Application Exist	The application status specified in the process monitoring settings. **NORMAL** or **ABNORMAL** is displayed. The application status is the result obtained based on the status specified in the monitoring targets. To check the status of the monitoring targets, refer to `Monitoring Status`, which is displayed in the Application Process Detail (`PD_APPD`) record. **NORMAL**: The status of least one of the monitoring targets is **NORMAL** **ABNORMAL**: The status of all of the monitoring targets is **ABNORMAL**
Application Name	The name specified in the process monitoring settings
Application Status	The application status specified in the process monitoring settings. **NORMAL** or **ABNORMAL** is displayed. The application status is the result obtained based on the status specified in the monitoring targets. To check the status of the monitoring targets, refer to `Monitoring Status`, which is displayed in the Application Process Detail (`PD_APPD`) record. **NORMAL**: The status of all of the monitoring targets is **NORMAL** **ABNORMAL**: The status of least one of the monitoring targets is **ABNORMAL**

Drilldown report (field level)

Report name	Explanation
Application Process Status	Displays the operating status of each process of the application on a real-time basis. To display this report, click the **Application Name** field.

Application Process Count

Overview

The Application Process Count report displays the operating status of each process of an application over the last hour (on a minute-by-minute basis). It is displayed as a table and a line graph. This is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/Drilldown Only/`

Record

Application Process Detail (`PD_APPD`)

Fields

Field name	Explanation
Application Name	The name specified in the process monitoring settings
Monitoring Condition	Condition expression for identifying the processes or services being monitored
Monitoring Count	Number of running processes or services that match the monitoring condition
Monitoring Field	The field being monitored
Monitoring Label	The name used to identify the monitoring condition
Monitoring Max	Upper limit on the monitoring count
Monitoring Min	Lower limit on the monitoring count
Monitoring Number	Monitoring condition number
Monitoring Status	Result of the monitoring count condition

Drilldown report (field level)

Report name	Explanation
Monitoring Process Detail	Displays performance information about a specific process. To display this report, click the **Monitoring Label** field.

Application Process Status

Overview

The Application Process Status report displays the operating status of each process and service of an application on a real-time basis. It is displayed as a table. This is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

Application Process Detail (`PD_APPD`)

Fields

Field name	Explanation
Application Name	The name specified in the process monitoring settings
Monitoring Condition	Condition expression for identifying the processes or services being monitored
Monitoring Count	Number of running processes or services that match the monitoring condition
Monitoring Field	The field being monitored
Monitoring Label	The name used to identify the monitoring condition
Monitoring Max	Lower limit on the monitoring count
Monitoring Min	Upper limit on the monitoring count
Monitoring Number	Monitoring condition number
Monitoring Status	Result of the monitoring count condition

Drilldown report (field level)

Report name	Explanation
Application Process Count	Displays the operating status of each process of the application. To display this report, click the **Monitoring Count** field.

Avg Service Time - Top 10 Devices

Overview

The Avg Service Time - Top 10 Devices report displays real-time information about the top 10 devices with the longest average operation hours. It is displayed as an aggregated bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Device Detail (`PI_DEVD`)

Fields

Field name	Explanation
Avg Service Time	The average I/O operation time for a device (seconds). To display this report, click the **Device Detail** report.
Device Name	Device name.

Drilldown report (report level)

Report name	Explanation
I/O Activity - Top 10 Processes	Displays real-time information about the top 10 I/O processes. This report cannot be used on HP-UX, AIX, and Linux.

Drilldown report (field level)

Report name	Explanation
Device Detail	Displays real-time information about the selected device. To display this report, click the **Avg Service Time** field.

Avg Service Time Status - Top 10 Devices

Overview

The Avg Service Time Status - Top 10 Devices report displays real-time information about the top 10 devices with the longest average operation hours. It is displayed as a table and an aggregated bar graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

Device Detail (`PI_DEVD`)

Fields

Field name	Explanation
Avg Service Time	Average I/O operation time for the device (seconds).
Device Name	Device name.
I/O Mbytes	Total transfer size of I/O processing (megabytes).
Mbytes Xferd/sec	Average I/O speed (megabytes per second).
Total I/O Ops/sec	Frequency of I/O processing (number of times per second).

Console Messages

Overview

The Console Messages report displays historical information about console messages displayed over the last hour. Japanese data cannot be displayed. The information is displayed as a table.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/Advanced/`

Record

Logged Messages (`PL_MESS`)

For details about this record, see *5. Collecting Log Information*. Note that Japanese data cannot be used in the messages in this record.

Fields

Field name	Explanation
Message Text	Message text that consists of one line of information extracted from a log file.

CPU Per Processor Status

Overview

The CPU Per Processor Status report displays real-time information about the processor status. The information is displayed as a stacked bar graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

CPU - Per Processor Detail (`PI_CPUP`)

Fields

Field name	Explanation
Idle %	Percentage of time in idle status (%).
Processor ID	Processor identifier.
System %	Percentage of operation time in kernel mode (%).
User %	Percentage of operation time in user mode (%).
Wait %	Percentage of time in I/O waiting status (%).

CPU Per Processor Usage

Overview

The CPU Per Processor Usage report displays real-time CPU usage information for each processor. The information is displayed as a stacked bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

CPU - Per Processor Detail (`PI_CPUP`)

Fields

Field name	Explanation
Idle %	Percentage of time in idle status (%).
Processor ID	Processor identifier.
System %	Percentage of operation time in kernel mode (%).
User %	Percentage of operation time in user mode (%).
Wait %	Percentage of time in I/O waiting status (%).

CPU Status

Overview

The CPU Status report displays real-time information about CPU usage. The information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
1-Minute Run Queue Avg	The average number of threads over the last one minute, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
15-Minute Run Queue Avg	The average number of threads over the last 15 minutes, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
5-Minute Run Queue Avg	The average number of threads over the last 5 minutes, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function). To display a CPU Per Processor Status report, click this field.
Context Switches	Number of times the context switch is executed.
Idle %	Percentage of time in idle status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Kernel CPU %	Percentage of operation time in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Record Time	Time when the record was created (Greenwich Mean Time).
System Calls	Number of times a system call was issued.
User CPU %	Percentage of operation time in user mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Wait %	Percentage of time in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Drilldown report (field level)

Report name	Explanation
CPU Per Processor Status	Displays real-time information about the processor status. To display this report, click the **CPU %** field.

CPU Status (Multi-Agent)

Overview

The CPU Status (Multi-Agent) report displays hourly historical information about CPU usage for multiple hosts over the last 24 hours. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
1-Minute Run Queue Avg	The average number of threads over the last one minute, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Context Switches	Number of times the context switch was executed.
Kernel CPU %	Percentage of operation time in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
System Calls	Number of times a system call was issued.
User CPU %	Percentage of operation time in user mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Wait %	Percentage of time in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

CPU Trend

Overview

The CPU Trend report displays historical information about CPU usage for a specific host over the last month that is summarized on a daily basis. This information is displayed as a line graph.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Kernel CPU %	Percentage of operation time in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
User CPU %	Percentage of operation time in user mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

CPU Trend (Multi-Agent)

Overview

The CPU Trend (Multi-Agent) report displays daily historical information about CPU usage for multiple hosts over the last month. The information is displayed as a line graph.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Agent Instance#	Name of the host on which PFM - Agent for Platform is operating.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function). To display a CPU Trend report, click this field.

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

Drilldown report (report level)

Report name	Explanation
CPU Trend	Displays historical information about CPU usage of a specific host over the last month (on a daily basis). To display this report, click the **CPU %** field.

CPU Usage - Top 10 Processes

Overview

The CPU Usage - Top 10 Processes report displays real-time information about the top 10 processes in terms of CPU usage. The information is displayed as an aggregated bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Process Detail Interval (`PD_PDI`)

Fields

Field name	Explanation
CPU %	Average value (%) obtained by dividing the CPU usage of processes by the number of processors (on AIX, the number of processors logically partitioned by the Micro-Partitioning function). To display a Process Detail report, click this field.
PID	Process ID.
Program	Running program name.

Drilldown report (report level)

Report name	Explanation
CPU Per Processor Usage	Displays real-time CPU usage information for each processor.

Drilldown report (field level)

Report name	Explanation
Process Detail	Displays real-time process information for the selected host. To display this report, click the **CPU %** field.

CPU Usage Summary

Overview

The CPU Usage Summary report displays minute-by-minute historical information about CPU usage for the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
15-Minute Run Queue Avg	The average number of threads over the last 15 minutes, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Context Switches	Number of times the context switch was executed.
Idle %	Percentage of time that the CPU is in idle status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Interrupts	Number of times interrupt occurred.
Kernel CPU %	Percentage of time that the CPU is operating in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Run Queue	The number of processes, including processes waiting in the queue for the kernel.
System Calls	Number of times a system call was issued.
User CPU %	Percentage of time that the CPU is operating in user mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Wait %	Percentage of time that the CPU is in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Device Detail

Overview

The Device Detail report displays real-time information about the selected device. The information is displayed as a list. This report is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

Device Detail (`PI_DEVD`)

Fields

Field name	Explanation
Avg Service Time	Average I/O operation time for the device (seconds).
Avg Wait Time	Average I/O waiting time for the device (seconds).
Device Name	Device name.
I/O Mbytes	Total transfer size of I/O processing (megabytes).
Mbytes Xferd/sec	Average I/O speed (megabytes/second).
Queue Length	Queue length of devices for which the I/O processing amount per second is 1.
Read Mbytes	Transfer size of read processing (megabytes).
Reads/sec	Frequency of read processing (number of times per second).
Total I/O Ops	Number of times I/O processing occurred.
Total I/O Ops/sec	Frequency of I/O processing (number of times per second).
Total Service Time	Total operation time for processing for a device (seconds). This value includes the waiting time. On HP-UX, because this is the sum of the total operation time of all I/O, if processing for a device is executed continuously, the value may greatly exceed the Interval value. With other OSs, if processing for a device is executed continuously, the Interval value may be exceeded.
Write Mbytes	Write processing transfer size (megabytes).
Writes/sec	Frequency of write processing (number of times per second).

Device Usage Status

Overview

The Device Usage Status report displays real-time usage status information for devices. It is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

Device Summary (`PI_DEVS`)

Fields

Field name	Explanation
Avg Service Time/op	Average I/O operation time for the device (seconds).
Devices	Number of devices.
I/O Mbytes	Total transfer size of I/O processing (megabytes).
Mbytes Xferd/sec	Average I/O processing speed (megabytes per second).
Read Ops %	Of I/O processing, the read processing Percentage (%).
Record Time	Time when the record was created (Greenwich Mean Time).
Total I/O Ops	Number of times I/O processing occurred.
Total I/O Ops/sec	Frequency of I/O processing (number of times per second).
Write Ops %	Of I/O processing, the write processing Percentage (%).

Device Usage Status (Multi-Agent)

Overview

The Device Usage Status (Multi-Agent) report displays hourly historical information about the usage status of devices used by multiple hosts over the last 24 hours. The information is displayed as a table.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/Advanced/`

Record

Device Summary (`PI_DEVS`)

Fields

Field name	Explanation
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
Avg Service Time/op	Average I/O operation time for the device (seconds).
Devices	Number of devices.
I/O Mbytes	Total transfer size of I/O processing (megabytes).
Mbytes Xferd/sec	Average I/O processing speed (megabytes per second).
Total I/O Ops	Number of times I/O processing occurred.
Total I/O Ops/sec	Frequency of I/O processing (number of times per second).

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

Free Space Mbytes - Top 10 Local File Systems

Overview

The Free Space Mbytes - Top 10 Local File Systems report displays real-time information about the top 10 local file systems with the largest amount of free space. The information is displayed as a table.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

File System Detail - Local (`PD_FSL`)

Fields

Field name	Explanation
File System	File system mount point.
Mbytes Free	Available size (megabytes).
Mbytes Rsvd	Reserved size for superuser (megabytes).
Mbytes in Use	Size being used by general user (megabytes).
Record Time	Time when the record was created (Greenwich Mean Time).
Total Inodes Available	Number of i nodes that can be used by general users.
Total Size Mbytes	File system size (megabytes).

I/O Activity - Top 10 Processes

Overview

The I/O Activity - Top 10 Processes report displays real-time information about the top 10 processes in terms of the number of I/O processing operations performed. It is displayed as an aggregated bar graph.

Note:

This report cannot be used on HP-UX, AIX, and Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Process Detail (`PD`)

Fields

Field name	Explanation
PID	Process ID.
Program	Program name, which is the same as the character string displayed in the `COMMAND (CMD)` row of the `ps -e` command. On AIX, this value is limited to a maximum of 32 bytes. On Linux, this value is limited to a maximum of 15 bytes. On HP-UX, this value is limited to a maximum of 14 bytes. On Solaris, this value is limited to a maximum of 15 bytes.
Total I/O Ops/sec	Frequency of I/O processing (number of times per second). To display a Process Detail report, click this field.

Drilldown report (field level)

Report name	Explanation
Process Detail	Displays real-time process information for the selected host. To display this report, click the **Total I/O Ops/sec** field.

I/O Overview

Overview

The I/O Overview report displays minute-by-minute historical information about the number of I/O operations performed over the last hour. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Block Ops	Number of times block I/O processing occurred.
Block Reads	Number of times block read processing occurred.
Block Writes	Number of times block write processing occurred.
Logical I/O Ops	Number of times logical I/O processing occurred.
Logical Reads	Number of times logical read processing occurred.
Logical Writes	Number of times logical write processing occurred.
Physical I/O Ops	Number of times physical I/O processing occurred.
Physical Reads	Number of times physical read processing occurred.
Physical Writes	Number of times physical write processing occurred.

Local File System Detail

Overview

The Local File System Detail report displays real-time information about the selected local file system. The information is displayed as a list. This report is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

File System Detail - Local (`PD_FSL`)

Fields

Field name	Explanation
Available Space %	Percentage of size available to general users (megabytes) (%).
Available Space Blocks	Number of logical blocks available to general users.
Available Space Mbytes	Size available to general user (megabytes).
Block Size	Logical block size (preferred file system block size) (bytes).
Blocks in Use	Number of logical blocks in use.
Device Name	Name of device where file system is placed.
File System	File system mount point.
Mbytes Free	Available size (megabytes).
Mbytes Rsvd	Reserved size for superusers (megabytes).
Mbytes in Use	Size used by general users (megabytes).
Mbytes in Use %	Percentage of size used by general users (megabytes) (%).
Total Inodes	Number of file system i nodes.
Total Inodes Available %	Percentage of number of i nodes available to general users (%).
Total Inodes Free %	Percentage of number of i nodes available (%).
Total Inodes Rsvd %	Percentage of number of reserved i nodes for superusers (%).
Total Inodes in Use %	Percentage of number of i nodes in use (%).
Total Size Blocks	Number of file system logical blocks.

Major Page Faults - Top 10 Processes

Overview

The Major Page Faults - Top 10 Processes report displays real-time information about the top 10 processes of page faults that invoke physical I/O. The information is displayed as an aggregated bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Process Detail Interval (`PD_PDI`)

Fields

Field name	Explanation
Major Faults	Number of page faults that invoked physical I/O. To display a Process Detail report, click this field.
PID	Process ID.
Program	Name of running program.

Drilldown report (field level)

Report name	Explanation
Process Detail	Displays real-time process information for the selected host. To display this report, click the **Major Faults** field.

Memory Paging

Overview

The Memory Paging report displays minute-by-minute historical information about memory usage over the last hour. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Alloc Swap %	Percentage of number of megabytes in swap area in use (%).
Cache Read %	Percentage of cache read processing of read processing (%).
Cache Write %	Percentage of cache write processing of write processing (%).
Free Mem %	Percentage of available physical memory (%) (megabytes).
Major Faults	Number of page faults that invoked physical I/O.
Minor Faults	Number of page faults that did not invoke physical I/O.
Page Ops/sec	Frequency of page-in and page-out processing operations (number of times per second).
Swap-In Ops	Number of swap-in processing operations that occurred.
Swap-Out Ops	Number of swap-out processing operations that occurred.
Total Page Scans	Number of pages checked by page scans.
Total Swaps	Number of swap processing operations that occurred.

Memory Paging (8.4)

Overview

The Memory Paging (8.4) report displays minute-by-minute historical information about memory usage over the last hour. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Alloc Swap %	Percentage of number of megabytes in swap area in use (%).
Cache Read %	Percentage of cache read processing of read processing (%).
Cache Write %	Percentage of cache write processing of write processing (%).
Effective Free Mem %	Percentage of the size (in megabytes) of physical memory that can be used by applications (%).
Major Faults	Number of page faults that invoked physical I/O.
Minor Faults	Number of page faults that did not invoke physical I/O.
Page Ops/sec	Frequency of page-in and page-out processing operations (number of times per second).
Swap-In Ops	Number of swap-in processing operations that occurred.
Swap-Out Ops	Number of swap-out processing operations that occurred.
Total Page Scans	Number of pages checked by page scans.
Total Swaps	Number of swap processing operations that occurred.

Memory Paging Status

Overview

The Memory Paging Status report displays real-time information about memory and paging. This information is displayed as a list and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Free Swap %	Percentage of available swap area size (megabytes) (%).
Free Swap Mbytes	Available swap area size (megabytes).
Major Faults/sec	Frequency of page faults that invoked physical I/O (number of times per second).
Minor Faults/sec	Frequency of page faults that did not invoke physical I/O (number of times per second).
Page Scans/sec	Frequency of page scans (number of times per second).
Pages In/sec	Frequency of page reads by page in processing (number of pages per second).
Pages Out/sec	Frequency of pages taken out by page out processing (number of pages per second).
Record Time	Time when the record was created (Greenwich Mean Time).

Memory Paging Status (8.4)

Overview

The Memory Paging Status (8.4) report displays real-time information about memory and paging. This information is displayed as a list and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Free Swap %	Percentage of available swap area size (megabytes) (%).
Free Swap Mbytes	Available swap area size (megabytes).
Major Faults/sec	Frequency of page faults that invoked physical I/O (number of times per second).
Minor Faults/sec	Frequency of page faults that did not invoke physical I/O (number of times per second).
Page Scans/sec	Frequency of page scans (number of times per second).
Pages In/sec	Frequency of page reads by page in processing (number of pages per second).
Pages Out/sec	Frequency of pages taken out by page out processing (number of pages per second).
Record Time	Time when the record was created (Greenwich Mean Time).

Memory Paging Status (Multi-Agent)

Overview

The Memory Paging Status (Multi-Agent) report displays hourly historical information about memory usage for multiple hosts over the last 24 hours. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
Free Swap %	Percentage of available swap area size (megabytes) (%).
Free Swap Mbytes	Available swap area size (megabytes).
Major Faults/sec	Frequency of the page faults that occurred that invoked physical I/O (number of times per second).
Page Scans/sec	Frequency of page scans (number of times per second).
Pages In/sec	Frequency at which pages are read in by page in processing (number of pages per second).
Pages Out/sec	Frequency at which pages are taken out by page out processing (number of pages per second).

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

Memory Paging Status (Multi-Agent) (8.4)

Overview

The Memory Paging Status (Multi-Agent) (8.4) report displays hourly historical information about memory usage for multiple hosts over the last 24 hours. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
Free Swap %	Percentage of available swap area size (megabytes) (%).
Free Swap Mbytes	Available swap area size (megabytes).
Major Faults/sec	Frequency of the page faults that occurred that invoked physical I/O (number of times per second).
Page Scans/sec	Frequency of page scans (number of times per second).
Pages In/sec	Frequency at which pages are read in by page in processing (number of pages per second).
Pages Out/sec	Frequency at which pages are taken out by page out processing (number of pages per second).

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

Monitoring Process Detail

Overview

The Monitoring Process Detail report displays performance information about a specific process over the last hour (on a minute-by-minute basis). It is displayed as a line graph. This is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/Drilldown Only/`

Record

Application Process Interval (`PD_APSI`)

Fields

Field name	Explanation
Application Name	The name specified in the process monitoring settings
CPU %	Percentage of the processor time used by the process (%). In a multi-processor environment, usage is displayed with *number of processors* × 100% as the maximum value.
Monitoring Field	The field being monitored
Monitoring Label	The name used to identify the monitoring condition
Monitoring Number	Monitoring condition number
PID	Process ID. Unique ID of the process being executed
Program Name	The name of the program
Real Mem Kbytes	Amount of physical memory being used (kilobytes)
Swaps	Number of times swapping occurred
System CPU	Time running in the kernel mode (sec)
User CPU	Time running in the user mode (sec)
User ID	The effective user ID of the process
Virtual Mem Kbytes	Amount of virtual memory being used (kilobytes)

Network Interface Detail

Overview

The Network Interface Detail report displays real-time information about network usage for the selected system. The information is displayed as a list. This report is a drilldown report.

Note:

The interface flags set in the Flags (`FLAGS`) field that can be evaluated are as follows:

- HP-UX
  `UP`, `LOOPBACK`
- Solaris
  `UP`, `BROADCAST`, `DEBUG`, `LOOPBACK`, `POINTOPOINT`, `NOTRAILERS`, `RUNNING`, `NOARP`, `PROMISC`, `ALLMULTI`, `INTELLIGENT`, `MULTICAST`, `MULTI_BCAST`, `UNNUMBERED`, `PRIVATE`
- AIX
  `UP`, `BROADCAST`, `DEBUG`, `LOOPBACK`, `POINTOPOINT`, `NOTRAILERS`, `RUNNING`, `NOARP`, `PROMISC`, `ALLMULTI`, `OACTIVE`, `SIMPLEX`, `MULTICAST`
- Linux
  `UP`, `BROADCAST`, `DEBUG`, `LOOPBACK`, `POINTOPOINT`, `NOTRAILERS`, `RUNNING`, `NOARP`, `PROMISC`, `ALLMULTI`, `MULTICAST`, `MASTER`, `SLAVE`, `PORTSEL`, `AUTOMEDIA`, `DYNAMIC`

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

Network Interface Detail (`PI_NIND`)

Fields

Field name	Explanation
Flags	Sets the interface flag. When the number of bytes exceeds 80, the last character is >. For HP-UX, all flags that indicate the interface status are included in this field value (the `ifconfig` command only displays part of the flags).
IP Address	IP address that was found first. This is displayed using the Internet address format.
Interface	Interface name.
Max Transmission Unit	Maximum packet size (bytes).
Network Mask	Subnet mask of the IP address that was found first. This is displayed using the Internet address format.
Network Name	Network name. Network names of 1,028 bytes or more cannot be handled. If a network name is 256 bytes or more, a maximum of 255 bytes is displayed. On HP-UX, AIX, and Linux, if NIS is operating and there are no entries in the NIS network address database, the result obtained by applying mask processing on the first IP address found is displayed.
Record Time	Time when the record was created (Greenwich Mean Time).
Type	Interface type.

Network Interface Summary (real-time report indicating network usage status)

Overview

The Network Interface Summary report displays real-time information about the network usage status. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

Network Interface Summary (`PI_NINS`)

Fields

Field name	Explanation
Interface Count	Number of interfaces. To display a Network Interface Detail report, click this field.
Pkt Collisions	Number of packet collisions.
Pkt Receive Errors	Number of packet receive errors.
Pkt Xmit Errors	Number of packet transmission errors.
Pkts Rcvd	Number of packets received.
Pkts Rcvd/sec	Frequency at which packets were received (number of packets per second).
Pkts Xmitd	Number of packets sent.
Pkts Xmitd/sec	Frequency at which packets are sent (number of packets per second).

Drilldown report (field level)

Report name	Explanation
Network Interface Detail	Displays real-time information about the network usage status for the selected system. To display this report, click the **Interface Count** field.

Network Interface Summary (historical report indicating network usage status)

Overview

The Network Interface Summary report displays minute-by-minute historical information about the network usage status over the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/Advanced/`

Record

Network Interface Summary (`PI_NINS`)

Fields

Field name	Explanation
Interface Count	Number of interfaces.
Pkt Collisions	Number of packet collisions.
Pkt Receive Errors	Number of packet receive errors.
Pkt Xmit Errors	Number of packet transmission errors.
Pkts Rcvd	Number of packets received.
Pkts Rcvd/sec	Frequency at which packets were received (number of packets per second).
Pkts Xmitd	Number of packets sent.
Pkts Xmitd/sec	Frequency at which packets are sent (number of packets per second).

Network Overview

Overview

The Network Overview report displays minute-by-minute historical information about the network usage status over the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
ICMP Pkts In	Number of IPv4 ICMP packets received. The combined total of local packets, remote packets, and packets for which an error occurred.
ICMP Pkts Out	Number of IPv4 ICMP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred.
IP Pkts In	Number of IPv4 IP packets received. The combined total of local and remote packets. This number does not include packets for which an error occurred.
IP Pkts Out	Number of IPv4 IP packets sent. The combined total of local and remote packets. This number does not include packets for which an error occurred.
TCP Pkts In	Number of IPv4 TCP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
TCP Pkts Out	Number of IPv4 TCP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
Total Pkts In	The combined total of IPv4 TCP packets received, IPv4 UDP packets received, and IPv4 ICMP packets received. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts In	Number of IPv4 UDP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts Out	Number of IPv4 UDP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.

Network Status

Overview

The Network Status report displays real-time information about the network usage status. The information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
ICMP Pkts In	Number of IPv4 ICMP packets received. The combined total of local packets, remote packets, and packets for which an error occurred.
ICMP Pkts Out	Number of IPv4 ICMP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred.
IP Pkts In	Number of IPv4 IP packets received. The combined total of local and remote packets. This number does not include packets for which an error occurred.
IP Pkts Out	Number of IPv4 IP packets sent. The combined total of local and remote packets. This number does not include packets for which an error occurred.
Record Time	Time when the record was created (Greenwich Mean Time).
TCP Pkts In	Number of IPv4 TCP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
TCP Pkts Out	Number of IPv4 TCP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts In	Number of IPv4 UDP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts Out	Number of IPv4 UDP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.

Network Status (Multi-Agent)

Overview

The Network Status (Multi-Agent) report displays hourly historical information about the network usage status for multiple hosts over the last 24 hours. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
ICMP Pkts In	Number of IPv4 ICMP packets received. The combined total of local packets, remote packets, and packets for which an error occurred.
ICMP Pkts Out	Number of IPv4 ICMP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred.
IP Pkts In	Number of IPv4 IP packets received. The combined total of local and remote packets. This number does not include packets for which an error occurred.
IP Pkts Out	Number of IPv4 IP packets sent. The combined total of local and remote packets. This number does not include packets for which an error occurred.
Kernel CPU %	Percentage of operation time in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
TCP Pkts In	Number of IPv4 TCP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
TCP Pkts Out	Number of IPv4 TCP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts In	Number of IPv4 UDP packets received. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.
UDP Pkts Out	Number of IPv4 UDP packets sent. The combined total of local packets, remote packets, and packets for which an error occurred. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

NFS Activity Overview

Overview

The NFS Activity Overview report displays minute-by-minute historical information about the operation status of the NFS client and NFS server over the last hour. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
NFS Client Lookup Ops	Number of times lookup processing occurred on the NFS client.
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Client Total Ops	Total number of processing operations that occurred on the NFS client.
NFS Server Lookup Ops	Number of times lookup processing was performed on the NFS server.
NFS Server Total Bad Ops	Number of times processing failed on the NFS server.
NFS Server Total Ops	Number of times processing occurred on the NFS server.

NFS Load Trend

Overview

The NFS Load Trend report displays daily historical information about the operation status of the NFS client and NFS server over the last month. The information is displayed as a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
NFS Client Ops/sec	Frequency at which processing occurred on the NFS client (number of times per second).
NFS Server Ops/sec	Frequency at which processing occurred on the NFS server (number of times per second).

NFS Usage Status

Overview

The NFS Usage Status report displays real-time information about the operation status of the NFS client and NFS server. The information is displayed as a list and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
NFS Client Lookup Ops	Number of times lookup processing occurred on the NFS client.
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Client Total Ops	Total number of processing operations that occurred on the NFS client.
NFS Server Lookup Ops	Number of times lookup processing occurred on the NFS server.
NFS Server Total Bad Ops	Number of times processing failed on the NFS server.
NFS Server Total Ops	Number of times processing occurred on the NFS server.
Record Time	Time when the record was created (Greenwich Mean Time).

NFS Usage Status (Multi-Agent)

Overview

The NFS Usage Status (Multi-Agent) report displays hourly historical information about the operation status of the NFS client and NFS server over the last 24 hours. The information is displayed as a table and a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Agent Instance#	Name of host on which PFM - Agent for Platform is operating.
NFS Client Lookup Ops	Number of times lookup processing occurred on the NFS client.
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Client Total Ops	Total number of processing operations that occurred on the NFS client.
NFS Server Lookup Ops	Number of times lookup processing occurred on the NFS server.
NFS Server Total Bad Ops	Number of times processing failed on the NFS server.
NFS Server Total Ops	Number of times processing occurred on the NFS server.

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

Paging Trend (Multi-Agent)

Overview

The Paging Trend (Multi-Agent) report displays daily historical information about page scans for multiple hosts over the last month. The information is displayed as a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Page Scans/sec	Frequency of page scans (number of times per second).

Paging Trend (Multi-Agent) (8.4)

Overview

The Paging Trend (Multi-Agent) (8.4) report displays daily historical information about page scans for multiple hosts over the last month. The information is displayed as a line graph.

Note:

This report cannot be used on Linux.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Page Scans/sec	Frequency of page scans (number of times per second).

Process Detail

Overview

The Process Detail report displays real-time information about processes for the selected host. The information is displayed as a list. This report is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

Process Detail Interval (`PD_PDI`)

Fields

Field name	Explanation
CPU %	Average value (%) obtained by dividing the CPU usage of processes by the number of processors (on AIX, the number of processors logically partitioned by the Micro-Partitioning function).

Executable Text Kbytes	Text size in use (kilobytes).
	You need to set the `SAUNIXPMAP` environment variable beforehand to collect this field on Solaris systems. Execute the following command from the command-line interface:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`

Major Faults	Number of page faults that invoked physical I/O.
Minor Faults	Number of page faults that did not invoke physical I/O.
Parent PID	Process ID of the parent process.
PID	Process ID.
Program	Name of running program.
Reads	AIX: Number of times RAW read processing occurred. Solaris: Number of times Block read processing occurred.
Real Mem Kbytes	Physical memory size in use (kilobytes).
Record Time	Time when the record was created (Greenwich Mean Time).
Shared Mem Kbytes	Shared memory size in use (kilobytes).
Stack Kbytes	Stack size in use (kilobytes).
Start Time	Process start time.
Swaps	Number of times swap occurred.
System CPU	Period of time that the system was operating in kernel mode (seconds).
Total I/O Kbytes	Total transfer size of I/O processing (kilobytes).
Total I/O Ops	Number of I/O processing operations.
Total Process Kbytes	Process size (kilobytes). AIX: The value of this field is the same as the value that is displayed in the `SZ` row obtained using the `ps -l` command.

Field name	Explanation
Total Process Kbytes	HP-UX and Solaris: This value is the same as the value obtained by multiplying the page size by the value displayed in the `SZ` row obtained using the `ps -l` command. Linux: This value is the same as the value displayed in the `SIZE` row obtained by using the `top` command.
User	Process effective user name.
User CPU	Period of time that the system was operating in user mode (seconds).
Writes	AIX: The number of times RAW write processing occurred. Solaris: The number of times Block write processing occurred.

Process Overview

Overview

The Process Overview report displays minute-by-minute historical information about the operation status of processes performed over the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/Advanced/`

Record

Process Summary (`PD_PDS`)

Fields

Field name	Explanation
Active System Processes	Number of active system processes. Displays `0` for the first real-time report.
Active User Processes	Number of active user processes. Displays `0` for the first real-time report. On AIX and Linux, the total number of active system processes and user processes is displayed.
Active Users	Number of users who are executing processes counted as active system processes or user processes. Displays `0` for the first real-time report.
New Processes	Number of processes started. To calculate the difference from the previous process information, displays `0` for the first real-time report.
Processes	Number of processes in the system.
Runnable Processes	Number of runnable processes.
Sleeping Processes	Number of sleeping processes.
Stopped Processes	Number of stopped processes.
Swapped Processes	Number of swapped processes. From the process table, checks the number of processes that are not in the core.
Terminated Processes	Number of terminated processes. To calculate the difference from the previous process information, displays `0` for the first real-time report.
Zombie Processes	Number of zombie processes.

Process Summary Status

Overview

The Process Summary Status report displays real-time information about the operation status of processes. The information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

Process Summary (`PD_PDS`)

Fields

Field name	Explanation
Active System Processes	Number of active system processes. Displays `0` for the first real-time report.
Active User Processes	Number of active user processes. Displays `0` for the first real-time report. On AIX and Linux, the total number of active system processes and user processes is displayed.
Processes	Number of processes in the system.
Record Time	Time when the record was created (Greenwich Mean Time).
Sleeping Processes	Number of sleeping processes.
Swapped Processes	Number of swapped processes. From the process table, checks the number of processes that are not in the core.
Users	Number of real users.
Zombie Processes	Number of zombie processes.

Process Trend

Overview

The Process Trend report displays daily historical information about the number of operating processes over the last month. This information is displayed as a line graph.

Storage destination

`Reports/UNIX/Monthly Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Processes	Number of processes in the system.

Remote File System Detail

Overview

The Remote File System Detail report displays real-time information about the selected remote file system. This information is displayed as a list. This report is a drilldown report.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/Drilldown Only/`

Record

File System Detail - Remote (`PD_FSR`)

Fields

Field name	Explanation
Available Blocks	Number of logical blocks available to general users.
Available Space %	Percentage of size available to general users (megabytes) (%).
Available Space Mbytes	Size available to general users (megabytes).
Block Size	Logical block size (preferred file system block size) (bytes).
Blocks in Use	Number of logical blocks in use.
File System	File system mount point.
Host	Name of the host on which the file system is placed.
Mbytes Free	Available size (megabytes).
Mbytes Rsvd	Reserved size for superuser (megabytes).
Mbytes in Use	Size used by general users (megabytes).
Mbytes in Use %	Percentage of size used by general users (megabytes) (%).
Remote File System	Remote file system name.
Total Size Blocks	Number of file system logical blocks.

Space Usage - Top 10 Local File Systems

Overview

The Space Usage - Top 10 Local File Systems report displays real-time information about the top 10 most used local file systems. It is displayed as an aggregated bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

File System Detail - Local (`PD_FSL`)

Fields

Field name	Explanation
File System	File system mount point.
Mbytes in Use %	Size used by general users (megabytes) (%). To display a Local File System Detail report, click this field.
Total Size Mbytes	File system size (megabytes).

Drilldown report (field level)

Report name	Explanation
Local File System Detail	Displays real-time information about the selected local file system. To display this report, click the **Mbytes in Use %** field.

Space Usage - Top 10 Remote File Systems

Overview

The Space Usage - Top 10 Remote File Systems report displays real-time information about the top 10 most used remote file systems. It is displayed as an aggregated bar graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

File System Detail - Remote (`PD_FSR`)

Fields

Field name	Explanation
Mbytes in Use %	Size used by general users (megabytes) (%). To display a Remote File System Detail report, click this field.
Remote File System	Remote file system name.
Total Size Mbytes	File system size (megabytes).

Drilldown report (field level)

Report name	Explanation
Remote File System Detail	Displays real-time information about the selected remote file system. To display this report, click the **Mbytes in Use %** field.

System Overview (real-time report indicating system operation status)

Overview

The System Overview report displays real-time information about the operation status of a system. This information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function). To display a CPU Usage - Top 10 Processes report, click this field.
Free Swap %	Percentage of available swap area size (megabytes) (%).
Kernel CPU %	Percentage of operation time in kernel mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Logical I/O Ops	Number of times logical I/O processing occurred. To display an I/O Activity - Top 10 Processes report, click this field.
Major Faults/sec	Frequency of page faults that invoked physical I/O (number of times per second). To display a Major Page Faults - Top 10 Processes report, click this field.
Minor Faults/sec	Frequency of page faults that did not invoke physical I/O (number of times per second).
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Server Total Bad Ops	Total number of times processing failed on the NFS server.
Physical I/O Ops	Number of times physical I/O processing occurred. To display an Avg Service Time - Top 10 Devices report, click this field.
Record Time	Time when the record was created (Greenwich Mean Time).
Run Queue	The number of processes, including processes waiting in the execution queue for the kernel.
Swapped-In Pages	Number of pages taken in by swap-in processing.
Swapped-Out Pages	Number of pages taken out by swap-out processing.
Total Page Scans	Number of pages checked by page scan.
Total Pkts In	Total number of IPv4 TCP packets received, IPv4 UDP packets received, and IPv4 ICMP packets received. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed. To display a Network Interface Summary report, click this field.
User CPU %	Percentage of operation time in user mode (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Field name	Explanation
Wait %	Percentage of time in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Drilldown report (report level)

Report name	Explanation
Space Usage - Top 10 Local File Systems	Displays real-time information for the top 10 most used local file systems.
Space Usage - Top 10 Remote File Systems	Displays real-time information for the top 10 most used remote file systems.

Drilldown report (field level)

Report name	Explanation
Avg Service Time - Top 10 Devices	Displays real-time information for the top 10 devices having the longest average operation time. To display this report, click the **Physical I/O Ops** field.
CPU Usage - Top 10 Processes	Displays real-time information for the top 10 processes with the highest CPU usage. To display this report, click the **CPU %** field.
I/O Activity - Top 10 Processes	Displays real-time information for the top 10 processes with the highest number of I/O operations. To display this report, click the **Logical I/O Ops** field. This report cannot be used on HP-UX, AIX, and Linux.
Major Page Faults - Top 10 Processes	Displays real-time information for the top 10 processes with the highest number of page faults that invoke physical I/O. To display this report, click the **Major Faults/sec** field.
Network Interface Summary	Displays network usage status information. To display this report, click the **Total Pkts In** field.
NFS Client Detail	Reserved report that cannot be used.
NFS Server Detail	Reserved report that cannot be used.

System Overview (historical report indicating system operation status)

Overview

The System Overview report displays minute-by-minute historical information about the operation status for a system over the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Block Reads	Number of times block read processing occurred. To display an I/O Overview report, click this field.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function). To display a CPU Usage Summary report, click this field.
Major Faults	Number of page faults that invoked physical I/O. To display a Memory Paging report, click this field.
NFS Client Total Bad Ops	The total number of times processing failed on the NFS client. To display an NFS Activity Overview report, click this field.
Swapped-In Pages/sec	Frequency at which pages were taken in by swap-in processing (number of pages per second). On AIX, the frequency at which paging area pages only were taken in by swap-in processing (number of pages per second).
Swapped-Out Pages/sec	Frequency at which pages were taken out by swap-out processing (number of pages per second). On AIX, the frequency at which paging area pages only were taken out by swap-out processing (number of pages per second).
Total Pkts In	The combined total of IPv4 TCP packets received, IPv4 UDP packets received, and IPv4 ICMP packets received. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed. To display a Network Overview report, click this field.
Wait %	Percentage of time in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Drilldown report (field level)

Report name	Explanation
CPU Usage Summary	Displays minute-by-minute historical information about the CPU usage over the last hour. To display this report, click the **CPU %** field.
I/O Overview	Displays minute-by-minute historical information about the number of I/Os over the last hour. To display this report, click the **Block Reads** field.

Report name	Explanation
I/O Overview	This report cannot be used on Linux.
Memory Paging	Displays minute-by-minute historical information about the memory usage status over the last hour. To display this report, click the **Major Faults** field. This report cannot be used on Linux.
Network Overview	Displays minute-by-minute historical information about the network usage status over the last hour. To display this report, click the **Total Pkts In** field.
NFS Activity Overview	Displays minute-by-minute historical information about the operation status of the NFS client and NFS server over the last hour. To display this report, click the **NFS Client Total Bad Ops** field. This report cannot be used on Linux.

System Overview (8.4) (historical report indicating system operation status)

Overview

The System Overview (8.4) report displays minute-by-minute historical information about the operation status for a system over the last hour. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Troubleshooting/Recent Past/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
Block Reads	Number of times block read processing occurred. To display an I/O Overview report, click this field.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function). To display a CPU Usage Summary report, click this field.
Major Faults	Number of page faults that invoked physical I/O. To display a Memory Paging (8.4) report, click this field.
NFS Client Total Bad Ops	The total number of times processing failed on the NFS client. To display an NFS Activity Overview report, click this field.
Swapped-In Pages/sec	Frequency at which pages were taken in by swap-in processing (number of pages per second). On AIX, the frequency at which paging area pages only were taken in by swap-in processing (number of pages per second).
Swapped-Out Pages/sec	Frequency at which pages were taken out by swap-out processing (number of pages per second). On AIX, the frequency at which paging area pages only were taken out by swap-out processing (number of pages per second).
Total Pkts In	The combined total of IPv4 TCP packets received, IPv4 UDP packets received, and IPv4 ICMP packets received. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed. To display a Network Overview report, click this field.
Wait %	Percentage of time in I/O waiting status (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).

Drilldown report (field level)

Report name	Explanation
CPU Usage Summary	Displays minute-by-minute historical information about the CPU usage over the last hour. To display this report, click the **CPU %** field.
I/O Overview	Displays minute-by-minute historical information about the number of I/Os over the last hour.

Report name	Explanation
I/O Overview	To display this report, click the **Block Reads** field. This report cannot be used on Linux.
Memory Paging (8.4)	Displays minute-by-minute historical information about the memory usage status over the last hour. To display this report, click the **Major Faults** field. This report cannot be used on Linux.
Network Overview	Displays minute-by-minute historical information about the network usage status over the last hour. To display this report, click the **Total Pkts In** field.
NFS Activity Overview	Displays minute-by-minute historical information about the operation status of the NFS client and NFS server over the last hour. To display this report, click the **NFS Client Total Bad Ops** field. This report cannot be used on Linux.

System Utilization Status

Overview

The System Utilization Status report displays real-time information about the utilization status of a system. The information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
15-Minute Run Queue Avg	The average number of threads over the last 15 minutes, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
Boot Time	Last boot time.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Free Swap %	Percentage of available swap area size (megabytes) (%).
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Server Total Bad Ops	Total number of times processing failed on the NFS server.
Page Scans/sec	Frequency of page scans (number of times per second).
Record Time	Time when the record was created (Greenwich Mean Time).
Total Pkts In	The combined total of IPv4 TCP packets received, IPv4 UDP packets received, and IPv4 ICMP packets received. In 09-00 or later, the total number of packets for IPv4 or IPv6 is displayed.

Workload Status

Overview

The Workload Status report displays real-time information about the system workload. This information is displayed as a list and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Real-Time/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
5-Minute Run Queue Avg	The average number of threads over the last 5 minutes, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Context Switches/sec	Frequency of context switches (number of times per second).
NFS Client Total Bad Ops	Total number of times processing failed on the NFS client.
NFS Server Total Bad Ops	Total number of times processing failed on the NFS server.
Processes	Number of processes in the system.
Record Time	Time when the record was created (Greenwich Mean Time).
System Calls/sec	Frequency of system calls (number of times per second).
Users	Number of real users.

Workload Status (Multi-Agent)

Overview

The Workload Status (Multi-Agent) report displays hourly historical information about the system workload for multiple hosts over the last 24 hours. The information is displayed as a table and a line graph.

Storage destination

`Reports/UNIX/Status Reporting/Daily Trend/`

Record

System Summary Overview (`PI`)

Fields

Field name	Explanation
1-Minute Run Queue Avg	The average number of threads over the last one minute, other than threads waiting in the execution queue for the kernel. On HP-UX, Solaris, and AIX, this value does not include the number of threads waiting for I/O. On Linux, this value includes the number of threads waiting for I/O.
Agent Instance#	Name of the host on which PFM - Agent for Platform is operating.
CPU %	CPU usage (%). This is also the average value of the percentage for each processor (for AIX, it includes processors after logical partitioning by the Micro-Partitioning function).
Context Switches/sec	Frequency of context switches (number of times per second).
Free Swap %	Percentage of available swap area size (megabytes) (%).
Page Scans/sec	Frequency of page scans (number of times per second).
Processes	Number of processes in the system.
System Calls/sec	Frequency of system calls (number of times per second).
Users	Number of real users.

#

This field is added only when data is recorded in the Store database. For details about adding fields only when data is recorded in the Store database, see *9. Records*.

9 Records

This chapter explains PFM - Agent for Platform records. For details about how to collect performance data for each record, see the chapter that explains the functions of Performance Management in the *JP1/Performance Management Planning and Configuration Guide*, or the chapter that explains the management of operation monitoring data in the *JP1/Performance Management User's Guide*.

Data model

The records and fields that each PFM - Agent has are summarily referred to as a *data model*. Each PFM - Agent and the data model it has are assigned unique version numbers. For details about data model versions of PFM - Agent for Platform, see *I. Version Compatibility*.

To check the data model version of each PFM - Agent, use the Agents window in PFM - Web Console to display the agent properties.

For details about data models, see the chapter explaining the functions of Performance Management in the *JP1/Performance Management Planning and Configuration Guide*.

Format of record explanations

This chapter describes PFM - Agent for Platform records in alphabetical order. The explanation of each record consists of the following items:

Function

Provides an overview and notes on the performance data stored in each record.

Default values and values that can be specified

The default values of the performance data collection conditions that are set in each record and the values that the user can specify in place of the default values are shown in a table. The table below explains the meaning of the items described in *Default values and values that can be specified*. For details about the items shown in this table, see the chapter explaining management of operation monitoring data in the *JP1/Performance Management User's Guide*.

Table 9–1: Default values and changeable values

Item	Meaning	Modifiable
`Collection Interval`	Performance data collection interval (seconds).	Yes: Can be modified No: Cannot be modified
`Collection Offset`#1	Offset value for starting the collection of performance data (seconds). For details about the offset value, see the chapter explaining management of operation monitoring data in the *JP1/Performance Management User's Guide*. For details about the start time of performance data collection, see the chapter explaining the functions of Performance Management in the *JP1/Performance Management Planning and Configuration Guide*.	
`Log`	Indicates whether to register the collected performance data in the Store database: Yes: Recorded. However, the data is not recorded if `Collection Interval=0`. No: Not recorded.	
`LOGIF`	Condition for determining whether to register the collected performance data in the Store database.	
`Over 10 Sec Collection Time`#2	Indicates whether collecting records takes 10 seconds or more, depending on the system configuration. `Yes`: It sometimes takes 10 seconds or more. `No`: It takes less than 10 seconds.	
`Sync Collection With`#3	Indicates whether to collect performance data synchronously with the record displayed in the `Description` record property. For details, see the chapter explaining management of operation monitoring data in the *JP1/Performance Management User's Guide*.	
`Realtime Report Data Collection Mode`#2	Specifies the real-time report display mode. `Reschedule`: Reschedule mode `Temporary Log`: Temporary log mode Note that you must specify the temporary log mode (`Temporary Log`) for records for which `Over 10 Sec Collection Time` is set to `Yes`.	

#1

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing load. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the value of Collection Offset, specify a value after taking the collection load into consideration.

#2

This item is displayed when the history collection priority function is enabled.

#3

If `Sync Collection With` is displayed, neither `Collection Interval` nor `Collection Offset` is displayed.

ODBC key fields

Indicates the primary key required for PFM - Manager or PFM - Base to use the record data stored in the Store database. Some ODBC key fields are common to all records while others are unique to some records. Only those ODBC key fields that are unique to each record are shown for each record. Only multi-instance records have unique ODBC key fields.

For details about the ODBC key fields common to all records, see *List of ODBC key fields*.

Lifetime

Indicates the duration over which the consistency of the performance data collected into each record is guaranteed. For details about lifetime, see the chapter explaining the functions of Performance Management in the *JP1/Performance Management Planning and Configuration Guide*.

Record size

Indicates the size of performance data that is stored in each record in a single collection operation.

Fields

The fields in each record are explained in a table. The individual items in a table are explained below.

- PFM - View name (PFM - Manager name)
  - PFM - View name

    Indicates the field name (PFM - View name) that is displayed in PFM - Web Console.
  - PFM - Manager name

    Indicates the field name (PFM - Manager name) described in an SQL statement when PFM - Manager uses an SQL statement to utilize the field data stored in the Store database.

    In an SQL statement, the Manager name is described with the record ID of each record added to the front. For example, for the Flags (`FLAGS`) field of the Process Detail (`PD`) record, `PD_FLAGS` is used.
- Explanation

  Describes the performance data stored in each field.
- Summary

  Indicates the method (summarization rules) to be used by Agent Store when summarizing data. For details about summarization rules, see *Summarization rules*.
- Format

  Indicates the data type of the values in each field, such as `char` or `float`. For details about data types, see *List of data types*.
- Delta

  Data that is expressed as a changed amount relative to data collected as an accumulative value is called a *delta*. For details about delta, see *Field values*.

- Not supported in

  Indicates a platform or platform version in which the field is not supported:

  - Two hyphens (--) indicate that the field can be used in all platforms that are supported by PFM - Agent for Platform.

- Data source

  Indicates the calculation method or collection source for the values in the applicable field. For details about field values, see *Field values*.

List of ODBC key fields

Some ODBC key fields are common to all records while others are unique to some records. This section shows the ODBC key fields that are common to all records. For PFM - Manager to use the record data stored in the Store database, ODBC key fields are required.

The table below shows a list of ODBC key fields common to all records. For details about the ODBC key fields unique to each record, see the explanation of each record.

Table 9–2: List of ODBC key fields common to all records

ODBC key fields	ODBC format	Data	Explanation
*record-id*_DATE	SQL_INTEGER	Internal	Record key indicating the date on which the record was created
*record-id*_DATETIME	SQL_INTEGER	Internal	Combination of the *record-id*_DATE field and the *record-id*_TIME field
*record-id*_DEVICEID	SQL_VARCHAR	Internal	Name of the host on which PFM - Agent is running
*record-id*_DRAWER_TYPE	SQL_VARCHAR	Internal	Category. The following values are valid: m: Minute H: Hour D: Day W: Week M: Month Y: Year
*record-id*_PROD_INST	SQL_VARCHAR	Internal	Name of the host on which PFM - Agent is running
*record-id*_PRODID	SQL_VARCHAR	Internal	PFM - Agent's product ID
*record-id*_RECORD_TYPE	SQL_VARCHAR	Internal	Record type identifier (4 bytes)
*record-id*_TIME	SQL_INTEGER	Internal	Time at which the record was created (Greenwich Mean Time)

Summarization rules

For records of the `PI` record type, two types of data are stored in the Store database: the data collected at the interval set in `Collection Interval`, and the data summarized for a specific period of time (minute, hour, day, week, month, or year) according to a predefined rule. The type of summarization is defined for each field. This definition is called a *summarization rule*.

Depending on the summarization rule, intermediate data used during the summarization period must sometimes be retained. In this case, a field for holding the intermediate data is added to a record in the Store database. This field is called an *added field*.

Part of an added field is displayed as a record field in PFM - Web Console. The added fields displayed on PFM - Web Console can be used as the fields to be displayed in a historical report.

To differentiate them from the fields that are added through summarization, the fields described in the explanation of each record in this chapter are called *fixed fields*.

The following field names are used for added fields:

- Added field name stored in the Store database
  A field name consisting of the PFM - Manager name in the fixed field and an added suffix
- Added field name displayed on PFM - Web Console
  A field name consisting of the PFM - View name in the fixed field and an added suffix

The table below shows the suffixes to be added to the PFM - Manager name, the corresponding suffixes to be added to the PFM - View name, and the data stored in the fields.

Table 9–3: List of suffixes for added fields

PFM - Manager name	PFM - View name	Stored data
`_TOTAL`	(Total)	Total field value of the records within the summarization period
`_TOTAL_SEC`	(Total)	Total field value of the records within the summarization period (for a `utime` field)
`_COUNT`	--	Number of records collected within the summarization period
`_HI`	(Max)	Maximum field value of the records within the summarization period
`_LO`	(Min)	Minimum field value of the records within the summarization period

Legend:
--: No added field

The table below shows a list of summarization rules.

Table 9–4: List of summarization rules

Summarization rule name	Summarization rule
`COPY`	Stores the field value of the latest record within the summarization period without any modification.
`AVG`	Stores the average field value within the summarization period. The computation formula follows: `(total-field-value)/(number-of-collected-records)`

Summarization rule name	Summarization rule
AVG	Added fields (Store database) • _TOTAL • _TOTAL_SEC (for a utime field) • _COUNT Added field (PFM - Web Console)#1, #2 • (Total)
%	Stores the average field value within the summarization period. Applied primarily to percentage fields. The computation formula follows: (*total-field-value*)/(*number-of-collected-records*) Added fields (Store database) • _TOTAL • _TOTAL_SEC (for a utime field) • _COUNT
R	Stores the average field value within the summarization period. Applied primarily to percentage fields that indicate the amount per second. The computation formula follows: (*total-field-value*)/(*number-of-collected-records*) When delta is specified for a real-time report, the special computation method that divides the difference by an interval is used. Added fields (Store database) • _TOTAL • _TOTAL_SEC (for a utime field) • _COUNT Added field (PFM - Web Console)#1, #2 • (Total)
--	No summarization

#1

A utime field containing _AVG in the PFM - Manager name cannot use a (Total) field added to PFM - Web Console in a historical report.

#2

A field containing any of the following character strings in the PFM - Manager name cannot use a (Total) field added to PFM - Web Console in a historical report:

PER, PCT, PERCENT, _AVG, and _RATE_TOTAL

List of data types

The table below lists the data types for various field values along with the corresponding C and C++ data types. The field values shown under *Data type* in this table are shown in the *Format* column in the field table for each record.

Table 9–5: List of data types

Data type		Bytes	Explanation
Field	C and C++		
`char`(*n*)	`char()`	Number inside ( )	Character string having a length of *n* bytes
`double`	`double`	8	Numerical value (1.7E±308 (15 digits))
`float`	`float`	4	Numerical value (3.4E±38 (7 digits))
`long`	`long`	4	Numerical value (-2,147,483,648 to 2,147,483,647)
`short`	`short`	2	Numerical value (-32,768 to 32,767)
`string`(*n*)	`char[]`	Number inside ( )	Character string having *n*-byte length. The last character is `null`.
`time_t`	`unsigned long`	4	Numerical value (0 to 4,294,967,295)
`timeval`	`struct`	8	Numerical value (the first 4 bytes indicate seconds, and the next 4 bytes indicate microseconds.)
`ulong`	`unsigned long`	4	Numerical value (0 to 4,294,967,295)
`utime`	`struct`	8	Numerical value (the first 4 bytes indicate seconds, and the next 4 bytes indicate microseconds.)
`word`	`unsigned short`	2	Numerical value (0 to 65,535)
(Not applicable)	`unsigned char`	1	Numerical value (0 to 255)

Note on data type

PFM - Agent for Platform cannot store a value that exceeds the maximum value defined for a data type in the data model. If a value exceeding the maximum value is collected, PFM - Agent for Platform does not retain the correct value.

Field values

This section explains the values stored in individual fields.

Data source

Each field stores a value acquired from Performance Management or monitoring target program, or a value computed from such a value based on a certain computation formula. The acquisition destination or computation method for each field value is shown under the *Data source* column.

If performance data acquired from UNIX is processed and set as a field value, the character string in PFM - Agent for Platform's *Data source* column shows the computation method for the value that is set in the field. For example, for the value of the Avg Wait Time (`AVG_WAIT_TIME`) field of the Device Detail (`PI_DEVD`) record, the value obtained by dividing the value of the Total Wait Time (`TOTAL_WAIT_TIME`) field by the value of the Total I/O Ops (`TOTAL_OPS`) field is stored.

Two hyphens (--) indicate that performance data is not processed and set as a field value.

Delta

Data expressed as an amount of change is called a *delta*. For example, if the performance data collected during the first interval is `3` and that collected during the second interval is `4`, `1` is stored as the amount of change. Whether each field value is a delta is indicated under the *Delta* column in the field table.

The performance data collected by PFM - Agent for Platform varies as shown in the table below.

Real-time reports for which **Indicate delta value** is selected#, historical reports, and alarms

Record type	Delta	Data source	Is there any field that has delta = Yes in its data source column?	Value displayed or value evaluated through alarm monitoring
`PI` record type	Yes	No	--	Changed amount
		Yes	No	Changed amount in the result computed based on the value at the time of collection
			Yes	Result computed based on the changed amount
	No	No	--	Value at the time of collection
		Yes	No	Result computed based on the value at the time of collection
			Yes	Result computed based on the changed amount
`PD` record type	Yes	No	--	Changed amount. Accumulated value for a historical report or alarm of the Process Detail (`PD`) record
		Yes	No	--
			Yes	Result computed based on the changed amount. Result computed based on the accumulated value for a historical report or alarm of the Process Detail (`PD`) record
	No	No	--	Value at the time of collection
		Yes	No	Result computed based on the value at the time of collection

Record type	Delta	Data source	Is there any field that has delta = Yes in its data source column?	Value displayed or value evaluated through alarm monitoring
PD record type	No	Yes	Yes	Result computed based on the changed amount. Result computed based on the accumulated value for the Process Detail (PD) record
PL record type	No	No	--	Differing-components information for each collection interval

Legend:

--: Not applicable

#

Indicates whether the following items in PFM - Web Console are selected:

- **Indicate delta value** in the Indication settings (Realtime) window, which opens when **Edit** is chosen in the Report wizard
- **Indicate delta value** under **Indication settings (Realtime)** under the **Properties** tab in the Report window

Real-time reports for which **Indicate delta value** is not selected#

Record type	Delta	Data source	Is there any field that has delta = Yes in its data source column?	Displayed value
PI record type	Yes	No	--	Accumulated value
		Yes	No	Result computed based on the value at the time of collection
			Yes	Result computed based on the accumulated value
	No	No	--	Value at the time of collection
		Yes	No	Result computed based on the value at the time of collection
			Yes	Result computed based on the accumulated value
PD record type	Yes	No	--	Accumulated value
		Yes	No	--
			Yes	Result computed based on the accumulated value
	No	No	--	Value at the time of collection
		Yes	No	Result computed based on the value at the time of collection
			Yes	Result computed based on the accumulated value
PL record type	No	No	--	Accumulated information since the service startup

Legend:

--: Not applicable

#

The following items in the dialog box in PFM - Web Console show whether **Indicate delta value** is selected:

- **Indicate delta value** in the Indication settings (Realtime) window, which opens when **Edit** is chosen in the Report wizard
- **Indicate delta value** under **Indication settings (Realtime)** under the **Properties** tab in the Report window

Example:

The `Idle %` field of the CPU - Per Processor Detail (`PI_CPUP`) record contains a `PI`-type record, and `delta = No` is set in this field. However, `delta = Yes` is set for the field indicated as the data source. Therefore, for real-time reports for which **Indicate delta value** is selected, historical reports, and alarms, a result computed based on the changed amount is displayed.

Note the following points about performance data collection:

- For a `PI`-type record to be saved, performance data must be collected at least twice.

  In each collection interval specified in PFM - Web Console, performance data is collected into a `PI`-type record. However, storing of performance data in the Store database is not executed at the point in time when performance data collection is specified in PFM - Web Console.
- The historical data of a `PI`-type record contains data that requires the difference from the previously collected data (for example, a delta value), and thus data from two collection intervals is required. Therefore, it may take as long as twice the specified time before the first historical data is stored in the Store database.

  For example, if the performance data collection interval is set to 300 seconds (5 minutes) at 18:32 in PFM - Web Console, the first data collection begins at 18:35. The next data collection begins at 18:40. Then, historical data is created based on the data collected at 18:35 and 18:40, and is stored in the Store database at 18:40.
- If **Indicate delta value** is selected for a real-time report, a value is displayed when the first data is collected. However, if the report needs the previous data, the first value is displayed as `0`. When the second or subsequent data is collected, the collected data value is displayed.
- If the OS counter value wraps around because, for example, the system has been operating for a long time, the value of the Delta field might be extremely large.

Averages and percentages in individual fields

Some averages and percentages in individual fields are determined as values on a second-by-second basis, while others are determined according to the data collection interval. Unless otherwise specified, the values determined according to the data collection interval are used (if the interval is 60 seconds, data is collected every minute).

Interval field values

Interval field values vary according to the record type as shown in the table below.

Table 9–6: Interval field values

Record type	Explanation
`PI` record type	• Real-time reports If **Indicate delta value** is selected for a real-time report, the first value is `0`. Thereafter, the value is the same as the refresh interval that is set in the report. The value is computed using the following formula: *interval-field-value = record-time-field-value - record-time-field-value-obtained-during-previous-collection-interval* If **Indicate delta value** is not selected for a real-time report, the value obtained by subtracting the system startup time from the current time is used. The value is computed using the following formula: *interval-field-value = record-time-field-value - system-startup-time* • Historical reports The same value as the Collection Interval value. Computed using the following formula: *interval-field-value = record-time-field-value - record-time-field-value-obtained-during-previous-collection-interval*
`PD` record type	Always `0`.
`PL` record type	Always `0`.

Fields added only when data is stored in the Store database

The table below shows the fields that are added only when data is stored in the Store database.

Table 9–7: Fields that are added only when a record is recorded in the Store database

PFM - View name (PFM - Manager name)	Explanation	Format	Delta	Supported version	Data source
Agent Host (DEVICEID)	Name of the host on which PFM - Agent is running	string(256)	No	All	--
Agent Instance (PROD_INST)	Name of the host on which PFM - Agent is running	string(256)	No	All	--
Agent Type (PRODID)	Product ID of PFM - Agent indicated as a 1-byte identifier	char	No	All	--
Date (DATE)	Date on which the record was created (Greenwich Mean Time)#1, #2	char (3)	No	All	--
Date and Time (DATETIME)	Combination of the Date (DATE) field and the Time (TIME) field#2	char (6)	No	All	--
Drawer Type (DRAWER_TYPE)	For records of the PI record type, the data summarization category is indicated.	char	No	All	--
GMT Offset (GMT_ADJUST)	Difference between the Greenwich Mean Time and the local time, in seconds	long	No	All	--
Time (TIME)	Time at which the record was created (Greenwich Mean Time)#1, #2	char (3)	No	All	--

Legend:

--: Indicates that the performance data acquired from UNIX is not processed and set as a field.

#1

In records of the PI record type, data is summarized, and therefore the time used as the reference point for summarization is set. The table below shows the setting for each record category.

Table 9–8: Setting for each record category

Category	Setting for each record category
Minute	0 seconds for the time at which the record is created
Hour	0 minutes and 0 seconds for the time at which the record is created
Day	0 hours, 0 minutes, and 0 seconds for the day on which the record is created
Week	0 hours, 0 minutes, and 0 seconds for the week in which the record is created
Month	0 hours, 0 minutes, and 0 seconds for the first day of the month in which the record is created
Year	0 hours, 0 minutes, and 0 seconds of January 1 of the year in which the record is created

#2

When data is displayed in a report, the Date field uses the *YYYYMMDD* format, the Date and Time field uses the *YYYYMMDD hh:mm:ss* format, and the Time field uses the *hh:mm:ss* format.

Fields that are output when data stored in the Store database is exported

When data stored in the Store database is exported by using the `jpctool db dump` command, the fields listed below are output. Although these fields are also added when a record is stored in the Store database, they are not displayed in PFM - Web Console and therefore cannot be used as fields displayed in a report. Do not use these fields for operations because they are used internally by PFM - Agent.

- *record-ID*_DATE_F
- *record-ID*_DEVICEID_F
- *record-ID*_DRAWER_TYPE_F
- *record-ID*_DRAWER_COUNT
- *record-ID*_DRAWER_COUNT_F
- *record-ID*_INST_SEQ
- *record-ID*_PRODID_F
- *record-ID*_PROD_INST_F
- *record-ID*_RECORD_TYPE
- *record-ID*_RECORD_TYPE_F
- *record-ID*_SEVERITY
- *record-ID*_SEVERITY_F
- *record-ID*_TIME_F
- *record-ID*_UOWID
- *record-ID*_UOWID_F
- *record-ID*_UOW_INST
- *record-ID*_UOW_INST_F
- *record-ID*_*PFM - Manager-name*_SEC
- *record-ID*_*PFM - Manager-name*_MSEC

Notes on records

Note the following points when collecting records.

Notes on performance information when modifying system resources

Notes are provided below for modifying system resources using the DLPAR functionality of AIX, the vPars functionality of HP-UX, the DR functionality of Solaris, and the CPU hotplug functionality of Linux.

- Regardless of whether a PFM - Agent for Platform service is running or stopped, if the system resources are modified, there is no continuity between the performance data before and after the modification. Therefore, the performance data before and after the modification must be treated as separate.

  If necessary, back up the Store database before modifying the system resources, then clear the Store database and begin collecting new performance data. For details about how to back up the Store database, see the chapter explaining backup and restoration in the *JP1/Performance Management User's Guide*.

- With respect to the DR functionality of Solaris or the CPU hotplug functionality of Linux, the performance data shown in the following table will not be collected correctly if system resources are modified while a PFM - Agent for Platform service is running. Therefore, before modifying the system resources, stop the PFM - Agent for Platform service. After the modification, restart the PFM - Agent for Platform service. For details about how to start and stop services, see the chapter explaining how to start and stop Performance Management in the *JP1/Performance Management User's Guide*.

  The table below shows the types of system resources that are affected if they are modified while a PFM - Agent for Platform service is running, and the actions that you can take to correctly display reports after the modification.

  Table 9–9: System resource types and actions to be taken to correctly display reports after modification

<table>
<tr><th>Record</th><th>OS and name of the function used to modify system resources</th><th>System resource type</th><th>Actions to be taken after modification</th></tr>
<tr><td>CPU - Per Processor Detail (PI_CPUP)#1</td><td>• DLPAR of AIX
• vPars of HP-UX
• DR of Solaris
• CPU hotplug of Linux</td><td>CPU</td><td rowspan="8">• For a real-time report:
Display the report again.
• For a historical report:
Specify the display period so that it does not include the date and time when the system resources were modified.</td></tr>
<tr><td>Device Detail (PI_DEVD)#1</td><td rowspan="4">DR of Solaris</td><td rowspan="2">Device</td></tr>
<tr><td>Device Summary (PI_DEVS)#1</td></tr>
<tr><td>Network Interface Detail (PI_NIND)#1</td><td rowspan="2">LAN board</td></tr>
<tr><td>Network Interface Summary (PI_NINS)#1</td></tr>
<tr><td rowspan="4">System Summary Overview (PI)#2</td><td>DLPAR of AIX</td><td>• CPU
• Memory</td></tr>
<tr><td>vPars of HP-UX</td><td>CPU</td></tr>
<tr><td>DR of Solaris</td><td>• CPU
• Memory</td></tr>
<tr><td>CPU hotplug of Linux</td><td>CPU</td><td></td></tr>
</table>

#1

Fields other than those listed below are affected by a system resource modification:

- Interval (INTERVAL)
- Record Time (RECORD_TIME)
- Record Type (INPUT_RECORD_TYPE)
- Fields that are added only when data is recorded in the Store database

#2

The table below shows the fields that are affected by a system resource modification.

Table 9–10: Fields that are affected by a system resource modification

System resource type	Target field
CPU	When modifying system resources using the DLPAR functionality of AIX, the vPars functionality of HP-UX, and the DR functionality of Solaris: • Active CPUs (NUMBER_OF_ACTIVE_CPUS) • CPU % (KERNELMODE_USERMODE_PERCENT) • Context Switches (CONTEXT_SWITCHES) • Context Switches/sec (CONTEXT_SWITCHES_PER_SECOND) • Idle % (IDLE_TIME_PERCENT) • Interrupts (INTERRUPTS) • Interrupts/sec (INTERRUPTS_PER_SECOND) • Kernel CPU % (KERNELMODE_PERCENT) • System Calls (SYSTEM_CALLS) • System Calls/sec (SYSTEM_CALLS_PER_SECOND) • Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME) • Total Idle Time (TOTAL_IDLE_TIME) • Total User-Mode Time (TOTAL_USERMODE_TIME) • Total Wait Time (TOTAL_WAIT_TIME) • Traps (TRAPS) • Traps/sec (TRAPS_PER_SECOND) • User CPU % (USERMODE_PERCENT) • Wait % (WAIT_TIME_PERCENT) When modifying system resources using the CPU hotplug functionality of Linux • Active CPUs (NUMBER_OF_ACTIVE_CPUS) • CPU % (KERNELMODE_USERMODE_PERCENT) • Idle % (IDLE_TIME_PERCENT) • Kernel CPU % (KERNELMODE_PERCENT) • Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME) • Total Idle Time (TOTAL_IDLE_TIME) • Total User-Mode Time (TOTAL_USERMODE_TIME) • Total Wait Time (TOTAL_WAIT_TIME) • User CPU % (USERMODE_PERCENT) • Wait % (WAIT_TIME_PERCENT)
Memory	• Alloc Mem % (ALLOCATED_MEMORY_PERCENT) • Alloc Mem Mbytes (ALLOCATED_MEMORY_MBYTES) • Effective Free Mem % (EFFECTIVE_FREE_MEM_PERCENT) • Effective Free Mem Mbytes (EFFECTIVE_FREE_MEM_MBYTES) • Effective Used Mem % (EFFECTIVE_USED_MEM_PERCENT) • Effective Used Mem Mbytes (EFFECTIVE_USED_MEM_MBYTES) • Free Mem % (FREE_MEMORY_PERCENT)

System resource type	Target field
Memory	• Free Mem Mbytes (FREE_MEMORY_MBYTES) • Total Physical Mem Mbytes (TOTAL_MEMORY_MBYTES)

Performance information about resources that are logically partitioned

When the Micro-Partitioning function is used in AIX, a record is created for each logically partitioned processor. Therefore, even when resources are part of the same physical resource, their performance is not affected by other logical resources. Note that for a single-instance record of the PI record type, the combined total value of all resources including logical resources is displayed.

The table below shows the records and fields that are affected by the Micro-Partitioning function.

Table 9–11: Records and fields that are affected by the Micro-Partitioning function

Record name	Field name
CPU - Per Processor Detail (PI_CPUP)	All
Application Process Interval (PD_APSI)	CPU % (CPU_PERCENT_USED)
Application Process Overview (PD_APS)	
Process Detail (PD)	
Process Detail Interval (PD_PDI)	
Program Summary (PD_PGM)	
System Summary Overview (PI)	Active CPUs (NUMBER_OF_ACTIVE_CPUS)
	CPU % (KERNELMODE_USERMODE_PERCENT)
	Idle % (IDLE_TIME_PERCENT)
	Kernel CPU % (KERNELMODE_PERCENT)
	Total Idle Time (TOTAL_IDLE_TIME)
	Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME)
	Total User-Mode Time (TOTAL_USERMODE_TIME)
	User CPU % (USERMODE_PERCENT)
	Wait % (WAIT_TIME_PERCENT)
Terminal Summary (PD_TERM)	CPU % (CPU_PERCENT_USED)
User Summary (PD_USER)	
Workgroup Summary (PI_WGRP)	

Record generation results when data cannot be collected

The following explains the record generation results when data to be stored in fields cannot be collected:

Records cannot be generated

When PFM - Agent for Platform cannot collect performance data to be stored in the fields defined as ODBC key fields, no records are generated.

Information on terminated processes

Functions for collecting information on terminated processes cannot be used. Therefore, the performance data for the fields shown below cannot be collected.

Table 9–12: Fields for which performance data cannot be collected

Record name	Field name
Process Detail (PD)	Accounting Flags (ACCOUNTING_FLAGS)
	End Time (END_TIME)
	Exit Status (EXIT_STATUS)
	Mem Charge (MEMORY_CHARGE)
Process Detail Interval (PD_PDI)	Accounting Flags (ACCOUNTING_FLAGS)
	End Time (END_TIME)
	Exit Status (EXIT_STATUS)

Notes on reserved records

Under **Record** in the New Report > Field window of the PFM - Web Console report wizard, reserved records are displayed, and you can create a report of these records. You cannot, however, display a report of reserved records.

In addition, if PFM - Agent for Platform 07-00 or earlier is upgraded to 08-00 or later, records reserved in 08-00 or later appear during history collection setup. These records can be modified. However, if they are modified and then specified for data collection, no records will be collected. The following message is issued to the common message log during the first collection attempt only:

KAVF10600-W An invalid collection event occurred. (record=*record-id*, rc=*maintenance-code*)

Note

When you perform a new installation of PFM - Agent for Platform 08-00 or later, the properties of the reserved records are not displayed.

Note on specifying a process name in an alarm condition

When you specify a process name in an alarm condition for the PD_APP or PI_WGRP record, only the string length that is within the range of data that can be collected for the PD record is used to determine the alarm condition. Accordingly, when the process name exceeds the data range as shown in the example below, and you specify *xyz in the alarm condition, the expected condition judgment cannot be performed, since PFM - Agent for Platform does not hold the xyz part. In this case, specify an alarm condition within the range of data that can be collected for the PD record by, for example, specifying abc*.

Example:

```
abc... xyz (the xyz part is outside the range that can be collected for th
e PD record)
```

Notes on cases in which record instances cannot be uniquely identified

For the PD_UPD, PD_UPDB, PI_UPI, PI_UPIB, and PI_XUI1 to PI_XUI5 records, the first instance of each record is used.

Notes about records

- When a value is not specified in the User Time 1 field of PD_UPD, PD_UPDB, PI_UPI, PI_UPIB, PI_XUI1, PI_XUI2, PI_XUI3, PI_XUI4, and PI_XUI5 record, it is displayed as "1970 01 01 09:00:00." in PFM - Web Console.
- Although PFM - Web Console displays values after the decimal point for the User Long *N*[#] field and User Long Roll *N*[#] fields of PD_UPD, PD_UPDB, PI_UPI, and PI_UPIB records, only integer values can be specified.

  #

  N is a user-chosen numerical value.

Other note

- Language environments other than Japanese, English, and Chinese support only the ASCII character set. In such environments, non-ASCII characters in performance data are not displayed correctly.

List of records

The table below shows the records that can be collected by PFM - Agent for Platform and the information stored in these records by category.

Table 9–13: PFM - Agent for Platform record list (by category)

Category	Record name	Record ID	Information stored
System Memory Processor Protocol NFS	System Summary Overview	PI	Performance data per unit time on the entire system
Disk	File System Detail - Local	PD_FSL	Performance data showing the status of the local file system capacity at a given point in time
	File System Detail - Remote	PD_FSR	Performance data showing the status of the remote file system capacity at a given point in time
	Device Detail	PI_DEVD	Performance data per unit time on the local disk device usage
	Device Summary	PI_DEVS	Performance data in which the data stored in the Device Detail (PI_DEVD) record is summarized per unit time
Network	Network Interface Detail	PI_NIND	Performance data per unit time on a network interface
	Network Interface Summary	PI_NINS	Performance data in which the data stored in the Network Interface Detail (PI_NIND) record is summarized per unit time
Process	Process Detail	PD	Performance data showing the status of a process at a given point in time
	Process Detail Interval	PD_PDI	Performance data per unit time on a process
	Process Summary	PD_PDS	Performance data in which the data stored in the Process Detail (PD) record is summarized as a state at a given point in time
	Program Summary	PD_PGM	Performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each program at a given point in time
	Terminal Summary	PD_TERM	Performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each terminal at a given point in time
	User Summary	PD_USER	Performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each user at a given point in time
	Workgroup Summary	PI_WGRP	Performance data per unit time on the processor's arithmetic operations, logical computation, initialization of peripheral device operations, the rate of interrupts such as process thread execution, and the time spent.
	Application Process Interval	PD_APSI	Performance data that shows the state at a given point in time of a process that has been set for process monitoring
	Application Process Overview	PD_APS	Performance data that shows the state of a process at a given point in time

Category	Record name	Record ID	Information stored
Processor	CPU - Per Processor Detail	PI_CPUP	Performance data per unit time on the processor usage
Message	Logged Messages	PL_MESS	Messages that are written into the log file specified by the Messages File property or the event file. This record cannot be used in Linux.
User-defined record	Application Summary	PD_APP	Performance data that summarizes, by application, the records stored in the Process Detail (PD) record at a given point in time
	User Data Detail	PD_UPD	User-specific performance data that indicates the status at a given point in time
	User Data Detail - Extended	PD_UPDB	
	User Data Interval	PI_UPI	User-specific performance data per unit time
	User Data Interval - Extended	PI_UPIB	
	User Data Interval - Expanded 1	PI_XUI1	
	User Data Interval - Expanded 2	PI_XUI2	
	User Data Interval - Expanded 3	PI_XUI3	
	User Data Interval - Expanded 4	PI_XUI4	
	User Data Interval - Expanded 5	PI_XUI5	
	Application Summary Extension	PD_APP2	Performance data that summarizes by application the records stored in the Application Process Overview (PD_APS) records at a given point in time
	Application Process Detail	PD_APPD	Performance data that summarizes by process being monitored on a per-application basis the records stored in the Application Process Overview (PD_APS) records at a given point in time
Reserved records	IPC Summary	PD_IPCS	These records are reserved and cannot be used.
	Message Queue Detail	PD_MSQD	
	NFS Client Detail	PI_NCD	
	NFS Client Overview	PI_NCO	
	NFS Server Detail	PI_NSD	
	NFS Server Overview	PI_NSO	
	Quotas	PD_UFSQ	
	Semaphore Detail	PD_SEMD	
	Shared Memory Detail	PD_SHMD	
	Tape Device Summary	PI_TAPS	
	User File System Storage	PD_UFSS	

Application Process Detail (PD_APPD)

Function

The Application Process Detail (PD_APPD) record stores performance data that summarizes by process being monitored on a per-application basis the state at a given point in time of the records stored in the Application Process Overview (PD_APS) records. One record is stored for each monitoring condition in the application. This record is a multi-instance record.

Note:

To change the application definition, you must change the settings from PFM - Web Console.

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, APP2	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key field

- PD_APPD_APPLICATION_NAME
- PD_APPD_MONITORING_NUMBER

Lifetime

From the time the application definition is added in PFM - Web Console until it is deleted.

Record size

- Fixed portion: 681 bytes
- Variable portion: 263 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Name (APPLICATION_NAME)	The name specified in the process monitoring settings	--	string(64)	No	--	--
Interval (INTERVAL)	Number of seconds in the interval during which Application Process Detail (PD_APPD) records were stored. Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Monitoring Condition (MONITORING_CONDITION)	Condition expression for identifying the processes to be monitored	--	string(128)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Monitoring Count (MONITORING_COUNT)	Number of running processes or services that match the monitoring condition.	--	ulong	No	--	--
Monitoring Field (MONITORING_FIELD)	Field to be monitored	--	string(16)	No	--	--
Monitoring Label (MONITORING_LABEL)	Name used to identify the monitoring condition	--	string(32)	No	--	--
Monitoring Max (MONITORING_MAX)	Upper limit on the monitoring count	--	ulong	No	--	--
Monitoring Min (MONITORING_MIN)	Lower limit on the monitoring count	--	ulong	No	--	--
Monitoring Number (MONITORING_NUMBER)	The monitoring condition number	--	word	No	--	--
Monitoring Status (MONITORING_STATUS)	Result of the monitoring count condition **NORMAL**: No error exists. **ABNORMAL**: An error exists.	--	string(9)	No	--	--
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich mean time)	--	time_t	No	--	--
Record Type (INPUT_RECORD_TYPE)	Type of record. Always `APPD`.	--	char(8)	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Application Process Interval (PD_APSI)

Function

The Application Process Interval (PD_APSI) record stores performance data that shows the state at a given point in time of a process that has been set for process monitoring. This record is a multi-instance record.

Notes:

- Each process represents a program that is being executed during data collection.
- If the process does not have a terminal name, ?? is displayed in the Terminal (TERMINAL_NAME) field.
- In AIX and Solaris, only data for the fields shown below is acquired for <defunct> processes (0 or n/a is displayed for all other fields):

  Application Name, Group ID, Monitoring Field, Monitoring Label, Monitoring Number, Parent PID, PID, Process Group ID, Program Name, Real Group ID, Real User ID, Record Time, Record Type, Snapshot Time, Start Time, State, Terminal, User ID, Virtual Env ID (Solaris or AIX)
- Any character in the source information that is not in the ASCII character set range of 0x20 to 0x7E will be converted to a hash mark (#: 0x23) before it is stored in the Program Name (PROGRAM_NAME) field. Note that multi-byte characters are processed in single-byte units during conversion. For example, the multi-byte (full-width) letter A is converted as follows:

Source information		Post-conversion information	
Character encoding	Binary	Binary	Character string
Shift-JIS	8260	2360	#`
EUC	A3C1	2323	##
UTF-8	EFBCA1	232323	###

- If the target process of an instance is a zombie process, the following are displayed for the Group ID, Real Group ID, and User ID fields in AIX and HP-UX:

OS	Field	Content displayed
AIX	Group ID	0
	Real Group ID	
HP-UX	Group ID	-3
	Real Group ID	
	User ID	

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, APP2	No

Item	Default value	Modifiable
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key field

- PD_APSI_START_TIME
- PD_APSI_PID
- PD_APSI_APPLICATION_NAME
- PD_APSI_MONITORING_NUMBER

Lifetime

From the time of execution of the process for which the application definition was set until the process terminates or the application definition is deleted.

Record size

- Fixed portion: 681 bytes
- Variable portion: 746 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Name (APPLICATION_NAME)	The name specified in the process monitoring settings	--	string(64)	No	--	--
Avg I/O Kbytes (AVG_IO_KBYTES)	Average transfer size in I/O operations (kilobytes)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ TOTAL_IO_OPS
CPU % (CPU_PERCENT_USED)	Average obtained by dividing he process's CPU usage by the processor count (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) ÷ processors-count) × 100
CPU % Each (CPU_PERCENT_EACH)	CPU usage of the process (%)	--	float	No	--	((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) × 100
CPU Limit (CPU_LIMIT)	CPU limit for the process. This value is defined by the RLIMIT_CPU parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Context Switches (CONTEXT_SWITCHES)	Number of context switch operations	--	ulong	Yes	HP-UX, Linux	--
Core Size Limit (CORE_SIZE_LIMIT)	Maximum size of a core file that can be created by the process (bytes). This value is defined by	--	double	No	HP-UX, Solaris, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Core Size Limit (CORE_SIZE_LIMIT)	the RLIMIT_CORE parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Data Size Limit (DATA_SIZE_LIMIT)	Process data size limit (bytes). This value is defined by the RLIMIT_DATA parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Elapsed Time (ELAPSED_TIME)	Time elapsed since the process started	--	utime	No	--	- If process is running, RECORD_TIME - START_TIME - If process is not running, --
Executable Data Kbytes (EXECUTABLE_DATA_KBYTES)	Size of the data being used (kilobytes). For Solaris to collect this field, the SAUNIXPMAP environment variable must have been set in advance. Execute the following commands from the command line: # SAUNIXPMAP=1 # export SAUNIXPMAP	--	ulong	No	Linux	--
Executable Text Kbytes (EXECUTABLE_TEXT_KBYTES)	Size of the text being used (kilobytes). For Solaris to collect this field, the SAUNIXPMAP environment variable must have been set in advance. Execute the following commands from the command line: # SAUNIXPMAP=1 # export SAUNIXPMAP	--	ulong	No	Linux	--
File Size Limit (FILE_SIZE_LIMIT)	Process file size limit (bytes). This value is defined by the RLIMIT_FSIZE parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Group ID (GROUP_ID)	Effective group ID of the process	--	long	No	--	--
Hard CPU Limit (HARD_CPU_LIMIT)	Process hard CPU limit (Hcpulimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Core Size Limit (HARD_CORE_SIZE_LIMIT)	Process hard core size limit (bytes) (Hcorelimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Data Size Limit (HARD_DATA_SIZE_LIMIT)	Process hard data size limit (bytes) (Hdatalimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard File Size Limit (HARD_FILE_SIZE_LIMIT)	Process hard file size limit (bytes) (Hfilesizelimit)	--	double	No	HP-UX, Solaris, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Hard Stack Size Limit (HARD_STACK_SIZE_LIMIT)	Process hard stack size limit (bytes) (`Hstacklimit`)	--	double	No	HP-UX, Solaris, Linux	--
Hard Virtual Mem Size Limit (HARD_VIRTUAL_MEMORY_SIZE_LIMIT)	Process hard virtual memory size limit (bytes) (`Hrsslimit`)	--	double	No	HP-UX, Solaris, Linux	--
Heap Kbytes (HEAP_KBYTES)	Heap size used (kilobytes)	--	ulong	No	HP-UX, AIX, Linux	--
Interval (INTERVAL)	Interval at which the Application Process Interval (`PD_APSI`) record was stored (seconds)	--	ulong	Yes	HP-UX, Solaris, AIX, Linux	--
Major Faults (MAJOR_FAULTS)	Number of page faults that cause physical I/Os	--	ulong	Yes	--	--
Minor Faults (MINOR_FAULTS)	Number of page faults that do not cause physical I/Os	--	ulong	Yes	--	--
Monitoring Field (MONITORING_FIELD)	Field to be monitored	--	string(16)	No	--	--
Monitoring Label (MONITORING_LABEL)	Name used to identify the monitoring condition	--	string(32)	No	--	--
Monitoring Number (MONITORING_NUMBER)	The monitoring condition number	--	word	No	--	--
Niceness (NICENESS)	Process niceness. The scheduler class assigned to the process affects the value. For example, if a task is assigned to scheduler class RT, its niceness value is `0`.	--	long	No	--	--
PID (PID)	Process ID	--	long	No	--	--
Parent PID (PARENT_PID)	Process ID of the parent process	--	long	No	--	--
Priority (PRIORITY)	Process priority	--	long	No	--	--
Process Group ID (PROCESS_GROUP_ID)	Process group ID	--	long	No	--	--
Program Name (PROGRAM_NAME)	Program name	--	string(257)	No	--	--
Reads (READ_OPS)	In AIX, the number of RAW read operations that occurred. In Solaris, the number of Block read operations that occurred.	--	ulong	Yes	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Reads/sec (READ_OPS_PER_SECOND)	Frequency of read operations (operations per second)	--	float	Yes	HP-UX, Linux	READ_OPS ÷ ELAPSED_TIME
Real Group ID (REAL_GROUP_ID)	Real group ID of the process	--	long	No	--	--
Real Mem Kbytes (REAL_MEMORY_KBYTES)	Physical memory size used (kilobytes)	--	ulong	No	--	--
Real User ID (REAL_USER_ID)	Real user ID of the process	--	long	No	--	--
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type (INPUT_RECORD_TYPE)	Record type. Always `APSI`.	--	char(8)	No	--	--
Shared Mem Kbytes (SHARED_MEMORY_KBYTES)	Size of the shared memory used (kilobytes)	--	ulong	No	Solaris, AIX	--
Signals Rcvd (NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	Yes	Linux	--
Snapshot Time (LAST_SNAPSHOT_TIME)	Time at which the last process table information was acquired. Shows the record time if the process table contains no information.	--	time_t	No	--	--
Stack Kbytes (STACK_KBYTES)	Size of the stack being used by the process (kilobytes)	--	ulong	No	AIX	--
Stack Size Limit (STACK_SIZE_LIMIT)	Process stack size limit (bytes). This value is defined by the `RLIMIT_STACK` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Start Time (START_TIME)	Start time of the process	--	time_t	No	--	--
State (STATE)	Process state. One of the following values is set: - For AIX: `IDLE`, `ZOMBIE`, `STOP`, `RUN`, `SWAP`, `NONE` - For HP-UX: `IDLE`, `OTHER`, `RUN`, `SLEEP`, `STOP`, `ZOMBIE`, `NONE` - For Solaris: `IDLE`, `ONCPU`, `RUN`, `SLEEP`, `STOP`, `ZOMBIE`, `NONE` - For Linux: `IDLE`, `RUN`, `SLEEP`, `SWAP`, `STOP`, `ZOMBIE`, `NONE`	--	string(10)	No	--	--
Swaps (SWAPS)	Number of times swapping occurred	--	ulong	Yes	Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
System CPU (SYSTEM_CPU_TIME)	Time running in kernel mode (sec)	--	utime	Yes	--	--
Terminal (TERMINAL_NAME)	Name of the executing terminal. If the process does not have a terminal name, `??` is displayed.	--	string(40)	No	--	--
Throughput/sec (IO_KBYTES_PER_SECOND)	I/O operation speed (kilobytes/sec)	--	float	Yes	AIX, Linux	TOTAL_IO_KBYTES ÷ ELAPSED_TIME
Total I/O Kbytes (TOTAL_IO_KBYTES)	Total size transferred in I/O operations (kilobytes)	--	float	Yes	AIX, Linux	--
Total I/O Ops (TOTAL_IO_OPS)	Total number of I/O operations	--	ulong	Yes	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec (TOTAL_IO_OPS_PER_SECOND)	Frequency of I/O operations (operations per sec)	--	float	Yes	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ ELAPSED_TIME
Total Process Kbytes (TOTAL_PROCESS_KBYTES)	Process size (kilobytes). In AIX, this value is the same as the value displayed in the `SZ` column obtained with the `ps -l` command. In HP-UX and Solaris, this value is the same as the value displayed in the `SZ` column obtained with the `ps -l` command multiplied by the page size. In Linux, this value is the same as the value displayed in the `SIZE` column obtained with the `top` command.	--	ulong	No	--	--
User CPU (USER_CPU_TIME)	Time running in user mode (sec)	--	utime	Yes	--	--
User ID (USER_ID)	Effective user ID of the process	--	long	No	--	--
Virtual Env ID (VIRTUAL_ENV_ID)	ID of the virtualized environment created by the virtualized system provided by the OS	--	string(65)	No	HP-UX	--
Virtual Mem Kbytes (VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (kilobytes)	--	ulong	No	Solaris	--
Virtual Mem Size Limit (VIRTUAL_MEMORY_SIZE_LIMIT)	Size limit of the process's virtual memory (bytes) (rsslimit). This value is defined by the `RLIMIT_VMEM` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Writes (WRITE_OPS)	In AIX, the number of RAW write operations that occurred. In Solaris, the number of Block write operations that occurred.	--	ulong	Yes	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Writes/sec (WRITE_OPS_PER_SECOND)	Frequency of write operations (operations per sec)	--	float	Yes	HP-UX, Linux	WRITE_OPS ÷ ELAPSED_TIME

Legend:

Smry: Summary

Not sprtd in: Not supported in

Application Process Overview (PD_APS)

Function

The Application Process Overview (PD_APS) record stores performance data that shows the state of a process at a given point in time. One record is created per existing process ID in the system. This record is a multi-instance record.

Notes:

- This record can be used for real-time reports only.
- Each process represents a program that is being executed during data collection.
- If the process does not have a terminal name, ?? is displayed in the Terminal (TERMINAL_NAME) field.
- In AIX and Solaris, only data for the fields shown below is acquired for <defunct> processes (0 or n/a is displayed for all other fields):
  Child Process List, Command Line, Flags, Group, Group ID, Parent PID, Parent Process List, PID, Process Group ID, Program Name, Program/PID, Real Group, Real Group ID, Real User, Real User ID, Record Time, Record Type, Scheduler Class, Snapshot Time, Start Time, State, Terminal, User, User ID, Virtual Env ID (Solaris or AIX)
- An error (KAVJS5001-I) results if you attempt to display a historical report of this record.
- Any character in the source information that is not in the ASCII character set range of 0x20 to 0x7E will be converted to a hash mark (#: 0x23) before it is stored in the Program Name (PROGRAM_NAME) field or Command Line (COMMAND_LINE) field. Note that multi-byte characters are processed in single-byte units during conversion. For example, the multi-byte (full-width) letter A is converted as follows:

Source information		Post-conversion information	
Character encoding	Binary	Binary	Character string
Shift-JIS	8260	2360	#`
EUC	A3C1	2323	##
UTF-8	EFBCA1	232323	###

- If the target process of an instance is a zombie process, the following are displayed for the Group, Group ID, Real Group, Real Group ID, User, and User ID fields in AIX and HP-UX:

OS	Field	Content displayed
AIX	Group	Name of group with ID of 0
	Real Group	
	Group ID	0
	Real Group ID	
HP-UX	Group	N/A gid = -3
	Real Group	
	User	N/A uid = -3
	Group ID	-3
	Real Group ID	
	User ID	

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	No
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, APP2	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key field

- PD_APS_START_TIME
- PD_APS_PID

Lifetime

From the time the process executes until it terminates.

Record size

- Fixed portion: 681 bytes
- Variable portion: 6,421 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg I/O Kbytes (AVG_IO_KBYTES)	Average transfer size in I/O operations (kilobytes)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ TOTAL_IO_OPS
CPU % (CPU_PERCENT_USED)	Average obtained by dividing the process's CPU usage by the processor count (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) ÷ processors-count) × 100
CPU % Each (CPU_PERCENT_EACH)	CPU usage of the process (%)	--	float	No	--	((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) × 100
CPU Limit (CPU_LIMIT)	Process CPU limit. This value is defined by the RLIMIT_CPU parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Child Process List (CHILD_PROCESS_LIST)	List of child processes created. Child processes are delimited by a space. If more than 128 bytes, the final character is a right angle bracket (>).	--	string(128)	No	--	Searches for child chain on process list.

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Command Line (COMMAND_LINE)	Command line	--	string(4097)	No	--	--
Context Switches (CONTEXT_SWITCHES)	Number of context switch operations	--	ulong	Yes	HP-UX, Linux	--
Core Size Limit (CORE_SIZE_LIMIT)	Maximum size of a core file that can be created by the process (bytes). This value is defined by the `RLIMIT_CORE` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Data Size Limit (DATA_SIZE_LIMIT)	Process data size limit (bytes). This value is defined by the `RLIMIT_DATA` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Elapsed Time (ELAPSED_TIME)	Time elapsed since the process started	--	utime	No	--	- If process is running, `RECORD_TIME` - `START_TIME` - If process is not running, --

Executable Data Kbytes (EXECUTABLE_DATA_KBYTES)	Size of the data being used (kilobytes).
	For Solaris to collect this field, the `SAUNIXPMAP` environment variable must have been set in advance. Execute the following commands from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

Executable Text Kbytes (EXECUTABLE_TEXT_KBYTES)	Size of the text being used (kilobytes).
	For Solaris to collect this field, the `SAUNIXPMAP` environment variable must have been set in advance. Execute the following commands from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

File Size Limit (FILE_SIZE_LIMIT)	Process file size limit (bytes). This value is defined by the `RLIMIT_FSIZE` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Flags (FLAGS)	Program flag list. If more than 120 bytes, the final character is a right angle bracket (>).	--	string(120)	No	Linux	--
Group (GROUP)	Effective group name of the process	--	string(256)	No	--	--
Group ID (GROUP_ID)	Effective group ID of the process	--	long	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Hard CPU Limit (HARD_CPU_LIMIT)	Process hard CPU limit (`Hcpulimit`)	--	double	No	HP-UX, Solaris, Linux	--
Hard Core Size Limit (HARD_CORE_SIZE_LIMIT)	Process hard core size limit (bytes) (`Hcorelimit`)	--	double	No	HP-UX, Solaris, Linux	--
Hard Data Size Limit (HARD_DATA_SIZE_LIMIT)	Process hard data size limit (bytes) (`Hdatalimit`).	--	double	No	HP-UX, Solaris, Linux	--
Hard File Size Limit (HARD_FILE_SIZE_LIMIT)	Process hard file size limit (bytes) (`Hfilesizelimit`)	--	double	No	HP-UX, Solaris, Linux	--
Hard Stack Size Limit (HARD_STACK_SIZE_LIMIT)	Process hard stack size limit (bytes) (`Hstacklimit`)	--	double	No	HP-UX, Solaris, Linux	--
Hard Virtual Mem Size Limit (HARD_VIRTUAL_MEMORY_SIZE_LIMIT)	Process hard virtual memory size limit (bytes) (`Hrsslimit`)	--	double	No	HP-UX, Solaris, Linux	--
Heap Kbytes (HEAP_KBYTES)	Heap size used (kilobytes)	--	ulong	No	HP-UX, AIX, Linux	--
Interval (INTERVAL)	Interval at which the Application Process Overview (`PD_APS`) record was stored (seconds)	--	ulong	Yes	HP-UX, Solaris, AIX, Linux	--
Major Faults (MAJOR_FAULTS)	Number of page faults that cause physical I/Os	--	ulong	Yes	--	--
Minor Faults (MINOR_FAULTS)	Number of page faults that do not cause physical I/Os	--	ulong	Yes	--	--
Niceness (NICENESS)	Process niceness. The scheduler class assigned to the process affects the value. For example, if a task is assigned to scheduler class RT, its niceness value is `0`.	--	long	No	--	--
PID (PID)	Process ID	--	long	No	--	--
Parent PID (PARENT_PID)	Process ID of the parent process	--	long	No	--	--
Parent Process List (PARENT_PROCESS_LIST)	List of parent processes. For example, `3867 < 1 < 0` means that process 0 created process 1, and process 1 created process 3867. If more than 128 bytes, the final character is a right angle bracket (>).	--	string(128)	No	--	Searches for parent chain on the list of processes.
Priority (PRIORITY)	Process priority	--	long	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process Group ID (PROCESS_GROUP_ID)	Process group ID	--	long	No	--	--
Program Name (PROGRAM_NAME)	Program name	--	string(257)	No	--	--
Program/PID (PROGRAM_PID)	Name of the program executed and a process ID	--	string(280)	No	--	--
Reads (READ_OPS)	In AIX, the number of RAW read operations that occurred. In Solaris, the number of Block read operations that occurred.	--	ulong	Yes	HP-UX, Linux	--
Reads/sec (READ_OPS_PER_SECOND)	Frequency of read operations (operations per sec)	--	float	Yes	HP-UX, Linux	READ_OPS ÷ ELAPSED_TIME
Real Group (REAL_GROUP)	Real group name of the process	--	string(256)	No	--	--
Real Group ID (REAL_GROUP_ID)	Real group ID of the process	--	long	No	--	--
Real Mem Kbytes (REAL_MEMORY_KBYTES)	Physical memory size used (kilobytes)	--	ulong	No	--	--
Real User (REAL_USER)	Real user name of the process	--	string(256)	No	--	--
Real User ID (REAL_USER_ID)	Real user ID of the process	--	long	No	--	--
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type (INPUT_RECORD_TYPE)	Record type. Always `APS`.	--	char(8)	No	--	--
Scheduler Class (SCHEDULER_CLASS)	Scheduler class name. One of the following values is set: `TS`, `IA`, `FX`, `FSS`, `RT`, `SYS`	--	string(12)	No	HP-UX, AIX, Linux	--
Shared Mem Kbytes (SHARED_MEMORY_KBYTES)	Size of the shared memory used (kilobytes)	--	ulong	No	Solaris, AIX	--
Signals Rcvd (NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	Yes	Linux	--
Snapshot Time (LAST_SNAPSHOT_TIME)	Time at which the last process table information was acquired. Shows the record time if the process table contains no information.	--	time_t	No	--	--
Stack Kbytes (STACK_KBYTES)	Size of the stack being used by the process (kilobytes)	--	ulong	No	AIX	--
Stack Size Limit (STACK_SIZE_LIMIT)	Process stack size limit (bytes). This value is defined by the	--	double	No	HP-UX, Solaris,	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Stack Size Limit (STACK_SIZE_LIMIT)	RLIMIT_STACK parameter of the setrlimit system call.	--	double	No	Linux	--
Start Time (START_TIME)	Start time of the process	--	time_t	No	--	--
State (STATE)	Process state. One of the following values is set: - For AIX: IDLE, ZOMBIE, STOP, RUN, SWAP, NONE - For HP-UX: IDLE, OTHER, RUN, SLEEP, STOP, ZOMBIE, NONE - For Solaris: IDLE, ONCPU, RUN, SLEEP, STOP, ZOMBIE, NONE - For Linux: IDLE, RUN, SLEEP, SWAP, STOP, ZOMBIE, NONE	--	string(10)	No	--	--
Swaps (SWAPS)	Number of times swapping occurred	--	ulong	Yes	Linux	--
System CPU (SYSTEM_CPU_TIME)	Time running in kernel mode (sec)	--	utime	Yes	--	--
Terminal (TERMINAL_NAME)	Name of the executing terminal. If the process does not have a terminal name, ?? is displayed.	--	string(40)	No	--	--
Throughput/sec (IO_KBYTES_PER_SECOND)	I/O operation speed (kilobytes/sec)	--	float	Yes	AIX, Linux	TOTAL_IO_KBYTES ÷ ELAPSED_TIME
Total I/O Kbytes (TOTAL_IO_KBYTES)	Total size transferred in I/O operations (kilobytes)	--	float	Yes	AIX, Linux	--
Total I/O Ops (TOTAL_IO_OPS)	Total number of I/O operations	--	ulong	Yes	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec (TOTAL_IO_OPS_PER_SECOND)	Frequency of I/O operations (operations per sec)	--	float	Yes	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ ELAPSED_TIME
Total Process Kbytes (TOTAL_PROCESS_KBYTES)	Process size (kilobytes). In AIX, this field is the same as the value displayed in the SZ column obtained with the ps -l command. In HP-UX and Solaris, this value is the same as the value displayed in the SZ column obtained with the ps -l command multiplied by the page size. In Linux, this value is the same as the value displayed in the SIZE column obtained with the top command.	--	ulong	No	--	--
User CPU (USER_CPU_TIME)	Time running in user mode (sec)	--	utime	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
User (USER)	Effective user name of the process	--	string(256)	No	--	--
User ID (USER_ID)	Effective user ID of the process	--	long	No	--	--
Virtual Env ID (VIRTUAL_ENV_ID)	ID of the virtualized environment created by the virtualized system provided by the OS	--	string(65)	No	HP-UX	--
Virtual Mem Kbytes (VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (kilobytes)	--	ulong	No	Solaris	--
Virtual Mem Size Limit (VIRTUAL_MEMORY_SIZE_LIMIT)	Size limit of the process virtual memory (bytes) (`rsslimit`). This value is defined by the `RLIMIT_VMEM` parameter of the `setrlimit` system call.	--	double	No	HP-UX, Solaris, Linux	--
Writes (WRITE_OPS)	In AIX, the number of RAW write operations that occurred. In Solaris, the number of Block write operations that occurred.	--	ulong	Yes	HP-UX, Linux	--
Writes/sec (WRITE_OPS_PER_SECOND)	Frequency of write operations (operations per sec)	--	float	Yes	HP-UX, Linux	WRITE_OPS ÷ ELAPSED_TIME

Legend:

Smry: Summary

Not sprtd in: Not supported in

Application Summary (PD_APP)

Function

The Application Summary (PD_APP) record stores performance data that summarizes, by application, the records stored in the Process Detail (PD) record at a given point in time. One record is stored for each application. This record is a multi-instance record.

Note:

To change the application definition, you must change the settings from PFM - Web Console.

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, PD	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key field

PD_APP_APPLICATION_NAME

Lifetime

From the time the setting for monitoring application operation is specified using PFM - Web Console until the setting is deleted.

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,000 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Exist(APPLICATION_EXIST)	The results of all Process Range conditions. NORMAL or ABNORMAL is displayed. When a blank is displayed for Process01 Status -Process15 Status, the blank is ignored. NORMAL: At least one of the results of Process01 Status - Process15 Status is NORMAL. ABNORMAL: All the results of Process01 Status -Process15 Status are ABNORMAL. Blank: The settings are not set.	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Name(APPLICATION_NAME)	The name specified by Application monitoring setting	--	string(64)	No	--	--
Application Status(APPLICATION_STATUS)	The results of all Process Range conditions. NORMAL or ABNORMAL is displayed. When a blank is displayed for Process01 Status -Process15 Status, the blank is ignored. NORMAL: All the results of Process01 Status -Process15 Status are NORMAL. ABNORMAL: At least one of the results of Process01 Status -Process15 Status is ABNORMAL Blank: The settings are not set.	--	string(10)	No	--	--
Interval(INTERVAL)	Interval for which the Application Summary (PD_APP) record was stored (in seconds). The normal value is 0.	--	ulong	No	--	--
Process01 Count(PROCESS01_COUNT)	The number of processes currently operating	--	word	No	--	--
Process01 Kind(PROCESS01_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process01 Name(PROCESS01_NAME)	Process name	--	string(32)	No	--	--
Process01 Range(PROCESS01_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process01 Status(PROCESS01_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process02 Count(PROCESS02_COUNT)	The number of processes currently operating	--	word	No	--	--
Process02 Kind(PROCESS02_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process02 Name(PROCESS02_NAME)	Process name	--	string(32)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process02 Range(PROCESS02_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process02 Status(PROCESS02_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process03 Count(PROCESS03_COUNT)	The number of processes currently operating	--	word	No	--	--
Process03 Kind(PROCESS03_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process03 Name(PROCESS03_NAME)	Process name	--	string(32)	No	--	--
Process03 Range(PROCESS03_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process03 Status(PROCESS03_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process04 Count(PROCESS04_COUNT)	The number of processes currently operating	--	word	No	--	--
Process04 Kind(PROCESS04_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process04 Name(PROCESS04_NAME)	Process name	--	string(32)	No	--	--
Process04 Range(PROCESS04_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process04 Status(PROCESS04_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process05 Count(PROCESS05_COUNT)	The number of processes currently operating	--	word	No	--	--
Process05 Kind(PROCESS05_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process05 Name(PROCESS05_NAME)	Process name	--	string(32)	No	--	--
Process05 Range(PROCESS05_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process05 Status(PROCESS05_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process06 Count(PROCESS06_COUNT)	The number of processes currently operating	--	word	No	--	--
Process06 Kind(PROCESS06_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process06 Name(PROCESS06_NAME)	Process name	--	string(32)	No	--	--
Process06 Range(PROCESS06_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process06 Status(PROCESS06_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process07 Count(PROCESS07_COUNT)	The number of processes currently operating	--	word	No	--	--
Process07 Kind(PROCESS07_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process07 Name(PROCESS07_NAME)	Process name	--	string(32)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process07 Range(PROCESS07_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process07 Status(PROCESS07_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process08 Count(PROCESS08_COUNT)	The number of processes currently operating	--	word	No	--	--
Process08 Kind(PROCESS08_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process08 Name(PROCESS08_NAME)	Process name	--	string(32)	No	--	--
Process08 Range(PROCESS08_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process08 Status(PROCESS08_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process09 Count(PROCESS09_COUNT)	The number of processes currently operating	--	word	No	--	--
Process09 Kind(PROCESS09_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process09 Name(PROCESS09_NAME)	Process name	--	string(32)	No	--	--
Process09 Range(PROCESS09_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process09 Status(PROCESS09_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process10 Count(PROCESS10_COUNT)	The number of processes currently operating	--	word	No	--	--
Process10 Kind(PROCESS10_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process10 Name(PROCESS10_NAME)	Process name	--	string(32)	No	--	--
Process10 Range(PROCESS10_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process10 Status(PROCESS10_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process11 Count(PROCESS11_COUNT)	The number of processes currently operating	--	word	No	--	--
Process11 Kind(PROCESS11_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process11 Name(PROCESS11_NAME)	Process name	--	string(32)	No	--	--
Process11 Range(PROCESS11_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process11 Status(PROCESS11_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process12 Count(PROCESS12_COUNT)	The number of processes currently operating	--	word	No	--	--
Process12 Kind(PROCESS12_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process12 Name(PROCESS12_NAME)	Process name	--	string(32)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process12 Range(PROCESS12_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process12 Status(PROCESS12_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process13 Count(PROCESS13_COUNT)	The number of processes currently operating	--	word	No	--	--
Process13 Kind(PROCESS13_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process13 Name(PROCESS13_NAME)	Process name	--	string(32)	No	--	--
Process13 Range(PROCESS13_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process13 Status(PROCESS13_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Process14 Count(PROCESS14_COUNT)	The number of processes currently operating	--	word	No	--	--
Process14 Kind(PROCESS14_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process14 Name(PROCESS14_NAME)	Process name	--	string(32)	No	--	--
Process14 Range(PROCESS14_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process14 Status(PROCESS14_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process15 Count(PROCESS15_COUNT)	The number of processes currently operating	--	word	No	--	--
Process15 Kind(PROCESS15_KIND)	The specified type of the condition is displayed	--	string(4)	No	--	--
Process15 Name(PROCESS15_NAME)	Process name	--	string(32)	No	--	--
Process15 Range(PROCESS15_RANGE)	The range conditions for each process. The conditions are displayed as "Minimum value-Maximum value".	--	string(12)	No	--	--
Process15 Status(PROCESS15_STATUS)	The Process Range condition result for each process. NORMAL, ABNORMAL, or a blank is displayed. NORMAL: No error exists. ABNORMAL: An error exists. Blank: The settings are not set.	--	string(10)	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always APP.	--	char(8)	No	--	--
Virtual Env ID(VIRTUAL_ENV_ID)	ID of the virtual environment made by the virtual system provided by the OS.	--	string(16)	No	HP-UX, Solaris (SPARC), Linux	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Application Summary Extension (PD_APP2)

Function

The Application Summary Extension (PD_APP2) record stores performance data that summarizes by application the records stored in the Application Process Overview (PD_APS) record at a given point in time. One record is stored for each application. This record is a multi-instance record.

Note:

To change the application definition, you must change the settings from PFM - Web Console.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key field

PD_APP2_APPLICATION_NAME

Lifetime

From the time the application definition is added in PFM - Web Console until it is deleted.

Record size

- Fixed portion: 681 bytes
- Variable portion: 153 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Exist (APPLICATION_EXIST)	The application status set from the process monitoring settings.	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Application Exist (APPLICATION_EXIST)	**NORMAL** or **ABNORMAL** is displayed. The application status is the result obtained based on the status set for the monitoring targets. To check the status of the monitoring targets, refer to Monitoring Status, which is displayed in the Application Process Detail (PD_APPD) record. **NORMAL**: The status of least one of the monitoring targets is NORMAL. **ABNORMAL**: The status of all of the monitoring targets is ABNORMAL.	--	string(10)	No	--	--
Application Name (APPLICATION_NAME)	The name set in the process monitoring settings	--	string(64)	No	--	--
Application Status (APPLICATION_STATUS)	The application status set from the process monitoring settings. **NORMAL** or **ABNORMAL** is displayed. The application status is the result obtained based on the status set for the monitoring targets. To check the status of the monitoring targets, refer to Monitoring Status, which is displayed in the Application Process Detail (PD_APPD) record. **NORMAL**: The status of all of the monitoring targets is NORMAL. **ABNORMAL**: The status of least one of the monitoring targets is ABNORMAL.	--	string(10)	No	--	--
Case Sensitive (CASE_SENSITIVE)	Distinguishes between upper- and lower-case letters: **Yes**: Case sensitive **No**: Not case sensitive	--	string(4)	No	--	--
Interval (INTERVAL)	Number of seconds in the interval during which the Application Process Detail (PD_APP2) records were stored. Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type (INPUT_RECORD_TYPE)	Record type. Always APP2.	--	char(8)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Virtual Env ID (VIRTUAL_ENV_ID)	ID of the virtualized environment created by the virtualized system provided by the OS	--	string(65)	No	HP-UX	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

CPU - Per Processor Detail (PI_CPUP)

Function

The CPU - Per Processor Detail (PI_CPUP) record stores the performance data per unit time on the processor usage. Every time performance data is collected, a record is created for each processor. This is a multi-instance record.

If you are collecting processor performance data from the results output by the sar command in the AIX environment, the displayed values will be for different fields. For details about the values displayed in the fields when collecting processor performance data from the results output by the sar command in the AIX environment, see *2.3.23 Monitoring processor performance based on results output by the sar command in an AIX environment*.

Note:

- If the DLPAR function of AIX, the vPars function of HP-UX, the DR function of Solaris, or the CPU hotplug function of Linux is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification:

System resource type	Report type	Actions to be taken after modification
CPU	Real-time	Display the report again.
	Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

  Fields other than those listed below are affected by a system resource modification:
  - Interval (INTERVAL)
  - Record Time (RECORD_TIME)
  - Record Type (INPUT_RECORD_TYPE)
  - Fields that are added only when data is recorded in the Store database

  For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*.
- In AIX, when the Status field of CPU - Per Processor Detail (PI_CPUP) record is changed from operating CPU to non-operating, since the operation duration exists in between the collection interval, [Online] is displayed at the time of the next collection after the change. After that, if the CPU state is not changed, it is further displayed as [Offline] during next collection time.
- If you collect in a non-global zone environment of Solaris, the value of the Boot Time field might not be displayed correctly.

  Display example: 1970 01 01 06:00:00

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No

Item	Default value	Modifiable
`Realtime Report Data Collection Mode`	`Reschedule`	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

`PI_CPUP_LOGICAL_PROCESSOR_ID`

Lifetime

None

Record size

- Fixed portion: 681 bytes
- Variable portion: 376 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Boot Time(SYSTEM_BOOT_TIME)	The last boot time	COPY	time_t	No	--	--
CPU % (KERNELMODE_USERMODE_PERCENT)	CPU usage per processor (%)	%	float	No	--	• ((PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME) ÷ (PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME + PROCESSOR_IDLE_TIME + PROCESSOR_WAIT_TIME)) × 100 • `%sys` column + `%usr` column when collecting results output by the `sar` command
Context Switches(PROCESSOR_CONTEXT_SWITCHES)	Number of times context switching was executed	AVG	ulong	Yes	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Context Switches/sec(PROCESSOR_CONTEXT_SWITCHES_PER_SECOND)	Frequency at which context switching was executed (number of times per second)	R	float	Yes	HP-UX, Linux	PROCESSOR_CONTEXT_SWITCHES ÷ INTERVAL
Idle %(PROCESSOR_IDLE_PERCENT)	Percentage of time the processor was idle (%)	%	float	No	--	• PROCESSOR_IDLE_TIME ÷ (PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME + PROCESSOR_IDLE_TIME + PROCESSOR_WAIT_TIME) × 100 • `%idle` when collecting results output by the `sar` command
Idle Time(PROCESSOR_IDLE_TIME)	Time during which the processor was idle (seconds)	AVG	utime	Yes	--	--
Interrupts(PROCESSOR_INTERRUPTS)	Number of interrupts that occurred	AVG	ulong	Yes	HP-UX, AIX	--
Interrupts/sec(PROCESSOR_INTERRUPTS_PER_SECOND)	Frequency at which interrupts occurred (number of times per second)	R	float	Yes	HP-UX, AIX	PROCESSOR_INTERRUPTS / INTERVAL
Interval(INTERVAL)	Interval at which the CPU - Per Processor Detail (PI_CPUP) record was stored (seconds)	COPY	ulong	Yes	--	• When delta values are not checked in real time, `RECORD_TIME` - *Last-boot-time* • Other cases, `RECORD_TIME` - *Previous-record-time*
Processor ID(LOGICAL_PROCESSOR_ID)	Processor identifier	COPY	string(12)	No	--	• Does not apply when not collecting results output by the `sar` command. • `cpu` column when collecting results output by the `sar` command
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always CPUP.	COPY	char(8)	No	--	--
Status(PROCESSOR_STATUS)	Processor status	COPY	string(12)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Sys Calls/sec(PROCESSOR_SYSTEM_CALLS_PER_SECOND)	Frequency at which system calls were issued (number of system calls per second)	R	float	Yes	HP-UX, Linux	PROCESSOR_SYSTEM_CALLS ÷ INTERVAL
System %(PROCESSOR_SYSTEM_PERCENT)	Percentage of time the system ran in the kernel mode (%)	%	float	No	--	• PROCESSOR_SYSTEM_TIME ÷ (PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME + PROCESSOR_IDLE_TIME + PROCESSOR_WAIT_TIME) × 100 • `%sys` column when collecting results output by the `sar` command
System Calls(PROCESSOR_SYSTEM_CALLS)	Number of system calls that were issued	AVG	ulong	Yes	HP-UX, Linux	--
System Time(PROCESSOR_SYSTEM_TIME)	Time during which the system ran in the kernel mode (seconds)	AVG	utime	Yes	--	--
Traps(PROCESSOR_TRAPS)	Number of traps that were executed	AVG	ulong	Yes	HP-UX, AIX, Linux	--
Traps/sec(PROCESSOR_TRAPS_PER_SECOND)	Frequency at which traps were executed (number of traps per second)	R	float	Yes	HP-UX, AIX, Linux	PROCESSOR_TRAPS ÷ INTERVAL
Type(PROCESSOR_TYPE)	Processor explanation	COPY	string(40)	No	--	--
Up Time(SYSTEM_UP_TIME)	Time elapsed since the last boot (seconds)	COPY	string(20)	No	--	RECORD_TIME - SYSTEM_BOOT_TIME
User %(PROCESSOR_USER_PERCENT)	Percentage of time the processor ran in the user mode (%)	%	float	No	--	• PROCESSOR_USER_TIME ÷ (PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME + PROCESSOR_IDLE_TIME + PROCESSOR_WAIT_TIME) × 100 • `%usr` column when collecting results output by the `sar` command
User Time(PROCESSOR_USER_TIME)	Time during which the processor ran in the user mode (seconds)	AVG	utime	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Wait % (PROCESSOR_WAIT_PERCENT)	Percentage of time the processor was waiting for I/O (%)	%	float	No	Solaris	• PROCESSOR_WAIT_TIME ÷ (PROCESSOR_USER_TIME + PROCESSOR_SYSTEM_TIME + PROCESSOR_IDLE_TIME + PROCESSOR_WAIT_TIME) × 100 • `%wio` column when collecting results output by the `sar` command
Wait Time(PROCESSOR_WAIT_TIME)	Time during which the processor was waiting for I/O (seconds)	AVG	utime	Yes	Solaris	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Device Detail (PI_DEVD)

Function

The Device Detail (PI_DEVD) record stores the performance data per unit time on the local disk device usage. Every time performance data is collected, a record is created for each local disk device. This is a multi-instance record.

Notes:

- In AIX, performance data of devices whose device names begin with vscsi or vhost is not collected.
- In Linux, the Device Detail (PI_DEVD) record is collected as performance data for block devices matching the following major numbers from the information recorded in /proc/partitions:

  • In Linux 6: 1 to 4, 7 to 9, 11 to 37, 40, 41, 43 to 59, 64 to 94, 96, 98, 99, 101 to 117, 128 to 147, 152, 153, 160, 161, 180, 199, 201, 202, 256

  • In Linux 7, Linux 8, Oracle Linux 7#, or Oracle Linux 8: 1 to 4, 7 to 9, 11, 15 to 35, 37, 43 to 59, 64 to 94, 96, 98, 99, 101 to 117, 128 to 147, 152, 153, 160, 161, 179, 180, 199, 201, 202, 256 to 259

  # When the Unbreakable Enterprise Kernel (UEK) is used, the performance data to be collected is the same as that defined for Linux 6 above.
- If the DR function of Solaris is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification.

System resource type	Report type	Actions to be taken after modification
Device	Real-time	Display the report again.
	Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

  Fields other than those listed below are affected by a system resource modification:

  • Interval (INTERVAL)

  • Record Time (RECORD_TIME)

  • Record Type (INPUT_RECORD_TYPE)

  • Fields that are added only when data is recorded in the Store database

  For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*.
- In HP-UX, the local disk device of Legacy DSF and Persistent DSF both are collected.
- In AIX, a PI_DEVD record's key field is a device name. Therefore, if the device name is changed (by using the rendev command), the continuity of performance data cannot be maintained even if the same device is used before and after the device name changes. In this case, that device is recognized as a different device and performance data is collected.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes

Item	Default value	Modifiable
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

PI_DEVD_DEVICE_NAME

Lifetime

None

Record size

- Fixed portion: 681 bytes
- Variable portion: 405 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg Service Time(AVG_SERVICE_TIME)	Average I/O service time for devices (seconds)	AVG	utime	No	--	TOTAL_SERVICE_TIME ÷ TOTAL_OPS
Avg Wait Time(AVG_WAIT_TIME)	Average I/O wait time for devices (seconds)	AVG	utime	No	AIX	TOTAL_WAIT_TIME ÷ TOTAL_OPS
Busy % (BUSY_PERCENT)	Percentage of time a disk was busy (%). This value may exceed 100 when processes are continuously executed on a device.	%	float	No	--	(TOTAL_BUSY_TIME ÷ INTERVAL) × 100
Device Name(DEVICE_NAME)	Device name	COPY	string(40)	No	--	--
Device Type(DEVICE_TYPE)	Device type. Disk (disk device) or Tape (tape device)	COPY	string(5)	No	--	--
I/O Mbytes(TOTAL_IO_MBYTES)	Total size of data transferred through I/O operations (MB)	AVG	float	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval at which the Device Detail (PI_DEVD) record was stored (seconds)	COPY	ulong	Yes	--	• When delta values are not checked in real time, `RECORD_TIME` - *last-boot-time* • Other cases, `RECORD_TIME` - *previous-record-time*
Mbytes Xferd/sec(MBYTES_TRANSFERRED_PER_SECOND)	Average I/O speed (MB per second)	R	float	Yes	--	TOTAL_IO_MBYTES ÷ INTERVAL
Queue Length(QUEUE_LENGTH)	Number of I/O requests in the device queue	AVG	ulong	No	AIX	--
Read % (READ_OPS_PERCENT)	Read operations as a percentage of all I/O operations (%)	%	float	No	HP-UX, AIX	READ_OPS ÷ (READ_OPS + WRITE_OPS) × 100
Read Mbytes(TOTAL_READ_MBYTES)	Size of data transferred in read operations (MB)	AVG	float	Yes	HP-UX	--
Read Ops(READ_OPS)	Number of read operations that occurred	AVG	ulong	Yes	HP-UX, AIX	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of read operations per second)	R	float	Yes	HP-UX, AIX	READ_OPS ÷ INTERVAL
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always DEVD.	COPY	char(8)	No	--	--
Seek Ops(SEEK_OPS)	Number of seek operations	AVG	ulong	Yes	Solaris, AIX, Linux	--
Total Busy Time(TOTAL_BUSY_TIME)	Total busy time for a device (seconds). This value may exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	Yes	--	--
Total I/O Ops(TOTAL_OPS)	Number of I/O operations that occurred	AVG	ulong	Yes	--	--
Total I/O Ops/sec(TOTAL_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of I/O operations per second)	R	float	Yes	--	TOTAL_OPS ÷ INTERVAL
Total Service Time(TOTAL_SERVICE_TIME)	Total service time for a device (seconds). This value includes wait time. In HP-UX, this value indicates the total service time of all I/Os, and	AVG	utime	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Service Time(TOTAL_SERVICE_TIME)	therefore, it may significantly exceed the value of Interval when processes are continuously executed on a device. Even with other OSs, the value may exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	Yes	--	--
Total Wait Time(TOTAL_WAIT_TIME)	Total wait time for a device (seconds). In HP-UX, this value indicates the total wait time of all I/Os, and therefore, it may significantly exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	Yes	AIX	--
Wait Length Time(WAIT_LEN_TIME)	Length of time I/Os waited for a device (seconds). The processing capacity of the I/Os that were waiting is divided by the I/Os' processing capacity per second, and is then integrated over the wait time.	AVG	utime	Yes	AIX	--
Write % (WRITE_OPS_PERCENT)	Write operations as a percentage of all I/O operations (%)	%	float	No	HP-UX, AIX	WRITE_OPS ÷ (READ_OPS + WRITE_OPS) × 100
Write Mbytes(TOTAL_WRITE_MBYTES)	Size of data transferred in write operations (MB)	AVG	float	Yes	HP-UX	--
Write Ops(WRITE_OPS)	Number of write operations that occurred	AVG	ulong	Yes	HP-UX, AIX	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of write operations per second)	R	float	Yes	HP-UX, AIX	WRITE_OPS ÷ INTERVAL

Legend:

Smry: Summary

Not sprtd in: Not supported in

Device Summary (PI_DEVS)

Function

The Device Summary (PI_DEVS) record stores the performance data in which the data stored in the Device Detail (PI_DEVD) record is summarized per unit time. Note that even if the Device Detail (PI_DEVD) record is not collected, performance data is stored in the Device Summary (PI_DEVS) record.

Notes:

- In Linux, the performance data of only the devices that have the following major numbers are collected from /proc/partitions and summarized in the Device Summary (PI_DEVS) record: 3, 8, 22, 33, 34, 56, 57, 65 to 71, 88 to 91, and 128 to 135
- If the DR function of Solaris is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification.

System resource type	Report type	Actions to be taken after modification
Device	Real-time	Display the report again.
	Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

  Fields other than those listed below are affected by a system resource modification:

  • Interval (INTERVAL)

  • Record Time (RECORD_TIME)

  • Record Type (INPUT_RECORD_TYPE)

  • Fields that are added only when data is recorded in the Store database

  For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*

- Only the local disk devices of Persistent DSF are collected in HP-UX.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

None

Lifetime

None

Record size

- Fixed portion: 1,069 bytes
- Variable portion: 0 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg Service Time/ device(SERVICE_TIME)	Average service time per device (seconds). In HP-UX, this value indicates the total service time of all I/Os, and therefore, it may significantly exceed the value of Interval when processes are continuously executed on a device. Even with other OSs, the value may exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	No	--	TOTAL_SERVICE_TIME ÷ DEVICE_COUNT
Avg Service Time/ op(AVG_SERVICE_TIME)	Average I/O service time for a device (seconds)	AVG	utime	No	--	TOTAL_SERVICE_TIME ÷ TOTAL_IO_OPS
Busy % (BUSY_PERCENT)	Average percentage of time a disk was busy (%). This value may exceed 100 when processes are continuously executed on a device.	%	float	No	--	((TOTAL_BUSY_TIME ÷ INTERVAL) ÷ DEVICE_COUNT) × 100
Devices(DEVICE_COUNT)	Device count	AVG	ulong	No	--	Number of Device Detail (PI_DEVD) records of the category "disk" summarized in the Device Summary (PI_DEVS) record
I/O Mbytes(TOTAL_IO_MBYTES)	Total size of data transferred through I/O operations (MB)	AVG	double	Yes	--	--
Interval(INTERVAL)	Interval at which the Device Summary (PI_DEVS) record was stored (seconds)	COPY	ulong	Yes	--	• When delta values are not checked in real time, `RECORD_TIME` - *last-boot-time* • Other cases, `RECORD_TIME` - *previous-record-time*
Mbytes Xferd/ sec(MBYTES_TRANSFERRED_PER_SECOND)	Average I/O speed (MB per second)	R	float	Yes	--	TOTAL_IO_MBYTES ÷ INTERVAL

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Queue Length(QUEUE_LENGTH)	Number of I/O requests in the device queue	AVG	ulong	No	AIX	sum-of-queue-lengths-of-all-devices ÷ DEVICE_COUNT
Read Ops % (READ_OPS_PERCENT)	Read operations as a percentage of all I/O operations (%)	%	float	No	HP-UX, AIX	TOTAL_READ_OPS ÷ TOTAL_IO_OPS × 100
Reads(TOTAL_READ_OPS)	Number of read operations that occurred	AVG	ulong	Yes	HP-UX, AIX	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of read operations per second)	R	float	Yes	HP-UX, AIX	TOTAL_READ_OPS ÷ INTERVAL
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always DEVS.	COPY	char(8)	No	--	--
Seek Ops(SEEK_OPS)	Number of seek operations	AVG	ulong	Yes	Solaris, AIX, Linux	--
Total Busy Time(TOTAL_BUSY_TIME)	Total busy time for a device (seconds). This value may exceed the value of Interval × Devices when processes are continuously executed on a device.	AVG	utime	Yes	--	--
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	AVG	ulong	Yes	--	TOTAL_READ_OPS + TOTAL_WRITE_OPS
Total I/O Ops/sec(TOTAL_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	R	float	Yes	--	TOTAL_IO_OPS ÷ INTERVAL
Total Service Time(TOTAL_SERVICE_TIME)	Total service time for a device (seconds). This value includes wait time. In HP-UX, this value indicates the total service time of all I/Os, and therefore, it may significantly exceed the value of Interval × Devices when processes are continuously executed on a device. Even with other OSs, the value may exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	Yes	--	--
Total Wait Length Time(TOTAL_WAIT_LEN_TIME)	Total length of time I/Os waited for a device (seconds). The processing capacity of the I/Os that were waiting is divided by the I/Os' processing capacity per second, and is then integrated over the wait time.	AVG	utime	Yes	AIX	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Wait Time(TOTAL_WAIT_TIME)	Total wait time for a device (seconds). In HP-UX, this value indicates the total wait time of all I/Os, and therefore, it may significantly exceed the value of Interval × Devices when processes are continuously executed on a device.	AVG	utime	Yes	AIX	--
Wait Length Time(WAIT_LEN_TIME)	Average length of time an I/O waited for a device (seconds). The processing capacity of the I/Os that were waiting is divided by the I/Os' processing capacity per second, is integrated over the wait time, and is then divided by the device count.	AVG	utime	No	AIX	TOTAL_WAIT_LEN_TIME ÷ DEVICE_COUNT
Wait Time(WAIT_TIME)	Average wait time for a device (seconds). In HP-UX, this value indicates the total wait time of all I/Os, and therefore, it may significantly exceed the value of Interval when processes are continuously executed on a device.	AVG	utime	No	AIX	TOTAL_WAIT_TIME ÷ DEVICE_COUNT
Write Ops % (WRITE_OPS_PERCENT)	Write operations as a percentage of all I/O operations (%)	%	float	No	HP-UX, AIX	TOTAL_WRITE_OPS ÷ TOTAL_IO_OPS × 100
Writes(TOTAL_WRITE_OPS)	Number of write operations that occurred	AVG	ulong	Yes	HP-UX, AIX	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of write operations per second)	R	float	Yes	HP-UX, AIX	TOTAL_WRITE_OPS ÷ INTERVAL

Legend:

Smry: Summary

Not sprtd in: Not supported in

File System Detail - Local (PD_FSL)

Function

The File System Detail - Local (PD_FSL) record stores the performance data showing the status of the local file system capacity at a given point in time. A record is created for each local file system. This is a multi-instance record.

Notes:

- In HP-UX, Solaris, and Linux, the file system area for a general user is defined as the sum of the file system area used and the file system area that is available. The file system area for a superuser also includes a reserved file system area. In AIX, because there is no reserved area, there is no difference between the file system area for a general user and that for a superuser.
- In HP-UX, Solaris, and Linux, the number of i nodes is defined as the sum of the i nodes used and the i nodes that are available. The number of i nodes for a superuser also includes the number of reserved i nodes. In AIX, because there are no reserved i nodes, there is no difference between the number of reserved i nodes for a general user and that for a superuser.
- In specialized file systems, it may not be possible to collect performance data. For example, a remote file system whose remote host name cannot be acquired as a virtual local file system may be treated as a local file system.
- The basic block size (FFBS: Fundamental Filesystem Block Size) used in the field table of this record is the allocation unit used during file system creation. The number of blocks that corresponds to the basic block size is called the *basic block count*. The logical block size (PFBS: Preferred Filesystem Block Size) is the allocation unit used during input/output operations. The number of blocks that corresponds to the logical block size is called the *logical block count*.
- In Solaris, /system/contract or /system/object is not collected.
- In Solaris, do not use a colon (:) in a file system name. If a colon is used by mistake, the file name is treated as a remote file system and collected as the PD_FSR record.
- When information is collected into the PD_FSL record, a system call is issued at the data collection interval. Because collecting information using the system call involves file system access, an attempt to execute the umount command to unmount the file system during information collection might fail. If execution of the command fails, re-execute the umount command at a time different from the collection interval.
- If a special file system (such as a pseudo file system) exists in the system, the PD_FSL record behaves in either of the following ways:

  • The PD_FSL record does not create instances of the record. (File systems such as /proc file systems perform this behavior.)

  • The PD_FSL record outputs the values collected from the OS directly as record information. In this case, performance data that is output is indeterminate.
- Depending on the system status, the number of instances in the PD_FSL record might be different from the result of the df command. If they are different, estimate the disk space requirements based on the actual collection results.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	3600	Yes
Collection Offset#	0	Yes
Log	No	Yes

Item	Default value	Modifiable
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

PD_FSL_FILESYSTEM_NAME

Lifetime

From the mounting of a file system until its unmounting

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,216 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Available Space % (TOTAL_MBYTES_AVAILABLE_PERCENT)	Percentage of size (MB) available to general users (%)	--	float	No	--	(TOTAL_MBYTES_AVAILABLE ÷ (TOTAL_SIZE_IN_MBYTES - TOTAL_MBYTES_RESERVED)) × 100
Available Space Blocks(TOTAL_BLOCKS_AVAILABLE)	Number of logical blocks available to general users	--	double	No	--	--
Available Space Mbytes(TOTAL_MBYTES_AVAILABLE)	Size available to general users (MB)	--	double	No	--	(Fundamental Block count available to the general user × Fundamental Block Size) ÷ 1MB
Block Size(BLOCKSIZE)	Logical block size (bytes)	--	double	No	--	--
Blocks Free(TOTAL_BLOCKS_FREE)	Number of free logical blocks	--	double	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Blocks in Use(TOTAL_BLOCKS_IN_USE)	Number of logical blocks being used	--	double	No	--	--
Device Name(DEVICE_NAME)	Name of the device where the file system is located	--	string(40)	No	--	--
File System(FILESYSTEM_NAME)	File system mounting point	--	string(1024)	No	--	--
File System Type(FILESYSTEM_TYPE)	File system type (e.g., UFS, HFS)	--	string(20)	No	--	--
Interval(INTERVAL)	Interval at which the File System Detail - Local (PD_FSL) record was stored (seconds)	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Mbytes Free(TOTAL_MBYTES_FREE)	Size that is free (MB)	--	double	No	--	(TOTAL_BLOCKS_FREE × Fundamental Block Size) ÷ 1MB
Mbytes Free % (TOTAL_MBYTES_FREE_PERCENT)	Percentage of size (MB) that is free (%)	--	float	No	--	(TOTAL_MBYTES_FREE ÷ TOTAL_SIZE_IN_MBYTES) × 100
Mbytes Rsvd(TOTAL_MBYTES_RESERVED)	Size reserved for superusers (MB)	--	double	No	AIX	TOTAL_MBYTES_FREE-TOTAL_MBYTES_AVAILABLE
Mbytes Rsvd % (TOTAL_MBYTES_RESERVED_PERCENT)	Percentage of size (MB) reserved for superusers (%)	--	float	No	AIX	(TOTAL_MBYTES_RESERVED ÷ TOTAL_SIZE_IN_MBYTES) × 100
Mbytes in Use(TOTAL_MBYTES_IN_USE)	The size (in megabytes) used by superusers and general users	--	double	No	--	((TOTAL_SIZE_IN_BLOCKS - TOTAL_BLOCKS_FREE) × Fundamental Block Size) ÷ 1MB
Mbytes in Use % (TOTAL_MBYTES_IN_USE_PERCENT)	The percentage (%) of the size (in megabytes) used by superusers and general users	--	float	No	--	(TOTAL_MBYTES_IN_USE ÷ (TOTAL_SIZE_IN_MBYTES - TOTAL_MBYTES_RESERVED)) × 100
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always FSL.	--	char(8)	No	--	--
Total Inodes(TOTAL_NUMBER_OF_INODES)	Number of i nodes in a file system	--	double	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Inodes Available(TOTAL_INODES_AVAILABLE)	Number of i nodes available to general users	--	double	No	--	--
Total Inodes Available % (TOTAL_INODES_AVAILABLE_PERCENT)	Percentage of i nodes available to general users (%)	--	float	No	--	(TOTAL_INODES_AVAILABLE ÷ (TOTAL_NUMBER_OF_INODES - TOTAL_INODES_RESERVED)) × 100
Total Inodes Free(TOTAL_INODES_FREE)	Number of free i nodes	--	double	No	--	--
Total Inodes Free % (TOTAL_INODES_FREE_PERCENT)	Percentage of i nodes that are free (%)	--	float	No	--	(TOTAL_INODES_FREE ÷ TOTAL_NUMBER_OF_INODES) × 100
Total Inodes Rsvd(TOTAL_INODES_RESERVED)	Number of i nodes reserved for superusers	--	double	No	AIX, Linux	TOTAL_INODES_FREE- TOTAL_INODES_AVAILABLE
Total Inodes Rsvd % (TOTAL_INODES_RESERVED_PERCENT)	Percentage of i nodes reserved for superusers (%)	--	float	No	AIX, Linux	(TOTAL_INODES_RESERVED ÷ TOTAL_NUMBER_OF_INODES) × 100
Total Inodes in Use(TOTAL_INODES_IN_USE)	Number of i nodes used	--	double	No	--	TOTAL_NUMBER_OF_INODES - TOTAL_INODES_FREE
Total Inodes in Use % (TOTAL_INODES_IN_USE_PERCENT)	Percentage of i nodes used (%)	--	float	No	--	(TOTAL_INODES_IN_USE ÷ (TOTAL_NUMBER_OF_INODES - TOTAL_INODES_RESERVED)) × 100
Total Size Blocks(TOTAL_SIZE_IN_BLOCKS)	Number of logical blocks in a file system	--	double	No	--	--
Total Size Mbytes(TOTAL_SIZE_IN_MBYTES)	File system size (MB)	--	double	No	--	(Fundamental Block Count × Fundamental Block Size) ÷ 1MB

Legend:

Smry: Summary

Not sprtd in: Not supported in

File System Detail - Remote (PD_FSR)

Function

The File System Detail - Remote (PD_FSR) record stores the performance data showing the status of the remote file system capacity at a given point in time. A record is created for each remote file system. This is a multi-instance record.

Notes:

- In HP-UX, Solaris, and Linux, the file system area for a general user is defined as the sum of the file system area used and the file system area that is available. The file system area for a superuser also includes a reserved file system area. In AIX, because there is no reserved area, there is no difference between the file system area for a general user and that for a superuser.
- In specialized file systems, it may not be possible to collect performance data. For example, a remote file system whose remote host name cannot be acquired as a virtual local file system may be treated as a local file system.
- The *Unsupported* column in the File System Detail - Remote (PD_FSR) record indicates the platform on the remote file system side.
- The basic block size (FFBS: Fundamental Filesystem Block Size) used in the field table of this record is the allocation unit used during file system creation. The number of blocks that corresponds to the basic block size is called the *basic block count*. The logical block size (PFBS: Preferred Filesystem Block Size) is the allocation unit used during input/output operations. The number of blocks that corresponds to the logical block size is called the *logical block count*.
- In PFM - Agent for Platform 09-00 or later, IPv6 is supported. The following table lists the fields in which numeric values are affected in an environment that contains both IPv4 and IPv6.

PFM-View name	Obtained IP information		
	IPv4 environment	IPv6 environment	Coexisting environment
IP Address	IP address resolved from the host name		

- When information is collected into the PD_FSR record, a system call is issued at the data collection interval. Because collecting information using the system call involves file system access, an attempt to execute the umount command to unmount the file system during information collection might fail. If the execution of the command fails, re-execute the umount command at a time different from the collection interval.
- If the NFS File System property is set to No, no information will be collected for the File System Detail - Remote (PD_FSR) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	3600	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- `PD_FSR_FILESYSTEM_NAME`
- `PD_FSR_IP_ADDRESS`

Lifetime

From the mounting of a file system until its unmounting

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,702 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Available Blocks(TOTAL_BLOCKS_AVAILABLE)	Number of logical blocks available to general users	--	double	No	--	--
Available Space % (TOTAL_MBYTES_AVAILABLE_PERCENT)	Percentage of size (MB) available to general users (%)	--	float	No	--	(TOTAL_MBYTES_AVAILABLE ÷ (TOTAL_SIZE_IN_MBYTES - TOTAL_MBYTES_RESERVED)) × 100
Available Space Mbytes(TOTAL_MBYTES_AVAILABLE)	Size available to general users (MB)	--	double	No	--	(Fundamental Block Count available to the general user × Fundamental Block Size) ÷ 1MB
Block Size(BLOCKSIZE)	Logical block size (bytes)	--	double	No	--	--
Blocks Free(TOTAL_BLOCKS_FREE)	Number of free logical blocks	--	double	No	--	--
Blocks in Use(TOTAL_BLOCKS_IN_USE)	Number of logical blocks used	--	double	No	--	--
File System(FILESYSTEM_NAME)	File system mounting point	--	string(1024)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
File System Type(FILESYSTEM_TYPE)	File system type (e.g., NFS)	--	string(20)	No	--	--
Host(HOSTNAME)	Name of the host where the file system is located	--	string(256)	No	--	--
IP Address(IP_ADDRESS)	IP address of the host where the file system is located	--	string(50)	No	--	--
Interval(INTERVAL)	Interval at which the File System Detail - Remote (PD_FSR) record was stored (seconds)	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Mbytes Free(TOTAL_MBYTES_FREE)	Size not used (MB)	--	double	No	--	(TOTAL_BLOCKS_FREE × Fundamental Block Size) ÷ 1MB
Mbytes Free %(TOTAL_MBYTES_FREE_PERCENT)	Percentage of size (MB) not used (%)	--	float	No	--	(TOTAL_MBYTES_FREE ÷ TOTAL_SIZE_IN_MBYTES) × 100
Mbytes Rsvd(TOTAL_MBYTES_RESERVED)	Size reserved for superusers (MB)	--	double	No	AIX	TOTAL_MBYTES_FREE-TOTAL_MBYTES_AVAILABLE
Mbytes Rsvd %(TOTAL_MBYTES_RESERVED_PERCENT)	Percentage of size (MB) reserved for superusers (%)	--	float	No	AIX	(TOTAL_MBYTES_RESERVED ÷ TOTAL_SIZE_IN_MBYTES) × 100
Mbytes in Use(TOTAL_MBYTES_IN_USE)	Size used by general users (MB)	--	double	No	--	((TOTAL_SIZE_IN_BLOCKS - TOTAL_BLOCKS_FREE) × Fundamental Block Size) ÷ 1MB
Mbytes in Use %(TOTAL_MBYTES_IN_USE_PERCENT)	Percentage of size (MB) used by general users (%)	--	float	No	--	(TOTAL_MBYTES_IN_USE ÷ (TOTAL_SIZE_IN_MBYTES - TOTAL_MBYTES_RESERVED)) × 100
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always FSR.	--	char(8)	No	--	--
Remote File System(REMOTE_FILESYSTEM_NAME)	Remote file system name	--	string(256)	No	--	--
Total Size Blocks(TOTAL_SIZE_IN_BLOCKS)	Number of logical blocks in a file system	--	double	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Size Mbytes(TOTAL_SIZE_IN_MBYTES)	File system size (MB)	--	double	No	--	(Fundamental Block Count × Fundamental Block Size) ÷ 1MB

Legend:

Smry: Summary

Not sprtd in: Not supported in

Logged Messages (PL_MESS)

Function

The Logged Messages (PL_MESS) record stores the messages that are written into the log file specified by the Messages File property or the event file. A record is created for each line of message that is written into the log file. This is a multi-instance record.

For details about the Messages File property, see *F.2 Agent Collector service properties*. For details about event files, see *5. Collecting Log Information*.

Notes:

- This record cannot be used in Linux.
- The only log files monitored by this record are files that monotonously increase. The increase that has occurred since the start of log file monitoring is collected as data. Therefore, if there has been no increase, no data is collected.
- Specify the absolute path of the log file for the monitoring target in the Messages File property or the event file. If log files are specified in both the Messages File property and the event file, both log files are monitored. If the log file specified in the Messages File property is invalid, this causes an OS error when PFM - Agent for Platform is started, and the KAVF10150-W message is displayed. If the log file specified in the event file is invalid, the KAVF10203-W message is displayed. If either of these messages is output, change the log file and restart PFM - Agent for Platform.
- If the same path and the same identifier (id) are specified multiple times inside the event file, the messages are summarized into a single record. However, if the same path and the same identifier (id) are specified in the Messages File property and the event file, separate records are created.
- If a space is specified for the log file, no log file is monitored.
- When this record is displayed as a real-time report, the display changes as follows.

  • If **Indicate delta value** is not selected, all message texts that have been added since the service startup are displayed.

  • If **Indicate delta value** is selected, only the message texts that were added during the display interval are displayed.
- This record does not support collection in a logical host environment. Collect this record in a physical host environment.
- PL_MESS records cannot be used to collect log file information for logfile items in the /opt/jp1pc/agtu/agent/evfile file that contain an asterisk (*).

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

PL_MESS_MESSAGE_TEXT

Lifetime

None

Record size

- Fixed portion: 681 bytes
- Variable portion: 512 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval at which the Logged Messages (PL_MESS) record was stored (seconds). Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Message Text(MESSAGE_TEXT)	Message text. Consists of a single line of information taken from the log file.	--	string(512)	No	Linux	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	Linux	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always MESS.	--	char(8)	No	Linux	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Network Interface Detail (PI_NIND)

Function

The Network Interface Detail (PI_NIND) record stores the performance data per unit time on a network interface. The available network interfaces include a loopback adapter, a local Token Ring adapter, and an Ethernet adapter, multiple network interfaces can be installed on a single machine. Each interface can have multiple IP addresses. Every time performance data is collected, a record is created for each network interface, but only the first IP address that is found is used. This is a multi-instance record.

Notes:

- The interface flags that are set in the Flags (FLAGS) field and that can be judged are listed below:

  • In AIX

  UP, BROADCAST, DEBUG, LOOPBACK, POINTOPOINT, NOTRAILERS, RUNNING, NOARP, PROMISC, ALLMULTI, OACTIVE, SIMPLEX, MULTICAST

  • In HP-UX

  UP, LOOPBACK

  • In Solaris

  UP, BROADCAST, DEBUG, LOOPBACK, POINTOPOINT, NOTRAILERS, RUNNING, NOARP, PROMISC, ALLMULTI, INTELLIGENT, MULTICAST, MULTI_BCAST, UNNUMBERED, PRIVATE

  • In Linux

  UP, BROADCAST, DEBUG, LOOPBACK, POINTOPOINT, NOTRAILERS, RUNNING, NOARP, PROMISC, ALLMULTI, MULTICAST, MASTER, SLAVE, PORTSEL, AUTOMEDIA, DYNAMIC

- If the DR function of Solaris is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification.

System resource type	Report type	Actions to be taken after modification
LAN board	Real-time	Display the report again.
	Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

  Fields other than those listed below are affected by a system resource modification:

  • Interval (INTERVAL)

  • Record Time (RECORD_TIME)

  • Record Type (INPUT_RECORD_TYPE)

  • Fields that are added only when data is recorded in the Store database

  For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*

- In Linux, only performance data on the network interface for which either an IPv4 address or an IPv6 address has been set is collected.
- This record cannot be used in Solaris shared-IP non-global zones, but can be used in exclusive-IP non-global zones. Note that no reports can be displayed in shared-IP non-global zones.
- In PFM - Agent for Platform 09-00 or later, IPv6 is supported. The following table shows the fields whose values are affected in an environment that contains both IPv4 and IPv6.

PFM-View name	OS environment	Obtained IP information		
		IPv4 environment	IPv6 environment	Coexisting environment
Broadcast Address	All	IPv4	Blank	IPv4
Flags	HP-UX, AIX	--	--	--
	Solaris, Linux	IPv4	IPv6	IPv4
IP Address	All	IPv4	Blank	IPv4
IP6 Address	All	Blank	IPv6	IPv6
Interface	All	IPv4	IPv6	IPv4
Interval	All	--	--	--
Max Transmission Unit	HP-UX, Solaris, Linux	IPv4	IPv6	IPv4
	AIX	--	--	--
Network Mask	All	IPv4	Blank	IPv4
Network Name	All	IPv4	Blank	IPv4
Pkt Collisions	HP-UX	IPv4	0	IPv4
	AIX, Solaris, Linux	IPv4	IPv6	Sum of the values for IPv4 and IPv6
Pkt Receive Errors	HP-UX	IPv4	0	IPv4
	AIX, Solaris, Linux	IPv4	IPv6	Sum of the values for IPv4 and IPv6
Pkt Xmit Errors	HP-UX	IPv4	0	IPv4
	AIX, Solaris, Linux	IPv4	IPv6	Sum of the values for IPv4 and IPv6
Pkts Rcvd	All	IPv4	IPv6	Sum of the values for IPv4 and IPv6
Pkts Rcvd/sec	All	IPv4/Interval	IPv6/Interval	(Sum of the values for IPv4 and IPv6)/ Interval
Pkts Xmitd	All	IPv4	IPv6	Sum of the values for IPv4 and IPv6
Pkts Xmitd/sec	All	IPv4/Interval	IPv6/Interval	(Sum of the values for IPv4 and IPv6)/ Interval
Record Time	All	--	--	--
Record Type	All	--	--	--
Total Pkt Errors	HP-UX	Sum of the values of `Pkt Receive Errors`, `Pkt Xmit Errors`, and `Pkt Collisions` in IPv4	0	Sum of the values of `Pkt Receive Errors`, `Pkt Xmit Errors`, and `Pkt Collisions` in IPv4
	AIX, Solaris, Linux	Sum of the values of `Pkt Receive`	Sum of the values of `Pkt Receive`	Sum of the values of `Pkt Receive`

PFM-View name	OS environment	Obtained IP information		
		IPv4 environment	IPv6 environment	Coexisting environment
Total Pkt Errors	AIX, Solaris, Linux	Errors, Pkt Xmit Errors, and Pkt Collisions for IPv4	Errors, Pkt Xmit Errors, and Pkt Collisions for IPV6	Errors, Pkt Xmit Errors, and Pkt Collisions for IPV4 and IPV6
Total Pkts	All	Sum of the values of pkts received and Pkts Xmitd for IPv4	Sum of the values of pkts received and Pkts Xmitd for IPv6	Sum of the values of pkts received and Pkts Xmitd for IPv4 and IPv6
Total Xmit Receive Errors	HP-UX	Sum of the values of Pkt Receive Errors and Pkt Xmit Errors for IPv4	0	Sum of the values of Pkt Receive Errors and Pkt Xmit Errors for IPv4
	AIX, Solaris, Linux	Sum of the values of Pkt Receive Errors and Pkt Xmit Errors for IPv4	Sum of the values of Pkt Receive Errors and Pkt Xmit Errors for IPv6	Sum of the values of Pkt Receive Errors and Pkt Xmit Errors for IPv4 and IPv6
Type	All	--	--	--

Legend:

--: Information that is not affected by the IPv4 or IPv6 environment is collected.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_NIND_INTERFACE_NAME
- PI_NIND_IP_ADDRESS

- PI_NIND_IP6_ADDRESS

Lifetime

None

Record size

- Fixed portion: 681 bytes
- Variable portion: 670 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Broadcast Address(BROADCAST_ADDRESS)	The first broadcast address that was found. Displayed in the Internet address format. In HP-UX and Solaris, the broadcast address of a loopback instance is not displayed.	COPY	string(20)	No	--	--
Flags(FLAGS)	Interface flag setting. If it is 80 bytes or longer, the last character becomes >. In HP-UX, all flags that indicate interface statuses are included in the value in this field (the ifconfig command displays only some of the flags).	COPY	string(80)	No	--	--
IP Address(IP_ADDRESS)	IP address of IPv4. Displayed in the Internet address format.	COPY	string(20)	No	--	--
IP6 Address(IP6_ADDRESS)	IP address of IPv6. Displayed in the Internet address format.	COPY	string(50)	No	--	--
Interface(INTERFACE_NAME)	Interface name	COPY	string(40)	No	--	--
Interval(INTERVAL)	Interval at which the Network Interface Detail (PI_NIND) record was stored (seconds)	COPY	ulong	Yes	--	• When delta values are not checked in real time, RECORD_TIME - *Last-boot-time* • Other cases, RECORD_TIME - *previous-record-time*
Max Transmission Unit(MAX_TRANSMISSION_UNIT)	Maximum packet size (bytes)	COPY	ulong	No	--	--
Network Mask(NETWORK_MASK)	Subnet mask of the first IP address that was found. Displayed in the Internet address format.	COPY	string(20)	No	--	--
Network Name(NETWORK_NAME)	Network name. A network name that is 1,028 bytes or longer cannot be handled. The network name is displayed up to the size defined in the data model. In AIX	COPY	string(256)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Network Name(NETWORK_NAME)	and HP-UX, if NIS is active and there is no entry in the NIS database for network addresses, the first IP address of the interface that is found is masked and the result is displayed.	COPY	string(256)	No	--	--
Pkt Collisions(PACKET_COLLISIONS)	Number of packet collisions that occurred	AVG	ulong	Yes	--	--
Pkt Receive Errors(PACKET_RECEIVE_ERRORS)	Number of packet receive errors	AVG	ulong	Yes	--	--
Pkt Xmit Errors(PACKET_TRANSMIT_ERRORS)	Number of packet transmission errors	AVG	ulong	Yes	--	--
Pkts Rcvd(PACKETS_RECEIVED)	Number of packets received	AVG	ulong	Yes	--	--
Pkts Rcvd/sec(PACKETS_RECEIVED_PER_SECOND)	Frequency at which packets were received (number of packets per second)	R	float	Yes	--	PACKETS_RECEIVED ÷ INTERVAL
Pkts Xmitd(PACKETS_TRANSMITTED)	Number of packets transmitted	AVG	ulong	Yes	--	--
Pkts Xmitd/sec(PACKETS_TRANSMITTED_PER_SECOND)	Frequency at which packets were transmitted (number of packets per second)	R	float	Yes	--	PACKETS_TRANSMITTED ÷ INTERVAL
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always NIND.	COPY	char(8)	No	--	--
Total Pkt Errors(TOTAL_PACKET_ERRORS)	Number of packet transmission and receiving errors (including the number of packet collisions)	AVG	ulong	Yes	--	PACKET_TRANSMIT_ERRORS + PACKET_RECEIVE_ERRORS + PACKET_COLLISIONS
Total Pkts(TOTAL_PACKETS)	Number of packets transmitted and received	AVG	ulong	Yes	--	PACKETS_RECEIVED + PACKETS_TRANSMITTED
Total Xmit Receive Errors(TOTAL_TRANS_RECEIVE_ERRORS)	Number of packet transmission and receiving errors (excluding the number of packet collisions)	AVG	ulong	Yes	--	PACKET_TRANSMIT_ERRORS + PACKET_RECEIVE_ERRORS
Type(INTERFACE_TYPE)	Interface type	COPY	string(20)	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Network Interface Summary (PI_NINS)

Function

The Network Interface Summary (PI_NINS) record stores the performance data in which the data stored in the Network Interface Detail (PI_NIND) record is summarized per unit time.

The available network interfaces include a loopback adapter, a local Token Ring adapter, and an Ethernet adapter; multiple network interfaces can be installed on a single machine. Each interface can have multiple IP addresses. Every time performance data is collected, a record is created for each network interface, but only the first IP address that is found is used. Even when the Network Interface Detail (PI_NIND) record is not collected, performance data is stored in the Network Interface Summary (PI_NINS) record.

Notes:

- If the DR function of Solaris is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification.

System resource type	Report type	Actions to be taken after modification
LAN board	Real-time	Display the report again.
	Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

  Fields other than those listed below are affected by a system resource modification:

  • Interval (INTERVAL)

  • Record Time (RECORD_TIME)

  • Record Type (INPUT_RECORD_TYPE)

  • Fields that are added only when data is recorded in the Store database

  For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*.

- This record cannot be used in Solaris shared-IP non-global zones, but can be used in exclusive-IP non-global zones. Although reports can be displayed in shared-IP non-global zones, 0 is displayed for all fields other than the Interval, Record Time, and Record Type fields.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

None

Lifetime

None

Record size

- Fixed portion: 797 bytes
- Variable portion: 0 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interface Count(NUMBER_OF_INTERFACES)	Interface count	COPY	ulong	No	--	Number of Network Interface Detail (PI_NIND) records summarized in Network Interface Summary (PI_NINS) record.
Interval(INTERVAL)	Interval at which the Network Interface Summary (PI_NINS) record was stored (seconds)	COPY	ulong	Yes	--	• When delta values are not checked in real time, `RECORD_TIME` - *last-boot-time* • Other cases, `RECORD_TIME` - *previous-record-time*
Pkt Collisions(PACKET_COLLISIONS)	Number of packet collisions that occurred	AVG	ulong	Yes	--	--
Pkt Receive Errors(PACKET_RECEIVE_ERRORS)	Number of packet receive errors	AVG	ulong	Yes	--	--
Pkt Xmit Errors(PACKET_TRANSMIT_ERRORS)	Number of packet transmission errors	AVG	ulong	Yes	--	--
Pkts Rcvd(TOTAL_PACKETS_RECEIVED)	Number of packets received	AVG	ulong	Yes	--	--
Pkts Rcvd/sec(PACKETS_RECEIVED_PER_SECOND)	Frequency at which packets were received (number of packets per second)	R	float	Yes	--	TOTAL_PACKETS_RECEIVED ÷ INTERVAL

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Pkts Xmitd(PACKETS_TRANSMITTED)	Number of packets transmitted	AVG	ulong	Yes	--	--
Pkts Xmitd/sec(PACKETS_TRANSMITTED_PER_SECOND)	Frequency at which packets were transmitted (number of packets per second)	R	float	Yes	--	PACKETS_TRANSMITTED ÷ INTERVAL
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always NINS.	COPY	char(8)	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Process Detail (PD)

Function

The Process Detail (PD) record stores the Performance data showing the status of a process at a given point in time. A record is created for each process ID within the system.

The performance data stored in this record is the same as that stored in the Process Detail Interval (PD_PDI) record. However, if performance data collected at a point in time is the same as the performance data collected in the previous interval for the same process, performance data is stored in the same record, overwriting the data in the database. This point is different from the Process Detail Interval (PD_PDI) record. After performance data is collected at a given point in time, if the process is terminated before the next performance data is collected, no performance data is collected on that process. Furthermore, no performance data is collected on processes that are not being executed.

When this record is collected on a real-time basis, performance data on all processes that are being executed by the system is stored. This is a multi-instance record.

Notes:

- If a process has no terminal name, ?? is displayed in the Terminal (TERMINAL_NAME) field.
- In AIX and Solaris, only the data in the fields listed below is collected on defunct processes. For all other fields, either 0 or n/a is displayed.

  Accounting Flags, Argument List, Child Process List, Exit Status, Flags, Group, Group ID, Parent PID, Parent Process List, PID, Process Group ID, Program, Program/PID, Real Group, Real Group ID, Real User, Real User ID, Record Time, Record Type, Scheduler Class, Snapshot Time, Start Time, Terminal, User, User ID, Virtual Env ID (Solaris or AIX)
- In the Priority (PRIORITY) field, process priority is determined based on the number. In Solaris, the greater the number, the higher the priority. In AIX and Linux, the smaller the number, the higher the priority. In HP-UX, the number that indicates priority differs with each process setting. You can use the following command to check the process setting:

  ```
  export UNIX95=1
  ps -eo pid,cls,pri
  ```

  When cls is one of the following, the smaller the number, the higher the priority:

  FIFO

  RR

  RR2

  When cls is one of the following, the greater the number, the higher the priority:

  HPUX

  RT
- The process flag lists that are set in the Flags (FLAGS) field and that can be judged are listed below.

  • In AIX

  SLOAD, SNOSWAP, SFORKSTACK, STRC, SFIXPRI, SKPROC, SSIGNOCHLD, SSIGSET, SLKDONE, STRACING, SMPTRACE, SEXIT, SORPHANPGRP, SNOCNTLPROC, SPPNOCLDSTOP, SEXECED, SJOBSESS, SJOBOFF, SEXECING, SPSEARLYALLOC

  • In HP-UX

  SLOAD, SSYS, SLOCK, STRC, SWTED_PARENT, SDEACTSELF, SOMASK, SWEXIT, SVFORK, SSEQL, SUANOM, SOUSIG, SOWEUPC, SSEL, SWANTS_ALLCPU, SSERIAL, SFAULTING, SDEACT, SWAITIO, STRAPPING, SWTED_DEBUGGER

  • In Solaris

SSYS, SITBUSY, SWATCHOK, SKILLED, SSCONT, SZONETOP, SEXTKILLED, SUGID, SEXECED, SJCTL, SNOWAIT, SVFORK, SEXITLWPS, SCOREDUMP, SMSACCT, SLWPWRAP

- Performance data on a terminated process cannot be collected.
- If the target process of an instance is a zombie process, the following are displayed for the Group, Group ID, Real Group, Real Group ID, User, and User ID fields in AIX and HP-UX:

OS	Field	Content displayed
AIX	Group	Name of group with ID of 0
	Real Group	
	Group ID	0
	Real Group ID	
HP-UX	Group	N/A gid = -3
	Real Group	
	User	N/A uid = -3
	Group ID	-3
	Real Group ID	
	User ID	

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PD_START_TIME
- PD_PID

Lifetime

From process execution until termination

Record size

- Fixed portion: 681 bytes
- Variable portion: 2,747 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Accounting Flags(ACCOUNTING_FLAGS)	Process operation history information. Valid only when the process is terminated (End Time is not n/a).	--	string(256)	No	HP-UX, Solaris, AIX, Linux	--
Argument List(ARGUMENT_LIST)	Process argument list. Corresponds to the character string displayed in the COMMAND string of the ps -ef command. In AIX, the maximum size is limited to 119 bytes. In Linux, the maximum size is limited to 119 bytes. In HP-UX, the maximum size is limited to 63 bytes. In Solaris, the maximum size is limited to 79 bytes.	--	string(120)	No	--	--
Avg I/O Kbytes(AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ TOTAL_IO_OPS
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) ÷ processors-count) × 100
CPU % Each(CPU_PERCENT_EACH)	Process CPU usage (%)	--	float	No	--	((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ ELAPSED_TIME) × 100
CPU Limit(CPU_LIMIT)	Process CPU limit. This value is defined by the RLIMIT_CPU parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Child Process List(CHILD_PROCESS_LIST)	Created child process list. The individual child processes are separated by blank spaces. If the list exceeds 128 bytes, the character becomes >.	--	string(128)	No	--	Traverse a list of processes using PID
Context Switches(CONTEXT_SWITCHES)	Number of times context switching was executed	--	ulong	Yes	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Core Size Limit(CORE_SIZE_LIMIT)	Maximum size of the core file that can be created by a process (bytes). This value is defined by the RLIMIT_CORE parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Data Size Limit(DATA_SIZE_LIMIT)	Process data size limit (bytes). This value is defined by the RLIMIT_DATA parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Elapsed Time(ELAPSED_TIME)	Time elapsed since a process was started until the Process Detail (PD) record is stored or the process is terminated (seconds)	--	utime	No	--	• When the process is executing RECORD_TIME - START_TIME • Other cases --
End Time(END_TIME)	Process termination time. n/a if the process is being executed.	--	time_t	No	HP-UX, Solaris, AIX, Linux	START_TIME + ELAPSED_TIME

Executable Data Kbytes(EXECUTABLE_DATA_KBYTES)	Size of data being used (KB).
	To collect data for this field in Solaris, the environment variable SAUNIXPMAP must be specified in advance. Execute the following command from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

Executable Text Kbytes(EXECUTABLE_TEXT_KBYTES)	Size of the text being used (KB).
	To collect data for this field in Solaris, the environment variable SAUNIXPMAP must be specified in advance. Execute the following command from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

Exit Status(EXIT_STATUS)	Process termination status. Valid only when the process is terminated (End Time is not n/a).	--	string(20)	No	HP-UX, Solaris, AIX, Linux	--
File Size Limit(FILE_SIZE_LIMIT)	Process file size limit (bytes). This value is defined by the RLIMIT_FSIZE parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Flags(FLAGS)	Process flag list. If the list exceeds 120 bytes, the character becomes >.	--	string(120)	No	Linux	--
Group(GROUP_NAME)	Effective group name of the process	--	string(256)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Group ID(GROUP_ID)	Effective group ID of the process	--	long	No	--	--
Hard CPU Limit(HARD_CPU_LIMIT)	Process hard CPU limit (Hcpulimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Core Size Limit(HARD_CORE_SIZE_LIMIT)	Process hard core size limit (bytes) (Hcorelimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Data Size Limit(HARD_DATA_SIZE_LIMIT)	Process hard data size limit (bytes) (Hdatalimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard File Size Limit(HARD_FILE_SIZE_LIMIT)	Process hard file size limit (bytes) (Hfilesizelimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Open Files Limit(HARD_OPEN_FILES_LIMIT)	Process hard open file limit (Hopenfileslimit)	--	double	No	HP-UX, Solaris, AIX, Linux	--
Hard Stack Size Limit(HARD_STACK_SIZE_LIMIT)	Process hard stack size limit (bytes) (Hstacklimit)	--	double	No	HP-UX, Solaris, Linux	--
Hard Virtual Mem Size Limit(HARD_VIRTUAL_MEMORY_SIZE_LIMIT)	Process hard virtual memory size limit (bytes) (Hrsslimit)	--	double	No	HP-UX, Solaris, Linux	--
Heap Kbytes(HEAP_KBYTES)	Heap size used (KB)	--	ulong	No	HP-UX, AIX, Linux	--
Interval(INTERVAL)	Interval at which the Process Detail (PD) record was stored (seconds)	--	ulong	Yes	HP-UX, Solaris, AIX, Linux	--
Major Faults(MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	--	ulong	Yes	--	--
Mem Charge(MEMORY_CHARGE)	Process service unit charge. Valid only when the process is terminated (End Time is not n/a).	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Minor Faults(MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	--	ulong	Yes	--	--
Niceness(NICENESS)	Process niceness value. The scheduler class assigned to the process affects this value. For example, when a task is assigned to the scheduler class RT, the niceness value is 0.	--	long	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Open Files Limit(OPEN_FILES_LIMIT)	Process open file limit. This value is defined by the RLIMIT_NOFILE parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, AIX, Linux	--
PID(PID)	Process ID	--	long	No	--	--
Parent PID(PARENT_PID)	Process ID of a parent process	--	long	No	--	--
Parent Process List(PARENT_PROCESS_LIST)	Parent process list of a process. For example, 3867<1<0 means process 0 created process 1, and process 1 created process 3867. If the list exceeds 128 bytes, the character becomes >.	--	string(128)	No	--	Traverse a list of processes using Parent PID
Priority(PRIORITY)	Process priority	--	long	No	--	--
Process Group ID(PROCESS_GROUP_ID)	Process group ID	--	long	No	--	--
Program(PROGRAM_NAME)	Program Name. Correspond to the character string displayed in the COMMAND string of the ps -e command. In AIX, the maximum size is limited to 32 bytes. In Linux, the maximum size is limited to 15 bytes. In HP-UX, the maximum size is limited to 14 bytes. In Solaris, the maximum size is limited to 15 bytes.	--	string(256)	No	--	--
Program/PID(PROGRAM_PID)	Name of the program executed and a process ID	--	string(280)	No	--	--
Reads(READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	--	ulong	Yes	HP-UX, Linux	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	--	float	Yes	HP-UX, Linux	READ_OPS ÷ ELAPSED_TIME
Real Group(REAL_GROUP_NAME)	Real group name of the process	--	string(256)	No	--	--
Real Group ID(REAL_GROUP_ID)	Real group ID of the process	--	long	No	--	--
Real Mem Kbytes(REAL_MEMORY_KBYTES)	Physical memory size used (KB)	--	ulong	No	--	--
Real User(REAL_USER_NAME)	Real user name of the process	--	string(256)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Real User ID(REAL_USER_ID)	Real user ID of the process	--	long	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always PD.	--	char(8)	No	--	--
Scheduler Class(SCHEDULER_CLASS)	Scheduling class name. One of the following values is set: TS, IA, FX, FSS, RT, SYS	--	string(12)	No	HP-UX, AIX, Linux	--
Shared Lib Kbytes(SHARED_LIBRARY_KBYTES)	Size of the shared library used (KB)	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Shared Mem Kbytes(SHARED_MEMORY_KBYTES)	Size of the shared memory used (KB)	--	ulong	No	Solaris, AIX	--
Signals Rcvd(NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	Yes	Linux	--
Snapshot Time(LAST_SNAPSHOT_TIME)	Time at which the last process table information was acquired. Shows the record time if the process table contains no information.	--	time_t	No	--	--
Stack Kbytes(STACK_KBYTES)	Size of the stack being used by the process (KB)	--	ulong	No	AIX	--
Stack Size Limit(STACK_SIZE_LIMIT)	Process stack size limit (bytes). This value is defined by the RLIMIT_STACK parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--
Start Time(START_TIME)	Process start time	--	time_t	No	--	--
State(STATE)	Process state. One of the following values is set: - In AIX IDLE, ZOMBIE, STOP, RUN, SWAP, NONE - In HP-UX IDLE, OTHER, RUN, SLEEP, STOP, ZOMBIE, NONE - In Solaris IDLE, ONCPU, RUN, SLEEP, STOP, ZOMBIE, NONE - In Linux IDLE, RUN, SLEEP, SWAP, STOP, ZOMBIE, NONE	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Swaps(SWAPS)	Number of times swapping occurred	--	ulong	Yes	Linux	--
System CPU(SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode (seconds)	--	utime	Yes	--	--
Terminal(TERMINAL_NAME)	Name of the executing terminal. Process without terminal name, ?? is displayed.	--	string(40)	No	--	--
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	Yes	AIX, Linux	TOTAL_IO_KBYTES ÷ ELAPSED_TIME
Total I/O Kbytes(TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	--	float	Yes	AIX, Linux	--
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	--	ulong	Yes	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec(TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	--	float	Yes	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ ELAPSED_TIME
Total Process Kbytes(TOTAL_PROCESS_KBYTES)	Process size (KB). In AIX, the value in this field is the same as the value displayed in the SZ string obtained by the ps -l command. In HP-UX and Solaris, the value in this field is the same as the value displayed in the SZ string obtained by the ps -l command, multiplied by the page size. In Linux, the same as the value obtained from the first column of `/proc/<PID>/statm` * *page-size* ÷ 1,024.	--	ulong	No	--	--
User(USER_NAME)	Effective user name of the process	--	string(256)	No	--	--
User CPU(USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	--	utime	Yes	--	--
User ID(USER_ID)	Effective user ID of the process	--	long	No	--	--
Virtual Env ID(VIRTUAL_ENV_ID)	ID of the virtual environment made by the virtual system provided by the OS.	--	string(65)	No	HP-UX	--
Virtual Mem Kbytes(VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (KB)	--	ulong	No	Solaris	--
Virtual Mem Size Limit(VIRTUAL_MEMORY_SIZE_LIMIT)	Size limit of the process' virtual memory (bytes) (rsslimit). This value is defined by the RLIMIT_VMEM parameter of the setrlimit system call.	--	double	No	HP-UX, Solaris, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Writes(WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	--	ulong	Yes	HP-UX, Linux	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	--	float	Yes	HP-UX, Linux	WRITE_OPS ÷ ELAPSED_TIME

Legend:

Smry: Summary

Not sprtd in: Not supported in

Process Detail Interval (PD_PDI)

Function

The Process Detail Interval (PD_PDI) record stores the performance data per unit time on a process. A record is created for each process ID within the system.

The performance data stored in this record is the same as that stored in the Process Detail (PD) record. However, if performance data collected at a point in time is the same as the performance data collected in the previous interval for the same process, performance data is not stored in the same record, and a new record is created. This point is different from the Process Detail (PD) record. Therefore, multiple records are created for the same process. After performance data is collected at a given point in time, if the process is terminated before the next performance data is collected, no performance data is collected on that process.

When this record is collected on a real-time basis, performance data on all processes that are being executed by the system is stored. This is a multi-instance record.

Notes:

- In AIX and Solaris, only the data in the fields listed below is collected on defunct processes. For all other fields, either 0 or n/a is displayed.

  Accounting Flags, Argument List, End Time, Exit Status, Group, Group ID, Parent PID, PID, Program, Real User, Real User ID, Record Time, Record Type, Snapshot Time, Start Time, State, Terminal, User, User ID
- The Interval (INTERVAL) field of this record is always 0. In the fields that show frequency, average, and percentage, the time elapsed since the process startup is used as the data source. To collect a delta value, the difference between the previous elapsed process time and the current elapsed process time is computed. If no delta value is to be collected, the cumulative elapsed time of the process is used for computation.
- In the Priority (PRIORITY) field, process priority is determined based on the size of the number. In Solaris, the greater the number, the higher the priority. In AIX and Linux, the smaller the number, the higher the priority. In HP-UX, the number that indicates priority differs with each process setting. You can use the following command to check the process setting:

  ```
  export UNIX95=1
  ps -eo pid,cls,pri
  ```

  When cls is one of the following, the smaller the number, the higher the priority:

  FIFO

  RR

  RR2

  When cls is one of the following, the greater the number, the higher the priority:

  HPUX

  RT
- Performance data on a terminated process cannot be collected.
- If the target process of an instance is a zombie process, the following are displayed for the Group, Group ID, User, and User ID fields in AIX and HP-UX:

OS	Field	Content displayed
AIX	Group	Name of group with ID of 0
	Group ID	0
HP-UX	Group	N/A gid = -3

OS	Field	Content displayed
HP-UX	User	N/A uid = -3
	Group ID	-3
	User ID	

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Interval Records, PI	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

- PD_PDI_PID
- PD_PDI_START_TIME

Lifetime

From process execution until termination

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,626 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Accounting Flags(ACCOUNTING_FLAGS)	Process operation history information. Valid only when the process is terminated (End Time is not n/a).	--	string(256)	No	HP-UX, Solaris, AIX, Linux	--
Argument List(ARGUMENT_LIST)	Process argument list. Correspond to the character string displayed in the COMMAND string of the ps -ef command. In AIX, the maximum size is limited to 119 bytes. In Linux, the maximum size is limited to 119 bytes. In HP-UX, the maximum size is limited to 63 bytes. In Solaris, the maximum size is limited to 79 bytes.	--	string(120)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg I/O Kbytes(AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ TOTAL_IO_OPS
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	--	float	No	--	Non-delta calculation or collection first time, (((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ Process Cumulative Elapsed time) ÷ Processor Count) × 100. Others, (((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ Interval) ÷ Processor Count) × 100
CPU % Each(CPU_PERCENT_EACH)	Process CPU usage (%)	--	float	No	--	Non-delta calculation or collection first time, ((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ Process Cumulative Elapsed time) × 100. Others, ((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ Interval) × 100
Context Switches(CONTEXT_SWITCHES)	Number of times context switching was executed	--	ulong	Yes	HP-UX, Linux	--
End Time(END_TIME)	Process termination time. n/a if the process is being executed.	--	time_t	No	HP-UX, Solaris, AIX, Linux	START_TIME + Process Operation Time

Executable Data Kbytes(EXECUTABLE_DATA_KBYTES)	Size of data used (KB).
	To collect data for this field in Solaris, the environment variable SAUNIXPMAP must be specified in advance. Execute the following command from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

Executable Text Kbytes(EXECUTABLE_TEXT_KBYTES)	Size of the text used (KB).
	To collect data for this field in Solaris, the environment variable SAUNIXPMAP must be specified in advance. Execute the following command from the command line:
	`# SAUNIXPMAP=1`
	`# export SAUNIXPMAP`
		--	ulong	No	Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Exit Status(EXIT_STATUS)	Process termination status. Valid only when the process is terminated (End Time is not n/a).	--	string(20)	No	HP-UX, Solaris, AIX, Linux	--
Group(GROUP_NAME)	Effective group name of the process	--	string(256)	No	--	--
Group ID(GROUP_ID)	Effective group ID of the process	--	long	No	--	--
Heap Kbytes(HEAP_KBYTES)	Heap size used (KB)	--	ulong	No	HP-UX, AIX, Linux	--
Interval(INTERVAL)	Always 0.	--	ulong	Yes	HP-UX, Solaris, AIX, Linux	--
Major Faults(MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	--	ulong	Yes	--	--
Minor Faults(MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	--	ulong	Yes	--	--
Niceness(NICENESS)	Process niceness value. The scheduler class assigned to the process affects this value. For example, when a task is assigned to the scheduler class RT, the niceness value is 0.	--	long	No	--	--
PID(PID)	Process ID	--	long	No	--	--
Parent PID(PARENT_PID)	Process ID of a parent process	--	long	No	--	--
Priority(PRIORITY)	Process priority	--	long	No	--	--
Program(PROGRAM_NAME)	Program Name. Correspond to the character string displayed in the COMMAND string of the ps -e command. In AIX, the maximum size is limited to 32 bytes. In Linux, the maximum size is limited to 15 bytes. In HP-UX, the maximum size is limited to 14 bytes. In Solaris, the maximum size is limited to 15 bytes.	--	string(256)	No	--	--
Reads(READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	--	ulong	Yes	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	--	float	Yes	HP-UX, Linux	When delta value is not collected, READ_OPS ÷ when collecting Process cumulative elapsed time delta value, READ_OPS ÷ (Previous process elapsed time - Present process elapsed time)
Real Mem Kbytes(REAL_MEMORY_KBYTES)	Physical memory size used (KB)	--	ulong	No	--	--
Real User(REAL_USER_NAME)	Real user name of the process	--	string(256)	No	--	--
Real User ID(REAL_USER_ID)	Real user ID of the process	--	long	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always PDI.	--	char(8)	No	--	--
Shared Lib Kbytes(SHARED_LIBRARY_KBYTES)	Size of the shared library used (KB)	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Shared Mem Kbytes(SHARED_MEMORY_KBYTES)	Size of the shared memory used (KB)	--	ulong	No	Solaris, AIX	--
Signals Rcvd(NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	Yes	Linux	--
Snapshot Time(LAST_SNAPSHOT_TIME)	Time at which the last process table information was acquired. Shows the record time if the process table contains no information.	--	time_t	No	--	--
Stack Kbytes(STACK_KBYTES)	Size of the stack used by the process (KB)	--	ulong	No	AIX	--
Start Time(START_TIME)	Process start time	--	time_t	No	--	--
State(STATE)	Process state. One of the following values is set: - In AIX IDLE, ZOMBIE, STOP, RUN, SWAP, NONE - In HP-UX IDLE, OTHER, RUN, SLEEP, STOP, ZOMBIE, NONE	--	string(10)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
State(STATE)	- In Solaris IDLE, ONCPU, RUN, SLEEP, STOP, ZOMBIE, NONE - In Linux IDLE, RUN, SLEEP, SWAP, STOP, ZOMBIE, NONE	--	string(10)	No	--	--
Swaps(SWAPS)	Number of times swapping occurred	--	ulong	Yes	Linux	--
System CPU(SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode (seconds)	--	utime	Yes	--	--
Terminal(TERMINAL_NAME)	Name of the executing terminal. Process without terminal name, ?? is displayed.	--	string(40)	No	--	--
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	Yes	AIX, Linux	When delta value is not collected,TOTAL_IO_KBYTES ÷ when collecting Process cumulative elapsed time delta value, TOTAL_IO_KBYTES ÷ (Previous process elapsed time - Present process elapsed time)
Total I/O Kbytes(TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	--	float	Yes	AIX, Linux	--
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	--	ulong	Yes	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec(TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	--	float	Yes	HP-UX, AIX, Linux	When delta value is not collected, TOTAL_IO_OPS ÷ when collecting Process cumulative elapsed time delta value, TOTAL_IO_OPS ÷ (Previous process elapsed time - Present process elapsed time)
Total Process Kbytes(TOTAL_PROCESS_KBYTES)	Process size (KB). In AIX, the value in this field is the same as the value displayed in the SZ string obtained by the ps -l command. In HP-UX and Solaris, the value in this field is the same as the value displayed in the SZ string obtained by the ps -l command, multiplied by the page size. In Linux, the same as the value obtained from the first column	--	ulong	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Process Kbytes(TOTAL_PROCESS_KBYTES)	of `/proc/<PID>/statm` * *page-size* ÷ 1,024.	--	ulong	No	--	--
User(USER_NAME)	Effective user name of the process	--	string(256)	No	--	--
User CPU(USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	--	utime	Yes	--	--
User ID(USER_ID)	Effective user ID of the process (for a zombie process, this field is `0` on Solaris and `-3` on AIX and HP-UX.)	--	long	No	--	--
Virtual Mem Kbytes(VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory being used (KB)	--	ulong	No	Solaris	--
Writes(WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	--	ulong	Yes	HP-UX, Linux	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	--	float	Yes	HP-UX, Linux	When delta value is not collected, WRITE_OPS ÷ when collecting Process cumulative elapsed time delta value, WRITE_OPS ÷ (Previous process elapsed time - Present process elapsed time)

Legend:

Smry: Summary

Not sprtd in: Not supported in

Process Summary (PD_PDS)

Function

The Process Summary (PD_PDS) record stores the performance data in which the data stored in the Process Detail (PD) record is summarized as a state at a given point in time.

The information for this record is collected from the process table information for the past minute. Additionally, the number of processes in which changes occurred in the values listed below during the collection interval is counted as active processes.

- In AIX
  CPU usage time, number of page faults, number of context switches, and process size
- In HP-UX
  CPU usage time, number of page faults, and process size
- In Solaris
  CPU usage time, number of page faults, number of context switches, number of system calls, number of I/O operations, and process size
- In Linux
  CPU usage time, number of page faults, process size, and current directory

Even when the Process Detail (PD) record is not collected, performance data is stored in the Process Summary (PD_PDS) record.

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Interval Records, PI	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

None

Lifetime

None

Record size

- Fixed portion: 741 bytes
- Variable portion: 0 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Active System Processes(NUMBER_OF_ACTIVE_SYSTEM_PROCESSES)	Number of active system processes. In the first real-time report, 0 is displayed.	--	ulong	No	AIX, Linux	--
Active User Processes(NUMBER_OF_ACTIVE_USER_PROCESSES)	Number of active user processes. In the first real-time report, 0 is displayed. In AIX, this field shows the combined total number of active system processes and active user processes.	--	ulong	No	--	--
Active Users(NUMBER_OF_ACTIVE_USERS)	Number of users who are executing the processes that are counted as active system or user processes. In the first real-time report, 0 is displayed.	--	ulong	No	--	--
Daemon Processes(NUMBER_OF_DAEMON_PROCESSES)	Number of daemon processes	--	ulong	No	--	--
Interval(INTERVAL)	Interval at which the Process Summary (PD_PDS) record was stored. Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Mem Processes(NUMBER_OF_MEMORY_PROCESSES)	Number of memory management processes. In Linux, the number of processes having process name as "kswapd0" are counted.	--	ulong	No	--	--
New Processes(NUMBER_OF_NEW_PROCESSES)	Number of processes started. Because the difference from the previous process information is computed, 0 is displayed in the first real-time report.	--	ulong	No	--	Present TOTAL_NUMBER_OF_PROCESSES - Previous TOTAL_NUMBER_OF_PROCESSES
Processes(TOTAL_NUMBER_OF_PROCESSES)	Number of processes within the system	--	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always PDS.	--	char(8)	No	--	--
Runnable Processes(NUMBER_OF_RUNNABLE_PROCESSES)	Number of processes that are executable	--	ulong	No	--	--
Sleeping Processes(NUMBER_OF_SLEEPING_PROCESSES)	Number of processes that are sleeping	--	ulong	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Stopped Processes(NUMBER_OF_STOPPED_PROCESSES)	Number of processes that are stopped	--	ulong	No	--	--
Swapped Processes(NUMBER_OF_SWAPPED_PROCESSES)	Number of processes that have been swapped out. The number of processes that are not inside the core is determined from the process table.	--	ulong	No	Solaris	--
Terminals(NUMBER_OF_ACTIVE_TERMINALS)	Number of TTYs that are executing the processes that are counted as active system or user processes. In the first real-time report, 0 is displayed.	--	ulong	No	--	--
Terminated Processes(NUMBER_OF_DIED_PROCESSES)	Number of terminated processes. Because the difference from the previous process information is computed, 0 is displayed in the first real-time report.	--	ulong	No	--	Previous TOTAL_NUMBER_OF_PROCESSES - Present TOTAL_NUMBER_OF_PROCESSES
Users(TOTAL_NUMBER_OF_USERS)	Number of real users	--	ulong	No	--	--
Zombie Processes(NUMBER_OF_ZOMBIE_PROCESSES)	Number of zombie processes	--	ulong	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

Program Summary (PD_PGM)

Function

The Program Summary (PD_PGM) record stores the performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each program at a given point in time. A record is created for each program that is being executed. This is a multi-instance record. Note that even when the Process Detail (PD) record is not collected, performance data is stored in the Program Summary (PD_PGM) record.

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, PD	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

PD_PGM_PROGRAM_NAME

Lifetime

From the time when at least one process is being executed by a particular program name until the time when the number of processes being executed becomes 0.

Record size

- Fixed portion: 681 bytes
- Variable portion: 344 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg I/O Kbytes(AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ (READ_OPS + WRITE_OPS)
CPU % (CPU_PERCENT_USED)	Average CPU usage by processors (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ total-elapsed-time-of-all-processes-executing-programs) ÷ processors-count) × 100
Context Switches(CONTEXT_SWITCHES)	Number of times context switching was executed	--	ulong	No	HP-UX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval at which the Program Summary (PD_PGM) record was stored (seconds). Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Major Faults(MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	--	ulong	No	--	--
Minor Faults(MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	--	ulong	No	--	--
Process Count(PROCESS_COUNT)	Number of processes that are executing programs. Number of processes that have this program name inside the process table.	--	ulong	No	--	--
Program(PROGRAM_NAME)	Program name	--	string(256)	No	--	--
Reads(READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	--	ulong	No	HP-UX, Linux	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	--	float	No	HP-UX, Linux	READ_OPS ÷ total-elapsed-time-of-all-processes-executing-programs
Real Mem Kbytes(REAL_MEMORY_KBYTES)	Physical memory size used (KB)	--	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always PGM.	--	char(8)	No	--	--
Signals Rcvd(NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	No	Linux	--
Swaps(SWAPS)	Number of times swapping occurred	--	ulong	No	Linux	--
System CPU(SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode	--	utime	No	--	--
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	No	AIX, Linux	TOTAL_IO_KBYTES ÷ total-elapsed-time-of-all-processes-executing-programs
Total I/O Kbytes(TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	--	double	No	AIX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	--	ulong	No	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec(TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ total-elapsed-time-of-all-processes-executing-programs
User CPU(USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	--	utime	No	--	--
Virtual Mem Kbytes(VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (KB)	--	ulong	No	Solaris	--
Writes(WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	--	ulong	No	HP-UX, Linux	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	--	float	No	HP-UX, Linux	WRITE_OPS ÷ total-elapsed-time-of-all-processes-that-invoke-programs

Legend:

Smry: Summary

Not sprtd in: Not supported in

System Summary Overview (PI)

Function

The System Summary Overview (`PI`) record stores the performance data per unit time on the entire system.

If you are collecting processor performance data from the results output by the `sar` command in the AIX environment, the displayed numeric values will be for different fields. For details about the values displayed in the fields when collecting processor performance data from the results output by the `sar` command in the AIX environment, see *2.3.23 Monitoring processor performance based on results output by the sar command in an AIX environment*.

Notes:

- If you are collecting data in a Solaris non-global zone environment or an AIX System WPAR environment, the following fields cannot be collected:

  `NFS Server Lookup Ops`, `NFS Server Ops/sec`, `NFS Server Read Ops`, `NFS Server Read Ops/sec`, `NFS Server Total Bad Ops`, `NFS Server Total Ops`, `NFS Server Write Ops`, `NFS Server Write Ops/sec`

- If you are collecting data in a Solaris non-global zone environment, the following fields cannot be collected:

  `Cache Mem %`, `Cache Mem Mbytes`, `Effective Free Mem %`, `Effective Free Mem Mbytes`

  Also, the correct value is not collected in the non-global zone environment for the following fields:

  `Effective Used Mem %`, `Effective Used Mem Mbytes`

  (For Effective Used Mem %, 100.0000 is collected, and for Effective Used Mem Mbytes, the same value as Total Physical Mem Mbytes is collected.)

- If you collect in a non-global zone environment of Solaris, the value of the Boot Time field might not be displayed correctly.

  Display example: 1970 01 01 06:00:00

- If the DLPAR function of AIX, the vPars function of HP-UX, the DR function of Solaris, or the CPU hotplug of Linux is used to modify a system resource while a PFM - Agent for Platform service is running, some types of performance data cannot be correctly collected. The table below shows the types of system resources that are affected and the actions that you can take to correctly display reports after the modification.

OS and the name of function used to modify system resources	System resource type	Report type	Actions to be taken after modification
• DLPAR of AIX • DR of Solaris	• CPU • Memory	Real-time	Display the report again.
		Historical	Specify the display period such that it does not include the date and time when the system resources were modified.
vPars of HP-UX	CPU	Real-time	Display the report again.
		Historical	Specify the display period such that it does not include the date and time when the system resources were modified.
CPU hotplug of Linux	CPU	Real-time	Display the report again.
		Historical	Specify the display period such that it does not include the date and time when the system resources were modified.

The table below shows the fields that are affected by a system resource modification.

System resource type	Target field
CPU	When modifying system resources using the DLPAR functionality of AIX, the vPars functionality of HP-UX, and the DR functionality of Solaris: • Active CPUs (NUMBER_OF_ACTIVE_CPUS) • CPU % (KERNELMODE_USERMODE_PERCENT) • Context Switches (CONTEXT_SWITCHES) • Context Switches/sec (CONTEXT_SWITCHES_PER_SECOND) • Idle % (IDLE_TIME_PERCENT) • Interrupts (INTERRUPTS) • Interrupts/sec (INTERRUPTS_PER_SECOND) • Kernel CPU % (KERNELMODE_PERCENT) • System Calls (SYSTEM_CALLS) • System Calls/sec (SYSTEM_CALLS_PER_SECOND) • Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME) • Total Idle Time (TOTAL_IDLE_TIME) • Total User-Mode Time (TOTAL_USERMODE_TIME) • Total Wait Time (TOTAL_WAIT_TIME) • Traps (TRAPS) • Traps/sec (TRAPS_PER_SECOND) • User CPU % (USERMODE_PERCENT) • Wait % (WAIT_TIME_PERCENT) When modifying system resources using the DLPAR functionality of AIX, the vPars functionality of HP-UX, and the DR functionality of Solaris: • Active CPUs (NUMBER_OF_ACTIVE_CPUS) • CPU % (KERNELMODE_USERMODE_PERCENT) • Idle % (IDLE_TIME_PERCENT) • Kernel CPU % (KERNELMODE_PERCENT) • Total Kernel-Mode Time (TOTAL_KERNELMODE_TIME) • Total Idle Time (TOTAL_IDLE_TIME) • Total User-Mode Time (TOTAL_USERMODE_TIME) • Total Wait Time (TOTAL_WAIT_TIME) • User CPU % (USERMODE_PERCENT) • Wait % (WAIT_TIME_PERCENT)
Memory	• Alloc Mem % (ALLOCATED_MEMORY_PERCENT) • Alloc Mem Mbytes (ALLOCATED_MEMORY_MBYTES) • Free Mem % (FREE_MEMORY_PERCENT) • Free Mem Mbytes (FREE_MEMORY_MBYTES) • Total Physical Mem Mbytes (TOTAL_MEMORY_MBYTES)

For details about modifying system resources, see *Notes on performance information when modifying system resources* in *Notes on records*.

- In PFM - Agent for Platform 09-00 or later, IPv6 is supported. The following table lists the fields in which numeric values are affected in an environment that contains both IPv4 and IPv6.

PFM-View name	Obtained IP information		
	IPv4 environment	IPv6 environment	Coexisting environment
ICMP Pkts In	IPv4	0	IPv4
ICMP Pkts Out	IPv4	0	IPv4
ICMP6 Pkts In	0	IPv6	IPv6

PFM-View name	Obtained IP information		
	IPv4 environment	IPv6 environment	Coexisting environment
ICMP6 Pkts Out	0	IPv6	IPv6
IP Pkts In	IPv4	0	IPv4
IP Pkts Out	IPv4	0	IPv4
IP6 Pkts In	0	IPv6	IPv6
IP6 Pkts Out	0	IPv6	IPv6
Total Pkts In	Sum of the values of `TCP Pkts In`, `UDP Pkts In`, `ICMP Pkts In`, and `ICMP6 Pkts In`		
Total Pkts Out	Sum of the values of `TCP Pkts Out`, `UDP Pkts Out`, `ICMP Pkts Out`, and `ICMP6 Pkts Out`		

- In AIX, the following fields of PI record treat assigned file cache as "used physical memory".
  - Alloc Mem %
  - Alloc Mem Mbytes
  - Free Mem %
  - Free Mem Mbytes
- If you want to monitor the amount of memory actually available, we recommend that you monitor the `Effective Free Mem Mbytes` field, not the `Free Mem Mbytes` field. (Excluding the non-global zone environment of Solaris)

  Similarly, for the `Free Mem %`, `Alloc Mem Mbytes`, and `Alloc Mem %` fields, we recommend that you monitor the `Effective Free Mem %`, `Effective Used Mem Mbytes`, and `Effective Used Mem %` fields, which all begin with `Effective`, respectively.

  For details about the difference between the fields, see *3.1.2(2)(e) Checking the amount of memory that the system can actually use*.

Default values and values that can be specified

Item	Default value	Modifiable
`Collection Interval`	`60`	Yes
`Collection Offset`#	`0`	Yes
`Log`	`Yes`	Yes
`LOGIF`	Blank	Yes
`Over 10 Sec Collection Time`	`Yes`	No
`Realtime Report Data Collection Mode`	`Reschedule`	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

None

Lifetime

None

Record size

- Fixed portion: 2,749 bytes
- Variable portion: 0 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
1-Minute Run Queue Avg (ONE_MINUTE_RUN_QUEUE_AVG)	The average number of threads over the last one minute, other than threads in the queue waiting for the kernel. In HP-UX, Solaris, and AIX, this value does not include the number of threads that were waiting for I/O. In Linux, this value includes the number of threads waiting for I/O.	AVG	`float`	No	--	--
15-Minute Run Queue Avg (FIFTEEN_MINUTE_RUN_QUEUE_AVG)	The average number of threads over the last 15 minutes, other than threads in the queue waiting for the kernel. In HP-UX, Solaris, and AIX, this value does not include the number of threads that were waiting for I/O. In Linux, this value includes the number of threads waiting for I/O.	AVG	`float`	No	--	--
5-Minute Run Queue Avg (FIVE_MINUTE_RUN_QUEUE_AVG)	The average number of threads over the last 5 minutes, other than threads waiting in the queue for the kernel. In HP-UX, Solaris, and AIX, this value does not include the number of threads that were waiting for I/O. In Linux, this value includes the number of threads waiting for I/O.	AVG	`float`	No	--	--
Active CPUs (NUMBER_OF_ACTIVE_CPUS)	Processor count	COPY	`ulong`	No	--	• CPU - Per Processor Detail (PI_CPUP) Record count. • When collecting results output by the `sar` command, the number of lines in which a number is displayed in the `cpu` column.

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Alloc Mem % (ALLOCATED_MEMORY_PERCENT)	Percentage of the physical memory size (MB) used (%)	%	`float`	No	--	(ALLOCATED_MEMORY_MBYTES ÷ TOTAL_MEMORY_MBYTES) × 100
Alloc Mem Mbytes (ALLOCATED_MEMORY_MBYTES)	Megabytes of the physical memory used	AVG	`float`	No	--	TOTAL_MEMORY_MBYTES - FREE_MEMORY_MBYTES
Alloc Swap % (ALLOCATED_SWAP_PERCENT)	Percentage of the megabytes of the swap area used (%)	%	`float`	No	--	(ALLOCATED_SWAP_MBYTES ÷ TOTAL_SWAP_MBYTES) × 100
Alloc Swap Mbytes (ALLOCATED_SWAP_MBYTES)	Megabytes of the swap area used	AVG	`float`	No	--	TOTAL_SWAP_MBYTES - FREE_SWAP_MBYTES
Block Ops (BLOCKIO_IO_OPS)	Number of block I/O operations that occurred	AVG	`ulong`	Yes	Linux	BLOCKIO_READ_OPS + BLOCKIO_WRITE_OPS
Block Reads (BLOCKIO_READ_OPS)	Number of block read operations that occurred	AVG	`ulong`	Yes	Linux	--
Block Reads/sec (BLOCKIO_READ_OPS_PER_SECOND)	Frequency at which block read operations occurred (number of operations per second)	R	`float`	Yes	Linux	BLOCKIO_READ_OPS ÷ INTERVAL
Block Writes (BLOCKIO_WRITE_OPS)	Number of block write operations that occurred	AVG	`ulong`	Yes	Linux	--
Block Writes/sec (BLOCKIO_WRITE_OPS_PER_SECOND)	Frequency at which block write operations occurred (number of operations per second)	R	`float`	Yes	Linux	BLOCKIO_WRITE_OPS ÷ INTERVAL
Boot Time (CURRENT_SYSTEM_BOOT_TIME)	Last boot time	COPY	`time_t`	No	--	--
Buffers Mem % (BUFFERS_MEM_PERCENT)	Percentage of the megabytes of the buffer area used (%)	%	`double`	No	HP-UX, Solaris, AIX	(BUFFERS_MEM_MBYTES ÷ TOTAL_MEMORY_MBYTES) × 100
Buffers Mem Mbytes (BUFFERS_MEM_MBYTES)	Megabytes of the buffer area used	AVG	`double`	No	HP-UX, Solaris, AIX	--
CPU % (KERNELMODE_USERMODE_PERCENT)	CPU usage (%). It also indicates the average value per processor.	%	`float`	No	--	• ((TOTAL_USERMODE_TIME + TOTAL_KERNELMODE_TIME) ÷ (TOTAL_IDLE_TIME + TOTAL_USERMODE_TIME + TOTAL_WAIT_TI

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
CPU % (KERNELMODE_USER MODE_PERCENT)	CPU usage (%). It also indicates the average value per processor.	%	float	No	--	ME + TOTAL_KERNEL MODE_TIME)) × 100 • %sys column + %usr column when collecting results output by the sar command
Cache Mem % (CACHE_MEM_PERCE NT)	Percentage of the megabytes of the cache area used (%)	%	double	No	HP-UX	(CACHE_MEM_MBY TES ÷ TOTAL_MEMORY_ MBYTES) × 100
Cache Mem Mbytes (CACHE_MEM_MBYT ES)	Megabytes of the cache area used	AVG	double	No	HP-UX	--
Cache Read % (CACHE_READ_PERC ENT)	Cache read operations as a percentage of all read operations (%)	%	float	No	Linux	((LOGICAL_READ_ OPS - BLOCKIO_READ_OP S) ÷ LOGICAL_READ_OP S) × 100
Cache Write % (CACHE_WRITE_PER CENT)	Cache write operations as a percentage of all write operations (%)	%	float	No	Linux	((LOGICAL_WRITE_ OPS - BLOCKIO_WRITE_O PS) ÷ LOGICAL_WRITE_O PS) × 100
Context Switches (CONTEXT_SWITCHE S)	Number of times context switching was executed	AVG	ulong	Yes	--	--
Context Switches/sec (CONTEXT_SWITCHE S_PER_SECOND)	Frequency at which context switching was executed (number of times per second)	R	float	Yes	--	CONTEXT_SWITCH ES ÷ INTERVAL
Effective Free Mem % (EFFECTIVE_FREE_M EM_PERCENT)	The percentage (in mbytes) of physical memory that can be used by applications	%	double	No	--	(EFFECTIVE_FREE_ MEM_MBYTES ÷ TOTAL_MEMORY_ MBYTES) × 100
Effective Free Mem Mbytes (EFFECTIVE_FREE_M EM_MBYTES)	The amount (in mbytes) of physical memory that can be used by applications	AVG	double	No	--	--
Effective Used Mem % (EFFECTIVE_USED_M EM_PERCENT)	The percentage of the physical memory size (MB) actually used (%)	%	double	No	--	(EFFECTIVE_USED_ MEM_MBYTES ÷ TOTAL_MEMORY_ MBYTES) × 100
Effective Used Mem Mbytes (EFFECTIVE_USED_M EM_MBYTES)	The number of megabytes of physical memory actually used	AVG	double	No	--	TOTAL_MEMORY_ MBYTES - EFFECTIVE_FREE_ MEM_MBYTES
Faults (TOTAL_FAULTS)	Number of page faults that occurred	AVG	ulong	Yes	Linux	MAJOR_FAULTS + MINOR_FAULTS

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Free Mem % (FREE_MEMORY_PERCENT)	Percentage of physical memory size (MB) that is free (%)	%	`float`	No	--	(FREE_MEMORY_MBYTES ÷ TOTAL_MEMORY_MBYTES) × 100
Free Mem Mbytes (FREE_MEMORY_MBYTES)	Megabytes of physical memory not used	AVG	`float`	No	--	--
Free Swap % (FREE_SWAP_PERCENT)	Percent of the swap area size (MB) not used (%)	%	`float`	No	--	(FREE_SWAP_MBYTES ÷ TOTAL_SWAP_MBYTES) × 100
Free Swap Mbytes (FREE_SWAP_MBYTES)	Size of the swap area not used (MB)	AVG	`float`	No	--	--
ICMP Pkts In (ICMP_PACKETS_IN)	Number of IPv4 ICMP packets received. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred.	AVG	`ulong`	Yes	--	--
ICMP Pkts Out (ICMP_PACKETS_OUT)	Number of IPv4 ICMP packets sent. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred.	AVG	`ulong`	Yes	--	--
ICMP6 Pkts In (ICMP6_PACKETS_IN)	Number of IPv6 ICMP packets received. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred.	AVG	`ulong`	Yes	--	--
ICMP6 Pkts Out (ICMP6_PACKETS_OUT)	Number of IPv6 ICMP packets sent. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred.	AVG	`ulong`	Yes	--	--
IP Pkts In (IP_PACKETS_IN)	Number of IPv4 IP packets received. Combined total of the number of local and remote packets. This value does not include the number of packets in which errors occurred.	AVG	`ulong`	Yes	--	--
IP Pkts Out (IP_PACKETS_OUT)	Number of IPv4 IP packets sent. Combined total of the number of local and remote packets. This value does not include the number of packets in which errors occurred.	AVG	`ulong`	Yes	--	--
IP6 Pkts In (IP6_PACKETS_IN)	Number of IPv6 IP packets received. Combined total of the number of local and remote packets. This value does not include the number of packets in which errors occurred.	AVG	`ulong`	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
IP6 Pkts Out (IP6_PACKETS_OUT)	Number of IPv6 IP packets sent. Combined total of the number of local and remote packets. This value does not include the number of packets in which errors occurred.	AVG	`ulong`	Yes	--	--
Idle % (IDLE_TIME_PERCENT)	Percentage of time a processor was idle (%). It also indicates the average value per processor.	%	`float`	No	--	• (TOTAL_IDLE_TIME / (TOTAL_IDLE_TIME + TOTAL_USERMODE_TIME + TOTAL_WAIT_TIME + TOTAL_KERNELMODE_TIME)) × 100 • `%idle` column when collecting results output by the `sar` command
Interrupts (INTERRUPTS)	Number of interrupts that occurred	AVG	`ulong`	Yes	AIX	--
Interrupts/sec (INTERRUPTS_PER_SECOND)	Frequency at which interrupts occurred (number of times per second)	R	`float`	Yes	AIX	INTERRUPTS ÷ INTERVAL
Interval (INTERVAL)	Interval at which the System Summary Overview (PI) record was stored (seconds)	COPY	`ulong`	Yes	--	• When delta values are not checked in real time, `RECORD_TIME` - *last-boot-time* • Other cases, • `RECORD_TIME` - *previous-record-time*
Kernel CPU % (KERNELMODE_PERCENT)	Percentage of time the CPU ran in the kernel mode (%). It also indicates the average value per processor.	%	`float`	No	--	• (TOTAL_KERNELMODE_TIME ÷ (TOTAL_IDLE_TIME + TOTAL_USERMODE_TIME + TOTAL_WAIT_TIME + TOTAL_KERNELMODE_TIME)) × 100 • `%sys` column when collecting results output by the `sar` command
Logical I/O Ops (LOGICAL_IO_OPS)	Number of logical I/O operations that occurred	AVG	`ulong`	Yes	Linux	LOGICAL_READ_OPS + LOGICAL_WRITE_OPS

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Logical Read Mbytes (LOGICAL_READ_MBYTES)	Size transferred in logical read operations (MB). In HP-UX, this field shows the size of data read from block devices. In Solaris and AIX, this field shows the size of data read in read system calls.	AVG	`float`	Yes	Linux	--
Logical Reads (LOGICAL_READ_OPS)	Number of logical read operations that occurred	AVG	`ulong`	Yes	Linux	--
Logical Reads/sec (LOGICAL_READ_MBYTES_PER_SECOND)	Speed of logical read operations (MB per second)	R	`float`	Yes	Linux	LOGICAL_READ_MBYTES ÷ INTERVAL
Logical Write Mbytes (LOGICAL_WRITE_MBYTES)	Size transferred in logical write operations (MB). In HP-UX, this field shows the size of data written into block devices. In Solaris and AIX, this field shows the size of data written in write system calls.	AVG	`float`	Yes	Linux	--
Logical Writes (LOGICAL_WRITE_OPS)	Number of logical write operations that occurred	AVG	`ulong`	Yes	Linux	--
Logical Writes/sec (LOGICAL_WRITE_MBYTES_PER_SECOND)	Speed of logical write operations (MB per second)	R	`float`	Yes	Linux	LOGICAL_WRITE_MBYTES ÷ INTERVAL
Major Faults (MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	AVG	`ulong`	Yes	Linux	--
Major Faults/sec (MAJOR_FAULTS_PER_SECOND)	Frequency at which page faults that cause physical I/O operations occurred (number of times per second)	R	`float`	Yes	Linux	MAJOR_FAULTS ÷ INTERVAL
Mem I/O Ops (MEMORY_IO_OPS)	Number of page faults that cause swapping and physical I/O operations	AVG	`ulong`	Yes	AIX, Linux	SWAP_IN_OPS + SWAP_OUT_OPS + MAJOR_FAULTS
Minor Faults (MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	AVG	`ulong`	Yes	AIX, Linux	--
Minor Faults/sec (MINOR_FAULTS_PER_SECOND)	Frequency at which page faults that do not cause physical I/O operations occurred (number of times per second)	R	`float`	Yes	AIX, Linux	MINOR_FAULTS ÷ INTERVAL
NFS Client Lookup Ops (NFS_CLIENT_LOOKUP_OPS)	Number of times lookup operations occurred on the NFS client	AVG	`ulong`	Yes	Linux	--
NFS Client Ops/sec (NFS_CLIENT_TOTAL_OPS_PER_SECOND)	Frequency at which operations occurred on the NFS client (number of operations per second)	R	`float`	Yes	Linux	NFS_CLIENT_TOTAL_OPS ÷ INTERVAL

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
NFS Client Read Ops (NFS_CLIENT_READ_OPS)	Number of times read operations occurred on the NFS client	AVG	ulong	Yes	Linux	--
NFS Client Read Ops/sec (NFS_CLIENT_READ_OPS_PER_SECOND)	Frequency at which read operations occurred on the NFS client (number of operations per second)	R	float	Yes	Linux	NFS_CLIENT_READ_OPS ÷ INTERVAL
NFS Client Total Bad Ops (NFS_CLIENT_TOTAL_BAD_OPS)	Total number of failed operations on the NFS client	AVG	ulong	Yes	Linux	--
NFS Client Total Ops (NFS_CLIENT_TOTAL_OPS)	Total number of operations that occurred on the NFS client	AVG	ulong	Yes	Linux	--
NFS Client Write Ops (NFS_CLIENT_WRITE_OPS)	Number of times write operations occurred on the NFS client	AVG	ulong	Yes	Linux	--
NFS Client Write Ops/sec (NFS_CLIENT_WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred on the NFS client (number of operations per second)	R	float	Yes	Linux	NFS_CLIENT_WRITE_OPS ÷ INTERVAL
NFS Server Lookup Ops (NFS_SERVER_LOOKUP_OPS)	Number of times lookup operations occurred on the NFS server	AVG	ulong	Yes	Linux	--
NFS Server Ops/sec (NFS_SERVER_TOTAL_OPS_PER_SECOND)	Frequency at which operations occurred on the NFS server (number of operations per second)	R	float	Yes	Linux	NFS_SERVER_TOTAL_OPS ÷ INTERVAL
NFS Server Read Ops (NFS_SERVER_READ_OPS)	Number of times read operations occurred on the NFS server	AVG	ulong	Yes	Linux	--
NFS Server Read Ops/sec (NFS_SERVER_READ_OPS_PER_SECOND)	Frequency at which read operations occurred on the NFS server (number of operations per second)	R	float	Yes	Linux	NFS_SERVER_READ_OPS ÷ INTERVAL
NFS Server Total Bad Ops (NFS_SERVER_TOTAL_BAD_OPS)	Total number of failed operations on the NFS server	AVG	ulong	Yes	Linux	--
NFS Server Total Ops (NFS_SERVER_TOTAL_OPS)	Total number of operations that occurred on the NFS server	AVG	ulong	Yes	Linux	--
NFS Server Write Ops (NFS_SERVER_WRITE_OPS)	Number of times write operations occurred on the NFS server	AVG	ulong	Yes	Linux	--
NFS Server Write Ops/sec (NFS_SERVER_WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred on the NFS server (number of operations per second)	R	float	Yes	Linux	NFS_SERVER_WRITE_OPS ÷ INTERVAL
Other Pkts In (OTHER_PACKETS_IN)	This field is always 0 because it is not supported	AVG	ulong	Yes	HP-UX, Solaris, AIX,	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Other Pkts In (OTHER_PACKETS_IN)	This field is always 0 because it is not supported	AVG	ulong	Yes	Linux	--
Other Pkts Out (OTHER_PACKETS_OUT)	This field is always 0 because it is not supported	AVG	ulong	Yes	HP-UX, Solaris, AIX, Linux	--
Page Ops/sec (PAGE_OPS_PER_SECOND)	Frequency at which page-in and page-out operations occurred (number of operations per second)	R	float	Yes	AIX, Linux	TOTAL_PAGE_OPS ÷ INTERVAL
Page Reclaims/sec (PAGE_RECLAIMS_PER_SECOND)	Frequency at which page reclaiming occurred (number of times per second)	R	float	Yes	AIX, Linux	TOTAL_PAGE_RECLAIMS ÷ INTERVAL
Page Scans/sec (PAGE_SCANS_PER_SECOND)	Frequency at which page scans occurred (number of times per second)	R	float	Yes	Linux	TOTAL_PAGE_SCANS ÷ INTERVAL
Page-In Ops (PAGE_IN_OPS)	Number of page-in operations that occurred	AVG	ulong	Yes	AIX, Linux	--
Page-In Ops/sec (PAGE_IN_OPS_PER_SECOND)	Frequency at which page-in operations occurred (number of operations per second)	R	float	Yes	AIX, Linux	PAGE_IN_OPS ÷ INTERVAL
Page-Out Ops (PAGE_OUT_OPS)	Number of page-out operations that occurred	AVG	ulong	Yes	AIX, Linux	--
Page-Out Ops/sec (PAGE_OUT_OPS_PER_SECOND)	Frequency at which page-out operations occurred (number of operations per second)	R	float	Yes	AIX, Linux	PAGE_OUT_OPS ÷ INTERVAL
Pages In (PAGE_IN_COUNT)	Number of pages loaded through page-in operations	AVG	ulong	Yes	--	--
Pages In/sec (PAGE_IN_COUNT_PER_SECOND)	Frequency at which pages were loaded through page-in operations (number of pages per second)	R	float	Yes	--	PAGE_IN_COUNT ÷ INTERVAL
Pages Out (PAGE_OUT_COUNT)	Number of pages extracted through page-out operations	AVG	ulong	Yes	--	--
Pages Out/sec (PAGE_OUT_COUNT_PER_SECOND)	Frequency at which pages were extracted through page-out operations (number of pages per second)	R	float	Yes	--	PAGE_OUT_COUNT ÷ INTERVAL
Physical I/O Ops (PHYSICAL_IO_OPS)	Number of physical I/O operations that occurred	AVG	ulong	Yes	Linux	PHYSICAL_READ_OPS + PHYSICAL_WRITE_OPS
Physical Reads (PHYSICAL_READ_OPS)	Number of physical read operations that occurred	AVG	ulong	Yes	Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Physical Writes (PHYSICAL_WRITE_OPS)	Number of physical write operations that occurred	AVG	`ulong`	Yes	Linux	--
Processes (CURRENT_PROCESS_COUNT)	Number of processes within the system	AVG	`ulong`	No	--	Sum of the processes in the process table.
Processes Ended (PROCESSES_ENDED)	Number of processes ended. Because the difference from the previous process information is computed, 0 is displayed in the first real-time report.	AVG	`ulong`	No	--	Earlier Interval Process Count - (CURRENT_PROCESS_COUNT - PROCESSES_STARTED)
Processes Started (PROCESSES_STARTED)	Number of processes started. Because the difference from the previous process information is computed, 0 is displayed in the first real-time report.	AVG	`ulong`	No	--	CURRENT_PROCESS_COUNT - PREVIOUS PROCESS COUNT
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	`time_t`	No	--	--
Record Type (INPUT_RECORD_TYPE)	Record type. Always PI.	COPY	`char(8)`	No	--	--
Run Queue (PROCESSES_IN_RUN_QUEUE)	The number of threads, including threads waiting in the execution queue for the kernel	AVG	`float`	No	Linux	--
Software Lock Faults (SOFTWARE_LOCK_FAULTS)	Number of times software lock failed	AVG	`ulong`	Yes	HP-UX, AIX, Linux	--
Software Lock Faults/sec (SOFTWARE_LOCK_FAULTS_PER_SECOND)	Frequency at which software lock failed (number of times per second)	R	`float`	Yes	HP-UX, AIX, Linux	SOFTWARE_LOCK_FAULTS ÷ INTERVAL
Swap-In Ops (SWAP_IN_OPS)	Number of swap-in operations that occurred	AVG	`ulong`	Yes	AIX, Linux	--
Swap-Ins/sec (SWAP_IN_OPS_PER_SECOND)	Frequency at which swap-in operations occurred (number of times per second)	R	`float`	Yes	AIX, Linux	SWAP_IN_OPS ÷ INTERVAL
Swap-Out Ops (SWAP_OUT_OPS)	Number of swap-out operations that occurred	AVG	`ulong`	Yes	AIX, Linux	--
Swap-Outs/sec (SWAP_OUT_OPS_PER_SECOND)	Frequency at which swap-out operations occurred (number of times per second)	R	`float`	Yes	AIX, Linux	SWAP_OUT_OPS ÷ INTERVAL
Swapped-In Pages (SWAP_IN_COUNT)	Number of pages loaded through swap-in operations	AVG	`ulong`	Yes	--	--
Swapped-In Pages/sec (SWAP_IN_COUNT_PER_SECOND)	Frequency at which pages were loaded through swap-in operations (number of pages per second). In AIX, this field shows the frequency at which pages were	R	`float`	Yes	--	SWAP_IN_COUNT ÷ INTERVAL

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Swapped-In Pages/sec (SWAP_IN_COUNT_PER_SECOND)	swapped in only from the paging area through swap-in operations (number of pages per second).	R	`float`	Yes	--	SWAP_IN_COUNT ÷ INTERVAL
Swapped-Out Pages (SWAP_OUT_COUNT)	Number of pages extracted through swap-out operations	AVG	`ulong`	Yes	--	--
Swapped-Out Pages/sec (SWAP_OUT_COUNT_PER_SECOND)	Frequency at which pages were extracted through swap-out operations (number of pages per second). In AIX, this field shows the frequency at which pages were swapped out only from the paging area through swap-out operations (number of pages per second).	R	`float`	Yes	--	SWAP_OUT_COUNT ÷ INTERVAL
System Calls (SYSTEM_CALLS)	Number of system calls that were issued	AVG	`ulong`	Yes	Linux	--
System Calls/sec (SYSTEM_CALLS_PER_SECOND)	Frequency at which system calls were issued (number of times per second)	R	`float`	Yes	Linux	SYSTEM_CALLS ÷ INTERVAL
System Up Time (CURRENT_BOOT_SYSTEM_UP_TIME)	Time elapsed since the last boot (seconds)	COPY	`string(20)`	No	--	RECORD_TIME - CURRENT_SYSTEM_BOOT_TIME
TCP Pkts In (TCP_PACKETS_IN)	Number of IPv4 TCP packets received. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	`ulong`	Yes	--	--
TCP Pkts Out (TCP_PACKETS_OUT)	Number of IPv4 TCP packets sent. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	`ulong`	Yes	--	--
Total Faults/sec (TOTAL_FAULTS_PER_SECOND)	Frequency at which page faults occurred (number of times per second)	R	`float`	Yes	Linux	TOTAL_FAULTS ÷ INTERVAL
Total Idle Time (TOTAL_IDLE_TIME)	Combined total idle time of all processors (seconds)	AVG	`utime`	Yes	--	--
Total Kernel-Mode Time(TOTAL_KERNEL MODE_TIME)	Combined total time for all processors in which they ran in the kernel mode (seconds)	AVG	`utime`	Yes	--	--
Total Page Ops (TOTAL_PAGE_OPS)	Number of times paging operations occurred	AVG	`ulong`	Yes	AIX, Linux	PAGE_IN_OPS + PAGE_OUT_OPS
Total Page Reclaims (TOTAL_PAGE_RECLAIMS)	Number of times page reclaiming occurred	AVG	`ulong`	Yes	AIX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total Page Scans (TOTAL_PAGE_SCANS)	Number of pages checked by page scan	AVG	ulong	Yes	Linux	--
Total Physical Mem Mbytes (TOTAL_MEMORY_MBYTES)	Physical memory size (MB)	COPY	ulong	No	--	--
Total Pkts (TOTAL_PACKETS)	Total number of IPv4 TCP, IPv4 UDP, and IPv4 ICMP packets that were sent and received. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	ulong	Yes	--	TOTAL_PACKETS_IN + TOTAL_PACKETS_OUT
Total Pkts In (TOTAL_PACKETS_IN)	Total number of IPv4 TCP, IPv4 UDP, and IPv4 ICMP packets that were received. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	ulong	Yes	--	TCP_PACKETS_IN + UDP_PACKETS_IN + ICMP_PACKETS_IN + ICMP6_PACKETS_IN
Total Pkts Out (TOTAL_PACKETS_OUT)	Total number of IPv4 TCP, IPv4 UDP, and IPv4 ICMP packets that were sent. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	ulong	Yes	--	TCP_PACKETS_OUT + UDP_PACKETS_OUT + ICMP_PACKETS_OUT + ICMP6_PACKETS_OUT
Total Swap Mbytes (TOTAL_SWAP_MBYTES)	Swap area size (MB)	COPY	ulong	No	--	--
Total Swaps (TOTAL_SWAP_OPS)	Number of swapping operations that occurred	AVG	ulong	Yes	AIX, Linux	SWAP_IN_OPS + SWAP_OUT_OPS
Total Swaps/sec (TOTAL_SWAP_OPS_PER_SECOND)	Frequency at which swapping operations occurred (number of times per second)	R	float	Yes	AIX, Linux	TOTAL_SWAP_OPS ÷ INTERVAL
Total User-Mode Time (TOTAL_USERMODE_TIME)	Combined total time for all processors in which they ran in the user mode (seconds)	AVG	utime	Yes	--	--
Total Wait Time (TOTAL_WAIT_TIME)	Combined total time for all processors in which they were waiting for I/O operations (seconds)	AVG	utime	Yes	Solaris	--
Traps (TRAPS)	Number of traps that occurred	AVG	ulong	Yes	AIX, Linux	--
Traps/sec (TRAPS_PER_SECOND)	Frequency at which traps occurred (number of times per second)	R	float	Yes	AIX, Linux	TRAPS ÷ INTERVAL
UDP Pkts In (UDP_PACKETS_IN)	Number of IPv4 UDP packets received. Combined total of the numbers of local packets, remote packets, and packets in which	AVG	ulong	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
UDP Pkts In (UDP_PACKETS_IN)	errors occurred. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	`ulong`	Yes	--	--
UDP Pkts Out (UDP_PACKETS_OUT)	Number of IPv4 UDP packets sent. Combined total of the numbers of local packets, remote packets, and packets in which errors occurred. The total of the number of packets that IPv4 or IPv6 corresponds is displayed since 0900.	AVG	`ulong`	Yes	--	--
User CPU % (USERMODE_PERCENT)	Percentage of time a processor ran in the user mode (%). It also indicates the average value per processor.	%	`float`	No	--	• (TOTAL_USERMODE_TIME ÷ (TOTAL_IDLE_TIME + TOTAL_USERMODE_TIME + TOTAL_WAIT_TIME + TOTAL_KERNELMODE_TIME)) × 100 • `%usr` column when collecting results output by the `sar` command
Users (CURRENT_USER_COUNT)	Number of real users	AVG	`ulong`	No	Solaris	--
Wait % (WAIT_TIME_PERCENT)	Percentage of time a processor was waiting for I/O operations (%). It also indicates the average value per processor.	%	`float`	No	Solaris	• (TOTAL_WAIT_TIME ÷ (TOTAL_IDLE_TIME + TOTAL_USERMODE_TIME + TOTAL_WAIT_TIME + TOTAL_KERNELMODE_TIME)) × 100 • `%wio` column when collecting results output by the `sar` command

Legend:

Smry: Summary

Not sprtd in: Not supported in

Terminal Summary (PD_TERM)

Function

The Terminal Summary (PD_TERM) record stores the performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each terminal at a given point in time. A record is created for each terminal. This is a multi-instance record. Note that even if the Process Detail (PD) record is not collected, performance data is stored in the Terminal Summary (PD_TERM) record.

Notes:

- If a process has no terminal name, ?? is displayed in the Terminal (TERMINAL_NAME) field.

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, PD	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

PD_TERM_TERMINAL_NAME

Lifetime

From the time when at least one process is being executed at a particular terminal until the time when the number of processes being executed becomes 0.

Record size

- Fixed portion: 681 bytes
- Variable portion: 128 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg I/O Kbytes (AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ TOTAL_IO_OPS
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ total-elapsed-time-of-all-processes-executed-from-a-terminal) ÷ processors-count) × 100

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Context Switches (CONTEXT_SWITCHES)	Number of times context switching was executed	--	ulong	No	HP-UX, Linux	--
Interval (INTERVAL)	Interval at which the Terminal Summary (PD_TERM) record was stored (seconds). Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Major Faults (MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	--	ulong	No	--	--
Minor Faults (MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	--	ulong	No	--	--
Process Count (PROCESS_COUNT)	Number of processes executed from the terminal	--	ulong	No	--	Number of processes in processes table that are using a terminal
Reads (READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	--	ulong	No	HP-UX, Linux	--
Reads/sec (READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	--	float	No	HP-UX, Linux	READ_OPS ÷ total-elapsed-time-of-all-processes-executed-from-terminal
Real Mem Kbytes (REAL_MEMORY_KBYTES)	Physical memory size used (KB)	--	float	No	--	--
Record Time (RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type (INPUT_RECORD_TYPE)	Record type. Always TERM.	--	char(8)	No	--	--
Signals Rcvd (NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	No	Linux	--
Swaps (SWAPS)	Number of times swapping occurred	--	ulong	No	Linux	--
System CPU (SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode (seconds)	--	utime	No	--	--
Terminal (TERMINAL_NAME)	Terminal name. Process without terminal name, ?? is displayed.	--	string(40)	No	--	--
Throughput/sec (IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	No	AIX, Linux	TOTAL_IO_KBYTES ÷ total-elapsed-time-of-all-processes-executed-from-terminal
Total I/O Kbytes (TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	--	double	No	AIX, Linux	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total I/O Ops (TOTAL_IO_OPS)	Number of I/O operations that occurred	--	`ulong`	No	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec (TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	--	`float`	No	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ total-elapsed-time-of-all-processes-executed-from-terminal
User CPU (USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	--	`utime`	No	--	--
Virtual Mem Kbytes (VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (KB)	--	`float`	No	Solaris	--
Writes (WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	--	`ulong`	No	HP-UX, Linux	--
Writes/sec (WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	--	`float`	No	HP-UX, Linux	WRITE_OPS ÷ total-elapsed-time-of-all-processes-executed-from-terminal

Legend:

Smry: Summary

Not sprtd in: Not supported in

User Data Detail (PD_UPD)

Function

The User Data Detail (PD_UPD) record is a user-defined record that stores user-specific performance data that indicates the status at a given point in time. For 08-00 or earlier, this is a reserved record (field) that cannot be used in any operating systems.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PD_UPD_TT
- PD_UPD_TS_KEY
- PD_UPD_TD_KEY

Lifetime

From the time a jpcuser command is executed until the next jpcuser command is executed

Record size

- Fixed portion: 681 bytes
- Variable portion: 284 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Collect Time(API_TIME)	Time when data was converted	--	time_t	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval for which the User Data Detail (PD_UPD) record was stored (in seconds). The normal value is 0.	--	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always UPD.	--	char(8)	No	--	--
Trans Data Key(TD_KEY)	Transaction Data Key	--	ulong	No	--	--
Trans String Key(TS_KEY)	Transaction String Key	--	string(20)	No	--	--
Trans Type(TT)	Transaction Type	--	string(20)	No	--	--
User Float 1(F1)	Floating-point value	--	double	No	--	--
User Float 2(F2)	Floating-point value	--	double	No	--	--
User Long 1(L1)	Long integer value	--	long	No	--	--
User Long 2(L2)	Long integer value	--	long	No	--	--
User String 1(S1)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 2(S2)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 3(S3)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 4(S4)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 5(S5)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 6(S6)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 7(S7)	Character-string 64 characters in length	--	string(64)	No	--	--
User Time 1(T1)	Time value (Time is displayed at the local time of the PFM - Web Console). If the value is `0`, `1970 01 01 09:00:00` is displayed.	--	time_t	No	--	--
User Unsigned Long 1(UL1)	Unsigned long value	--	ulong	No	--	--
User Unsigned Long 2(UL2)	Unsigned long value	--	ulong	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

User Data Detail - Extended (PD_UPDB)

Function

The User Data Detail - Extended (PD_UPDB) record is a user-defined record that stores user-specific performance data that indicates the status at a given point in time. This record has more fields than the User Data Detail (PD_UPD) record. For 08-00 or earlier, this is a reserved record (field) that cannot be used in any operating systems.

Default values and values that can be specified

Item	Item	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PD_UPDB_TT
- PD_UPDB_TS_KEY
- PD_UPDB_TD_KEY

Lifetime

From the time a jpcuser command is executed until the next jpcuser command is executed

Record size

- Fixed portion: 681 bytes
- Variable portion: 712 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Collect Time(API_TIME)	Time when data was converted	--	time_t	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval for which the User Data Detail - Extended (PD_UPDB) record was stored (in seconds). The normal value is 0.	--	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always UPDB.	--	char(8)	No	--	--
Trans Data Key(TD_KEY)	Transaction Data Key	--	ulong	No	--	--
Trans String Key(TS_KEY)	Transaction String Key	--	string(20)	No	--	--
Trans Type(TT)	Transaction Type	--	string(20)	No	--	--
User Float 1(F1)	Floating-point value	--	double	No	--	--
User Float 2(F2)	Floating-point value	--	double	No	--	--
User Float 3(F3)	Floating-point value	--	double	No	--	--
User Float 4(F4)	Floating-point value	--	double	No	--	--
User Float 5(F5)	Floating-point value	--	double	No	--	--
User Long 1(L1)	Long integer value	--	long	No	--	--
User Long 2(L2)	Long integer value	--	long	No	--	--
User Long 3(L3)	Long integer value	--	long	No	--	--
User Long 4(L4)	Long integer value	--	long	No	--	--
User Long 5(L5)	Long integer value	--	long	No	--	--
User String 1(S1)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 10(S10)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 11(S11)	Character-string 64 characters in length	--	string(64)	No	--	--
User String 12(S12)	Character-string 64 characters in length	--	string(64)	No	--	--
User String 13(S13)	Character-string 64 characters in length	--	string(64)	No	--	--
User String 14(S14)	Character-string 64 characters in length	--	string(64)	No	--	--
User String 15(S15)	Character-string 64 characters in length	--	string(64)	No	--	--
User String 2(S2)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 3(S3)	Character-string 16 characters in length	--	string(16)	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
User String 4(S4)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 5(S5)	Character-string 16 characters in length	--	string(16)	No	--	--
User String 6(S6)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 7(S7)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 8(S8)	Character-string 32 characters in length	--	string(32)	No	--	--
User String 9(S9)	Character-string 32 characters in length	--	string(32)	No	--	--
User Time 1(T1)	Time value (Time is displayed at the local time of the PFM - Web Console). If the value is `0`, `1970 01 01 09:00:00` is displayed.	--	time_t	No	--	--
User Unsigned Long 1(UL1)	Unsigned long value	--	ulong	No	--	--
User Unsigned Long 2(UL2)	Unsigned long value	--	ulong	No	--	--
User Unsigned Long 3(UL3)	Unsigned long value	--	ulong	No	--	--
User Unsigned Long 4(UL4)	Unsigned long value	--	ulong	No	--	--
User Unsigned Long 5(UL5)	Unsigned long value	--	ulong	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

User Data Interval (PI_UPI)

Function

The User Data Interval (PI_UPI) record is a user-defined record that stores user-specific performance data per unit time. For 08-00 or earlier, this is a reserved record (field), which cannot be used in any operating systems.

Default values and values that can be specified

Item	Item	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_UPI_TT
- PI_UPI_TS_KEY
- PI_UPI_TD_KEY

Lifetime

From the time a jpcuser command is executed until the next jpcuser command is executed

Record size

- Fixed portion: 681 bytes
- Variable portion: 396 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Collect Time(API_TIME)	Time when data was converted	COPY	time_t	No	--	--
Interval(INTERVAL)	Interval for which the User Data Interval (PI_UPI) record was	COPY	ulong	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	stored (in seconds). The normal value is 0.	COPY	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always UPI.	COPY	char(8)	No	--	--
Trans Data Key(TD_KEY)	Transaction Data Key	COPY	ulong	No	--	--
Trans String Key(TS_KEY)	Transaction String Key	COPY	string(20)	No	--	--
Trans Type(TT)	Transaction Type	COPY	string(20)	No	--	--
User Float 1(F1)	Floating-point value	AVG	double	No	--	--
User Float 2(F2)	Floating-point value	AVG	double	No	--	--
User Float Roll 1(F1_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Roll 2(F2_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Long 1(L1)	Long integer value	AVG	long	No	--	--
User Long 2(L2)	Long integer value	AVG	long	No	--	--
User Long Roll 1(L1_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User Long Roll 2(L2_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User String 1(S1)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 2(S2)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 3(S3)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 4(S4)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 5(S5)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 6(S6)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 7(S7)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User Time 1(T1)	Time value (Time is displayed at the local time of the PFM - Web Console).	COPY	time_t	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
User Time 1(T1)	If the value is `0`, `1970 01 01 09:00:00` is displayed.	COPY	time_t	No	--	--
User Unsigned Long 1(UL1)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long 2(UL2)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long Roll 1(UL1_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--
User Unsigned Long Roll 2(UL2_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

User Data Interval - Expanded 1 (PI_XUI1)

Function

User Data Interval - Expanded 1 (PI_XUI1) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval - Extended (PI_UPIB) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

You can specify a value in the range from 0 to 32,767 seconds (which must not exceed the range specified by Collection Interval). This item is used to distribute the processing load resulting from the collection of multiple data items at a time. Note that the recorded data collection duration is the same as the value set for Collection Interval, regardless of the value set for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_XUI1_TT
- PI_XUI1_TS_KEY
- PI_XUI1_TD_KEY

Lifetime

From the execution of the jpcuser command until the execution of the next jpcuser command

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,148 bytes

Fields

PFM-View name (PFM - Manager name name)	Description	Smry	Format	Delta	Not sprtd on	Data source
Record Type(INPUT_RECORD_TYPE)	Record type (XUI1)	COPY	char (8)	No	--	--

PFM-View name (PFM - Manager name name)	Description	Smry	Format	Delta	Not sprtd on	Data source
Record Time(RECORD_TIME)	Time at which the record was created (GMT)	COPY	time_t	No	--	--
Interval(INTERVAL)	Interval for which the record was stored (in seconds). The normal value is 0.	COPY	ulong	No	--	--
Trans String Key(TS_KEY)	Transaction String Key	COPY	string(20)	No	--	--
Trans Data Key(TD_KEY)	Transaction Data Key	COPY	ulong	No	--	--
Trans Type(TT)	Transaction Type	COPY	string(20)	No	--	--
Collect Time(API_TIME)	Time when data was converted	COPY	time_t	No	--	--
User Float 01(F01)	Floating-point value	AVG	double	No	--	--
User Float 02(F02)	Floating-point value	AVG	double	No	--	--
User Float 03(F03)	Floating-point value	AVG	double	No	--	--
User Float 04(F04)	Floating-point value	AVG	double	No	--	--
User Float 05(F05)	Floating-point value	AVG	double	No	--	--
User Float 06(F06)	Floating-point value	AVG	double	No	--	--
User Float 07(F07)	Floating-point value	AVG	double	No	--	--
User Float 08(F08)	Floating-point value	AVG	double	No	--	--
User Float 09(F09)	Floating-point value	AVG	double	No	--	--
User Float 10(F10)	Floating-point value	AVG	double	No	--	--
User Float 11(F11)	Floating-point value	AVG	double	No	--	--
User Float 12(F12)	Floating-point value	AVG	double	No	--	--
User Float 13(F13)	Floating-point value	AVG	double	No	--	--
User Float 14(F14)	Floating-point value	AVG	double	No	--	--
User Float 15(F15)	Floating-point value	AVG	double	No	--	--
User Float 16(F16)	Floating-point value	AVG	double	No	--	--
User Float 17(F17)	Floating-point value	AVG	double	No	--	--
User Float 18(F18)	Floating-point value	AVG	double	No	--	--
User Float 19(F19)	Floating-point value	AVG	double	No	--	--
User Float 20(F20)	Floating-point value	AVG	double	No	--	--
User Float 21(F21)	Floating-point value	AVG	double	No	--	--
User Float 22(F22)	Floating-point value	AVG	double	No	--	--
User Float 23(F23)	Floating-point value	AVG	double	No	--	--

PFM-View name (PFM - Manager name name)	Description	Smry	Format	Delta	Not sprtd on	Data source
User Float 24(PI_XUI1_F24)	Floating-point value	AVG	double	No	--	--
User Float 25(F25)	Floating-point value	AVG	double	No	--	--
User Float 26(F26)	Floating-point value	AVG	double	No	--	--
User Float 27(F27)	Floating-point value	AVG	double	No	--	--
User Float 28(F28)	Floating-point value	AVG	double	No	--	--
User Float 29(F29)	Floating-point value	AVG	double	No	--	--
User Float 30(F30)	Floating-point value	AVG	double	No	--	--
User Float Add 01(F01_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 02(F02_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 03(F03_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 04(F04_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 05(F05_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 06(F06_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 07(F07_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 08(F08_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 09(F09_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 10(F10_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 11(F11_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 12(F12_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--

PFM-View name (PFM - Manager name name)	Description	Smry	Format	Delta	Not sprtd on	Data source
User Float Add 13(F13_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 14(F14_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 15(F15_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 16(F16_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 17(F17_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 18(F18_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 19(F19_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 20(F20_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 21(F21_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 22(F22_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 23(F23_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 24(F24_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 25(F25_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 26(F26_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 27(F27_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 28(F28_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--

PFM-View name (PFM - Manager name name)	Description	Smry	Format	Delta	Not sprtd on	Data source
User Float Add 29(F29_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Add 30(F30_A)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Time 1(T1)	Time value (Time is displayed at the local time of the PFM - Web Console) If the value is `0`, `1970 01 01 09:00:00` is displayed.	COPY	time_t	No	--	--
User String 1(S1)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 2(S2)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 3(S3)	Character-string 128 characters in length	COPY	string(128)	No	--	--

User Data Interval - Expanded 2 (PI_XUI2)

Function

User Data Interval - Expanded 2 (PI_XUI2) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval - Extended (PI_UPIB) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

You can specify a value in the range from 0 to 32,767 seconds (which must not exceed the range specified by Collection Interval). This item is used to distribute the processing load resulting from the collection of multiple data items at a time. Note that the recorded data collection duration is the same as the value set for Collection Interval, regardless of the value set for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_XUI2_TT
- PI_XUI2_TS_KEY
- PI_XUI2_TD_KEY

Lifetime

From the execution of the jpcuser command until the execution of the next jpcuser command

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,148 bytes

Fields

The value in the Record Type field is XUI2. The value in other fields is the same as that stored in the PI_XUI1 record.

User Data Interval - Expanded 3 (PI_XUI3)

Function

User Data Interval - Expanded 3 (PI_XUI3) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval - Extended (PI_UPIB) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

You can specify a value in the range from 0 to 32,767 seconds (which must not exceed the range specified by Collection Interval). This item is used to distribute the processing load resulting from the collection of multiple data items at a time. Note that the recorded data collection duration is the same as the value set for Collection Interval, regardless of the value set for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_XUI3_TT
- PI_XUI3_TS_KEY
- PI_XUI3_TD_KEY

Lifetime

From the execution of the jpcuser command until the execution of the next jpcuser command

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,148 bytes

Fields

The value in the Record Type field is XUI3. The value in other fields is the same as that stored in the PI_XUI1 record.

User Data Interval - Expanded 4 (PI_XUI4)

Function

User Data Interval - Expanded 4 (PI_XUI4) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval - Extended (PI_UPIB) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

You can specify a value in the range from 0 to 32,767 seconds (which must not exceed the range specified by Collection Interval). This item is used to distribute the processing load resulting from the collection of multiple data items at a time. Note that the recorded data collection duration is the same as the value set for Collection Interval, regardless of the value set for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_XUI4_TT
- PI_XUI4_TS_KEY
- PI_XUI4_TD_KEY

Lifetime

From the execution of the jpcuser command until the execution of the next jpcuser command

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,148 bytes

Fields

The value in the Record Type field is XUI4. The value in other fields is the same as that stored in the PI_XUI1 record.

User Data Interval - Expanded 5 (PI_XUI5)

Function

User Data Interval - Expanded 5 (PI_XUI5) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval - Extended (PI_UPIB) record.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

You can specify a value in the range from 0 to 32,767 seconds (which must not exceed the range specified by Collection Interval). This item is used to distribute the processing load resulting from the collection of multiple data items at a time. Note that the recorded data collection duration is the same as the value set for Collection Interval, regardless of the value set for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_XUI5_TT
- PI_XUI5_TS_KEY
- PI_XUI5_TD_KEY

Lifetime

From the execution of the jpcuser command until the execution of the next jpcuser command

Record size

- Fixed portion: 681 bytes
- Variable portion: 1,148 bytes

Fields

The value in the Record Type field is XUI5. The value in other fields is the same as that stored in the PI_XUI1 record.

User Data Interval - Extended (PI_UPIB)

Function

The User Data Interval - Extended (PI_UPIB) record is a user-defined record that stores user-specific performance data per unit time. This record has more fields than the User Data Interval (PI_UPI) record. For 08-00 or earlier, this is a reserved record (field), which cannot be used in any operating systems.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

- PI_UPIB_TT
- PI_UPIB_TS_KEY
- PI_UPIB_TD_KEY

Lifetime

From the time a jpcuser command is executed until the next jpcuser command is executed

Record size

- Fixed portion: 681 bytes
- Variable portion: 992 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Collect Time(API_TIME)	Time when data was converted	COPY	time_t	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Interval(INTERVAL)	Interval for which the User Data Interval - Extended (PI_UPIB) record was stored (in seconds). The normal value is 0.	COPY	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always UPIB.	COPY	char(8)	No	--	--
Trans Data Key(TD_KEY)	Transaction Data Key	COPY	ulong	No	--	--
Trans String Key(TS_KEY)	Transaction String Key	COPY	string(20)	No	--	--
Trans Type(TT)	Transaction Type	COPY	string(20)	No	--	--
User Float 1(F1)	Floating-point value	AVG	double	No	--	--
User Float 2(F2)	Floating-point value	AVG	double	No	--	--
User Float 3(F3)	Floating-point value	AVG	double	No	--	--
User Float 4(F4)	Floating-point value	AVG	double	No	--	--
User Float 5(F5)	Floating-point value	AVG	double	No	--	--
User Float Roll 1(F1_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Roll 2(F2_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Roll 3(F3_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Roll 4(F4_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Float Roll 5(F5_R)	Accumulated floating-point value (This data is added when this data is summarized)	ADD	double	No	--	--
User Long 1(L1)	Long integer value	AVG	long	No	--	--
User Long 2(L2)	Long integer value	AVG	long	No	--	--
User Long 3(L3)	Long integer value	AVG	long	No	--	--
User Long 4(L4)	Long integer value	AVG	long	No	--	--
User Long 5(L5)	Long integer value	AVG	long	No	--	--
User Long Roll 1(L1_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
User Long Roll 2(L2_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User Long Roll 3(L3_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User Long Roll 4(L4_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User Long Roll 5(L5_R)	Accumulated long integer value (This data is added when this data is summarized)	ADD	long	No	--	--
User String 1(S1)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 10(S10)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 11(S11)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 12(S12)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 13(S13)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 14(S14)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 15(S15)	Character-string 64 characters in length	COPY	string(64)	No	--	--
User String 2(S2)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 3(S3)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 4(S4)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 5(S5)	Character-string 16 characters in length	COPY	string(16)	No	--	--
User String 6(S6)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 7(S7)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 8(S8)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User String 9(S9)	Character-string 32 characters in length	COPY	string(32)	No	--	--
User Time 1(T1)	Time value (Time is displayed at the local time of the PFM - Web Console).	COPY	time_t	No	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
User Time 1(T1)	If the value is `0`, `1970 01 01 09:00:00` is displayed.	COPY	time_t	No	--	--
User Unsigned Long 1(UL1)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long 2(UL2)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long 3(UL3)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long 4(UL4)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long 5(UL5)	Unsigned long value	AVG	ulong	No	--	--
User Unsigned Long Roll 1(UL1_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--
User Unsigned Long Roll 2(UL2_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--
User Unsigned Long Roll 3(UL3_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--
User Unsigned Long Roll 4(UL4_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--
User Unsigned Long Roll 5(UL5_R)	Accumulated unsigned long value (This data is added when this data is summarized)	ADD	ulong	No	--	--

Legend:

Smry: Summary

Not sprtd in: Not supported in

User Summary (PD_USER)

Function

The User Summary (PD_USER) record stores the performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each user at a given point in time. A record is created for each user ID. This is a multi-instance record. Note that even if the Process Detail (PD) record is not collected, performance data is stored in the User Summary (PD_USER) record.

Note:

- If the target process of an instance is a zombie process, 0 is set in the field listed below for Solaris, and -3 is set for AIX and HP-UX.
  User ID

Default values and values that can be specified

Item	Default value	Modifiable
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	No	No
Sync Collection With	Detail Records, PD	No
Realtime Report Data Collection Mode	Reschedule	Yes

ODBC key fields

PD_USER_REAL_USER_ID

Lifetime

From the time when at least one process with a particular user name is being executed until the time when the number of processes being executed becomes 0.

Record size

- Fixed portion: 681 bytes
- Variable portion: 348 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Avg I/O Kbytes(AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ READ_OPS + WRITE_OPS
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	--	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ total-elapsed-time-of-all-processes-

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	--	float	No	--	executed-by-user) ÷ processors-count) × 100
Context Switches(CONTEXT_SWITCHES)	Number of times context switching was executed	--	ulong	No	HP-UX, Linux	--
Interval(INTERVAL)	Interval at which the User Summary (PD_USER) record was stored (seconds). Always 0.	--	ulong	No	HP-UX, Solaris, AIX, Linux	--
Major Faults(MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	--	ulong	No	--	--
Minor Faults(MINOR_FAULTS)	Number of page faults that do not cause physical I/O operations	--	ulong	No	--	--
Process Count(PROCESS_COUNT)	Number of processes executed by users	--	ulong	No	--	--
Reads(READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	--	ulong	No	HP-UX, Linux	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	--	float	No	HP-UX, Linux	READ_OPS ÷ total-elapsed-time-of-all-processes-belonging-to-a-user-account
Real Mem Kbytes(REAL_MEMORY_KBYTES)	Physical memory size used (KB)	--	float	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	--	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always USER.	--	char(8)	No	--	--
Signals Rcvd(NUMBER_OF_SIGNALS_RECEIVED)	Number of signals received	--	ulong	No	Linux	--
Swaps(SWAPS)	Number of times swapping occurred	--	ulong	No	Linux	--
System CPU(SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode (seconds)	--	utime	No	--	--
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	No	AIX, Linux	TOTAL_IO_KBYTES ÷ total-elapsed-time-of-all-processes-

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	--	float	No	AIX, Linux	belonging-to-a-user-account
Total I/O Kbytes(TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	--	double	No	AIX, Linux	--
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	--	ulong	No	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS
Total I/O Ops/sec(TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	--	float	No	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ total-process-time-of-all-processes-belonging-to-a-user-account
User(REAL_USER_NAME)	Effective user name	--	string(256)	No	--	--
User CPU(USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	--	utime	No	--	--
User ID(REAL_USER_ID)	Effective user ID	--	long	No	--	--
Virtual Mem Kbytes(VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (KB)	--	float	No	Solaris	--
Writes(WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	--	ulong	No	HP-UX, Linux	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	--	float	No	HP-UX, Linux	WRITE_OPS ÷ total-elapsed-time-of-all-processes-belonging-to-a-user-account

Legend:

Smry: Summary

Not sprtd in: Not supported in

Workgroup Summary (PI_WGRP)

Function

The Workgroup Summary (PI_WGRP) record stores the Performance data in which the data stored in the Process Detail (PD) record is summarized as the state for each workgroup at a given point in time. A record is created for each workgroup. For details about how to collect workgroup information, see *7. User-Defined Record Collection*.

A workgroup named Other is automatically created for all process records that do not belong to any of the defined workgroups.

This is a multi-instance record.

Note that even if the Process Detail (PD) record is not collected, performance data is stored in the Workgroup Summary (PI_WGRP) record.

Notes:

- When there are many objects to be collected because, for example, there are many processes on the machine, increased throughput results in a heavier processing load.
- To change a workgroup definition, you need to restart PFM - Agent for Platform.
- Group names are based on actual group names, and not on effective group names.
- This record does not support collection in a logical host environment. Collect this record in a physical host environment.

Default values and values that can be specified

Item	Default value	Modifiable
Collection Interval	60	Yes
Collection Offset#	0	Yes
Log	No	Yes
LOGIF	Blank	Yes
Over 10 Sec Collection Time	Yes	No
Realtime Report Data Collection Mode	Reschedule	Yes

#

A value between 0 and 32,767 seconds can be specified (a value within the range specified for Collection Interval). Use this item to distribute the collection load, because executing data collection all at once results in concentration of the processing workload. Note that the data collection duration to be recorded is the same as the Collection Interval, regardless of the value specified for Collection Offset.

When changing the Collection Offset value, specify a value that takes into account the impact of the collection processing load.

ODBC key fields

PI_WGRP_WORKGROUP_NAME

Lifetime

None

Record size

- Fixed portion: 681 bytes
- Variable portion: 278 bytes

Fields

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Argument Lists(ARGUMENT_LISTS)	Process argument list. If the value exceeds 30 bytes, the character becomes >. This field shows the character string specified in the arguments_02 parameter of the wgfile file.	COPY	string(30)	No	--	--
Avg I/O Kbytes(AVG_IO_KBYTES)	Average transfer size in I/O operations (KB)	COPY	float	No	HP-UX, AIX, Linux	TOTAL_IO_KBYTES ÷ (READ_OPS + WRITE_OPS)
CPU % (CPU_PERCENT_USED)	Average value obtained by dividing CPU usage by the number of processors (%)	COPY	float	No	--	(((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ INTERVAL) ÷ processors-count) × 100
CPU % Each(CPU_PERCENT_USED_EACH)	CPU usage by workgroups	COPY	float	No	--	((USER_CPU_TIME + SYSTEM_CPU_TIME) ÷ INTERVAL) × 100
Context Switches(CONTEXT_SWITCHES)	Number of times context switching was executed	COPY	ulong	Yes	HP-UX, Linux	--
Groups(GROUPS)	Group name list. If the value exceeds 30 bytes, the character becomes >. This field shows the character string specified in the groups parameter of the wgfile file. This field shows the character string specified in the groups or groups_02 parameter of the wgfile file in version 08-00 or later.	COPY	string(30)	No	--	PFM - Agent for Platform adds the process group name to this field for each process the workgroup executes.
Interval(INTERVAL)	Interval at which the Workgroup Summary (PI_WGRP) record was stored (seconds)	COPY	ulong	Yes	--	When the delta value is not collected, total of the operating time of each process. When the delta value is collected, RECORD_TIME - previous record time
Major Faults(MAJOR_FAULTS)	Number of page faults that cause physical I/O operations	COPY	ulong	Yes	--	--

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Process Count(PROCESS_COUNT)	Number of processes executed by workgroups	COPY	ulong	No	--	Number of processes in the process table that have user properties, group properties, or program properties that match the workgroup definition
Programs(PROGRAMS)	Program name list. If the value exceeds 30 bytes, the character becomes >. This field shows the character string specified in the programs parameter of the wgfile file. This field shows the character string specified in the programs or programs_02 parameter of the wgfile file in version 08-00 or later.	COPY	string(30)	No	--	PFM - Agent for Platform adds the process program name to this field for each process the workgroup executes.
Reads(READ_OPS)	In AIX, this field shows the number of times RAW read operations occurred. In Solaris, this field shows the number of times Block read operation occurred.	COPY	ulong	Yes	HP-UX, Linux	--
Reads/sec(READ_OPS_PER_SECOND)	Frequency at which read operations occurred (number of operations per second)	R	float	Yes	HP-UX, Linux	READ_OPS ÷ total-elapsed-time-of-all-processes-executed-by-workgroup
Real Mem Kbytes(REAL_MEMORY_KBYTES)	Physical memory size used (KB)	COPY	ulong	No	--	--
Record Time(RECORD_TIME)	Time at which the record was created (Greenwich Mean Time)	COPY	time_t	No	--	--
Record Type(INPUT_RECORD_TYPE)	Record type. Always WGRP.	COPY	char(8)	No	--	--
Swaps(SWAPS)	Number of times swapping occurred	COPY	ulong	Yes	Linux	--
System CPU(SYSTEM_CPU_TIME)	Time during which the system ran in the kernel mode (seconds)	COPY	utime	Yes	--	--
Throughput/sec(IO_KBYTES_PER_SECOND)	I/O operation speed (KB per second)	R	float	Yes	AIX, Linux	TOTAL_IO_KBYTES ÷ total-elapsed-time-of-all-processes-executed-by-workgroup
Total I/O Kbytes(TOTAL_IO_KBYTES)	Total size transferred in I/O operations (KB)	COPY	double	Yes	AIX, Linux	--
Total I/O Ops(TOTAL_IO_OPS)	Number of I/O operations that occurred	COPY	ulong	Yes	HP-UX, AIX, Linux	READ_OPS + WRITE_OPS

PFM-View name (PFM-Manager name)	Description	Smry	Format	Delta	Not sprtd in	Data source
Total I/O Ops/sec(TOTAL_IO_OPS_PER_SECOND)	Frequency at which I/O operations occurred (number of operations per second)	R	float	Yes	HP-UX, AIX, Linux	TOTAL_IO_OPS ÷ total-elapsed-time-of-all-processes-executed-by-workgroup
User CPU(USER_CPU_TIME)	Time during which the CPU ran in the user mode (seconds)	COPY	utime	Yes	--	--
Users(USERS)	User name list. If the value exceeds 30 bytes, the character becomes >. This field shows the character string specified in the users parameter of the wgfile file. This field shows the character string specified in the users or users_02 parameter of the wgfile file in version 08-00 or later.	COPY	string(30)	No	--	PFM - Agent for Platform adds the process user name to this field for each process the workgroup executes.
Virtual Mem Kbytes(VIRTUAL_MEMORY_KBYTES)	Size of the virtual memory used (KB)	COPY	ulong	No	Solaris	--
Workgroup(WORKGROUP_NAME)	Workgroup name. Other is displayed if a process does not belong to any of the defined workgroups. This field shows the character string specified in the workgroup parameter of the wgfile file.	COPY	string(30)	No	--	--
Writes(WRITE_OPS)	In AIX, this field shows the number of times RAW write operations occurred. In Solaris, this field shows the number of times Block write operation occurred.	COPY	ulong	Yes	HP-UX, Linux	--
Writes/sec(WRITE_OPS_PER_SECOND)	Frequency at which write operations occurred (number of operations per second)	R	float	Yes	HP-UX, Linux	WRITE_OPS ÷ total-elapsed-time-of-all-processes-executed-by-workgroup

10 Commands

This chapter describes the syntax of the command provided by PFM - Agent for Platform.

Command description format

This section describes the command descriptive format, including the command specification format and the notations used in explaining the command syntax.

Command specification format

The command specification format is as follows:

```
jpcxxx  [-option-A [value-a[, value-b[, value-c...]]]] ...(1)
        [-option-B [value-a[, value-b[, value-c...]]]] ...(1)  } ...(2)
        [any-name-X [any-name-Y [any-name-Z...]]]
```

The items indicated by (1) and (2) are called *options* and *arguments*, respectively.

Notations used in command syntax descriptions

The notations used in command syntax descriptions are as follows.

[ ] (square brackets)

Square brackets enclose items whose specification can be omitted (optional items).

jpcappcvt

Format

```
jpcappcvt      [-n]
```

Description

The `jpcappcvt` command converts application definitions that were set in version 09-00 or earlier (settings for collecting information about application operating status) to application definitions that will work in 10-00 and later versions (settings for collecting information about the operating status of processes).

Execution permission

Users with root user permissions.

Storage directory

`/opt/jp1pc/agtu/agent/`

Arguments

`-n`

Specification of this option suppresses output of query messages that interrupt program execution, thereby eliminating the need for user response. Specify this option to execute the command non-interactively.

Notes

- This command cannot be executed on a logical host.
- If the `-n` option is not specified, the command is executed interactively, in which case the user must enter responses to the query messages that are displayed on the screen in order for the conversion processing to be performed.
- When you execute this command, the settings are enabled automatically as soon as conversion occurs, so there is no need to restart the Agent Collector service.
- Although application definitions are created by converting earlier definitions to a format that is compatible with version 10-00 and later, the application definitions from version 09-00 or earlier are retained.
- If application definitions from version 10-00 or later already exist when this command is executed, the `KAVF10806-Q` message is output, requesting confirmation that you want to overwrite the existing settings. However, if the command is being executed in the non-interactive mode, this message will not be output, and the existing settings will be forcibly overwritten.
- The methods of collecting process information are different for the Application Summary (`PD_APP`) record, which is used to collect information about the application operating status, from the Application Process Detail (`PD_APPD`) and Application Summary Extension (`PD_APP2`) records, which are used to collect information about the operating status of processes. A consequence of this is that the values displayed in some fields will not be the same after this command has executed.

  If the values of the following fields are different from the values of the corresponding fields of the Application Summary (`PD_APP`) record, review and, if necessary, revise the monitoring conditions:

  - Application Process Detail (`PD_APPD`) record

    `Monitoring Count` field

    `Monitoring Status` field

- Application Summary Extension (PD_APP2) record
  Application Status field
  Application Exist field

The following table shows the correspondence between the fields of the Application Summary (PD_APP) record and the fields of the Application Process Detail (PD_APPD) and Application Summary Extension (PD_APP2) records.

Table 10–1: Correspondence between the fields of the Application Summary (PD_APP) record and the fields of the Application Process Detail (PD_APPD) and Application Summary Extension (PD_APP2) records

Field of the Application Summary (PD_APP) record	Field of the Application Process Detail (PD_APPD) record	Field of the Application Summary Extension (PD_APP2) record
Application Name	Application Name	Application Name
Application Status	--	Application Status
Application Exist	--	Application Exist
Process*XX* Count#	Monitoring Count	--
Process*XX* Range#	Monitoring Min Monitoring Max	--
Process*XX* Status#	Monitoring Status	--
Process*XX* Kind#	Monitoring Field	--
Process*XX* Name#	Monitoring Condition	--
Virtual Env ID	--	Virtual Env ID

Legend:

--: No corresponding field exists.

#

XX is a two-digit numeric value in the range from 01 to 15. *XX* corresponds to the numeric value in the Monitoring Label field (Monitoring*XX*) of the Application Process Detail (PD_APPD) record.

Return values

0	Normal termination
Other than 0	Abnormal termination

Usage example

The following is an example of executing the command interactively:

```
# /opt/jp1pc/agtu/agent/jpcappcvt
KAVF10800-Q Do you want to convert? (Y/N)
KAVF10806-Q A settings for collecting process operation and non-operation in
formation already exists. Would you like to overwrite it? (Y/N)
KAVF10801-I Conversion of a setting was successful.
```

11 Messages

This chapter explains the PFM - Agent for Platform message format, message output destinations, syslogs, and messages.

11.1 Message format

This section shows the format of the messages that are output by PFM - Agent for Platform and how they are described in the manual.

11.1.1 Format of output messages

This section explains the format of the messages that are output by PFM - Agent for Platform. A message consists of a message ID followed by message text. The message format follows:

```
KAVFnnnnn-Y message-text
```

The message ID indicates the following content:

`K`

Indicates the system identifier.

`AVF`

Indicates that this is a PFM - Agent message.

nnnnn

Indicates a message serial number. The message number for PFM - Agent for Platform is `10`*xxx*.

Y

Indicates a message type.

- `E`: Error
  Processing is halted.
- `W`: Warning
  Processing continues after the message is issued.
- `I`: Information
  Provides information to the user.
- `Q`: Response
  Prompts the user to respond.

The correspondence between the message type and the syslog priority level is explained below.

`-E`

- Level: `LOG_ERR`
- Meaning: Error message

`-W`

- Level: `LOG_WARNING`
- Meaning: Warning message

`-I`

- Level: `LOG_INFO`
- Meaning: Additional information message

`-Q`

(Not output)

11.1.2 Format of message explanations

This section explains the message description format in this manual. The italicized portion of the message text means the display content differs depending on the circumstance under which the message is issued. Messages are listed in the order of message ID. An example of the message description format follows.

Message ID

Message text

Message explanation

(S)

Indicates the action taken by the system.

(O)

Indicates the action that should be taken by the operator when the message is displayed.

Reference note

The system administrator who has been contacted by an operator must collect log information as described in *12. Error Handling Procedures*, and conduct an initial investigation.

When conducting an initial investigation of the problem, refer to the OS log information (`syslog`) and the various types of log information that is output by PFM - Agent for Platform. From these types of log information, determine what happened during the period when the problem occurred and take actions to avoid or correct the problem. Also record the operating method used that led to the problem. At the same time, try to determine whether the problem can be reproduced.

11.2 Message output destinations

This section lists the output destinations of the messages that are output by PFM - Agent for Platform.

The following conventions are used in the table:

Legend:

Y: Message is output.

--: No message is output.

Table 11–1: PFM - Agent for Platform message output destinations

Message ID	Output destination								
	syslog	Cmn msg log	Std out	Std err	Public log	Debug log	Cnvt log	JP1 system event#1	Agent event#2
KAVF10001	Y	Y	--	--	--	--	--	--	--
KAVF10002	Y	Y	--	--	--	--	--	--	--
KAVF10003	Y	Y	--	--	--	--	--	--	--
KAVF10004	Y	Y	--	--	--	--	--	Y	Y
KAVF10010	--	Y	--	--	--	--	--	--	--
KAVF10011	--	Y	--	--	--	--	--	--	--
KAVF10013	--	Y	--	--	--	--	--	--	--
KAVF10100	--	Y	--	--	--	--	--	--	--
KAVF10101	Y	Y	--	--	--	--	--	--	--
KAVF10102	Y	Y	--	--	--	--	--	--	--
KAVF10103	Y	Y	--	--	--	--	--	--	--
KAVF10105	--	Y	--	--	--	--	--	--	--
KAVF10106	--	Y	--	--	--	--	--	--	--
KAVF10107	--	Y	--	--	--	--	--	--	--
KAVF10150	--	Y	--	--	--	--	--	Y	Y
KAVF10200	Y	Y	--	--	--	--	--	--	--
KAVF10201	Y	Y	--	--	--	--	--	--	--
KAVF10202	Y	Y	--	--	--	--	--	--	--
KAVF10203	--	Y	--	--	--	--	--	--	--
KAVF10204	--	Y	--	--	--	--	--	Y	Y
KAVF10205	--	Y	--	--	--	--	--	Y	Y
KAVF10206	--	Y	--	--	--	--	--	Y	Y
KAVF10300	--	Y	--	--	--	--	--	Y	Y
KAVF10301	--	Y	--	--	--	--	--	Y	Y
KAVF10500	--	Y	--	--	--	--	--	Y	Y
KAVF10501	--	Y	--	--	--	--	--	Y	Y

Message ID	Output destination								
	syslog	Cmn msg log	Std out	Std err	Public log	Debug log	Cnvt log	JP1 system event#1	Agent event#2
KAVF10502	--	Y	--	--	--	--	--	Y	Y
KAVF10503	--	Y	--	--	--	--	--	--	--
KAVF10504	Y	Y	--	--	--	--	--	--	--
KAVF10550	--	--	--	Y	--	--	--	--	--
KAVF10551	--	--	--	Y	--	--	--	--	--
KAVF10552	--	--	--	Y	--	--	--	--	--
KAVF10600	--	Y	--	--	--	--	--	--	--
KAVF10700	--	Y	--	--	--	--	--	--	--
KAVF10701	--	Y	--	--	--	--	--	--	--
KAVF10703	--	Y	--	--	--	--	--	--	--
KAVF10704	--	Y	--	--	--	--	--	--	--
KAVF10705	--	Y	--	--	--	--	--	--	--
KAVF10706	--	Y	--	--	--	--	--	--	--
KAVF10707	--	Y	--	--	--	--	--	--	--
KAVF10709	--	Y	--	--	--	--	--	--	--
KAVF10710	--	Y	--	--	--	--	--	--	--
KAVF10711	--	Y	--	--	--	--	--	--	--
KAVF10713	--	Y	--	--	--	--	--	--	--
KAVF10800	--	--	Y	--	--	--	--	--	--
KAVF10801	--	--	Y	--	--	--	Y	--	--
KAVF10802	--	--	--	Y	--	--	Y	--	--
KAVF10803	--	--	Y	--	--	--	Y	--	--
KAVF10804	--	--	Y	--	--	--	Y	--	--
KAVF10805	--	--	Y	--	--	--	Y	--	--
KAVF10806	--	--	Y	--	--	--	--	--	--
KAVF10807	--	--	--	Y	--	--	Y	--	--
KAVF10808	--	--	--	Y	--	--	Y	--	--
KAVF10809	--	--	--	Y	--	--	Y	--	--
KAVF10810	--	--	--	Y	--	--	Y	--	--
KAVF10850	--	--	--	Y	--	--	--	--	--
KAVF10851	--	--	--	Y	--	--	--	--	--
KAVF10852	--	--	--	Y	--	--	--	--	--
KAVF10853	--	--	--	Y	--	--	--	--	--
KAVF10901	--	--	--	--	--	Y	--	--	--

Message ID	Output destination								
	syslog	Cmn msg log	Std out	Std err	Public log	Debug log	Cnvt log	JP1 system event#1	Agent event#2
KAVF10902	--	--	--	--	--	Y	--	--	--
KAVF10904	--	--	--	--	--	Y	--	--	--
KAVF10905	--	--	--	--	--	Y	--	--	--
KAVF10906	--	--	--	--	--	Y	--	--	--
KAVF10907	--	--	--	--	--	Y	--	--	--
KAVF10908	--	--	--	--	--	Y	--	--	--
KAVF10909	--	--	--	--	--	Y	--	--	--
KAVF10910	--	--	--	--	--	Y	--	--	--
KAVF10911	--	--	--	--	--	Y	--	--	--
KAVF10912	--	--	--	--	--	Y	--	--	--
KAVF10913	--	--	--	--	--	Y	--	--	--
KAVF10914	--	--	--	--	--	Y	--	--	--
KAVF10915	--	--	--	--	--	Y	--	--	--
KAVF10916	--	--	--	Y	Y	--	--	--	--
KAVF10917	--	--	--	Y	Y	--	--	--	--
KAVF10919	--	--	--	Y	Y	--	--	--	--
KAVF10920	--	--	--	Y	Y	--	--	--	--
KAVF10923	--	--	--	Y	Y	--	--	--	--
KAVF10924	--	--	--	--	Y	--	--	--	--
KAVF10925	--	--	--	--	Y	--	--	--	--
KAVF10926	--	--	--	--	Y	--	--	--	--
KAVF10927	--	--	--	--	Y	--	--	--	--
KAVF10928	--	--	--	--	Y	--	--	--	--
KAVF10929	--	--	--	--	--	Y	--	--	--
KAVF10930	--	--	--	--	--	Y	--	--	--
KAVF10931	--	--	--	--	--	Y	--	--	--
KAVF10932	--	--	--	--	--	Y	--	--	--
KAVF10935	--	--	--	Y	Y	--	--	--	--
KAVF10941	--	--	--	Y	--	--	--	--	--
KAVF10943	--	--	--	Y	Y	--	--	--	--
KAVF10948	--	--	--	--	Y	--	--	--	--
KAVF10950	--	--	--	--	--	Y	--	--	--
KAVF10951	--	--	--	--	--	Y	--	--	--
KAVF10952	--	--	--	--	Y	--	--	--	--

Message ID	Output destination								
	syslog	Cmn msg log	Std out	Std err	Public log	Debug log	Cnvt log	JP1 system event#1	Agent event#2
KAVF10953	--	--	--	--	--	Y	--	--	--
KAVF10954	--	--	--	--	Y	--	--	--	--
KAVF10957	--	--	Y	--	--	--	--	--	--
KAVF10982	--	--	--	Y	Y	--	--	--	--
KAVF10983	--	--	--	Y	Y	--	--	--	--
KAVF10984	--	--	--	Y	Y	--	--	--	--
KAVF10985	--	--	--	Y	Y	--	--	--	--
KAVF10986	--	--	--	--	--	Y	--	--	--
KAVF10987	--	--	--	--	--	Y	--	--	--
KAVF10988	--	--	--	--	--	Y	--	--	--
KAVF10989	--	--	--	--	--	Y	--	--	--
KAVF10990	--	--	--	--	--	Y	--	--	--
KAVF10991	--	--	--	--	--	Y	--	--	--
KAVF10992	--	--	--	--	--	Y	--	--	--
KAVF10993	--	--	--	--	--	Y	--	--	--
KAVF10994	--	--	--	--	--	Y	--	--	--
KAVF10995	--	--	--	--	--	Y	--	--	--
KAVF10996	--	--	--	--	--	Y	--	--	--
KAVF10997	--	--	--	--	--	Y	--	--	--
KAVF10998	--	--	--	--	--	Y	--	--	--

Legend:

Cmn msg log: Common message log

Std out: Standard output

Std err: Standard error output

Cnvt log: Convert log

#1

A JP1 system event notifies JP1/IM of changes in the agent status. For details about the JP1 system events, see the chapter explaining operation monitoring in linkage with the integrated management product (JP1/IM) in the *JP1/Performance Management User's Guide*.

The following table lists the prerequisite programs for issuing JP1 system events.

Table 11–2: Prerequisite programs for issuing JP1 system events

Host type	Prerequisite program	Version
Monitoring manager	PFM - Manager	09-00 or later
Monitoring console server	PFM - Web Console	08-00 or later
Monitoring agent	PFM - Agent for Platform	08-00 or later

Host type	Prerequisite program	Version
Monitoring agent	PFM - Agent for Platform	(To issue an event output by PFM - Agent, the version must be 09-00 or later.)
	PFM - Manager or PFM - Base	09-00 or later
	JP1/Base	09-00 or later

#2

An agent event notifies PFM - Manager of changes in the agent status. For details about the agent events, see the chapter explaining event display in the *JP1/Performance Management User's Guide*.

The following table lists the prerequisite programs for issuing agent events.

Table 11–3: Prerequisite programs for issuing agent events

Host type	Prerequisite program	Version
Monitoring manager	PFM - Manager	09-00 or later
Monitoring console server	PFM - Web Console	08-00 or later
Monitoring agent	PFM - Agent for Platform	09-00 or later
	PFM - Manager or PFM - Base	09-00 or later

11.3 List of messages output to syslog

This section shows the list of messages that PFM - Agent for Platform outputs to syslog.

Syslogs are output to the syslog file. For details about the syslog file storage location, see the syslog daemon configuration file (the default is `/etc/syslogd.conf`).

The table below shows the message information that is output by PFM - Agent for Platform to syslog.

Table 11–4: Message information that is output to syslog

Message ID	syslog	
	Facility	Level
KAVF10001-I	LOG_DAEMON	LOG_INFO
KAVF10002-E	LOG_DAEMON	LOG_ERR
KAVF10003-I	LOG_DAEMON	LOG_INFO
KAVF10004-E	LOG_DAEMON	LOG_ERR
KAVF10101-E	LOG_DAEMON	LOG_ERR
KAVF10102-E	LOG_DAEMON	LOG_ERR
KAVF10103-E	LOG_DAEMON	LOG_ERR
KAVF10200-E	LOG_DAEMON	LOG_ERR
KAVF10201-E	LOG_DAEMON	LOG_ERR
KAVF10202-E	LOG_DAEMON	LOG_ERR
KAVF10504-E	LOG_DAEMON	LOG_ERR

11.4 Messages

This section explains the messages that are output by PFM - Agent for Platform and how to handle them. The messages that are output by PFM - Agent for Platform are listed below.

KAVF10001-I

Agent Collector has stopped. (host=*host-name*, service=*host-name*<UNIX>)

The Agent Collector service terminated normally.

(S)

Terminates processing of the Agent Collector service.

KAVF10002-E

Agent Collector failed to start.

Startup of the Agent Collector service failed.

(S)

Terminates processing of the Agent Collector service.

(O)

Check the preceding message that was output to the common message log and take the action indicated in it.

KAVF10003-I

Agent Collector has started. (host=*host-name*, service=*host-name*<UNIX>)

Startup of the Agent Collector service was completed.

(S)

Begins collecting performance data on the Agent Collector service.

KAVF10004-E

Agent Collector stopped abnormally.

Agent Collector service terminated abnormally.

(S)

Terminates processing of the Agent Collector service.

(O)

Check the preceding message that was output to the common message log and take the action indicated in it. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10010-I

Name Server has been connected.

During the start of the Agent Collector service, the Name Server service was successfully connected.

(S)

Continues the start of the Agent Collector service.

KAVF10011-I

Agent Store has been connected.

During the start of the Agent Collector service, the Agent Store service was successfully connected.

(S)

Continues the start of the Agent Collector service.

KAVF10013-W

A user command failed to start. (record=*record-id*, en=*OS-detail-code*)

An attempt to start a user command has failed.

(S)

Continues subsequent processing.

(O)

To find out if there are any problems with the specified user command, check the following:

- Is the user command correctly specified?
- Does the specified user command exist?
- Do you have the permission to execute the specified user command?

KAVF10100-E

An error occurred in an OS API (*API-name*). (en=*OS-detail-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

An error occurred in the OS API. The code indicated by `en` is the error number (`errno`) of the system call.

(S)

If processing cannot continue because of an error during startup, the system terminates the processing of the Agent Collector service.

(O)

If this message is output during the start of the Agent Collector service, check whether the log file specified in the Messages File property exists. If not, change the log file name and then restart PFM - Agent for Platform. If the log file does exist, or for all other cases, check the OS detail code. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10101-E

An error occurred in a function (*function-name*). (rc=*maintenance-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

An error occurred in a function being controlled.

(S)

Terminates the processing of the Agent Collector service.

(O)

Collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10102-E

A signal has been accepted. (sn=*signal-number*)

A signal interrupt occurred. The code indicated by `sn` is the OS signal number.

(S)

Terminates the processing of the Agent Collector service.

(O)

Collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10103-E

An exception occurred. (msg=*exception-message*)

An exception occurred during the processing of the Agent Collector service.

(S)

Terminates the processing of the Agent Collector service.

(O)

Check whether a shortage has not occurred in the system resources such as memory. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10105-W

The file does not exist in the path specified by using Web Console. (PL_MESS) (*API-name*) (en= *OS-detail-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

The file indicated by *argument-1* does not exist in the path specified for collecting the record.

(S)

Continues the processing of the Agent Collector service.

(O)

Use PFM - Web Console to specify the file path, and then restart Agent Collector.

KAVF10106-W

The information to be collected has not been set in Web Console. (PD_APP)

The information to be collected for the record has not been set in Web Console.

(S)

Continues the processing of the Agent Collector service.

(O)

Use PFM - Web Console to set the information to be collected, and then collect the record.

KAVF10107-W

The path name specified by using Web Console exceeds the 256-byte limit. (*record-ID*) (filename)

The length of the characters specified as the path name for collecting the record indicated by *record-ID* is too long.

(S)

Continues the processing of the Agent Collector service.

(O)

Use PFM - Web Console to specify the file path, and then restart Agent Collector.

KAVF10150-W

An error occurred in an OS API (*API-name*). (en=*OS-detail-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

An error occurred in the OS API. The code indicated by `en` is the error number (`errno`) of the system call.

(S)

Continues the processing of the Agent Collector service.

If *argument-1* is `jpc_ufss`, the Agent Collector service cannot collect the User File System Storage (`PD_UFSS`) record.

(O)

Check the OS detail code, remove the error cause, and then restart the Agent Collector service. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10200-E

An attempt to initialize a connection failed.

An attempt to establish connection failed during the start of the Agent Collector service.

(S)

Terminates the start of the Agent Collector service.

(O)

Check whether the applicable host can be connected to the network. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10201-E

Initialization of Agent Configuration failed.

Initialization of Agent Configuration failed during the start of the Agent Collector service.

(S)

Terminates the start of the Agent Collector service.

(O)

Check whether the service startup initialization file (`jpcagt.ini`) is located under the following directory:

- For a physical host: `/opt/jp1pc/agtu/agent`
- For a logical host: *environment-directory*`/jp1pc/agtu/agent`

If the `jpcagt.ini` file does not exist, copy (into the relevant directory) the backup copy of the `jpcagt.ini` file. If the `jpcagt.ini` file that was backed up does not exist, copy the `jpcagt.ini.model` file into `/opt/jp1pc/agtu/agent` or *environment-directory*`/jp1pc/agtu/agent`, and then rename the file to `jpcagt.ini`.

Even when a logical host is used, use the `jpcagt.ini.model` file in the `/opt/jp1pc/agtu/agent` directory.

If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10202-E

An attempt to initialize a record failed.

An attempt to initialize a record failed during the start of the Agent Collector service.

(S)

Terminates the start of the Agent Collector service.

(O)

Collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10203-W

Invalid entry in *file-name*.

An error was found in the definition file or specification of the user command properties.

(S)

Ignores the definition content and continues the start of the Agent Collector service.

(O)

Correct the definition in the displayed file and then restart the service. Alternatively, correct the user command properties.

KAVF10204-W

An attempt to collect the OS information failed.

An attempt to collect the OS version information failed.

(S)

Continues the processing of the Agent Collector service.
However, the field values of the following records cannot be correctly collected:

CPU - Per Processor Detail (PI_CPUP)

- Boot Time (SYSTEM_BOOT_TIME)
- Status (PROCESSOR_STATUS)
- Up Time (SYSTEM_UP_TIME)

IPC Summary (PD_IPCS)

- Max Shared Mem Size (SHMMAX)
- Min Shared Mem Size (SHMMIN)
- Shared Mem IDs (SHMMNI)

System Summary Overview (PI)

- Boot Time (CURRENT_SYSTEM_BOOT_TIME)
- Free Swap Mbytes (FREE_SWAP_MBYTES)
- ICMP Pkts In (ICMP_PACKETS_IN)
- ICMP Pkts Out (ICMP_PACKETS_OUT)
- IP Pkts In (IP_PACKETS_IN)
- IP Pkts Out (IP_PACKETS_OUT)
- System Up Time (CURRENT_BOOT_SYSTEM_UP_TIME)
- TCP Pkts In (TCP_PACKETS_IN)
- TCP Pkts Out (TCP_PACKETS_OUT)
- Total Physical Mem Mbytes (TOTAL_MEMORY_MBYTES)
- Total Pkts (TOTAL_PACKETS)
- Total Pkts In (TOTAL_PACKETS_IN)
- Total Pkts Out (TOTAL_PACKETS_OUT)
- Total Swap Mbytes (TOTAL_SWAP_MBYTES)
- UDP Pkts In (UDP_PACKETS_IN)
- UDP Pkts Out (UDP_PACKETS_OUT)

(O)

After allocating free space (at least 50 kilobytes) under /tmp, restart the Agent Collector service.

KAVF10205-W

Initialization of interprocess communication failed.

An attempt to initialize communication with a child process of the jpcagtu process failed.

(S)

Continues the processing of the Agent Collector service.
However, the following fields cannot be collected:

- The IP Address (IP_ADDRESS) field of the File System Detail - Remote (PD_FSR) record
- The Network Mask (NETWORK_MASK) field of the Network Interface Detail (PI_NIND) record
- All fields of the following records (only in the Solaris 64-bit environment):
  Application Process Detail (PD_APPD), Application Process Interval (PD_APSI), Application Process Overview (PD_APS), Application Summary (PD_APP), Application Summary Extension (PD_APP2), Process Detail (PD), Process Detail Interval (PD_PDI), Process Summary (PD_PDS), Program Summary (PD_PGM), Terminal Summary (PD_TERM), User Summary (PD_USER), Workgroup Summary (PI_WGRP)

(O)

Ignore the message if the fields of the records listed above are not being collected.

If the fields of the records listed above are being collected, check whether a system resource shortage has occurred. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10206-W

Interprocess communication failed.

Communication with a child process of the jpcagtu process failed.

(S)

Continues the processing of the Agent Collector service.

However, the following fields cannot be collected:

- The IP Address (IP_ADDRESS) field of the File System Detail - Remote (PD_FSR) record
- The Network Mask (NETWORK_MASK) field of the Network Interface Detail (PI_NIND) record
- All fields of the following records (only in the Solaris 64-bit environment):
  Application Process Detail (PD_APPD), Application Process Interval (PD_APSI), Application Process Overview (PD_APS), Application Summary (PD_APP), Application Summary Extension (PD_APP2), Process Detail (PD), Process Detail Interval (PD_PDI), Process Summary (PD_PDS), Program Summary (PD_PGM), Terminal Summary (PD_TERM), User Summary (PD_USER), Workgroup Summary (PI_WGRP)

(O)

Ignore the message if the fields of the records listed above are not being collected.

If the fields of the records listed above are being collected, check whether a system resource shortage has occurred. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10300-W

An attempt to collect a record failed. (record=*record-id*)

There was a record that the Agent Collector service could not collect.

(S)

Continues the processing of the Agent Collector service.

(O)

If this message is issued continuously, check whether there is an invalid setting in the system environment of the monitoring target system. If the cause cannot be determined, collect maintenance information and then contact the

system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10301-W

An attempt to collect performance data used for *record-id* record has failed.

An error occurred during the collection of the performance data to be used by the displayed record.

(S)

Continues the processing of the Agent Collector service.

However, the performance data of the displayed record cannot be correctly collected.

(O)

Ignore the message if the displayed record is not being collected.

If the displayed record is being collected, check the preceding message that was output to the common message log and take the action indicated in it. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10500-W

Memory is insufficient. (required memory=*requested-size*)

Memory allocation failed.

(S)

Continues the processing of the Agent Collector service.

(O)

Stop those applications that are not being used or expand the memory size.

KAVF10501-W

The disk capacity is insufficient.

File access failed because of insufficient disk capacity.

(S)

Continues the processing of the Agent Collector service.

(O)

Delete unneeded files or expand the disk capacity.

KAVF10502-W

An I/O error occurred.

An I/O error occurred.

(S)

Continues the processing of the Agent Collector service.

(O)

Check whether a problem such as a hard disk error is preventing data from being written to the file system. If so, remove the error cause.

KAVF10503-W

Reception of a signal caused the service to stop. (signal=*signal-number*)

The Agent Collector service stops when it receives the signal. The code indicated by `signal` is an OS signal number.

(S)

Terminates the processing of the Agent Collector service.

(O)

The system may have been terminated without the service automatic stop script being set up. Check whether the service automatic stop script has been set up.

If the service automatic stop script is not set up, set it up. For details about how to set up the service automatic stop script, see the chapter explaining how to start and stop Performance Management in the *JP1/Performance Management User's Guide*.

If the service automatic stop script is set up, or if this message was issued during operations other than system termination, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10504-E

The logical host environment of Agent for Platform is not set up correctly, or the logical host environment is invalid. (lhost=*logical-host-name*)

The logical host environment of Agent for Platform has not been set up correctly, or the logical host environment is invalid.

(S)

Terminates processing of Agent Collector.

(O)

Check whether the version of PFM - Manager or PFM - Base on the same machine is earlier than 11-01. If the version is earlier than 11-01, set up the logical host environment according to the following procedure:

1. Unset up the logical host environment of PFM - Agent for Platform.
2. Update PFM - Manager or PFM - Base to version 11-01 or later.
3. Set up the logical host environment of PFM - Agent for Platform.

If the cause cannot be determined, collect maintenance information and then contact the system administrator.

KAVF10550-E

An attempt to create a directory failed. (dir=*directory-name*, cmd=*command-name*, rc=*return-code*)

An attempt to create a directory failed.

(S)

Terminates processing of the command.

(O)

Confirm the following, and then execute the command again:

- Necessary access permissions are granted.
- Disk capacity is sufficient.

KAVF10551-E

An attempt to copy a file failed. (from *copy-source-path* to *copy-destination-path*, cmd=*command-name*, rc=*return-code*)

An attempt to copy a file failed.

(S)

Terminates processing of the command.

(O)

Confirm the following, and then execute the command again:

- Necessary access permissions are granted.
- Disk capacity is sufficient.
- The copy source file exists.

KAVF10552-E

The system environment is invalid.

The system environment is invalid.

(S)

Terminates processing of the command.

(O)

Collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10600-W

An invalid collection event occurred. (record=*record-id*, rc=*maintenance-code*)

An unsupported collection event indicated by the record ID occurred.

(S)

Does not collect the record indicated by the record ID. Continues the processing of Agent Collector.

(O)

An attempt is being made for an unsupported record to collect a real-time report, perform alarm monitoring, or write to the Store database.

- For a real-time report, stop the attempt to collect the record.
- For an alarm, unbind the alarm that is using the record not supported by the host. For details about how to unbind an alarm, see the chapter explaining operation monitoring using alarms in the *JP1/Performance Management User's Guide*.

- For details about how to stop writing to the Store database, see the chapter explaining the management of operation monitoring data in the *JP1/Performance Management User's Guide*.

KAVF10700-W

It failed to occur JP1 system event or Agent event extension.

An attempt to issue a JP1 system event or Agent event extension has failed.

(S)

Continues the processing of the Agent Collector service.

(O)

Check the preceding message that was output to the common message log and take the action indicated in it. If the cause cannot be determined, collect maintenance information and then contact the system administrator.

For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10701-W

It failed to issue JP1 system event or Agent event extension, because Memory is insufficient.

An attempt to issue a JP1 system event or Agent event extension has failed due to insufficient memory.

(S)

Continues the processing of the Agent Collector service.

(O)

Check whether system resources such as memory are sufficient. If the cause cannot be determined, collect maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10703-W

The user command did not end in the seconds specified in the time-out value (*nn* seconds) and will now be forcibly ended. (record=*record-id,* pid=*process-id*)

Forcibly terminating the user command because it did not end within the time specified by the `User Command Timeout` property.

(S)

The applicable user command will now be forcibly terminated. Because the user command did not end successfully, the collection of the applicable records will be skipped, and the Agent Collector processing continues.

(O)

Remove the problem that is preventing the user command from ending or set the `User Command Timeout` property to a value greater than the time it takes for the user command to execute.

KAVF10704-I

The user command that was executed during the previous record collection is still running and will now be forcibly ended. (record=*record-id,* pid=*process-id*)

Forcibly terminating the user command because the previous user command executed during the previous process of record collection did not end.

(S)

The applicable user command will now be forcibly terminated. In addition, the record collection process for the record for which the user command was forcibly terminated is skipped, and the Agent Collector processing continues.

KAVF10705-W

The user command execution and collection processing of the corresponding record will be skipped because the user command does not end. (record=*record-id,* pid=*process-id*)

Because the forcible termination operation failed to end the previous user command, both the execution of the current user command and the collection of the applicable records will be skipped.

(S)

The collection of the applicable records will be skipped, and the Agent Collector processing continues.

(O)

Forcibly terminate the user command that would not end.

KAVF10706-I

The user command ended. (record=*record-id,* pid=*process-id*)

Forcible termination operation ended the user command.

(S)

The Agent Collector processing continues.

KAVF10707-W

Record collection processing will be skipped, because no user data file has been created. (record=*record-id*)

Collection of the applicable records is skipped because no user data file has been created since the record was last collected.

(S)

The collection of the applicable records will be skipped, and the Agent Collector processing continues.

(O)

Remove the cause of the problem that prevented the creation of a user data file. The possible problems are as follows:

- The user command was not executed.
- An attempt to execute the user command failed.
- The user command execution was not completed.

KAVF10709-I

Failed to connect to Docker Engine (en=*OS-detail-code*). Note that, if this problem reoccurs during the next attempt to collect information, this message will not be output again. The same applies for all consecutive attempts hereafter.

PFM - Agent for Platform failed to establish a connection with the Docker Engine. If PFM - Agent for Platform is running in a Docker environment, `N/A` is stored in the `Virtual Env ID` field of the `PD`, `PD_APSI`, and `PD_APS` records.

This message is only output the first time PFM - Agent for Platform fails to establish a connection with the Docker Engine after the start of the Agent Collector service. If the connection attempt fails again afterwards, the KAVF10710-W message is output.

(S)

The Agent Collector processing continues.

(O)

If you are not using the Docker environment, ignore this message.

If you are using the Docker environment, check whether any of the problems listed below are preventing PFM - Agent for Platform from establishing a connection with the Docker Engine. If they are, remove their causes:

- The Docker Engine is not up and running.
- The UNIX domain socket settings that are used for establishing a connection with the Docker Engine has been modified.

If you cannot identify the cause of the problem, collect maintenance information, and then contact the system administrator.

KAVF10710-W

Failed to connect to Docker Engine (en=*OS-detail-code*). Note that, if this problem reoccurs during the next attempt to collect information, this message will not be output again. The same applies for all consecutive attempts hereafter.

PFM - Agent for Platform failed to establish a connection with the Docker Engine. If PFM - Agent for Platform is running in a Docker environment, `N/A` is stored in the `Virtual Env ID` field of the `PD`, `PD_APSI`, and `PD_APS` records.

(S)

The Agent Collector processing continues.

(O)

Check whether any of the problems listed below are preventing PFM - Agent for Platform from establishing a connection with the Docker Engine. If they are, remove their causes:

- The Docker Engine is not up and running.
- The UNIX domain socket settings that are used for establishing a connection with the Docker Engine has been modified.

If you cannot identify the cause of the problem, collect maintenance information, and then contact the system administrator.

KAVF10711-W

An error occurred in the communication with Docker Engine. Note that, if this problem reoccurs during the next attempt to collect information, this message will not be output again. The same applies for all consecutive attempts hereafter.

An error occurred during communication with the Docker Engine. Due to this, `N/A` is stored in the `Virtual Env ID` field of the `PD`, `PD_APSI`, and `PD_APS` records of the processes whose execution environment could not be identified.

(S)

The Agent Collector processing continues.

(O)

The Docker Engine might have been busy during record collection by Agent Collector. Avoid operating the Docker Engine (for example, executing a Docker command) while Agent Collector is collecting records.

KAVF10713-W

Failed to determine the execution environment of the process because the information of the process inside the Podman container could not be collected. Note that, the output of this message will be suppressed while the same error occurs consecutively.

An error occurred in the Podman command or the command was not completed during a given time period. Due to this, `N/A` is stored in the `Virtual Env ID` field of the `PD`, `PD_APSI`, and `PD_APS` records of the processes whose execution environment could not be identified.

(S)

The Agent Collector processing continues.

(O)

Check that the Podman container is working properly and the system is not heavily loaded. If you cannot determine the cause, collect maintenance data, and then contact a system administrator.

KAVF10800-Q

Do you want to convert? (Y/N)

Indicate whether you want to create settings for collecting information about the operating status of processes from settings for collecting information about the application operating status.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was provided in version 09-00 and earlier.

(S)

Waits for a response.

(O)

To perform the conversion, enter `Y` or `y`. To not perform the conversion, enter `N` or `n`. If you specify any other value, the same message will be displayed again.

KAVF10801-I

Conversion of a setting was successful.

Settings for collecting information about the operating status of processes were created successfully from settings for collecting information about the application operating status.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Terminates processing of the `jpcappcvt` command.

KAVF10802-E

An attempt to convert a setting has failed.

The attempt to create settings for collecting information about the operating status of processes from settings for collecting information about the application operating status failed.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Terminates processing of the `jpcappcvt` command.

(O)

Check the previous message.

KAVF10803-W

An attempt to convert a setting will now halt.

The attempt to create settings for collecting information about the operating status of processes from settings for collecting information about the application operating status was suspended.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Suspends processing of the `jpcappcvt` command.

(O)

To perform the conversion, retry the `jpcappcvt` command.

KAVF10804-W

A settings for collecting application operation and non-operation information does not exist.

No settings could be found for collecting information about the application operating status.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Continues processing the `jpcappcvt` command.

(O)

Specify from the Agents or Services window in PFM - Web Console the appropriate settings for collecting information about the operating status of processes.

KAVF10805-W

A settings for collecting application operation and non-operation information is not correct.

The settings for collecting information about the application operating status are not correct.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Continues processing the `jpcappcvt` command.

(O)

The settings file might have become corrupted. Re-specifying the settings for collecting information about the application operating status from the Services window in PFM - Web Console might repair the file.

Alternatively, from the Agents or Services window in PFM - Web Console, specify settings for collecting information about the operating status of processes.

KAVF10806-Q

A settings for collecting process operation and non-operation information already exists. Would you like to overwrite it? (Y/N)

Confirm whether you want to overwrite the settings for collecting information about the operating status of processes.

Collecting information about the application operating status is functionality that existed in versions 09-00 and earlier. Collecting information about the operating status of processes is an extension of the functionality for collecting information about the application operating status that was packaged with versions 09-00 and earlier.

(S)

Waits for a response.

(O)

To overwrite the settings for collecting information about the operating status of processes, enter `Y` or `y`. To not overwrite, enter `N` or `n`. If you specify any other value, the same message will be displayed again.

KAVF10807-E

Superuser permissions are required.

A user who does not have superuser permissions attempted to execute the `jpcappcvt` command.

(S)

Terminates processing of the `jpcappcvt` command.

(O)

Execute the `jpcappcvt` command as a user with superuser permissions.

KAVF10808-E

A fatal error has occurred.

A fatal error occurred.

(S)

Terminates processing of the `jpcappcvt` command.

(O)

Collect maintenance data, and then contact a system administrator.

KAVF10809-E

An attempt to allocate memory failed.

An attempt to allocate memory failed.

(S)

Terminates processing of the `jpcappcvt` command.

(O)

Try closing other applications or increasing the amount of memory.

If you cannot determine the cause, collect maintenance data, and then contact a system administrator. For details about how to collect data, see the chapter that explains troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF10810-E

Convert command is being executed.

An attempt was made to execute the `jpcappcvt` command while it was already executing.

(S)

Terminates processing of the `jpcappcvt` command.

(O)

Retry the command after processing of the `jpcappcvt` command terminates.

KAVF10850-E

Error: Multiple logical host name options cannot be specified.

Multiple logical host name options cannot be specified.

(S)

Terminates processing of the `jpcuser` command.

(O)

Do not specify more than one logical host name option.

KAVF10851-E

Error: The specification of the logical host name is incorrect.

The logical host name has not been specified, the length of the specified logical host name exceeds the upper limit, or the specified logical host name is wrong.

(S)

Terminates processing of the `jpcuser` command.

(O)

Confirm the logical host name, and then execute the command again.

KAVF10852-E

Error: The specified logical host name is not set up. (lhost=*logical-host-name*)

The specified logical host name is not set up.

(S)

Terminates processing of the `jpcuser` command.

(O)

Confirm the logical host name, and then execute the command again.

KAVF10853-E

Error: The environment directory cannot be accessed. (dir=*environment-directory-name*)

The environment directory cannot be accessed.

(S)

Terminates processing of the `jpcuser` command.

(O)

Confirm the following:

- There is an environment directory that corresponds to the specified logical host.
- The command is not executed on the standby node.

KAVF10901-W

Warning: The specified value (*specified-value*) for the option (tt) is too long. (filename = *user-defined-data-file-path*)

The value specified for the option (`tt`) is too long.

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Shorten the specified value.

KAVF10902-W

Warning: The specified value (*specified-value*) for the option (ks) is too long. (filename = *user-defined-data-file-path*)

The value specified for the option (`ks`) is too long.

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Shorten the specified value.

KAVF10904-W

Warning: The data value count (*number-of-specified-values*) does not match the option count (*number-of-options*). (filename = *user-defined-data-file-path*)

The number of specified values does not match the number of specified options.

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Specify the same number of values and options.

KAVF10905-W

Warning: The data line is too long. (filename = *user-defined-data-file-path*)

The data line is too long.

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Shorten the data line.

KAVF10906-W

Warning: The specified value for the option must be 0 or a positive number. (*option-name* = *specified-value*, filename = *user-defined-data-file-path*)

A value smaller than 0 was specified for the option *option-name*.

(S)

The `jpcuser` command sets 0 and continues processing.

(O)

Specify 0 or a greater value.

KAVF10907-W

Warning: The specified value for the option is too long. (*option-name* = *specified-value*, filename = *user-defined-data-file-path*)

The value specified for the option *option-name* is too long.

(S)

The `jpcuser` command truncates the excess characters of the value specified as the option (the portion in excess of the maximum length) and then allows the processing to continue.

(O)

Shorten the specified value.

KAVF10908-W

Warning: The specified value (*specified-value*) for the option (t) is invalid. (filename = *user-defined-data-file-path*)

The value specified for the option (`t`) is invalid.

(S)

The `jpcuser` command sets `n/a` and continues processing.

(O)

Correct the specified value.

KAVF10909-W

Warning: The specified value for the option must be a numeric value. (*option-name* = *specified-value*, filename = *user-defined-data-file-path*)

A value that is not a numeric value was specified for the option *option-name*.

(S)

The `jpcuser` command sets 0 and continues processing.

(O)

Specify a numeric value.

KAVF10910-W

Warning: The specified value for a floating-point number option is invalid. (*option-name* = *specified-value*, filename = *user-defined-data-file-path*)

The value specified for the floating-point option *option-name* is invalid.

(S)

The `jpcuser` command sets 0 and continues processing.

(O)

Correct the specified value.

KAVF10911-W

Warning: An overflow or an underflow occurred for the value specified for the option. (*option-name* = *specified-value*, filename = *user-defined-data-file-path*)

An overflow or an underflow occurred due to the value specified for the option *option-name*.

(S)

The `jpcuser` command continues processing.

(O)

Correct the specified value.

KAVF10912-W

Warning: "" is specified for the option (*option-name*). (filename = *user-defined-data-file-path*)

No value is specified for the option *option-name*.

(S)

The `jpcuser` command sets 0 and continues processing.

(O)

Specify a value.

KAVF10913-W

Warning: The specified value (*specified-value*) for the option (ki) must be 0 or a positive number. (filename = *user-defined-data-file-path*)

A value smaller than 0 was specified for the option (`ki`).

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Specify 0 or a larger value.

KAVF10914-W

Warning: An overflow or an underflow occurred for the value (*specified-value*) specified for the option (ki). (filename = *user-defined-data-file-path*)

An overflow or an underflow occurred due to the value specified for the option (`ki`).

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Correct the specified value.

KAVF10915-W

Warning: "" is specified for the option (tt). (filename = *user-defined-data-file-path*)

No value is specified for the option (`tt`).

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Specify a value.

KAVF10916-E

Error: The specified value (*specified-value*) for record-ID is invalid.

The value (*specified-value*) specified for the record ID is invalid.

(S)

Terminates processing of the `jpcuser` command.

(O)

Correct the specified value.

KAVF10917-E

Error: An invalid number of arguments was specified.

An invalid number of arguments was specified.

(S)

Terminates processing of the `jpcuser` command.

(O)

Correct the number of specified values.

KAVF10919-E

Error: An invalid arguments (*specified-value*) were specified.

An invalid argument was specified.

(S)

Terminates processing of the `jpcuser` command.

(O)

Correct the specified value.

KAVF10920-E

Error: The user-defined data file option (-file) is not specified.

The user-defined data file option (`-file`) is not specified.

(S)

Terminates processing of the `jpcuser` command.

(O)

Specify the user-defined data file option (`-file`).

KAVF10923-E

Error: The user-defined data file cannot be opened. (filename = *file-name*)

The user-defined data file cannot be accessed.

(S)

Terminates processing of the `jpcuser` command.

(O)

Check whether the specified file name is correct and whether the file specified by *file-name* exists and is accessible. If the cause cannot be determined, collect maintenance information and then contact the system administrator.

KAVF10924-E

Error: There is no product information section in the user-defined data file.

There is no product information section in the user-defined data file.

(S)

Terminates processing of the `jpcuser` command.

(O)

Specify a product information section.

KAVF10925-E

Error: The product information section of the user-defined data file is invalid. Product Name=*product-name*, FormVer=*format-version*

The product information section in the user-defined data file is invalid.

(S)

Terminates processing of the `jpcuser` command.

(O)

Correct the product information section.

KAVF10926-E

Error: The user-defined data file does not contain an option header line.

The user-defined data file does not contain an option header line.

(S)

Terminates processing of the `jpcuser` command.

(O)

Specify an option header line.

KAVF10927-E

Error: The user-defined data file does not contain any data lines.

The user-defined data file does not contain any data lines.

(S)

Terminates processing of the `jpcuser` command.

(O)

Specify a data line.

KAVF10928-E

Error: The option header line is too long.

The option header line is too long.

(S)

Terminates processing of the `jpcuser` command.

(O)

Shorten the option header line.

KAVF10929-E

Error: Too many options (*specified-value*) were specified.

Too many options were specified.

(S)
Terminates processing of the `jpcuser` command.

(O)
Reduce the number of specified options.

KAVF10930-E

Error: An invalid option (*option-name*) was specified.

An invalid option was specified for the option *option-name*.

(S)
Terminates processing of the `jpcuser` command.

(O)
Correct the option.

KAVF10931-E

Error: The option (*option-name*) is not supported for the specified record (*record-ID*).

The specified option is not supported for the specified record ID.

(S)
Terminates processing of the `jpcuser` command.

(O)
Correct the record ID or option.

KAVF10932-E

Error: The option (tt) is not specified.

The option (`tt`) is not specified.

(S)
Terminates processing of the `jpcuser` command.

(O)
Specify the option (`tt`).

KAVF10935-E

Error: A fatal error has occurred.

A fatal error has occurred.

(S)

Terminates processing of the `jpcuser` command.

(O)

Collect maintenance information and then contact the system administrator.

KAVF10941-E

Error: Superuser permissions are required.

A user who does not have superuser permissions attempted to execute the `jpcuser` command.

(S)

Terminates processing of the `jpcuser` command.

(O)

Execute the `jpcuser` command as a user who has superuser permissions.

KAVF10943-E

Error: jpcuser terminated with error.

The `jpcuser` command terminated abnormally.

(S)

Terminates processing of the `jpcuser` command.

(O)

Check the preceding message that was output to the public log, and take the action indicated in it.

KAVF10948-I

Information: jpcuser terminated successfully.

The `jpcuser` command terminated normally.

(S)

Terminates processing of the `jpcuser` command.

KAVF10950-W

Warning: "" is specified for the option (ki). (filename = *user-defined-data-file-path*)

`""` is specified for the option (`ki`).

(S)

The `jpcuser` command skips the current line and continues processing.

(O)

Specify a valid value for the option (`ki`).

KAVF10951-W

Warning: "" is specified for the option (ks). (filename = *user-defined-data-file-path*)

"" is specified for the option (ks).

(S)

The jpcuser command skips the current line and continues processing.

(O)

Specify a valid value for the option (ks).

KAVF10952-E

Error: The option header line specified in the user-defined data file is incorrect. (*specified-value*)

The option header line specified in the user-defined data file is incorrect.

(S)

Terminates processing of the jpcuser command.

(O)

Correct the option header line.

KAVF10953-W

Warning: The data line specified in the user-defined data file is incorrect. (*specified-value*, filename = *user-defined-data-file-path*)

The data line specified in the user-defined data file is incorrect.

(S)

The jpcuser command skips the current line and continues processing.

(O)

Correct the data line.

KAVF10954-W

Warning: jpcuser terminated with a warning.

The jpcuser command terminated with a warning.

(S)

Terminates processing of the jpcuser command.

(O)

Correct the problem generating the warning indicated in the preceding message.

KAVF10957-I

Information: Usage: jpcuser <Record Name> -file <User-defined Data File Path> [-file <User-defined Data File Path>]... [-debug <value>] [-lhost <logical host name>]

This message explains how to use the jpcuser command.

(S)

Terminates processing of the jpcuser command.

KAVF10982-E

Error: The directory (*directory-name*) is specified for the user-defined data file.

A directory is specified for the user-defined data file.

(S)

Terminates processing of the `jpcuser` command.

(O)

Specify a file name.

KAVF10983-E

Error: The name of a user-defined data file is duplicated. (filename = *file-name*)

The user-defined data file name is already being used.

(S)

Terminates processing of the `jpcuser` command.

(O)

Correct the duplication of user-defined data files.

KAVF10984-E

Error: The debug option is duplicated.

The debug option has been specified twice.

(S)

Terminates processing of the `jpcuser` command.

(O)

Either do not specify the debug option, or specify it only once.

KAVF10985-E

Error: There are no user-defined data files.

There are no user-defined data files.

(S)

Terminates processing of the `jpcuser` command.

(O)

Check whether the specified file name is correct and whether the file specified by *file-name* exists and is accessible. If the cause cannot be determined, collect maintenance information and then contact the system administrator.

KAVF10986-W

Warning: The option header line specified in the user-defined data file is incorrect. (option = *specified-value*, filename = *user-defined-data-file-path*)

The option header line specified in the user-defined data file is incorrect.

(S)

Skips this user-defined data file.

(O)

Correct the option header line.

KAVF10987-W

Warning: The user-defined data file cannot be opened. (filename = *file-name*)

The user-defined data file cannot be accessed.

(S)

Skips this user-defined data file.

(O)

Check whether the specified file name is correct and whether the file specified by *file-name* exists and is accessible. If the cause cannot be determined, collect maintenance information and then contact the system administrator.

KAVF10988-W

Warning: There is no product information section in the user-defined data file. (filename = *user-defined-data-file-path*)

There is no product information section in the user-defined data file.

(S)

Skips this user-defined data file.

(O)

Specify a product information section.

KAVF10989-W

Warning: The product information section of the user-defined data file is invalid. Product Name = *product-name*, FormVer = *format-version*, filename = *user-defined-data-file-path*

The product information section in the user-defined data file is invalid.

(S)

Skips this user-defined data file.

(O)

Correct the product information section.

KAVF10990-W

Warning: The user-defined data file does not contain an option header line. (filename = *user-defined-data-file-path*)

The user-defined data file does not contain an option header line.

(S)

Skips this user-defined data file.

(O)

Specify an option header line.

KAVF10991-W

Warning: The user-defined data file does not contain any data lines. (filename = *user-defined-data-file-path*)

The user-defined data file does not contain any data lines.

(S)

Skips this user-defined data file.

(O)

Specify a data line.

KAVF10992-W

Warning: The option header line is too long. (filename = *user-defined-data-file-path*)

The option header line is too long.

(S)

Skips this user-defined data file.

(O)

Shorten the option header line.

KAVF10993-W

Warning: Too many options (*specified-value*) were specified. (filename = *user-defined-data-file-path*)

Too many options were specified.

(S)

Skips this user-defined data file.

(O)

Correct the specification.

KAVF10994-W

Warning: An invalid option (*option-name*) was specified. (filename = *user-defined-data-file-path*)

An invalid value was specified for the option indicated by *option-name*.

(S)

Skips this user-defined data file.

(O)

Correct the option.

KAVF10995-W

Warning: The option (*option-name*) is not supported for the specified record (*record-ID*). (filename = *user-defined-data-file-path*)

The specified option is not supported for the specified record ID.

(S)
Skips this user-defined data file.

(O)
Correct the record ID or option.

KAVF10996-W

Warning: The option (tt) is not specified. (filename = *user-defined-data-file-path*)

The option (`tt`) is not specified.

(S)
Skips this user-defined data file.

(O)
Specify the option (`tt`).

KAVF10997-W

Warning: The user-defined data file was skipped. (filename = *user-defined-data-file-path*)

The user-defined data file was skipped.

(S)
Skips the file and continues the `jpcuser` command processing.

(O)
Check the message that was output immediately before this message and correct the error.

KAVF10998-W

Warning: The directory (*directory-name*) is specified for the user-defined data file.

A directory is specified for the user-defined data file.

(S)
Skips the file and continues the `jpcuser` command processing.

(O)
Correct the specification.

12 Error Handling Procedures

This chapter explains how to handle the errors that occur during the operation of Performance Management. It describes how to handle the errors that occur mainly in PFM - Agent. For details about how to handle errors that occur in the overall Performance Management system, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

12.1 Error handling procedures

This section explains how to handle the errors that occur in Performance Management.

Checking the operating conditions

Check the following details:

- Operating conditions when the error occurred
- Message content (if a message has been issued)
- Log information in the common message log, for example

For details about the cause of each message and how to handle it, see *11. Messages*. For details about the log information that is output by Performance Management, see *12.3 Log information to be collected for troubleshooting*.

Collecting data

You must collect data to identify the error cause. See *12.4 UNIX-related data to be collected for troubleshooting* and *12.5 Procedures for collecting UNIX-related data for troubleshooting*, and collect the necessary data.

Investigating the error

Based on the collected data, investigate the error cause and isolate the area in which the error occurred or determine the scope of the error.

12.2 Troubleshooting

If a problem occurs when you are using Performance Management, first check the symptoms of the problem. If a message is output, check the contents of the message. For log information output by Performance Management, see *12.3 Log information to be collected for troubleshooting*.

You might be unable to resolve an error by taking the steps described in the chapter that describes troubleshooting in the *JP1/Performance Management User's Guide*, or an error other than those described in this manual might occur. In such cases, collect the data needed to investigate the cause of the error, and then contact the system administrator.

For details about the data you need to collect and how to collect it, see *12.4 UNIX-related data to be collected for troubleshooting* and *12.5 Procedures for collecting UNIX-related data for troubleshooting*.

12.3 Log information to be collected for troubleshooting

When an error occurs in Performance Management, check the log information to determine how to handle the error. When Performance Management is being used, the following four types of log information are output:

- System log
- Common message log
- Operation status log
- Trace log

These four types of log information and the log options that can be set in each type of log information are explained below.

12.3.1 Types of log information to be collected

(1) System log

The system log is used to report system statuses and errors. This log is output to the `syslog` file.

For the output format, see the chapter explaining log information in the manual *JP1/Performance Management Reference*.

Notes about logical host operation

In addition to the system log for Performance Management, you might need the log information for the cluster software in order to check such information as Performance Management control by the cluster software.

(2) Common message log

The common message log is used to report system statuses and errors. The log information that is output to the common message log is more detailed than the log information that is output to the system log. For details about the output destination file name for the common message log and file size, see *12.3.2 Log files and directories to check*. For the output format, see the chapter explaining log information in the manual *JP1/Performance Management Reference*.

Notes about logical host operation

When Performance Management is under logical host operation, the common message log is output to the shared disk. Because a log file on the shared disk is inherited together with the system during failover, messages are recorded in the same log file.

(3) Operation status log

The operation status log is output by PFM - Web Console. For details about the output destination file name for the operation status log and file size, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*. For the output format, see the chapter explaining log information in the manual *JP1/Performance Management Reference*.

(4) Trace log

The trace log is collected to investigate the process leading up to the error and to measure the time that was used for each process.

The trace log is output to the log file that each service of a Performance Management program has.

Notes about logical host operation

When Performance Management is under logical host operation, the trace log is output to the shared disk. Because a log file on the shared disk is inherited together with the system during failover, trace logs are recorded in the same log file.

12.3.2 Log files and directories to check

This section explains the log information that is output by Performance Management. For details about the operation status log output destination file name and file size, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

(1) Common message log

For details about the common message log, see the chapter that describes details on log information in the *JP1/Performance Management User's Guide*.

(2) Trace log

This section describes the trace log, which is one of the logs that are output by Performance Management. The table below shows the service name or control name, which is the output source for the trace log of PFM - Agent, and storage destination directory name.

Table 12–1: Trace log storage destination directory name

No.	Log type	Output source	Directory name
1	Trace log	Performance Management command	`/opt/jp1pc/tools/log/`
2		Agent Collector service	`/opt/jp1pc/agtu/agent/log/`
3		Agent Store service	`/opt/jp1pc/agtu/store/log/`
4	Trace log (for logical host operation)	Performance Management command	*environment-directory*#`/jp1pc/tools/log/`
5		Agent Collector service	*environment-directory*#`/jp1pc/agtu/agent/log/`
6		Agent Store service	*environment-directory*#`/jp1pc/agtu/store/log/`

#

The environment directory is on the shared disk that was specified when the logical host was created.

12.4 UNIX-related data to be collected for troubleshooting

If you cannot eliminate the error even when you have taken the steps described in *12.2 Troubleshooting*, you must collect data to identify the error cause and contact the system administrator. This section explains the data that must be collected when an error occurs.

Performance Management provides a command for collecting the necessary data in a batch. To collect PFM - Agent data, use the `jpcras` command. Data that can be collected using the `jpcras` command is indicated by a symbol in the table.

Note:

Data that can be collected using the `jpcras` command differs depending on the operations specified during command execution. For details about the options that can be specified for the command, see the chapter explaining commands in the manual *JP1/Performance Management Reference*.

Notes about logical host operation

The following notes apply to logical host operation:

- During logical host operation, Performance Management log information is stored on the shared disk. If the shared disk is online, you can use the `jpcras` command to also collect the log information on the shared disk in the batch mode.
- To investigate problems during failover, you need the information existing before and after the failover. This means that you need information from both the executing system and the standby system.
- When Performance Management runs on a logical host, its startup and termination are controlled by the cluster software. Therefore, you need information about the cluster software in order to investigate a Performance Management that is running on a logical host. Compare the cluster software operation and the Performance Management operation.

12.4.1 OS log information to be collected

You need to collect the OS-related log information indicated in the following table.

Table 12–2: OS-related log information

Information type	Overview	Default file name	Can be collected using the jpcras command?

System log	syslog	• For HP-UX
		`/var/adm/syslog/syslog.log`
		• For Solaris
		`/var/adm/messages*`
		• For AIX
		`/var/adm/syslog*`
		• For Linux
		`/var/log/messages*`
			Y[#1]

Process information	List of processes	--	Y
System files	`hosts` file	`/etc/hosts`	Y
		`/etc/inet/ipnodes`[#2]	Y[#3]
	`services` file	`/etc/services`	Y

Information type	Overview	Default file name	Can be collected using the jpcras command?
OS information	Patch information	--	Y
	Kernel information	--	Y
	Version information	--	Y
	Network status	--	Y
	Environment variable	--	Y
	Host name	--	Y
Dump information	`core` file#4	--	Y#5

Legend:

Y: Can be collected.

--: Not applicable

#1

You cannot collect this data when using a system that is set to output data using the default path and file name. Collect the data manually.

#2

The `/etc/inet/ipnodes` file exists only in Solaris. Collect this file together with the `/etc/hosts` file.

#3

Only the `jpcras` command in PFM - Manager 09-00 or later, or PFM - Base 09-00 or later, can collect this data.

#4

In HP-UX, you can use the `coreadm` command to change the name of a `core` file. After renaming, the `jpcras` command cannot collect any files whose name does not begin with `core`. For these files, collect information manually.

#5

Dump information is not collected if `systemd-coredump` is enabled in Linux.

See the chapter explaining Error Handling Procedures in the *JP1/PerformanceManagement User's Guide*.

12.4.2 Performance Management information to be collected

You need to collect the types of information related to Performance Management that are described below. If the error occurred in network connection, you also need to collect information from the connection destination machine. The following table describes the information related to Performance Management.

Table 12–3: Performance Management information

Information type	Description	Default file name	Collection by the jpcras command possible
Common message log	Message log output by Performance Management	`/opt/jp1pc/log/jpclog{01|02}`#1	Y

Information type	Description	Default file name	Collection by the jpcras command possible
Common message log	(sequential file method)	`/opt/jp1pc/log/jpclog{01|02}`#1	Y
	Message log output by Performance Management (wrap-around method)	`/opt/jp1pc/log/jpclogw{01|02}`#1	Y
Configuration information	Information file for each configuration	--	Y
	Output results of the `jpctool service list` command	--	Y
Version information	Product version	--	Δ
	History information	--	Δ

Database information	Agent Store	• For Store 1.0
		`/opt/jp1pc/agtu/store/*.DB`
		`/opt/jp1pc/agtu/store/*.IDX`
		• For Store 2.0
		`/opt/jp1pc/agtu/store/STPD`
		`/opt/jp1pc/agtu/store/STPI`
		The following files under the `/opt/jp1pc/agtu/store/STPL` directory:
		`*.DB`
		`*.IDX`
			Y

Trace log	Trace information on each service of a Performance Management program	--#2	Y
Definition file	Application definition file (09-00 or earlier)	`/opt/jp1pc/agtu/agent/jpcapp`	Y
	Application definition file (10-00 or later)	`/opt/jp1pc/agtu/agent/jpcapp2`	Y
Information about the function for collecting user-specific performance data	Configuration information	`/opt/jp1pc/agtu/agent/jpcuser/jpcuser.ini`	Y
	Debug log	`/opt/jp1pc/agtu/agent/jpcuser/debug/jpcuser_dbg_{01|02|03|04|05}.log`	Y
	Trace log	`/opt/jp1pc/agtu/agent/jpcuser/log/trace/msglog{01|02}`	Y
	Public log	`/opt/jp1pc/agtu/agent/jpcuser/log/public/jpclog{01|02}`	Y
	User data file	`/opt/jp1pc/agtu/agent/jpcuser/userdata/jpcuser_{UPI|UPIB|UPD|UPDB|XUI1|XUI2|XUI3|XUI4|XUI5}`	Y
	Backup file of the user data file	`/opt/jp1pc/agtu/agent/jpcuser/userdata/jpcuser_{UPI|UPIB|UPD|UPDB|XUI1|XUI2|XUI3|XUI4|XUI5}_bak`	Y

Information type	Description	Default file name	Collection by the jpcras command possible
Information about the function for collecting user-specific performance data	Core dump file#3	/opt/jp1pc/agtu/agent/jpcuser/core	Δ#4
Installation log#5	Default log of Hitachi PP Installer	/etc/.hitachi/.hitachi.log /etc/.hitachi/.hitachi.log{01|02|03|04|05} /etc/.hitachi/.install.log /etc/.hitachi/.install.log{01|02|03|04|05}	N

Legend:

Y: Can be collected (physical host and logical host).

Δ: Can be collected (physical host only).

N: Cannot be collected.

--: Not applicable

Note:

When a logical host is used, replace /opt with *environment-directory*.

#1

For details about log file output methods, see the chapter explaining error detection in Performance Management in the *JP1/Performance Management User's Guide*.

#2

For details about the trace log storage destination directory, see *12.3.2 Log files and directories to check*.

#3

In HP-UX, you can use the coreadm command to change the name of a core dump file. After renaming, the jpcras command cannot collect any files whose name does not begin with core. For these files, collect information manually.

#4

Dump information is not collected if systemd-coredump is enabled in Linux.

See the chapter explaining Error Handling Procedures in the *JP1/PerformanceManagement User's Guide*.

#5

Collect this log if installation fails.

12.4.3 Operation information to be collected

You need the following types of information on the operation that was taking place when the error occurred:

- Operation details
- Time at which the error occurred
- Machine configuration (including each OS version, host name, and the configurations of PFM - Manager and PFM - Agent)
- Whether the error can be reproduced
- Performance Management user name used for logon if the user logged on from PFM - Web Console

12.4.4 Error information to be collected

You need to collect the following error information:

- Messages output to the console if the error occurred during command execution

12.4.5 Performance data information to be collected

You also need to collect the following information related to performance data.

The following table shows the performance data information specific to PFM - Agent for Platform that needs to be collected in an environment in which PFM - Agent for Platform is installed.

Table 12–4: Performance data information specific to PFM - Agent for Platform

Collected item	OS	Information that needs to be collected (file name or collection command name)	Can be collected using the jpcras command?
Processor information	HP-UX	`ioscan -fnC processor`	Y
	Solaris	`psrinfo -v`	Y
	AIX	• `lsdev -Cc processor` • `lsattr -El proc#`[#1]	Y
	Linux	`/proc/cpuinfo`	Y
System paging space information	HP-UX	`swapinfo -a`	Y
	Solaris	`swap -l`	Y
	AIX	`lsps -a`	Y
	Linux	• `/proc/meminfo` • `free` • `/proc/swaps` • `procinfo`	Y
I/O statistics information	HP-UX	• `lsdev -C disk` • `iostat` • `/etc/netconfig`	Y
	Solaris	• `iostat` • `prtconf -D`	Y
	AIX	• `lsdev -Cc disk` • `iostat` • `lscfg`	Y
	Linux	• `lsdev` • `iostat` • `procinfo`	Y
Status information for inter-process communication function	HP-UX	`ipcs -a`	Y
	Solaris		
	AIX		

Collected item	OS	Information that needs to be collected (file name or collection command name)	Can be collected using the jpcras command?
Status information for inter-process communication function	Linux	• ipcs -a • ipcs -at • ipcs -ap • ipcs -ac • ipcs -al • ipcs -au	Y
Static information about file system	HP-UX	/etc/fstab	Y
	Solaris	/etc/vfstab	Y
	AIX	/etc/filesystems	Y
	Linux	/etc/fstab	Y
Network status information	HP-UX	• netstat -s • netstat -in • ioscan -fnC lan • /etc/rc.config.d/netconf • /etc/inetd.conf	Y
	Solaris	• netstat -s • netstat -in • /etc/inet/* • /etc/defaultrouter • /etc/inetd.conf	Y
	AIX	• netstat -s • netstat -in • lsattr -E -l inet0 • /etc/inetd.conf	Y
	Linux	• netstat -s • netstat -ni • ip -s link • ss -s	Y
LAN device configuration and status display	HP-UX	lanscan -v	Y
	Solaris	ifconfig -a	Y
	AIX		
	Linux	• ifconfig -a • ip addr	Y
hosts file information	HP-UX	--	--
	Solaris		
	AIX		
	Linux	/etc/hosts	Y
NIS configuration file information	HP-UX	/etc/nsswitch.conf	Y
	Solaris	/etc/inet/hosts	Y
	AIX	/etc/netsvc.conf	Y

Collected item	OS	Information that needs to be collected (file name or collection command name)	Can be collected using the jpcras command?
NIS configuration file information	Linux	`/etc/yp.conf`	Y
Summary information for file system assignment	HP-UX	`repquota -a`	Y
	Solaris		
	AIX		
	Linux		
Storage information collected by agent	HP-UX	`agtu/agent/storage.d`	Y
	Solaris		
	AIX		
	Linux		
Bit count information for kernel	HP-UX	`getconf KERNEL_BITS`	Y
	Solaris	`isainfo -kv`	Y
	AIX	`bootinfo -K`	Y
	Linux	`getconf LONG_BIT`	Y
Implemented memory size information	HP-UX	`grep -i Physical /var/adm/syslog/syslog.log`	Y
	Solaris	• `prtconf` • `echo "::memstat" | mdb -k`	Y
	AIX	`bootinfo -r`	Y
	Linux	• `/proc/meminfo` • `free` • `procinfo`	Y
System information (OS, version, and model)	HP-UX	`uname -a`	Y
	Solaris		
	AIX	• `uname -a` • `oslevel`	Y
	Linux	• `uname -a` • `/etc/redhat-release` • `/etc/os-release`	Y
Execution level information	HP-UX	`who -r`	Y
	Solaris		
	AIX		
	Linux	`runlevel`	Y
System time	HP-UX	`date`	Y
	Solaris		
	AIX		
	Linux		

Collected item	OS	Information that needs to be collected (file name or collection command name)	Can be collected using the jpcras command?
Time zone	HP-UX	/etc/TIMEZONE	Y
	Solaris	/etc/TIMEZONE	Y
	AIX	/etc/environment	Y
	Linux	/etc/sysconfig/clock	Y
Directory information to be exported onto NFS client	HP-UX	• /etc/exports • /etc/xtab	Y
	Solaris	• /etc/dfs/dfstab • /etc/dfs/sharetab#2 • /etc/rmtab#2	Y
	AIX	• /etc/exports • /etc/xtab	Y
	Linux	• /etc/exports • /var/lib/nfs/xtab	Y
Workgroup monitor settings information	HP-UX	agtu/agent/wgfile	Y
	Solaris		
	AIX		
	Linux		
Event file monitor settings information	HP-UX	agtu/agent/evfile	Y
	Solaris		
	AIX		
	Linux	--	--
Installed software information	HP-UX	swlist	Y
	Solaris	pkginfo	Y
	AIX	lslpp -L all	Y
	Linux	• rpm -qai • rpm -qa -last	Y
OS patch information	HP-UX	• swlist -l patch • what /stand/vmunix	Y
	Solaris	showrev -a	Y
	AIX	instfix -a	Y
	Linux	• rpm -qai • rpm -qa -last	Y
OS setup information	HP-UX	--	--
	Solaris		
	AIX	• /usr/bin/ls -l /unix • /usr/bin/ls -la /usr/lib/boot	Y

Collected item	OS	Information that needs to be collected (file name or collection command name)	Can be collected using the jpcras command?
OS setup information	Linux	--	--
Process list	HP-UX	--	--
	Solaris#3	`/bin/ps -elfZ`	Y
	AIX	--	--
	Linux		
Solaris zone information	HP-UX	--	--
	Solaris#3	`/usr/sbin/zoneadm list -cv`	Y
	AIX	--	--
	Linux		
Linux Docker information#4	HP-UX	--	--
	Solaris		
	AIX		

	Linux#5	`docker version`
		`docker ps -a --no-trunc`
		`docker top` *container-id*#6
		`docker inspect` *container-id*#6
		`cat /etc/docker/daemon.json`
		`cat /etc/sysconfig/docker`
			Y

Linux Podman information	HP-UX	--	--
	Solaris		
	AIX		

	Linux#7	`podman version`
		`podman ps --all --no-trunc --format={{.ID}}`
		`podman inspect` *container-id*#6
		`podman top` *container-id*#6 `hpid args`
			Y

Legend:

Y: Can be collected.

--: Not applicable.

#1

The hash mark (#) represents the processor number.

#2

When the zone functionality is used in Solaris, this information cannot be collected in a non-global zone environment.

#3

Needs to be collected for Solaris only.

#4

When Performance Management is used in a Docker container environment, this information is not collected. In this case, you have to manually collect information on the Docker host.

#5

This information needs to be collected only when the Docker environment is used with Linux 7 or CentOS 7.

#6

This information needs to be collected for all containers.

#7

This information needs to be collected only when the Podman environment is used with Linux 8 or CentOS 8.

12.4.6 Other information to be collected

The following types of information are also necessary:

- The argument specified for the command if the error occurred during command execution

12.5 Procedures for collecting UNIX-related data for troubleshooting

For details about how to collect UNIX data for troubleshooting, see the chapter that describes how to handle problems in the *JP1/Performance Management User's Guide*.

12.6 Performance Management error detection

You can use the health check function to detect errors in Performance Management itself. This function monitors the operating status of the monitoring agent and the host on which the monitoring agent is running, and displays the monitoring results in PFM - Web Console as changes in the status of the monitoring agent.

You can also use the PFM service Auto-restart functionality to automatically restart the PFM service when it abnormally terminates for some reason, or to restart it on a regular basis.

To use the health check function to monitor the operating status of the monitoring agent or use the PFM service Auto-restart functionality, you need to use the status management function. This function checks the detailed status of the Performance Management service. Therefore, the version of the target monitoring agent must support the status management function and the status management function must be enabled. There are no requirements when you monitor the operating status of the host.

Using JP1/Base, which is an integrated system monitoring and management product, to monitor Performance Management log files can also detect errors in Performance Management itself. This allows the system administrator to detect an error if a problem occurs, determine the cause of the error, and take an appropriate recovery measure.

For details about detecting errors in Performance Management itself, see the chapter explaining error detection in Performance Management in the *JP1/Performance Management User's Guide*.

12.7 Performance Management system error recovery

When an error occurs on a Performance Management server, you need to restore the server to the normal state that was present before the error occurred, based on a backup file.

For details about how to restore a server to its pre-error state, see the chapter explaining troubleshooting in the *JP1/Performance Management User's Guide*.

Appendixes

A. Estimating System Requirements

Before you build a system that uses PFM - Agent for Platform, Hitachi recommends that you estimate the system requirements to determine whether the performance of the machine to be used is sufficient to operate PFM - Agent for Platform.

A.1 Estimating memory requirements

Memory requirements change according to the settings and usage of PFM - Agent for Platform. For details about the formulas to use for estimating memory requirements, see the *Release Notes*.

A.2 Estimating disk space requirements

Disk space requirements vary according to the number of records for which performance data is collected. When estimating the amount of disk space required, note that the estimated disk space required for the entire system is a function of the estimated disk space required for the Store database (version 1.0 or 2.0). For the equations to use to estimate disk space requirements, see the *Release Notes*.

B. List of System Resources for Which Kernel Parameters Need to Be Adjusted

When you use PFM - Agent for Platform in a UNIX environment, you need to adjust the kernel parameters of the OS to allocate the resources necessary for executing PFM - Agent for Platform. Adjust the kernel parameters of the OS to allocate the resources required for executing PFM - Agent for Platform.

For details about kernel parameter adjustment when you use PFM - Manager, PFM - Base, and PFM - Web Console in a UNIX environment, see the list of system resources for which kernel parameters need to be adjusted in the appendix to the *JP1/Performance Management Planning and Configuration Guide*.

B.1 For HP-UX

(1) System resources required by PFM - Agent for Platform

The following table describes the system resources required for PFM - Agent for Platform.

System resource	Parameter	Estimate
File system	`ninode`	26[#1]
	`maxfiles_lim`	19[#1]
Semaphore	`semaem`	10[#2]
	`semmni`	20[#3]
	`semmns`	60[#3]
	`semmnu`	12[#2]
	`semume`	30
	`semvmx`	10,000
Process	`nproc`	3[#3 #4]
	`nkthread`	3[#3 #4]
	`maxdsiz_64bit`	78,643,200
	`maxssiz_64bit`	2,048,000

#1

This value does not include the number of files opened by the Store service. For the number of files opened by the Store service, see the system resources required by PFM - Base in the appendixes in the *JP1/Performance Management Planning and Configuration Guide*.

#2

If the number of instances of the `jpcuser` command that have been started on one host is larger than 10, replace the value with the value obtained from the following expression:

The number of instances of the jpcuser command that have been started on one host

#3

This value applies to a single host. To use multiple hosts (physical and logical hosts, or logical hosts) on one machine, replace the value with the value obtained from the following expression:

*Value of Estimate column * Number of hosts*

#4

If the `jpcuser` command is to be used, replace the value with the value obtained from the following expression:

Value of Estimate column + Number of instances of the jpcuser command that have been started on one machine

B.2 For Solaris

(1) System resources required by PFM - Agent for Platform

The following table describes the system resources required for PFM - Agent for Platform.

System resource	Parameter	Estimate
File system	`process.max-file-descriptor`	23[#1]
Common memory	`project.max-shm-ids`	1[#2]
	`project.max-shm-memory`	4[#2 #3]
Semaphore	`project.max-sem-ids`	20[#2]
	`process.max-sem-nsems`	3
	`process.max-sem-ops`	3
Memory	`process.max-stack-size`	2,048,000

#1

This value does not include the number of files opened by the Store service. For the number of files opened by the Store service, see the system resources required by PFM - Base in the appendixes in the *JP1/Performance Management Planning and Configuration Guide*.

#2

This value applies to a single host. To use multiple hosts (physical and logical hosts, or logical hosts) on one machine, replace the value with the value obtained from the following expression:

*Value of Estimate column * Number of hosts*

#3

This value indicates the size used by PFM - Agent for Platform.

B.3 For AIX

(1) System resources required by PFM - Agent for Platform

The following table describes the system resources required for PFM - Agent for Platform.

System resource	Parameter	Estimate
Process	`maxuproc`	4[#1 #2]
File system	`nofiles`	21[#3]
	`nofiles_hard`	21[#3]
Memory	`data_hard`	174,080
	`stack_hard`	4,096

#1

This value applies to a single host. To use multiple hosts (physical and logical hosts, or logical hosts) on one machine, replace the value with the value obtained from the following expression:

*Value of Estimate column * Number of hosts*

#2

If the `jpcuser` command is to be used, replace the value with the value obtained from the following expression:

Value of Estimate column + Number of instances of the jpcuser command that have been started on one machine

#3

This value does not include the number of files opened by the Store service. For the number of files opened by the Store service, see the system resources required by PFM - Base in the appendixes in the *JP1/Performance Management Planning and Configuration Guide*.

B.4 For Linux

(1) System resources required by PFM - Agent for Platform

The following table describes the system resources required for PFM - Agent for Platform.

System resource	Parameter	Estimate
File system	`fs.file-max`	20[#1 #2]
Semaphore	First parameter of `kernel.sem` (`SEMMSL`)	3
	Second parameter of `kernel.sem` (`SEMMNS`)	60[#1]
	Third parameter of `kernel.sem` (`SEMOPM`)	3
	Fourth parameter of `kernel.sem` (`SEMMNI`)	20[#1]
Process	`kernel.threads-max`	3[#1 #3]

#1

This value applies to a single host. To use multiple hosts (physical and logical hosts, or logical hosts) on one machine, replace the value with the value obtained from the following expression:

*Value of Estimate column * Number of hosts*

#2

This value does not include the number of files opened by the Store service. For the number of files opened by the Store service, see the system resources required by PFM - Base in the appendixes in the *JP1/Performance Management Planning and Configuration Guide*.

#3

If the `jpcuser` command is to be used, replace the value with the value obtained from the following expression:

Value of Estimate column + Number of instances of the jpcuser command that have been started on one machine

C. List of Identifiers

When operating PFM - Agent for Platform or extracting performance data from the Store database of PFM - Agent for Platform, identifiers that indicate PFM - Agent for Platform are necessary in some cases. The following table shows the identifiers of PFM - Agent for Platform.

Table C–1: PFM - Agent for Platform identifier list

Application	Name	Identifier	Explanation
Command, etc.	Product ID	`U`	Product ID is part of a service ID. You need a service ID when you use a command to check the Performance Management system configuration or back up performance data. For details about service IDs, see the section that explains service naming conventions in the appendix of the *JP1/Performance Management Planning and Configuration Guide*.
	Service key	`agtu` or `UNIX`	You need a service key when you use a command to start or stop PFM - Agent for Platform. For details about service keys, see the section that explains service naming conventions in the appendix of the *JP1/Performance Management Planning and Configuration Guide*.
Help	Help ID	`pcau`	Indicates help for PFM - Agent for Platform.

D. List of Processes

This appendix lists the processes in PFM - Agent for Platform.

For details about the processes in PFM - Manager, PFM - Base, and PFM - Web Console, see the appendix to the manual *JP1/Performance Management Reference*.

The following table shows the processes in PFM - Agent for Platform. The value provided next to the process name indicates the number of processes that can be concurrently started.

Table D–1: List of processes in PFM - Agent for Platform

Process name (Number of processes)	Function
`jpcagtu` (1)	Agent Collector service process. One Agent Collector service process is started for each PFM - Agent for Platform.
`jpc_hostutil` (1)#1	Host information acquisition program
`jpc_process` (1)#2	64-bit process information acquisition program
`jpcsto` (1)	Agent Store service process. One Agent Store service process is started for each PFM - Agent for Platform.
`stpqlpr` (1)#3	Process for executing a backup or export of the Store database

#1

Child process of the `jpcagtu` process.

#2

Child process of the `jpcagtu` process. In HP-UX, AIX, or Linux, this process is not installed.

#3

Child process of the `jpcsto` process.

E. List of Port Numbers

This appendix lists the port numbers used by PFM - Agent for Platform.

For details about the port numbers used by PFM - Manager, PFM - Web Console, and PFM - Base, and directions to pass through the firewall, see the appendix to the manual *JP1/Performance Management Reference*.

You can also change port numbers to the desired numbers as needed by the user environment.

For details about how to change port numbers, see the chapter explaining installation and setup in the *JP1/Performance Management Planning and Configuration Guide*. The protocol used is TCP/IP.

Note:

Performance Management supports static NAT (Basic NAT), which converts addresses on a one-to-one basis.

Performance Management does not support dynamic NAT or NAPT (IP Masquerade, NAT+), which includes a port conversion function.

E.1 Port numbers of PFM - Agent for Platform

Port numbers used by PFM - Agent for Platform are preferred port numbers.

If you do not change port numbers from their defaults when executing the `jpcconf port define` command, these preferred port numbers are assigned. If the `jpcconf port define` command is not executed, port numbers not being used by the system are automatically assigned when the service is restarted. If the logical host and physical host on a device (or multiple logical hosts) have Performance Management series program services, unique port numbers must be assigned among those hosts.

The following table lists the preferred port numbers used by PFM - Agent for Platform.

Table E–1: Preferred port numbers used by PFM - Agent for Platform

Preferred port number	Service name	Parameter#1	Application
20277	Agent Store service	`jp1pcstou[_`*logical host name*`]`#2	Used for recording performance data and acquiring historical reports.
20278	Agent Collector service	`jp1pcagtu[_`*logical host name*`]`#2	Used for binding an alarm and acquiring real-time reports.

#1

Service name of the `services` file

#2

_*logical host name* is added for a logical host service, and is not added for a physical host service.

E.2 Firewall passage directions

For details about locating PFM - Manager and PFM - Agent for Platform across a firewall, see the section that describes firewall passage directions in the manual *JP1/Performance Management Reference*.

F. Properties of PFM - Agent for Platform

This appendix lists the properties of the Agent Store and Agent Collector services of PFM - Agent for Platform that are displayed on PFM - Web Console.

F.1 Agent Store service properties

The following table shows the properties of the Agent Store service of PFM - Agent for Platform.

Table F–1: Properties of the Agent Store service of PFM - Agent for Platform

Directory name	Property name	Explanation
--	First Registration Date	Shows the first date on which the service was recognized by PFM - Manager.
	Last Registration Date	Shows the last date on which the service was recognized by PFM - Manager.
`General`	--	Stores information such as the host name and directories. The properties stored in this directory cannot be changed.
	Directory	Shows the name of the current directory in which the service runs.
	Host Name	Shows the name of the host on which the service runs.
	Process ID	Shows the process ID of the service.
	Physical Address	If the IPv6 communication functionality is disabled, this property shows the IP address and port number of the host on which the service runs.
	Physical Address (IPv4)	If the IPv6 communication functionality is enabled, this property shows the IP address (IPv4) of the host on which the service runs.
	Physical Address (IPv6)	If the IPv6 communication functionality is enabled, this property shows the IP address (IPv6) of the host on which the service runs.
	Port Number	If the IPv6 communication functionality is enabled, this property shows the port number of the host on which the service runs.
	User Name	Shows the name of the user who executed the service process.
	Time Zone	Shows the time zone used by the service.
`System`	--	Stores the OS information of the OS under which the service has been started. The properties stored in this directory cannot be changed.
	CPU Type	Shows the CPU type.
	Hardware ID	Shows the hardware ID.
	OS Type	Shows the OS type.
	OS Name	Shows the OS name.
	OS Version	Shows the OS version.
`Network Services`	--	Stores information on the Performance Management communication common library. The properties stored in this directory cannot be changed.
	Build Date	Shows the Agent Store service creation date.

Directory name		Property name	Explanation
Network Services		INI File	Shows the name of the directory storing the jpcns.ini file.
Network Services	Service	--	Stores information on the service. The properties stored in this directory cannot be changed.
		Description	Shows the host name.
		Local Service Name	Shows the service ID.
		Remote Service Name	Shows the service ID of the Master Manager service of the connection destination PFM - Manager host.
		EP Service Name	Shows the service ID of the Correlator service of the connection destination PFM - Manager host.
Retention		--	Specifies the data retention period when the Store version is 1.0. For details, see the chapter explaining management of operation monitoring data in the *JP1/Performance Management User's Guide*.
		Product Interval - Minute Drawer	Specifies the minute-by-minute record retention period for PI-type records. One of the following can be selected: • Minute • Hour • Day • 2 Days • 3 Days • 4 Days • 5 Days • 6 Days • Week • Month • Year
		Product Interval - Hour Drawer	Specifies the hourly record retention period for PI-type records. One of the following can be selected: • Hour • Day • 2 Days • 3 Days • 4 Days • 5 Days • 6 Days • Week • Month • Year
		Product Interval - Day Drawer	Specifies the daily record retention period for PI-type records. One of the following can be selected: • Day • 2 Days • 3 Days • 4 Days • 5 Days • 6 Days • Week • Month

Directory name		Property name	Explanation
Retention		Product Interval - Day Drawer	• Year
		Product Interval - Week Drawer	Specifies the weekly record retention period for PI-type records. One of the following can be selected: • Week • Month • Year
		Product Interval - Month Drawer	Specifies the monthly record retention period for PI-type records. One of the following can be selected: • Month • Year
		Product Interval - Year Drawer	Specifies the yearly record retention period for PI-type records. Fixed to Year.
		Product Detail - *record-ID-of-PD-type-record*	Specifies the number of records retained for each PD-type record. An integer in the range from 0 to 2,147,483,647 can be specified. *Note:* If an invalid numerical value or a character such as a letter is specified, an error message is displayed.
		Product Log - MESS	Specifies the number of Log Messages (PL_MESS) records that can be retained. An integer in the range from 0 to 2,147,483,647 can be specified. *Note:* If an invalid numerical value or a character such as a letter is specified, an error message is displayed.
RetentionEx		--	Specifies the data retention period when the Store version is 2.0. For details, see the chapter explaining management of operation monitoring data in the *JP1/Performance Management User's Guide*.
RetentionEx	Product Interval - *record-ID-of-PI-type-record*	--	Specifies the record retention period for PI-type records.
		Period - Minute Drawer (Day)	Specifies the minute-by-minute performance data retention period for each PI-type record ID. An integer in the range from 0 to 366 can be specified for the retention period (number of days).
		Period - Hour Drawer (Day)	Specifies the hourly performance data retention period for each PI-type record ID. An integer in the range from 0 to 366 can be specified for the retention period (number of days).
		Period - Day Drawer (Week)	Specifies the daily performance data retention period for each PI-type record ID. An integer in the range from 0 to 522 can be specified for the retention period (number of weeks).
		Period - Week Drawer (Week)	Specifies the weekly performance data retention period for each PI-type record ID. An integer in the range from 0 to 522 can be specified for the retention period (number of weeks).
		Period - Month Drawer (Month)	Specifies the monthly performance data retention period for each PI-type record ID. An integer in the range from 0 to 120 can be specified for the retention period (number of months).
		Period - Year Drawer (Year)	Displays the yearly performance data retention period for each PI-type record ID.

Directory name		Property name	Explanation
RetentionEx	Product Detail - *record-ID-of-PD-type-record*	Period (Day)	Specifies the performance data retention period for each PD-type record ID. An integer in the range from 0 to 366 can be specified for the retention period (number of days).
	Product Log - *record-ID-of-PL-type-record*	Period (Day)#	Specifies the performance data retention period for each PL-type record ID. An integer in the range from 0 to 366 can be specified for the retention period (number of days).
Disk Usage		--	Stores the disk capacity used by each database. The properties stored in this directory show the disk usage at the time when these properties are displayed. The properties stored in this directory cannot be modified.
		Product Interval	Shows the disk capacity used by PI-type records.
		Product Detail	Shows the disk capacity used by PD-type records.
		Product Alarm	Shows the disk capacity used by PA-type records. Not used in PFM - Agent for Platform (UNIX).
		Product Log	Shows the disk capacity used by PL-type records.
		Total Disk Usage	Shows the disk capacity used by the entire database.
Configuration		--	Shows the properties of the Agent Store service.
		Store Version	Shows the Store database version. • For Store version 1.0: 1.0 • For Store version 2.0: 2.0
Multiple Manager Configuration		Primary Manager	Displays the host name of the monitoring manager specified as the primary manager for multiple monitoring. You cannot change this property.
		Secondary Manager	Displays the host name of the monitoring manager specified as the secondary manager for multiple monitoring. You cannot change this property.

Legend:

--: Not applicable

#

This is fixed to 0 if the record ID is APS or ASVC.

F.2 Agent Collector service properties

The following table shows the properties of the Agent Collector service of PFM - Agent for Platform.

Table F–2: Properties of the Agent Collector service of PFM - Agent for Platform

Directory name	Property name	Explanation
--	First Registration Date	Shows the first date on which the service was recognized by PFM - Manager.

Directory name	Property name	Explanation
--	Last Registration Date	Shows the first date on which the service was recognized by PFM - Manager.
	Data Model Version	Shows the version of the data model.
`General`	--	Stores information such as the host name and directories. The properties stored in this directory cannot be changed.
	Directory	Shows the name of the current directory in which the service runs.
	Host Name	Shows the name of the host on which the service runs.
	Process ID	Shows the process ID of the service.
	Physical Address	If the IPv6 communication functionality is disabled, this property shows the IP address and port number of the host on which the service runs.
	Physical Address (IPv4)	If the IPv6 communication functionality is enabled, this property shows the IP address (IPv4) of the host on which the service runs.
	Physical Address (IPv6)	If the IPv6 communication functionality is enabled, this property shows the IP address (IPv6) of the host on which the service runs.
	Port Number	If the IPv6 communication functionality is enabled, this property shows the port number of the host on which the service runs.
	User Name	Shows the name of the user who executed the service process.
	Time Zone	Shows the time zone used by the service.
`System`	--	Stores the OS information of the OS under which the service has been started. The properties stored in this directory cannot be changed.
	CPU Type	Shows the CPU type.
	Hardware ID	Shows the hardware ID.
	OS Type	Shows the OS type.
	OS Name	Shows the OS name.
	OS Version	Shows the OS version.
`Network Services`	--	Stores information on the Performance Management communication common library. The properties stored in this directory cannot be changed.
	Build Date	Shows the Agent Collector service creation date.
	INI File	Shows the name of the directory storing the `jpcns.ini` file.

Directory name		Property name	Explanation
Network Services	Service	--	Stores information on the service. The properties stored in this directory cannot be changed.
		Description	Shows the host name.
		Local Service Name	Shows the service ID.
		Remote Service Name	Shows the service ID of the Agent Store service to which the Agent Collector service connects.
		AH Service Name	Shows the service ID of the Action Handler service that is located on the same host.
JP1 Event Configurations		--	Sets the condition for issuing JP1 events.
		service	Yes or No is selected from a list for the Agent Collector service, Agent Store service, Action Handler service, and Status Server service to specify whether to output JP1 system events for each service.
		JP1 Event Send Host	Specifies the name of the event server to which J1/Base connects. However, only the event server running on the logical host or physical host on the same machine as for the Action Handler service can be specified. From 0 to 255 bytes of alphanumeric characters, periods (.), and hyphens (-) can be specified. If a value outside this range is specified, it is assumed that no value has been specified. If no value is specified, the host on which the Action Handler service is running is used as the host that issues events. If localhost is specified, the physical host is assumed.
		Monitoring Console Host	Specifies the PFM - Web Console host that is to be started when the PFM - Web Console is started by starting the JP1/IM - Manager monitor. From 0 to 255 bytes of alphanumeric characters, periods (.), and hyphens (-). If a value outside this range is specified, it is assumed that no value has been specified. If no value is specified, the PFM - Manager host to be connected is assumed.
		Monitoring Console Port	Specifies the port number of PFM - Web Console (http request port number) to be started. A value in the range from 1 to 65,535 can be specified. If a value outside this range is specified, it is assumed that no value has been specified. If no value is specified, 20358 is set.
		Monitoring Console Https	Specifies whether to connect to PFM - Web Console via HTTPS-based encrypted communications when PFM - Web Console is started by the startup of the JP1/IM - Manager monitor. The default is No. • Yes: Uses encrypted communications. • No: Does not use encrypted communications.
JP1 Event Configurations	Alarm	JP1 Event Mode	Specifies which of the following events is issued when the alarm status changes:

Directory name		Property name	Explanation
JP1 Event Configurations	Alarm	JP1 Event Mode	• JP1 User Event: A JP1 user event is issued. • JP1 System Event: A JP1 system event is issued.
Detail Records		--	Stores the properties of PD-type records. The record ID of the record being collected is displayed in bold letters.
Detail Records	*record-ID*[#1]	--	Stores record properties.
		Description	Shows record description. This property cannot be modified.
		Log	Yes or No is selected from a list to specify whether to register records in the Store database. If this value is Yes and the Collection Interval is greater than 0, records are registered in the database.
		Log (ITSLM)	If there is a link to JP1/SLM - Manager, this property displays Yes or No indicating whether records from JP1/SLM - Manager are to be registered in the Store database of PFM - Agent Platform. If there is no linkage, this value is fixed to No. This property cannot be changed.
		Monitoring (ITSLM)	If there is a link to JP1/SLM - Manager, a setting in JP1/SLM - Manager displays Yes or No indicating whether records are to be sent to JP1/SLM - Manager. If there is no linkage, this value is fixed to No. This property cannot be changed.
		Collection Interval[#2]	Specifies the data collection interval. A value in the range from 0 to 2,147,483,647 seconds can be specified in 1-second increments. If 0 is specified, the collection interval is 0 and no data is collected.
		Collection Offset[#2]	Specifies the offset value for starting data collection. A value in the range from 0 to 32,767 seconds can be specified in 1-second increments, but cannot exceed the range specified for Collection Interval. Regardless of the value specified for Collection Offset, the data collection recording time is the same as the value specified for Collection Interval.
		Over 10 Sec Collection Time	This property is displayed only when historical data collection takes priority over real-time report display processing. (The history collection priority function is enabled.) Whether collection of records takes 10 seconds or longer is displayed as Yes or No. • Yes: It sometimes takes 10 seconds or longer. • No: It takes less than 10 seconds. You cannot change this property.

Directory name		Property name	Explanation
`Detail Records`	*record-ID*#1	Realtime Report Data Collection Mode	This property is displayed only when historical data collection takes priority over real-time report display processing. (The history collection priority function is enabled.) Specify either of the following real-time report display modes: • `Reschedule`: Reschedule mode • `Temporary Log`: Temporary log mode Note that you must specify the temporary log mode (`Temporary Log`) for records for which `Over 10 Sec Collection Time` is set to `Yes`.
		LOGIF	Specifies the condition for registering records in the database. Only those records satisfying the condition are registered in the database. Displays the conditional expression (character string) created in the LOGIF Expression Setting window, which is displayed by clicking **LOGIF** on the bottom frame of the Property window of the service that is displayed under the **Services** tab in PFM - Web Console.
		Sync Collection With	Synchronizes collection with the record specified in the following format: `Sync Collection With=`*record-type*`,` *record-ID* *Example:* `Sync Collection With=Interval Records, PI`
`Interval Records`		--	Stores the properties of `PI`-type records. The record ID of the record being collected is displayed in bold letters.
`Interval Records`	*record-ID*#1	--	Stores record properties.
		Description	Shows record description. This property cannot be modified.
		Log	`Yes` or `No` is selected from a list to specify whether to register records in the Store database. If this value is `Yes` and the Collection Interval is greater than `0`, records are registered in the database.
		Log (ITSLM)	If there is a link to JP1/SLM - Manager, this property displays `Yes` or `No` indicating whether records from JP1/SLM - Manager are to be registered in the Store database of PFM - Agent Platform. If there is no linkage, this value is fixed to `No`. This property cannot be changed.
		Monitoring (ITSLM)	If there is a link to JP1/SLM - Manager, a setting in JP1/SLM - Manager displays `Yes` or `No` indicating whether records are to be sent to JP1/SLM - Manager. If there is no linkage, this value is fixed to `No`. This property cannot be changed.
		Collection Interval	Specifies the data collection interval. A value in the range from 0 to 2,147,483,647 seconds

Directory name		Property name	Explanation
`Interval Records`	*record-ID*[#1]	Collection Interval	can be specified in 1-second increments. If `0` is specified, the collection interval is `0` and no data is collected.
		Collection Offset	Specifies the offset value for starting data collection. A value in the range from 0 to 32,767 seconds can be specified in 1-second increments, but cannot exceed the range specified for Collection Interval. Regardless of the value specified for Collection Offset, the data collection recording time is the same as the value specified for Collection Interval.
		Over 10 Sec Collection Time	This property is displayed only when historical data collection takes priority over real-time report display processing. (The history collection priority function is enabled.) Whether collection of records takes 10 seconds or longer is displayed as `Yes` or `No`. • `Yes`: It sometimes takes 10 seconds or longer. • `No`: It takes less than 10 seconds. You cannot change this property.
		Realtime Report Data Collection Mode	This property is displayed only when historical data collection takes priority over real-time report display processing. (The history collection priority function is enabled.) Specify either of the following real-time report display modes: • `Reschedule`: Reschedule mode • `Temporary Log`: Temporary log mode Note that you must specify the temporary log mode (`Temporary Log`) for records for which `Over 10 Sec Collection Time` is set to `Yes`.
		LOGIF	Specifies the condition for registering records in the database. Only those records satisfying the condition are registered in the database. Displays the conditional expression (character string) created in the LOGIF Expression Setting window, which is displayed by clicking **LOGIF** on the bottom frame of the Property window of the service that is displayed under the **Services** tab in PFM - Web Console.
		Sync Collection With	Synchronizes collection with the record specified in the following format: `Sync Collection With=`*record-type*`,` *record-ID* *Example:* `Sync Collection With=Interval Records, PI`
`Log Records`		--	Stores the properties of `PL`-type records. The record ID of the record being collected is displayed in bold letters.
`Log Records`	*record-ID*[#1]	--	Stores record properties.

Directory name		Property name	Explanation
`Log Records`	*record-ID*[#1]	Description	Shows record description. This property cannot be modified.
		Log	`Yes` or `No` is selected from a list to specify whether to register records in the Agent Store database. If this value is `Yes` and the Collection Interval is greater than `0`, records are registered in the database.
		Collection Interval	Specifies the data collection interval. A value in the range from 0 to 2,147,483,647 seconds can be specified in 1-second increments. If `0` is specified, the collection interval is `0` and no data is collected.
		Collection Offset	Specifies the offset value for starting data collection. A value in the range from 0 to 32,767 seconds can be specified in 1-second increments, but cannot exceed the range specified for Collection Interval. Regardless of the value specified for Collection Offset, the data collection recording time is the same as the value specified for Collection Interval.
		LOGIF	Specifies the condition for registering records in the database. Only those records satisfying the condition are registered in the database. Displays the conditional expression (character string) created in the LOGIF Expression Setting window, which is displayed by clicking **LOGIF** on the bottom frame of the Property window of the service that is displayed under the **Services** tab in PFM - Web Console.
		Sync Collection With	Synchronizes collection with the record specified in the following format: `Sync Collection With=`*record-type*`,` *record-ID* *Example:* `Sync Collection With=Interval Records, PI`
`Restart Configurations`		--	Sets the condition for automatically restarting the PFM service. The condition can be set when the version of PFM - Manager or PFM - Base is 09-00 or later. For details about the PFM service Auto-restart functionality, see the chapter explaining the Performance Management functions in the *JP1/Performance Management Planning and Configuration Guide*.
		Restart when Abnormal Status	Specifies whether to automatically restart the service if the Status Server service cannot correctly obtain the statuses of the Action Handler service, Agent Collector service, and Agent Store service.
		Restart when Single Service Running	Specifies whether to automatically restart the service if only the Agent Store service or only the Agent Collector service is running.
`Restart Configurations`	`Action Handler`	Auto Restart	Specifies whether to use the Auto-restart functionality for the Action Handler service.

Directory name		Property name	Explanation
Restart Configurations	Action Handler	Auto Restart - Interval (Minute)	When using the Auto-restart functionality, sets the interval for confirming the service's operating status. Values between 1 and 1,440 minutes can be set, in 1 minute increments.
		Auto Restart - Repeat Limit	When using the Auto-restart functionality, sets an integer between 1 and 10 as the number of consecutive restart retries.
		Scheduled Restart	Yes or No is selected from a list to specify whether to use the Scheduled restart functionality on the Action Handler service.
		Scheduled Restart - Interval	When using the Scheduled restart functionality, sets an integer between 1 and 1,000 as the restart interval.
		Scheduled Restart - Interval Unit	When using the Scheduled restart functionality, Hour, Day, Week, or Month is selected from a list to specify the unit for the restart interval.
		Scheduled Restart - Origin - Year	The restart year can be specified as an integer in the range from 1971 to 2035.
		Scheduled Restart - Origin - Month	The restart month can be specified as an integer in the range from 1 to 12.
		Scheduled Restart - Origin - Day	The restart day can be specified as an integer in the range from 1 to 31.
		Scheduled Restart - Origin - Hour	The restart time (hour) can be specified as an integer in the range from 0 to 23.
		Scheduled Restart - Origin - Minute	The restart time (minute) can be specified as an integer in the range from 0 to 59.
	Agent Collector	Auto Restart	Specifies whether to use the Auto-restart functionality for the Agent Collector service.
		Auto Restart - Interval (Minute)	Sets the interval (in minutes) for checking the operating status of the service when the Auto-restart functionality is used. A value in the range from 1 to 1,440 minutes can be set in 1-minute increments.
		Auto Restart - Repeat Limit	Sets (as an integer from 1 to 10) the number of times in succession a restart is to be attempted when the Auto-restart functionality is used.
		Scheduled Restart	Yes or No is selected from a list to specify whether to use the scheduled restart functionality for the Action Handler service.
		Scheduled Restart - Interval	Sets (as an integer from 1 to 1000) the restart interval when the scheduled restart functionality is used.
		Scheduled Restart - Interval Unit	When the scheduled restart functionality is used, sets the unit of the restart interval to Hour, Day, Week, or Month, as selected from a list.

Directory name		Property name	Explanation
Restart Configurations	Agent Collector	Scheduled Restart - Origin - Year	The restart year can be specified as an integer in the range from 1971 to 2035.
		Scheduled Restart - Origin - Month	The restart month can be specified as an integer in the range from 1 to 12.
		Scheduled Restart - Origin - Day	The restart day can be specified as an integer in the range from 1 to 31.
		Scheduled Restart - Origin - Hour	The restart time (hour) can be specified as an integer in the range from 0 to 23.
		Scheduled Restart - Origin - Minute	The restart time (minute) can be specified as an integer in the range from 0 to 59.
	Agent Store	Auto Restart	Specifies whether to use the Auto-restart functionality for the Agent Store service.
		Auto Restart - Interval (Minute)	Sets the interval (in minutes) for checking the operating status of the service when the Auto-restart functionality is used. A value in the range from 1 to 1,440 minutes can be set in 1-minute increments.
		Auto Restart - Repeat Limit	Sets (as an integer from 1 to 10) the number of times in succession a restart is to be attempted when the Auto-restart functionality is used.
		Scheduled Restart	Yes or No is selected from a list to specify whether the scheduled restart functionality for the Agent Collector service is to be used.
		Scheduled Restart - Interval	Sets (as an integer from 1 to 1000) the restart interval when the scheduled restart functionality is used.
		Scheduled Restart - Interval Unit	When the scheduled restart functionality is used, sets the unit of the restart interval to Hour, Day, Week, or Month, as selected from a list.
		Scheduled Restart - Origin - Year	The restart year can be specified as an integer in the range from 1971 to 2035.
		Scheduled Restart - Origin - Month	The restart month can be specified as an integer in the range from 1 to 12.
		Scheduled Restart - Origin - Day	The restart day can be specified as an integer in the range from 1 to 31.
		Scheduled Restart - Origin - Hour	The restart time (hour) can be specified as an integer in the range from 0 to 23.
		Scheduled Restart - Origin - Minute	The restart time (minute) can be specified as an integer in the range from 0 to 59.
ITSLM Connection Configuration		--	Shows information about the connection to the JP1/SLM - Manager.
ITSLM Connection Configuration	ITSLM Connection	--	Shows information about the JP1/SLM - Manager connection destination.
		ITSLM Host	Shows the host name of the connected JP1/SLM - Manager. If there is no connection

Directory name		Property name	Explanation
ITSLM Connection Configuration	ITSLM Connection	ITSLM Host	to JP1/SLM - Manager, this property is not shown.
		ITSLM Port	Shows the port number of the connected JP1/SLM - Manager. If there is no connection to JP1/SLM - Manager, this property is not shown.
	MANAGE ITSLM CONNECTION	--	Sets whether to stop the connection to JP1/SLM - Manager.
		DISCONNECT ITSLM CONNECTION	Specifies from a list of items the host name of the JP1/SLM - Manager to disconnect from. If you specify the empty string from the list, nothing happens. If there is no connection to JP1/SLM - Manager, the list shows only the empty string.
Multiple Manager Configuration		Primary Manager	Displays the host name of the monitoring manager specified as the primary manager for multiple monitoring. You cannot change this property.
		Secondary Manager	Displays the host name of the monitoring manager specified as the secondary manager for multiple monitoring. You cannot change this property.
Agent Configuration		--	Stores the properties that define information specific to PFM - Agent for Platform.
		ALL Zone Collection for Process#3	Selects whether process information only for the global zone is to be collected. If PFM - Agent for Platform is running in a non-global zone, process information only for that non-global zone is collected, regardless of the property setting. • Yes: Process information for the global zone and non-global zones is collected. • No: Process information only for the global zone is collected.
		ALL WPAR Collection for Process#4	Selects whether process information only for the global environment is to be collected. If PFM - Agent for Platform is running in a system WPAR environment, process information only for that system WPAR environment is collected, regardless of the property setting. • Yes: Process information for the global environment and WPAR environments is collected. • No: Process information only for the global environment is collected.
		ALL Container Collection for Process#5	Selects whether process information only for the Docker host environment or the Podman host environment is to be collected. If PFM - Agent for Platform is running in a Docker container environment or the Podman container environment, process information only for that container environment is collected, regardless of the property setting.

Directory name	Property name	Explanation
Agent Configuration	ALL Container Collection for Process[#5]	• Yes: Process information for the host environment and all container environments is collected. • No: Process information only for the host environment is collected.
	Messages File[#6]	Specifies the syslog file name as an absolute path. To change the setting, you must use PFM - Web Console. You can specify a path name that does not exceed 256 bytes. The default values are as follows: • HPUX : /var/adm/syslog/syslog.log • Solaris : /var/adm/messages • AIX : /var/adm/syslogfile
	User Defined Record DataMaxCount[#7]	Specifies the maximum number of instances for each user record.
	sar Command Monitoring[#4]	Specifies whether to collect processor performance data from sar command output results (the default is No): • Yes: Collect from sar command output results. • No: Do not collect from sar command output results.
	sar Command Interval[#4]	When Yes is specified in sar Command Monitoring, specifies the interval (sec) specified as a sar command argument. Specify (as an integer between 1 and 50) the numeric value set in the INTERVAL argument of the following AIX sar command (the default is 5): /usr/sbin/sar -P ALL INTERVAL 1
	Discard Dummy File System	Specifies whether to completely discard information about dummy file systems during PD_FSL record collection. • Yes: Completely discards information about dummy file systems. • No: Does not discard information about dummy file systems. This property is displayed only in Linux. If this property is set to No, some information about dummy file systems might be collected. To completely discard information about dummy file systems, specify Yes.
	NFS File System	Specifies whether to collect information about NFS file systems during collection of file system information. • Yes: Collects information about NFS file systems. • No: Does not collect information about NFS file systems.

Directory name			Property name	Explanation
Application monitoring setting	*instance-name*#8		--	Shows the monitoring instance name that is added.
			Process[01-15] Kind#9	Shows the process type: • None: No process is specified. • Execute: See the value of the ps -e command. • Command Line: See the value of the ps -ef command.
			Process[01-15] Name#9	Enter a process name that does not exceed 127 bytes.
			Process[01-15] Range#9	Insert a hyphen between the limits (for example, 1-2). If a single value without a hyphen is specified, the same value can be specified for the upper and lower limits. For example, when 10 is specified, 10-10 is set. A value in the range from 0 to 65535 can be specified.
ADDITION OR DELETION A SETTING			ADD AN APPLICATION MONITORING SETTING	Shows the monitoring instance name to be added.
			DELETE AN APPLICATION MONITORING SETTING	Selects the monitoring instance name to be deleted. Specify None if you do not want to delete a monitoring instance.
			ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT	Sets the range of data to be collected for the processes in the PD_APP record. A value in the range from 0 to 2147483647 can be specified. Characters cannot be entered.
Advanced application monitoring			--	Specifies settings for collecting information about the operating status of processes.
Advanced application monitoring	Application monitoring setting		Case Sensitive	Specifies whether to distinguish letter-case in comparisons to monitoring conditions. • Yes: Case sensitive • No: Not case sensitive
	Application monitoring setting	*application-name*#10	--	The name of an added application.
			Virtual Environment ID#11	Sets the identifier of a virtualized environment (container, etc.). This setting is used to specify the range of process data to be collected for the PD_APP2 and PD_APPD records. Sets 0 if you collect data only from the global environment, the Docker host environment, or the Podman host environment. • In the case of AIX and Solaris A value in the range from 0 to 2147483647 can be set. • In the case of Linux A 64-byte hexadecimal value can be set. If this property is not set, data collection is performed for all processes.

Directory name			Property name	Explanation
Advanced application monitoring	Application monitoring setting	*application-name*#10	Monitoring[01-15] Label#12	Specifies a name of no more than 31 bytes for identifying the monitoring condition. Monitoring[01-15] is set by default. If nothing is specified, Monitoring[01-15] is set. The value specified for this property must be unique.
			Monitoring[01-15] Field#12	The field to be monitored. • None: No field specified. • Program Name: See the Program Name field of the PD_APS record. • Command Line: See the Command Line field of the PD_APS record. The default is None.
			Monitoring[01-15] Condition#12	Specifies the monitoring condition (as a maximum of 4,096 bytes). The default is one space.
			Monitoring[01-15] Range#12	Specifies the minimum and maximum thresholds for the monitoring count, connected by a hyphen (-) (for example, 1-2). You can specify values in the range from 0 to 65535.
User Command Setting			User Command Execution Timing	By using the functionality for periodically executing user commands, specify when to execute user commands. • After: Execute user commands after record collection. • Before: Execute user commands before record collection. By default, After is set.
			User Command Timeout	If you select Before as the User Command Execution Timing property of the functionality for periodically executing user commands, set the time (seconds)#13 when the execution of user commands is discontinued by specifying an integer in the range from 1 to 86400. By default, 5 is set.
User Command Setting	*user-record-name*#14		Execute	Specifies whether to use the function for periodically executing user commands. • Yes: Use. • No: Do not use.
			UserCommand	Specifies the absolute path for user commands. The maximum length of the string that can be specified for an absolute path is 255 bytes. Alphanumeric characters and symbols can be specified, except for the following characters: | < >

Legend:

--: Not applicable

#1

For the directory name, a record ID without a database ID is displayed. For details about record IDs of individual records, see *9. Records*.

#2

If Sync Collection With is displayed, neither Collection Interval nor Collection Offset is displayed.

#3

Displayed only in Solaris.

#4

Displayed only in AIX.

#5

Displayed only in Linux.

#6

Not displayed in Linux.

#7

The target records for these properties are `PD_UPD`, `PD_UPDB`, `PI_UPI`, `PI_UPIB`, and `PI_XUI1` to `PI_XUI5` records.

#8

The instance name set in `ADD AN APPLICATION MONITORING SETTING` in `ADDITION OR DELETION A SETTING` is displayed for the directory name. For details about how to set the instance name, see *7.3 Settings for collecting information about the application operating status*.

#9

For Process[01-15] Kind, Process[01-15] Name, and Process[01-15] Range, a value in the range from 01 to 15 appears in place of [01-15] in each property name. For example, the property items that are actually displayed will be `Process01 Kind`, `Process07 Name`, and `Process15 Range`.

#10

The application name set in the property `ADD AN APPLICATION MONITORING SETTING` in **Advanced Application Monitoring** > **ADDITION OR DELETION A SETTING** is displayed for the directory name.

#11

This property exists only in Solaris, AIX and Linux.

#12

In `Monitoring[01-15] Label`, `Monitoring[01-15] Field`, `Monitoring[01-15] Condition`, and `Monitoring[01-15] Range`, a value in the range from `01` and `15` is set for `[01-15]` in each property name. Examples of the property items that are actually displayed include `Monitoring01 Label`, `Monitoring06 Field`, `Monitoring10 Condition`, and `Monitoring15 Range`.

#13

The specified time when the execution of user commands is discontinued must not affect the timing of other record collection processing.

#14

`PD_UPD`, `PD_UPDB`, `PI_UPI`, `PI_UPIB`, and `PI_XUI[1-5]` are displayed as folder names. `[1-5]` shown beside the directory name `PI_XUI` denotes a number in the range from 1 to 5 that is displayed beside the actual folder name, as in `PI_XUI1`, `PI_XUI3`, and `PI_XUI5`.

G. List of Files and Directories

This appendix lists the files and directories of PFM - Agent for Platform.

The installation directory for Performance Management is `/opt/jp1pc/`.

The following table lists the files and directories of PFM - Agent for Platform.

Table G–1: List of files and directories of PFM - Agent for Platform

Directory name	File name	Explanation
`/opt/jp1pc/agtu/`	--	Base directory of PFM - Agent for Platform (UNIX)
`/opt/jp1pc/agtu/agent/`	--	Base directory of Agent Collector service
	`COSLMMI.DB`	JP1/SLM linkage setting data file
	`COSLMMI.IDX`	Index file for JP1/SLM linkage setting data files
	`COSLMMI.LCK`	Lock file for JP1/SLM linkage setting data files
	`evfile`	Event file[#1]
	`evfile.model`	Model file of event file[#1]
	`jpc_container`	Script file for collecting information of processes on the Podman container[#2]
	`jpc_hostutil`	Host information acquisition program
	`jpc_process`	64-bit process information acquisition program[#3]
	`jpcagt.ini`[#4]	Service startup initialization file of Agent Collector
	`jpcagt.ini.model`	Model file of service startup initialization file of Agent Collector
	`jpcagtu`	Service executing program of Agent Collector
	`jpcapp`[#5]	Application definition file (09-00 and earlier)
	`jpcapp2`[#5]	Application definition file (10-00 and later)
	`jpcappcvt`	Convert command executable program
	`jpcappcvt.ini`	Convert command settings file
	`jpcappcvt.ini.model`	Model file for the convert command settings file
	`wgfile`	Workgroup file
	`wgfile.model`	Model file of workgroup file
`/opt/jp1pc/agtu/agent/log/`	--	Storage directory for internal log files of the Agent Collector service
`/opt/jp1pc/agtu/agent/jpcuser/`	--	Base directory of the function for collecting user-specific performance data
	`jpcuser`	Execution program for the function for collecting user-specific performance data
	`jpcuser.ini`	Configuration file for the function for collecting user-specific performance data

Directory name	File name	Explanation
/opt/jp1pc/agtu/agent/jpcuser/	jpcusercommand.ini	Configuration file for periodically executing user commands
	core	Core dump file for the function for collecting user-specific performance data
/opt/jp1pc/agtu/agent/jpcuser/debug/	--	jpcuser command debug log directory
/opt/jp1pc/agtu/agent/jpcuser/log/trace/	--	Trace log file (internal log file) storage directory
/opt/jp1pc/agtu/agent/jpcuser/log/public/	--	Public log file storage directory
/opt/jp1pc/agtu/agent/jpcuser/userdata/	--	User data file output directory
/opt/jp1pc/agtu/nls/	--	Message catalog installation directory
/opt/jp1pc/agtu/store/	--	Base directory of Agent Store service
	*.DB	Performance data file
	*.IDX	Index file of the performance data file
	*.LCK	Lock file of the performance data file
	jpcsto.ini#4	Service startup initialization file of Agent Store
	jpcsto.ini.model	Model file of service startup initialization file of Agent Store
	*.DAT	Data model definition file
/opt/jp1pc/agtu/store/backup/	--	Default database backup destination directory
/opt/jp1pc/agtu/store/partial/	--	Default database partial-backup destination directory
/opt/jp1pc/agtu/store/dump/	--	Default database export destination directory
/opt/jp1pc/agtu/store/import/	--	Default database import destination directory
/opt/jp1pc/agtu/store/log/	--	Storage directory for the Agent Store service internal log files
/opt/jp1pc/agtu/store/STPD/	--	PD database specific directory
/opt/jp1pc/agtu/store/STPI/	--	PI database specific directory
/opt/jp1pc/agtu/store/STPL/	--	PL database specific directory
/opt/jp1pc/auditlog/	--	Default output directory for action log files
	jpcauditn.log#6	Action log file
/opt/jp1pc/setup/	--	Setup file installation directory
	jpcagtuu.Z	Add archive file for PFM - Agent additional setup (UNIX)
	jpcagtuw.EXE	Add archive file for PFM - Agent additional setup (Windows)
/opt/jp1pc/setup/extract/	--	Setup file expansion directory
/opt/jp1pc/setup/update/	--	Version upgrade work directory

Directory name	File name	Explanation
/opt/jp1pc/setup/update/agtu/	--	Directory for PFM - Agent for Platform (UNIX) version upgrade
	.	Files for PFM - Agent for Platform (UNIX) version upgrade

Legend:

--: Not applicable

#1

This file is not installed in Linux.

#2

This file is not installed in HP-UX, Solaris, and AIX.

#3

This file is not installed in HP-UX, AIX, and Linux.

#4

Files having the lck extension may be created in the same directory. This file is internally used by Performance Management. You must not modify or delete the file.

Example: /opt/jp1pc/agtu/agent/jpcagt.ini
/opt/jp1pc/agtu/agent/jpcagt.ini.lck

#5

This file exists only when process monitoring settings are specified.

#6

n is a numeric value. The number of log files can be changed in the jpccomm.ini file.

H. Upgrade Procedure and Notes on Upgrading

To upgrade PFM - Agent for Platform to a newer version, install the newer version by overwriting the older version. For details about installation operations, see *4. Installation and Setup in UNIX*.

For notes about upgrading the Performance Management programs, see the section that presents the notes on upgrading in the chapter and appendix describing installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

The following are notes about upgrading PFM - Agent for Platform:

- When upgrading the PFM - Agent for Platform version, do not uninstall the older version. If you uninstall it, performance data and other data created with the older version are also deleted, and as a result, the newer version will not work.
- When a newer version of PFM - Agent for Platform program is installed by overwriting an older version, the following items are automatically updated:
  - Store database file of the Agent Store service
  - `ini` file
- When the PFM - Agent for Platform program is upgraded, the existing Store database is also automatically upgraded. It may take a few dozen of minutes for the installation to finish, depending on the size of the Store database.
- The maximum size of Store database files (*.DB and *.IDX) after upgrading is 1.8 GB. However, if a limit is set for the file size or file system by using the ulimit command, then the maximum file size is 90% of that limit, or 1.8 GB, whichever is less.

  If the size of Store database files exceeds this maximum size, after upgrading, the data that does not fit in the maximum size is deleted, and the message KAVE05834-W is output to the common message log.

  The data to be deleted is selected by using the following rules:

  - For the PD database or PL database

    Data is deleted sequentially, starting with the oldest data.
  - For the PI database

    Data is deleted sequentially based on the period by which it is summarized, in the following order: minute, hour, day, week, month, and year. When the summary period is the same, the data is deleted sequentially, starting with the oldest data.

  As mentioned above, old data may get deleted. If necessary, output the performance data to a CSV file before upgrading. For details on outputting a CSV file, see the chapter explaining outputting reports in the manual *JP1/Performance Management User's Guide*.
- When Store databases are upgraded, the data model changes, and therefore the disk usage of the Store databases changes. Based on the new data model, please review, and if necessary change, the collection items and frequency as well as the storage conditions of the Store databases.
- When an upgrade installation is performed, the Store database is automatically updated. Because of this, free space equal to twice the size of the Store database is temporarily needed on the disk where the database is stored. Therefore, before you perform an upgrade installation, make sure there is sufficient space on the disk where the Store database is stored.
- When PFM-Agent for Platform version 07-50 or earlier has already been installed, and an overwrite installation of PFM-Agent for Platform 08-00 or later is performed, some records are no longer supported. When you perform the upgrade installation, the following messages about this change are displayed by the PP installer:

  In the Performance Management - Agent Option for Platform that you are attempting to install, there is a limit on supported records.

For details, please see the "Release Notes."

For more information about the records that are no longer supported, please see *M. Version Changes*.

- If an overwrite installation from version 07-50 or earlier is performed, unsupported records are shown in both the `Detail Records` and `Interval Records` displayed in the properties of the PFM - Web Console's Agent Collector service, as well as in `Retention` displayed in the properties of the Agent Store service.

  You can change the properties of these records, but you cannot edit them, nor store them in the Agent Store database. To prevent unsupported records from being shown in `Detail Records` and `Interval Records`, you must copy the `jpcagt.ini.model` file that is in the in `/opt/jp1pc/agtu/agent` directory to `jpcagt.ini`, and reset the properties.

  Similarly, to prevent unsupported records from being shown in `Retention`, you must copy the `jpcsto.ini.model` file that is in the in `/opt/jp1pc/agtu/store` directory to `jpcsto.ini`, and reset the properties.

  The procedure for the above is as follows:

  1. Stop the service by using the `jpcspm stop -key agtu` command.
  2. Copy the `jpcagt.ini.model` file to `jpcagt.ini`.
  3. Copy the `jpcsto.ini.model` file to `jpcsto.ini`.
  4. Start the service by using the `jpcspm start -key agtu` command.
  5. Reset the property for each record by using PFM - Web Console, because the most recent settings are assumed as the default settings.

- In Linux 8 or CentOS 8, when you upgrade PFM - Agent for Platform to version 12-50 or later, you may need to review the `ALL Container Collection for Process` property setting. For details, see *2.3.21(5)(b) Notes on the Podman host environment*.

I. Version Compatibility

In PFM - Agent, there are product versions as well as data model versions.

Because data models maintain upward compatibility, report definitions and alarm definitions created in the older version can also be used in the newer version of the data model.

The following table shows version compatibility of PFM - Agent for Platform.

Table I–1: PFM - Agent for Platform version compatibility

PFM - Agent for Platform version	Data model version	Alarm table version of the monitoring template
06-70	3.0	6.70
07-00	4.0	7.00
07-10	5.0	7.10
07-50	5.0	7.50
	5.1	
08-00	6.0	8.00
08-10	7.0	8.10
08-11	7.2	8.11
08-50	7.4	8.50
09-00	7.6	09.00
09-10	7.8	09.10
10-00		
11-00		
11-01		
11-10	8.0	
11-50	8.2	
12-00	8.4	
12-50		

In PFM - Agent for Platform, an alarm table of a version that is not available in the Performance Management system being used may sometimes be displayed under **Alarms** in the PFM - Web Console window. Therefore, check the alarm table version being used by the Performance Management system and its compatibility before using an alarm table.

For details about version compatibility, see the version compatibility description in the appendix to the *JP1/Performance Management Planning and Configuration Guide*.

J. Outputting Action Log Data

Action log data of Performance Management is history information that is output in cooperation with the alarm function related to exceeded thresholds, such as for a system load.

For example, if an alarm indicating an error such as an exceeded threshold occurs, data indicating when, and by which service, the alarm was issued is output to the action log.

Action log data can be output when the version of PFM - Manager or PFM - Base is 08-11 or later.

An action log is a CSV text file. By saving action logs periodically and converting them with spreadsheet software, you can use the action logs as analysis data.

Output of action log data is specified in the `jpccomm.ini` file. This appendix describes the contents of the action log data output by PFM - Agent and PFM - Base, and how to specify the settings for outputting the action log.

J.1 Types of events output to the action log

The following table shows the types of events output to the action log and the times at which PFM - Agent and PFM - Base output data to the action log. The event types are the identifiers used within the action log to classify.

Table J–1: Types of events output to the action log

Event type	Explanation	PFM - Agent and PFM - Base output the action log when:
`StartStop`	Events indicating that software has been started or terminated.	• A PFM service has been started or stopped. • Stand-alone mode has been started or terminated.
`ExternalService`	Events indicating the result of communication between a JP1 product and an external service. This event type also indicates that an abnormal communication has occurred.	The status of a connection with PFM - Manager has changed.
`ManagementAction`	Events indicating that an important program action has been executed. This event type also indicates that the action was executed in response to another audit category.	An automated action is executed.

J.2 Format for saving the action log files

This section explains the format for saving the action log files.

Action log data is output to a specified file (the current output file). When the current output file becomes full, the action log data in that file is saved to another file (the shift file). The procedure for switching the file for storing action log data is as follows:

1. Action log data is output sequentially to the current output file `jpcaudit.log`.
2. When the current output file becomes full, the action log data is saved in a shift file.
   Shift files are named by suffixing a number to the name of the current output file. Each time the current output file becomes full, each shift file is renamed by incrementing its suffix by 1. Therefore, the file whose name has the largest number is the oldest log file.

Example:

When the current output file `jpcaudit.log` becomes full, the contents of the file are saved to the shift file `jpcaudit1.log`.

When the current output file becomes full again, the information is moved to `jpcaudit1.log`, and the existing shift file `jpcaudit1.log` is renamed to `jpcaudit2.log`.

Note that when the number of log files exceeds the number of saved log files (specified in the `jpccomm.ini` file), the oldest log file is deleted.

3. The current output file is initialized, and new action log data is written.

Whether action log data is to be output, the output destination, and the number of output files are specified in the `jpccomm.ini` file. For details about how to specify the `jpccomm.ini` file, see *J.4 Settings for outputting action log data*.

J.3 Format of output action log data

Data related to audit events is output to the Performance Management action log. Action log data is output to one file for one host. The action log data is output to a file on either of the following hosts:

- When a service is executed: The data is output to the file on the host on which the service runs.
- When a command is executed: The data is output to the file on the host on which the command was executed.

The following describes the format of the action log, the output destination, and the items that are output.

(1) Output format

```
CALFHM x.x,output-item-1=value-1,output-item-2=value-2,...,output-item-n=val
ue-n
```

(2) Output destination

```
/opt/jp1pc/auditlog/
```

The output destination for action log data can be changed in the `jpccomm.ini` file. For details about how to specify the `jpccomm.ini` file, see *J.4 Settings for outputting action log data*.

(3) Output items

There are two types of output items:

- Common output items
  Items that are always output by all JP1 products that output action log data
- Fixed output items
  Items that are optionally output by a JP1 product that outputs action log data

(a) Common output items

The following table lists and describes the common output items and their values. This table also includes the items and information output by PFM - Manager.

Table J–2: Common output items in action logs

No.	Output item		Value	Explanation
	Item name	Output attribute name		
1	Common specification identifier	--	`CALFHM`	Indicates the action log format
2	Common specification revision number	--	*x.x*	Revision number for managing action logs
3	Serial number	`seqnum`	*serial-number*	Serial number of the action log record
4	Message ID	`msgid`	`KAVE`*xxxxx-x*	Message ID of the product
5	Date and time	`date`	*YYYY-MM-DD*`T`*hh*`:`*mm*`:`*ss*`.`*sssTZD*#	Date, time, and time zone indication identifying when the action log was output
6	Program name	`progid`	`JP1PFM`	Name of the program for which the event occurred
7	Component name	`compid`	*service-ID*	Name of the component for which the event occurred
8	Process ID	`pid`	*process-ID*	Process ID of the process for which the event occurred
9	Location	`ocp:host`	• *host-name* • *IP-address*	Location where the event occurred
10	Event type	`ctgry`	• `StartStop` • `Authentication` • `ConfigurationAccess` • `ExternalService` • `AnomalyEvent` • `ManagementAction`	Category name used to classify the event output to the action log
11	Event result	`result`	• `Success` • `Failure` • `Occurrence`	Result of the event
12	Subject identification information	`subj:pid`	*process-ID*	One of the following: • Process ID of a process running as a user operation • Process ID of the process that caused the event • Name of the user who caused the event • Identification information in a one-to-one correspondence with the user
		`subj:uid`	*account-identifier* (PFM user/JP1 user)	
		`subj:euid`	*effective-user-ID* (OS user)	

Legend:

--: None

#

`T` is a separator between the date and the time.

TZD is the time zone specifier. One of the following values is output.

+*hh*:*mm*: The time zone is *hh*:*mm* ahead of UTC.

−*hh*:*mm*: The time zone is *hh*:*mm* behind UTC.

`Z`: The time zone is the same as UTC.

(b) Fixed output items

The following table lists and describes the fixed output items and their values. This table also includes the items and information output by PFM - Manager.

Table J–3: Fixed output items in action logs

No.	Output item		Value	Explanation
	Item name	Output attribute name		
1	Object information	`obj`	• *PFM - Agent-service-ID* • *added-deleted-or-updated-user-name* (PFM user)	Intended object for the operation
		`obj:table`	*alarm-table-name*	
		`obj:alarm`	*alarm-name*	
2	Action information	`op`	• `Start` • `Stop` • `Add` • `Update` • `Delete` • `Change Password` • `Activate` • `Inactivate` • `Bind` • `Unbind`	Information about the action that caused the event

3	Permissions information	`auth`	• Administrator:
			`Management`
			• General user:
			`Ordinary`
			• Windows:
			`Administrator`
			• UNIX:
			`SuperUser`
				Permissions information of the user who executed the command or service

		`auth:mode`	• PFM authentication mode:
			`pfm`
			• JP1 authentication mode:
			`jp1`
			• OS user:
			`os`
				Authentication mode of the user who executed the command or service

4	Output source	`outp:host`	*PFM - Manager-host-name*	Host that output the action log
5	Instruction source	`subjp:host`	• *login-host-name* • *execution-host-name* (only when the `jpctool alarm` command is executed)	Host that issued the instruction for the operation

No.	Output item		Value	Explanation
	Item name	Output attribute name		
6	Descriptive text	msg	*message*	Message that is output when an alarm occurs or when an automated action is executed

Whether the fixed output items are output and what they contain differ depending on when the action log data is output. The following describes the message ID and output data for each case.

■ A PFM service starts or stops (StartStop)

- Output host: The host on which the service is running
- Output component: The service that started or stopped

Item name	Attribute name	Value
Message ID	msgid	Started: KAVE03000-I Stopped: KAVE03001-I
Action information	op	Started: Start Stopped: Stop

■ Stand-alone mode starts or stops (StartStop)

- Output host: PFM - Agent host
- Output component: Agent Collector service and Agent Store service

Item name	Attribute name	Value
Message ID	msgid	Stand-alone mode has started: KAVE03002-I Stand-alone mode has terminated: KAVE03003-I

Notes:

1. No fixed output items are output.
2. When PFM - Agent is started, PFM - Agent services connect to the PFM - Manager host, register node information, and obtain the latest alarm definition information. If a connection with the PFM - Manager host cannot be established, PFM - Agent starts in stand-alone mode, in which only part of its functionality, such as collection of operating information, is enabled. In addition, KAVE03002-I is output to indicate that PFM - Agent has started in stand-alone mode. From this point, the PFM - Agent services periodically attempt to connect to PFM - Manager. When the services are able to successfully register node information or obtain definition information, PFM - Agent leaves stand-alone mode and KAVE03003-I is output. In this way, the action log enables you to understand that PFM - Agent was running in an imperfect condition for the period from the output of KAVE03002-I to the output of KAVE03003-I.

■ The status of the connection with PFM - Manager changes (ExternalService)

- Output host: PFM - Agent host
- Output component: Agent Collector service and Agent Store service

Item name	Attribute name	Value
Message ID	msgid	Sending of an event to PFM - Manager failed (queuing was started): KAVE03300-I.

Item name	Attribute name	Value
Message ID	msgid	An event was resent to PFM - Manager: KAVE03301-I.

Notes:

1. No fixed output items are output.
2. If the Agent Store service is unable to send an event to PFM - Manager, the Agent Store service starts queuing events, up to a maximum of three. When queuing is started after a failure to send an event, KAVE03300-I is output. When the connection with PFM - Manager is restored and all queued events have been sent, KAVE03301-I is output. In this way, the action log enables you to understand that real-time sending of events to PFM - Manager was disabled for the period from the output of KAVE03000-I to the output of KAVE03001-I.
3. The Agent Collector service usually sends events to PFM - Manager via the Agent Store service. It directly sends events to PFM - Manager only when the Agent Store service has stopped for some reason. If sending of events fails, KAVE03300-I is output but KAVE03301-I is not output, since no events are queued. In this way, the action log enables you to understand that some events have not been sent to PFM - Manager.

■ An automated action is executed (ManagementAction)

- Output host: The host on which the action was executed
- Output component: Action Handler service

Item name	Attribute name	Value
Message ID	msgid	The command execution process was created successfully: KAVE03500-I. An attempt to create a command execution process failed: KAVE03501-W. Email was send successfully: KAVE03502-I. Sending of email failed: KAVE03503-W
Free description	msg	Command execution: cmd=*executed-command-line*. Email sending: mailto=*destination-email-address*.

Note:

KAVE03500-I is output when the command execution process is created successfully. Thereafter, log information about whether the command was executed and about the execution results is not output to the action log.

(4) Output example

The following is an example of output action log data:

```
CALFHM 1.0, seqnum=1, msgid=KAVE03000-I, date=2007-01-18T22:46:49.682+09:00,
progid=JP1PFM, compid=UA1host01, pid=2076,
ocp:host=host01, ctgry=StartStop, result=Occurrence,
subj:pid=2076,op=Start
```

J.4 Settings for outputting action log data

The settings for outputting action log data are defined in the jpccomm.ini file. If no settings are specified, no action log data is output. The following describes the settings required to output action log data, and how to specify the settings.

(1) Setting procedure

To specify the settings for outputting action log data:

1. Stop all PFM services on the host.
2. Using a text editor, edit the `jpccomm.ini` file.
3. Save and close the `jpccomm.ini` file.

(2) Details about the jpccomm.ini file

The following describes the `jpccomm.ini` file in detail.

(a) Storage directory

`/opt/jp1pc/`

(b) Format

In the `jpccomm.ini` file, define the following information:

- Whether to output action log data
- Output destination of the action log
- Number of action logs that can be saved
- File size of the action log

The specification format is as follows:

"*item-name*"=*value*

The following table shows the items that you can specify.

Table J–4: Items specified in the jpccomm.ini file and their initial values

No.	Item	Explanation
1	`[Action Log Section]`	The section name, which cannot be changed.
2	`Action Log Mode`	Specify whether to output action log data. You must specify this item. • Initial value: `0` (Data not output) • Specifiable value: `0` (Data not output) or `1` (Information output) If any other value is specified, an error message is output and action log data will not be output.
3	`Action Log Dir`	Specify the action log output destination. If a path longer than the limit is specified or if access to the directory fails, an error message is output to the command log and action log data will not be output. • Initial value: None set • Default value used when no specification is made: `/opt/jp1pc/auditlog/` • Specifiable value: A character string from 1 to 185 bytes

No.	Item	Explanation
4	`Action Log Num`	Specify the upper limit on the total number of log files (number of saved files). Specify the sum of the number of current output files and shift files. • Initial value: None set • Default value used when no specification is made: `5` • Specifiable value: An integer in the range from 2 to 10 If a character string containing a non-numeric value is specified, an error message is output and the default value `5` is set. If a numeric value outside the valid range is specified, an error message is output and the integer nearest the specified value in the range from 2 to 10 is set.
5	`Action Log Size`	Specify the log file size in kilobytes. • Initial value: None set • Default value used when no specification is made: `2048` • Specifiable value: An integer in the range from 512 to 2096128 If a character string containing a non-numeric value is specified, an error message is output and the default value `2048` is set. If a numeric value outside the valid range is specified, an error message is output and the integer nearest the specified value in the range from 512 to 2096128 is set.

K. Linking to JP1/SLM

The monitoring performed by PFM - Agent for Platform can be enhanced by linking to JP1/SLM.

PFM - Agent for Platform provides for PFM - Manager a set of default JP1/SLM monitoring items to facilitate monitoring on JP1/SLM.

For multi-instance records, this monitoring collects records that match the values specified in keys. Check the collection results of each record associated with a collection target key.

Effective Free Memory is output on HP-UX and Solaris as `0`.

Table K–1: Default monitoring items that PFM - Agent for Platform provides to PFM - Manager

Display name in JP1/SLM	Description	Record (record ID)	Key (PFM - Manager name)	Field name
CPU Usage	CPU usage (%)	System Summary Overview (`PI`)	--	`KERNELMODE_USERMODE_PERCENT`
Available Memory	Total size of unused physical memory (megabytes)	System Summary Overview (`PI`)	--	`FREE_MEMORY_MBYTES`
Effective Free Memory	Megabytes of physical memory that applications can actually use	System Summary Overview (`PI`)	--	`EFFECTIVE_FREE_MEM_MBYTES`
File System Free Size	Amount of file system not in use (megabytes)	File System Detail - Local (`PD_FSL`)	File System (`FILESYSTEM_NAME`)	`TOTAL_MBYTES_FREE`
Disk Busy %	Disk busy rate (%)	Device Detail (`PI_DEVD`)	Device Name (`DEVICE_NAME`)	`BUSY_PERCENT`
Network Packets	Number of packets sent and received	Network Interface Detail (`PI_NIND`)	Interface (`INTERFACE_NAME`) IP Address (`IP_ADDRESS`) IP6 Address (`IP6_ADDRESS`)	`TOTAL_PACKETS`

Legend:

--: None

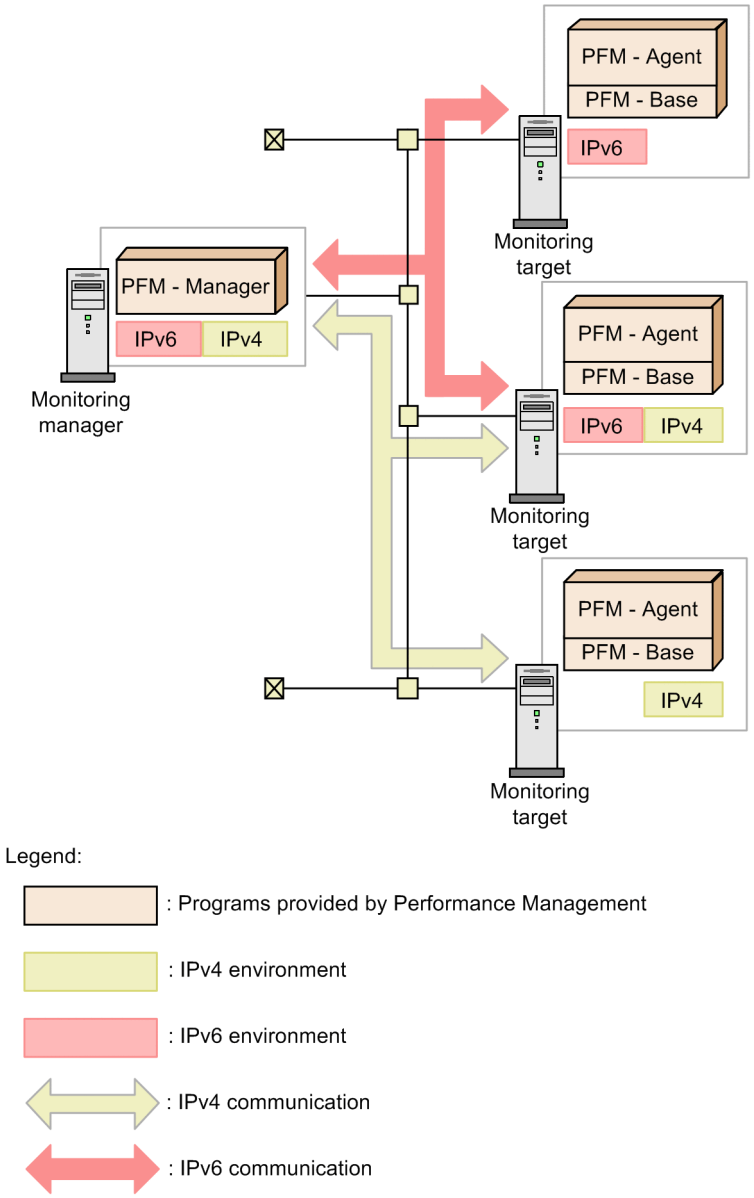
You can provide the default monitoring items to PFM - Manager by copying the setup files and executing the setup command. For details, see *4.4.2 Registering PFM - Agent for Platform in PFM - Manager and PFM - Web Console*.

L. Communication in IPv4 and IPv6 Environments

Performance Management supports IPv6 network configurations in addition to IPv4 environments. Performance Management can also be used in network configurations that support a mixture of IPv4 and IPv6 environments.

PFM - Agent for Platform can communicate with PFM - Manager over IPv6, but only if both are hosted on Windows or Linux.

Figure L–1: Application scope for communication in IPv4 and IPv6 environments



You execute the `jpcconf ipv6 enable` command to enable communication over IPv6. For details about the `jpcconf ipv6 enable` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*. For details about the timing and conditions for executing the `jpcconf ipv6 enable` command, see the chapter that describes network configurations that include IPv6 environments in the *JP1/Performance Management Planning and Configuration Guide*.

M. Version Changes

This appendix presents the changes in the manuals for each version.

M.1 Changes in 12-50

- The functionality of monitoring processes on host and container environments just like the Docker environment when the Podman environment is used in Linux.

  Based on this change, descriptions common to the Docker and Podman environments were changed to the descriptions on the *Linux container*.

  In addition, a script file to collect information of processes on the Podman container in Linux was added.
- The following message was added:
  `KAVF10713-W`
- A note on collecting dump information when troubleshooting was added.
- Commands used to collect system information (Linux) during troubleshooting were added.

M.2 Changes in 12-00#

- The following OSs are no longer supported:

  PFM - Manager and PFM - Web Console
  - Microsoft(R) Windows Server(R) 2008 R2
  - AIX V6.1
  - AIX V6.2
  - AIX V7.1

  PFM - Base
  - Microsoft(R) Windows Server(R) 2008 R2
  - AIX V6.1
  - Solaris 10 (SPARC)

  PFM - Agent for Platform
  - AIX V6.1
  - Solaris 10 (SPARC)
- Notes for usage that were listed in the *Release Notes* were added.
- The following fields in the `PI` record are now available in all OSs supported as a monitoring target OS by PFM - Agent for Platform:
  - Effective Free Mem %
  - Effective Free Mem Mbytes

  In addition, the following fields were added to the `PI` record:
  - Effective Used Mem %
  - Effective Used Mem Mbytes

- The following fields in the `PI` record are now available in Solaris:
  - Cache Mem %
  - Cache Mem Mbytes

  In addition, the description by OS about availability of the following fields was modified:
  - Buffers Mem %
  - Buffers Mem Mbytes
  - Cache Mem %
  - Cache Mem Mbytes
- The following reports were added:
  - Memory Paging (8.4)
  - Memory Paging Status (8.4)
  - Memory Paging Status (Multi-Agent) (8.4)
  - Paging Trend (Multi-Agent) (8.4)
  - System Overview (8.4)
- The explanation for the common message log was changed.
- The command `echo "::memstat" | mdb -k` was added to the implemented memory size information (Solaris) in the performance data information that needs to be collected when an error occurs.
- The description about the system resources that require adjustment of the kernel parameters was added.
- The following OSs are now supported:
  - Red Hat(R) Enterprise Linux(R) Server 8.1 and later
  - CentOS 8.1 and later
  - Oracle Linux(R) Operating System 8.1 and later
  - SUSE Linux(R) Enterprise Server 15
- The description of when you use the CPU hotplug function of Linux was added.

#

These changes include the functions supported after the release of 12-00.

M.3 Changes in 11-50

- Processes running in each container can now be monitored when the Docker environment is used with Linux.
- A modification has been introduced so that when no user data file has been created due to a failure to execute a user command or for other reasons, collection of user-specific performance data is skipped.
- The file size of `PD` records, `PD_APP2` records, `PD_APSI` records, and `PD_APS` records was modified.
- A description of Oracle Linux 7 was added to the note on `PI_DEVD` records.
- The following messages were added:
  `KAVF10707-W`, `KAVF10709-I`, `KAVF10710-W`, `KAVF10711-W`
- The following messages were changed:
  `KAVF10013-W`, `KAVF10907-W`

- The following properties were added to the Agent Collector service:
  - ALL Container Collection for Process

  The description of the following property was modified.
  - Virtual Environment ID

M.4 Changes in 11-10

- The following OS is now supported:
  - AIX V7.2
- The following records were added to user-specific performance data:
  `PI_XUI1`, `PI_XUI2`, `PI_XUI3`, `PI_XUI4`, and `PI_XUI5`
- By using the functionality for periodically executing user commands, the user can now select when to execute user commands.
- The following messages were added:
  `KAVF10703-W`, `KAVF10704-I`, `KAVF10705-W`, `KAVF10706-I`

M.5 Changes in 11-01

- A logical host environment is now supported and cluster operation is now possible.
- The following messages were added:
  `KAVF10504-E`, `KAVF10550-E` to `KAVF10552-E`, and `KAVF10850-E` to `KAVF10853-E`
- The following messages were changed:
  `KAVF10201-E` and `KAVF10957-I`
- A description indicating that unique port numbers must be assigned within the device was added.

M.6 Changes in 11-00

(1) Changes from the manual (3021-3-057(E)) to the manual (3021-3-A52(E))

- The following OSs are no longer supported:
  PFM - Manager and PFM - Web Console
  - Microsoft(R) Windows Server(R) 2003
  - Microsoft(R) Windows Server(R) 2008 (except R2)
  - AIX 6 (32-bit)
  - AIX 7 (32-bit)
  - HP-UX 11i V3 (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 (x86)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)

  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (x86)
  - Red Hat(R) Enterprise Linux(R) Server 6 (32-bit x86)
  - Solaris 10

  PFM - Base and PFM - Agent for Platform

  - AIX 6 (32-bit)
  - AIX 7 (32-bit)
  - Red Hat(R) Enterprise Linux(R) 5 (x86)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) Server 6 (32-bit x86)
- The following OSs are now supported:
  - CentOS 6.1 (x64) or later
  - CentOS 7.1 or later
  - Red Hat(R) Enterprise Linux(R) Server 7.1 or later
  - Oracle Linux(R) Operating System 6.1 (x64) or later
  - Oracle Linux(R) Operating System 7.1 or later
  - SUSE Linux(R) Enterprise Server 12
- The following properties were added:

  Agent Store service properties

  - Multiple Manager Configuration

  Agent Collector service properties

  - Discard Dummy File System
  - Monitoring Console Https
  - Multiple Manager Configuration
  - NFS File System
  - Over 10 Sec Collection Time
  - Realtime Report Data Collection Mode
- The product name was changed from JP1/ITSLM to JP1/SLM.
- Linkage with Network Node Manager (NNM) was discontinued.
- ODBC-based application programs were discontinued.
- The following languages were added as languages available in Performance Management:
  - Korean
  - Spanish
  - Chinese (simplified characters)
  - German
  - French
  - Russian
- It is now possible to check alarm application status and apply alarm information.

- The correspondence between the `LANG` environment variable for the OS and the character encoding used in the file to be exported was changed for the following commands:
  - `jpcconf agttree export`
  - `jpcconf bgdef export`
  - `jpctool alarm export`
- Notes on migration (for 11-00 or later) were added.
- Notes were added for when processor performance is monitored in an AIX environment according to the output results of the `sar` command.

(2) Changes from the manual (3021-3-355(E)) to the manual (3021-3-A52(E))

- The following OSs are no longer supported:
  PFM - Manager and PFM - Web Console
  - Microsoft(R) Windows Server(R) 2003
  - Microsoft(R) Windows Server(R) 2008 (except R2)
  - AIX 6 (32-bit)
  - AIX 7 (32-bit)
  - HP-UX 11i V3 (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 (x86)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (x86)
  - Red Hat(R) Enterprise Linux(R) Server 6 (32-bit x86)
  - Solaris 10

  PFM - Base and PFM - Agent for Platform
  - AIX 6 (32-bit)
  - AIX 7 (32-bit)
  - Red Hat(R) Enterprise Linux(R) 5 (x86)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) Server 6 (32-bit x86)
- The following OSs are now supported:
  - CentOS 6.1 (x64) or later
  - CentOS 7.1 or later
  - Red Hat(R) Enterprise Linux(R) Server 7.1 or later
  - Oracle Linux(R) Operating System 6.1 (x64) or later
  - Oracle Linux(R) Operating System 7.1 or later
  - SUSE Linux(R) Enterprise Server 12
- The following products were added as monitoring agents:

  - PFM - Agent for Cosminexus
  - PFM - Agent for DB2
  - PFM - Agent for Domino
  - PFM - Agent for Exchange Server
  - PFM - Agent for HiRDB
  - PFM - Agent for IIS
  - PFM - Agent for OpenTP1
  - PFM - Agent for WebLogic Server
  - PFM - Agent for WebSphere Application Server
- The following properties were added:
  Agent Store service properties
  - Multiple Manager Configuration

  Agent Collector service properties
  - Discard Dummy File System
  - Monitoring Console Https
  - Multiple Manager Configuration
  - NFS File System
  - Over 10 Sec Collection Time
  - Realtime Report Data Collection Mode
- The product name was changed from JP1/ITSLM to JP1/SLM.
- Linkage with Network Node Manager (NNM) was discontinued.
- ODBC-based application programs were discontinued.
- The following languages were added as languages available in Performance Management:
  - Korean
  - Spanish
  - Chinese (simplified characters)
  - German
  - French
  - Russian
- It is now possible to check alarm application status and apply alarm information.
- The correspondence between the `LANG` environment variable for the OS and the character encoding used in the file to be exported was changed for the following commands:
- `jpcconf agttree export`
- `jpcconf bgdef export`
- `jpctool alarm export`
- Notes on migration (for 11-00 or later) were added.
- Notes were added for when processor performance is monitored in an AIX environment according to the output results of the `sar` command.

M.7 Changes in 10-00

(1) Changes in the manual (3021-3-057(E))

- The following OSs are no longer supported:
  - AIX 5L V5.3
  - HP-UX 11i V2 (IPF)
  - Solaris 9 (SPARC)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 (IPF)
  - Red Hat(R) Enterprise Linux(R) AS 4 (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) AS 4 (AMD64 & Intel EM64T)
  - Red Hat(R) Enterprise Linux(R) ES 4 (AMD64 & Intel EM64T)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (x86)
  - Red Hat(R) Enterprise Linux(R) 5 (x86)
  - Red Hat(R) Enterprise Linux(R) AS 4 (x86)
  - Red Hat(R) Enterprise Linux(R) ES 4 (x86)
- Process monitoring conditions can now be specified by using up to 4,096 bytes of data.
- Linkage to JP1/SLM was facilitated by providing the following default monitoring items to PFM - Manager:
  - `PCT_TOTAL_PROCESSOR_TIME`
  - `AVAILABLE_BYTES`
  - `FREE_MEGABYTES`
  - `PCT_DISK_TIME`
  - `BYTES_TOTAL_PER_SEC`
- Performance data can now be collected in the IPv6 environment when the host on which PFM - Agent for Platform is installed is Linux.

(2) Changes in the manual (3021-3-355(E))

- The following OSs are no longer supported:
  - AIX 5L V5.3
  - HP-UX 11i V2 (IPF)
  - Solaris 9 (SPARC)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 (IPF)
  - Red Hat(R) Enterprise Linux(R) AS 4 (IPF)
  - Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (AMD/Intel 64)
  - Red Hat(R) Enterprise Linux(R) 5 (AMD/Intel 64)

- Red Hat(R) Enterprise Linux(R) AS 4 (AMD64 & Intel EM64T)
- Red Hat(R) Enterprise Linux(R) ES 4 (AMD64 & Intel EM64T)
- Red Hat(R) Enterprise Linux(R) 5 Advanced Platform (x86)
- Red Hat(R) Enterprise Linux(R) 5 (x86)
- Red Hat(R) Enterprise Linux(R) AS 4 (x86)
- Red Hat(R) Enterprise Linux(R) ES 4 (x86)

- Process monitoring conditions can now be specified by using up to 4,096 bytes of data.
- Linkage to JP1/SLM was facilitated by providing the following default monitoring items to PFM - Manager:
  - `PCT_TOTAL_PROCESSOR_TIME`
  - `AVAILABLE_BYTES`
  - `FREE_MEGABYTES`
  - `PCT_DISK_TIME`
  - `BYTES_TOTAL_PER_SEC`
- Performance data can now be collected in the IPv6 environment when the host on which PFM - Agent for Platform is installed is Linux.

M.8 Changes in 09-10

(1) Changes in the manual (3020-3-R49-21(E))

- Capability to monitor the operating status of processes was added.
- As part of the capability to monitor the operating status of processes, the following records were added.
  - Application Process Detail (`PD_APPD`)
  - Application Process Interval (`PD_APSI`)
  - Application Process Overview (`PD_APS`)
  - Application Summary Extension (`PD_APP2`)
- As part of the capability to monitor the operating status of processes, the following alarms were added:
  - Application Status
  - Process Existence
- As part of the capability to monitor the operating status of processes, the PFM UNIX Template Alarms [PS] 09.10 alarm table version was changed from 9.00 to 09.10.
- A functionality for monitoring processor performance based on results output by the `sar` command in an AIX environment has been added.
- Setup commands can now be executed non-interactively.
- As part of the capability to monitor the operating status of processes, the `jpcapp2` application definition file was added.
- As part of the capability to monitor the operating status of processes, the PFM UNIX Template Alarms [APP] 09.10 alarm table was added.
- As part of the capability to monitor the operating status of processes, the following reports were added:

- Application Process Count
- Application Process Status
- Application Status
- Monitoring Process Detail

- As part of the capability to monitor the operating status of processes, the `jpcappcvt` command was added.
- The following messages were added:
  `KAVF10800-Q`, `KAVF10801-I`, `KAVF10802-E`, `KAVF10803-W`, `KAVF10804-W`, `KAVF10805-W`, `KAVF10806-Q`, `KAVF10807-E`, `KAVF10808-E`, `KAVF10809-E`, `KAVF10810-E`
- The description of the `KAVF10100-E` message was changed.
- The `jpcapp2` file was added to the information that is required to be collected in Performance Management.
- A list of items about performance data to be collected when a problem occurs was added.
- Estimates of memory requirements were changed.
- Estimates of disk space requirements were changed.
- As part of the capability to monitor the operating status of processes, the Advanced Application Monitoring directory was added to the Agent Collector service properties.
- As part of the capability to monitor the operating status of processes, the following files were added:
  - `jpcapp2`
  - `jpcappcvt`
  - `jpcappcvt.ini`
  - `jpcappcvt.ini.model`
- The version of the data model has changed from 7.6 to 7.8.
- The monitoring template alarm table version was changed from 9.00 to 09.10.

M.9 Changes in 09-00

(1) Changes in the manual (3020-3-R49(E))

- The name *solution set* has changed to *monitoring template*.
- The following OSs have been added as the OSs on which PFM - Agent for Platform operates and as OSs for programs that can be monitored:
  - Solaris 10 (x86)
  - Solaris 10 (x64)
  - AIX V6.1
  - Linux 5 Advanced Platform (x86)
  - Linux 5 (x86)
  - Linux 5 Advanced Platform (AMD64 & EM64T)
  - Linux 5 (AMD64 & EM64T)
  - Linux 5 Advanced Platform (IPF)
  - Linux 5 (IPF)

- The monitoring items in a virtual environment have been increased. Accordingly, the following field has been added to the PD and PD_APP records:

  Virtual Env ID

  In addition, the following properties have been added to the list of properties for the Agent Collector service:

  ALL WPAR Collection for Process, ADD AN APPLICATION MONITORING SETTING FOR VIRTUAL ENVIRONMENT
- The alarm table version for the monitoring template has been changed from 8.11 to 09.00. Accordingly, the following alarm tables have been added:
  - PFM UNIX Template Alarms [CPU]09.00
  - PFM UNIX Template Alarms [MEM]09.00
  - PFM UNIX Template Alarms [DSK]09.00
  - PFM UNIX Template Alarms [NET]09.00
  - PFM UNIX Template Alarms [PS]09.00

  The following alarms have also been added:
  - CPU Per Processor(K)
  - CPU Per Processor(U)
  - Alloc Mem Mbytes
  - File System Free(L)
  - File System Free(R)
  - Disk Busy %
  - Disk Queue
  - Network Rcvd/sec
  - Process End
  - Process Alive
- Linkage with JP1/IM has been enhanced. Accordingly, the following directory has been added to the list of properties for the Agent Collector service:

  JP1 Event Configurations
- The following messages have been added:

  KAVF10013-W, KAVF10700-W to KAVF10701-W
- The data model version has changed from 7.2 to 7.6. Accordingly, the format of the following fields has been changed from ulong to double:

  Program Summary (PD_PGM) record
  - Total I/O Kbytes (TOTAL_IO_KBYTES)

  Terminal Summary (PD_TERM) record
  - Total I/O Kbytes (TOTAL_IO_KBYTES)

  User Summary (PD_USER) record
  - Total I/O Kbytes (TOTAL_IO_KBYTES)

  Workgroup Summary (PI_WGRP) record
  - Total I/O Kbytes (TOTAL_IO_KBYTES)

- With the addition of new-format commands compatible with 08-11 or earlier commands, the commands for 09-00 or later is now indicated as follows:
  command-for-09-00-or-later（*command-for-08-11-or-earlier*）

N. Reference Material for This Manual

This appendix provides reference information, including various conventions, for this manual.

N.1 Related publications

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

For JP1/Performance Management publications:

- *JP1 Version 12 Performance Management: Getting Started* (3021-3-D75(E))
- *JP1 Version 12 JP1/Performance Management Planning and Configuration Guide* (3021-3-D76(E))
- *JP1 Version 12 JP1/Performance Management User's Guide* (3021-3-D77(E))
- *JP1 Version 12 JP1/Performance Management Reference* (3021-3-D78(E))

For JP1 publications:

- *Job Management Partner 1/Software Distribution Client Description and User's Guide* (3020-3-S85(E)), for UNIX systems
- *Job Management Partner 1/Software Distribution SubManager Description and Administrator's Guide* (3020-3-L42(E))
- *Job Management Partner 1/Software Distribution Manager* (3000-3-841(E))

N.2 Conventions: Abbreviations for product names

This manual uses the following abbreviations for product names:

Abbreviation		Full name or meaning
AIX		AIX V7.1
		AIX V7.2
HP-UX	HP-UX 11i	HP-UX 11i V3 (IPF)
JP1/Base		JP1/Base
JP1/IM	JP1/IM - Manager	JP1/Integrated Management - Manager
	JP1/IM - View	JP1/Integrated Management - View
JP1/ITSLM (10-50 or earlier)	JP1/ITSLM - Manager	Job Management Partner 1/IT Service Level Management - Manager
	JP1/ITSLM - UR	Job Management Partner 1/IT Service Level Management - User Response
JP1/SLM	JP1/SLM - Manager	JP1/Service Level Management - Manager

Abbreviation			Full name or meaning
JP1/SLM	JP1/SLM - UR		JP1/Service Level Management - User Response
JP1/Software Distribution			Job Management Partner 1/Software Distribution Client
			Job Management Partner 1/Software Distribution Manager
			Job Management Partner 1/Software Distribution SubManager
Linux	CentOS	CentOS 6 (x64)	CentOS 6.1 (x64) and later
		CentOS 7	CentOS 7.1 and later
		CentOS 8	CentOS 8.1 and later
	Linux 6 (x64)		Red Hat(R) Enterprise Linux(R) Server 6.1 (64-bit x86_64) and later
	Linux 7		Red Hat(R) Enterprise Linux(R) Server 7.1 and later
	Linux 8		Red Hat(R) Enterprise Linux(R) Server 8.1 and later
	Oracle Linux	Oracle Linux 6 (x64)	Oracle Linux(R) Operating System 6.1 (x64) and later
		Oracle Linux 7	Oracle Linux(R) Operating System 7.1 and later
		Oracle Linux 8	Oracle Linux(R) Operating System 8.1 and later
	SUSE Linux	SUSE Linux 12	SUSE Linux(R) Enterprise Server 12
		SUSE Linux 15	SUSE Linux(R) Enterprise Server 15
NNM	HP NNM		HP Network Node Manager Software version 6 or earlier
			HP Network Node Manager Starter Edition Software version 7.5 or earlier
Performance Management			JP1/Performance Management

Abbreviation			Full name or meaning
PFM - Agent	PFM - Agent for JP1/AJS#	PFM - Agent for JP1/AJS3	JP1/Performance Management - Agent Option for JP1/AJS3
	PFM - Agent for Cosminexus#		JP1/Performance Management - Agent Option for uCosminexus Application Server
	PFM - Agent for DB2		JP1/Performance Management - Agent Option for IBM DB2
	PFM - Agent for Domino		JP1/Performance Management - Agent Option for IBM Lotus Domino
	PFM - Agent for Enterprise Applications		JP1/Performance Management - Agent Option for Enterprise Applications
	PFM - Agent for Exchange Server#		JP1/Performance Management - Agent Option for Microsoft(R) Exchange Server
	PFM - Agent for HiRDB#		JP1/Performance Management - Agent Option for HiRDB
	PFM - Agent for WebSphere MQ#		JP1/Performance Management - Agent Option for IBM WebSphere MQ
	PFM - Agent for IIS#		JP1/Performance Management - Agent Option for Microsoft(R) Internet Information Server
	PFM - Agent for Microsoft SQL Server		JP1/Performance Management - Agent Option for Microsoft(R) SQL Server
	PFM - Agent for OpenTP1#		JP1/Performance Management - Agent Option for OpenTP1
	PFM - Agent for Oracle		JP1/Performance Management - Agent Option for Oracle
	PFM - Agent for Platform	PFM - Agent for Platform (UNIX)	JP1/Performance Management - Agent Option for Platform(for UNIX(R) systems)
		PFM - Agent for Platform (Windows)	JP1/Performance Management - Agent Option for Platform(for Windows(R) systems)

Abbreviation		Full name or meaning
PFM - Agent	PFM - Agent for Service Response	JP1/Performance Management - Agent Option for Service Response
	PFM - Agent for WebLogic Server#	JP1/Performance Management - Agent Option for Oracle(R) WebLogic Server
	PFM - Agent for WebSphere Application Server#	JP1/Performance Management - Agent Option for IBM WebSphere Application Server
PFM - Base		JP1/Performance Management - Base
PFM - Manager		JP1/Performance Management - Manager
PFM - RM	PFM - RM for Microsoft SQL Server	JP1/Performance Management - Remote Monitor for Microsoft(R) SQL Server
	PFM - RM for Oracle	JP1/Performance Management - Remote Monitor for Oracle
	PFM - RM for Platform	JP1/Performance Management - Remote Monitor for Platform
	PFM - RM for Virtual Machine	JP1/Performance Management - Remote Monitor for Virtual Machine
PFM - Web Console		JP1/Performance Management - Web Console
Solaris	Solaris 11	Solaris 11 (SPARC)

- PFM - Manager, PFM - Agent, PFM - Base, PFM - Web Console, and PFM - RM may be referred to collectively as *Performance Management*.
- HP-UX, Solaris, AIX, and Linux may be referred to collectively as UNIX.

#
This product works in a Japanese environment only.

N.3 Conventions: Acronyms

This manual also uses the following acronyms:

Acronym	Full name or meaning
API	Application Programming Interface
CSV	Comma Separated Values

Acronym	Full name or meaning
DHCP	Dynamic Host Configuration Protocol
DLPAR	Dynamic Logical Partitioning
DNS	Domain Name System
DR	Dynamic Reconfiguration
FQDN	Fully Qualified Domain Name
I/O	Input/Output
IP	Internet Protocol
IPF	Itanium Processor Family
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IVM	Integrated Virtualization Manager
I-VM	Integrity Virtual Machines
KVM	Kernel-based Virtual Machine
LAN	Local Area Network
LPAR	Logical Partitioning
NAPT	Network Address Port Translation
NAT	Network Address Translation
NFS	Network File System
NIS	Network Information Service
OS	Operating System
SCSI	Small Computer System Interface
SMT	Simultaneous MultiThreading
UEK	Unbreakable Enterprise Kernel
VM	Virtual Machine
vPars	Virtual Partitions
WPAR	Workload Partition

N.4 Conventions: Format of product names, service IDs, and service keys in this manual

In Performance Management 09-00 or later, the service ID and service key can be displayed in the product name when the product name display functionality is enabled.

Identifier	Product name display functionality	
	Disabled	Enabled
Service ID	US1*host-name*	*host-name*<UNIX>(Store)

Identifier	Product name display functionality	
	Disabled	Enabled
Service ID	UA1*host-name*	*host-name*<UNIX>
Service key	agtu	UNIX

This manual uses the format that is used when the product name display functionality is enabled.

Note that the product name display functionality can be enabled when both of the following conditions exist:

- The version of the prerequisite program (PFM - Manager or PFM - Base) on the same machine as PFM-Agent is 09-00 or later.
- The versions of PFM - Web Console and the PFM - Manager to which it connects are 09-00 or later.

N.5 Conventions: KB, MB, GB and TB

This manual uses the following conventions:

- 1 KB (kilobyte) is 1,024 bytes.
- 1 MB (megabyte) is $1{,}024^2$ bytes
- 1 GB (gigabyte) is $1{,}024^3$ bytes.
- 1 TB (terabyte) is $1{,}024^4$ bytes.

N.6 Conventions: Performance Management installation folders

In this manual, installation folders for the Windows version of Performance Management are referred to as *installation folders*, and installation directories for the UNIX version of Performance Management are referred to as *installation directories*.

The default installation folders for the Windows version of Performance Management are as follows:

Default installation folder for PFM - Base

system-drive`\Program Files (x86)\Hitachi\jp1pc`

In this manual, *installation-folder* refers to the PFM - Base installation folder.

Default installation folder for PFM - Manager

system-drive`\Program Files (x86)\Hitachi\jp1pc`

Default installation folder for PFM - Web Console

system-drive`\Program Files (x86)\Hitachi\jp1pcWebCon`

The default installation directories for the UNIX version of Performance Management are as follows:

Default installation directory for PFM - Base

`/opt/jp1pc`

Default installation directory for PFM - Manager

`/opt/jp1pc`

Default installation directory for PFM - Web Console

```
/opt/jp1pcwebcon
```

O. Glossary

action

The operation that is automatically executed by Performance Management when the monitored data reaches the threshold. The following types of actions are available:

- E-mail transmission
- Command execution
- SNMP trap issuance
- JP1 event issuance

Action Handler

One of the PFM - Manager or PFM - Base services. It is a service for executing an action.

administrative tools

Various types of commands and GUI functions that are used for checking service statuses and manipulating performance data. The following operations can be performed:

- Displaying service configuration and status
- Saving and restoring performance data
- Exporting of performance data to a text file
- Erasing of performance data

Agent

The PFM - Agent service that collects performance data.

Agent Collector

One of the PFM - Agent services. It is a service for collecting performance data and evaluating performance data based on a threshold value that is set in an alarm.

Agent Store

One of the PFM - Agent services. It is a service for storing performance data. The Agent Store service uses a database for recording performance data. Each PFM - Agent has its own Agent Store service.

alarm

Information that defines the action to be taken or the event message to be issued when the monitored data reaches the threshold.

alarm table

A table into which one or more alarms are collected and which defines the following types of information:

- Objects to be monitored (Process, TCP, WebService, or the like)
- Information to be monitored (CPU usage, number of bytes received per second, or the like)
- Monitoring condition (threshold)

binding

The operation of associating an alarm with an agent. Once binding occurs, when the performance data collected by the agent reaches the threshold defined in the alarm, the user can be notified.

cluster system

A single system configured from multiple linked server systems. There are two major types of cluster systems: an HA (High Availability) cluster system and a load-balancing cluster system.

In this manual, a cluster system means an HA cluster system.

Correlator

One of the PFM - Manager services. It is a service for controlling event delivery among services. When the alarm status being monitored by this service exceeds the threshold, this service sends an alarm event and an agent event to the Trap Generator service and PFM - Web Console.

data model

A general term for records and fields that each PFM - Agent has. Data models are managed according to version.

database ID

An ID that is assigned to each record in PFM - Agent that indicates the database for storing records. A database ID indicates the type of records to be stored in the database. The following database IDs are used:

- `PI`: Indicates a database for records of the `PI` record type.
- `PD`: Indicates a database for records of the `PD` record type.

drilldown report

A report that is associated with a report or report field. You use a drilldown report to display the detailed information or related information of a report.

executing node

Of the server systems in a cluster system, the node that is currently executing applications (node whose logical host is active).

failover

The process by which the standby node takes over processing if a failure occurs on the node that is executing applications in a cluster system.

field

Individual operating information included in records. Fields correspond to monitoring items in Performance Management. For example, monitoring items such as `CPU %` or `Page Faults/sec` in System Overview(PI) records correspond to fields.

HA cluster system

A cluster system designed to implement high availability by continuing operation even if one system fails. If a failure occurs on the server currently executing applications, a separate standby server takes over and continues the processing of applications. Accordingly, because application processing is not interrupted when a failure occurs, availability improves.

In this manual, a cluster system means an HA cluster system.

historical report

A report indicating the status of the monitoring target from the past to the present.

instance

In this manual, the term *instance* is used in the following ways:

- To indicate a recording format for records

  A record that is recorded in a single line is called a *single-instance record*; a record that is recorded in multiple lines is called a *multi-instance record*; and each line inside a record is called an *instance*.

- To indicate a PFM - Agent startup method

  When a single agent monitors the monitoring targets on the same host, it is called a *single-instance agent*. In contrast, when the monitoring targets support multiple instances, each agent can monitor each monitoring target instance. This is called a *multi-instance agent*. Each agent of a multi-instance agent is called an *instance*.

JP1/SLM

A product that helps maintain service levels by monitoring system performance from various perspectives, such as a business system's performance as experienced by a service user.

Monitoring of the status of operations is enhanced by linking to JP1/SLM.

lifetime

The duration in which the integrity of the performance data collected into each record can be guaranteed.

logical host

A logical server that provides the JP1 execution environment for operation in a cluster system. If a failure occurs on the executing node, the logical host is switched to the standby node. Each logical host has a unique IP address. At failover, the IP address is inherited by the standby node. Thus, when the physical server is failed over, clients can still access the logical host using the same IP address. To the clients, it appears that one server is operating continuously.

Master Manager

One of the PFM - Manager services. It is the main service of PFM - Manager.

Master Store

One of the PFM - Manager services. It is a service for managing the alarm events issued by each PFM - Agent. The Master Store service uses a database for holding event data.

monitoring template

Predefined alarms and reports provided in PFM - Agent. The monitoring template enables you to easily prepare for monitoring the operating status of PFM - Agent without writing any complicated definitions.

multi-instance record

A record that is recorded in multiple lines. Such records have a specific ODBC key field.

See *instance*.

Name Server

One of the PFM - Manager services. It is a service for managing the service configuration information inside a system.

non-interactive (commands)

A mode of executing commands in which user entries required for the execution of commands are provided by means of specification of options or by reading from a definition file rather than by prompting the user to enter responses.

Executing commands non-interactively reduces the burden on the user by helping to automate the installation of the operation monitoring system.

ODBC key fields

A primary key required for PFM - Manager or PFM - Base to use the record data stored in a Store database. Some ODBC key fields are common to all records, while others are unique to some records.

PD record type

See *Product Detail record type*.

performance data

Resource usage data collected from a monitoring target system.

Performance Management

A general term for a group of software programs necessary for monitoring and analyzing issues related to system performance. Performance Management consists of the following five program products:

- PFM - Manager
- PFM - Web Console
- PFM - Base
- PFM - Agent
- PFM - RM

PFM - Agent

One of the program products comprising Performance Management. PFM - Agent that is equivalent to a system-monitoring facility. Various types of PFM - Agent are available according to the applications, databases, and OSs to be monitored. PFM - Agent has the following functions:

- Monitoring of the performance of the monitoring target
- Collection and recording of data from the monitoring target

PFM - Base

One of the program products comprising Performance Management. PFM - Base provides the basic functions for Performance Management to monitor operations. It is a prerequisite program product for PFM - Agent. PFM - Base provides the following functionality:

- Management tools such as various types of commands
- Common functions needed for linking Performance Management to other systems

PFM - Manager

One of the program products comprising Performance Management. PFM - Manager is equivalent to a manager facility and has the following functions:

- Management of Performance Management program products
- Management of events

PFM - Manager name

A name that identifies the field stored in the Store database. This name is used for specifying a field in a command.

PFM - View name

The alias of a PFM - Manager name. The PFM - View name is more intuitive than the PFM - Manager name. For example, the PFM - View name of the PFM - Manager name `INPUT_RECORD_TYPE` is `Record Type`. The PFM - View names are useful when specifying fields on the GUI of PFM - Web Console.

PFM - Web Console

One of the program products comprising Performance Management. PFM - Web Console provides the web application server functions necessary for centrally monitoring the Performance Management system from a web browser. PFM - Web Console has the following functions:

- GUI display
- Integrated monitoring and management functions
- Report and alarm definition

physical host

An environment unique to each server in a cluster system. When a failover occurs, the environment of the physical host is not inherited by the other server.

PI record type

See *Product Interval record type*.

PL record type

See *Product Log record type*.

Product Detail record type

The record type that stores the performance data that indicates the system status at a given point in time, such as detailed information on the processes that are currently active. Use the `PD` record type when you want to determine the system statuses at a given point in time, such as those listed below:

- System's operating status
- Capacity of the file system currently being used

Product Interval record type

The record type that stores the performance data for a specific duration (interval), such as the number of active processes during every minute. Use the `PI` record type when you want to analyze the changes and trends in the system status over time, such as those listed below:

- Trend in the number of system calls that occurred within a specific time span

Product Log record type

The record type that stores the log information on applications that are being executed or databases being used on UNIX.

real-time report

A report indicating the current status of the monitoring target.

record

A collection of operating information that is classified according to the purpose. For example, System Overview(PI) records are a collection of pieces of operating information that are used to provide a system overview, such as CPU usage rates or the size of unused physical memory. Monitoring agents collect operating information by record. The records that can be collected differ according to the agent program.

report

A report defines the information to be used when graphically displaying the performance data collected by PFM - Agent. A report mainly defines the following types of information:

- Records to be displayed in a report
- Performance data display items
- Performance data display format (table, graph, or the like)

service ID

A unique ID assigned to the service of a Performance Management program. For example, to use a command to check the system configuration of Performance Management or to back up the performance data of individual agents, you specify the service ID of the Performance Management program and execute the command.

The format of the service ID differs depending on whether the product name display functionality is enabled. For details about the service ID format, see the chapter explaining the Performance Management functions in the *JP1/Performance Management Planning and Configuration Guide*.

single-instance record

A record that is recorded in a single line. Such records do not have a specific ODBC key field.

See *instance*.

stand-alone mode

The mode in which PFM - Agent is running alone. Even when the Master Manager service and Name Server service of PFM - Manager cannot be started because of an error, you can start PFM - Agent alone and collect performance data.

standby node

Of the server systems in a cluster system, a node that is waiting to take over applications if the executing node fails.

status management function

A function for managing the statuses of all services that run on PFM - Manager and PFM - Agent. The status management function enables the system administrator to correctly determine the service startup and stop statuses at each host, making it possible to take the appropriate error recovery measures.

Store database

The database that stores the performance data collected by the Agent Collector service.

Trap Generator

One of the PFM - Manager services. It is a service that issues SNMP traps.

user defined record

A user-specified record for storing performance data that is not provided by default. When a user-defined record is specified, the collected performance data can be monitored from PFM - Web Console. The following types of user defined records are available:

- Application Summary (`PD_APP`) record
- User Data Detail (`PD_UPD`) record
- User Data Detail - Extended (`PD_UPDB`) record
- User Data Interval (`PI_UPI`) record
- User Data Interval - Extended (`PI_UPIB`) record
- User Data Interval - Expanded *n* (`PI_XUI`*n*) record (where *n* is a number in the range from 1 to 5)
- Workgroup Summary (`PI_WGRP`) record

workgroup

The unit in which PFM - Agent for Platform monitors the processes being executed. A workgroup can be specified in the following units:

- UNIX user
- UNIX group
- Programs executed by a process

Index

A

abbreviations for products 619
acronyms 622
action 22, 626
Action Handler 626
action log
  format for saving file of 598
  setting up 140
  types of events output to 598
action log data
  format of output 599
  outputting 598
  settings for outputting 603
actual host name 124
administrative tool 626
agent 626
Agent Collector 626
Agent Collector service property 578
Agent Store 626
Agent Store service property 575
AIX environment, monitoring processor performance based on results output by sar command in 82
alarm 22, 626
  Alloc Mem Mbytes 273
  Application Status 290
  CPU Per Processor(K) 269
  CPU Per Processor(U) 271
  Disk Busy % 279
  Disk Queue 281
  Disk Service Time 262
  File System Free(L) 275
  File System Free(R) 277
  format of explanation 259
  I/O Wait Time 263
  Kernel CPU 264
  Network Rcvd/sec 283
  Pagescans 265
  Process Alive 286
  Process End 285
  Process Existence 288
  Run Queue 266
  Swap Outs 267
  User CPU 268
alarms, list of 260
alarm table 22, 626
alias name 124
Alloc Mem Mbytes alarm 273
application definition, converting 508
application operating status
  checking or changing user-defined records for collecting 223
  deleting user-defined records for collecting 224
  example of using alarm when collecting information about 224
  settings for collecting information about 220
  user-defined record settings for collecting 220
Application Process Count report 300
Application Process Status report 301
Application Status alarm 290
Application Status report 299
Avg Service Time Status - Top 10 Devices report 303
Avg Service Time - Top 10 Devices report 302

B

backup 149
baseline, determining 25
binding 22, 627
browsing manual 152
  settings for 151
  settings for, in Web browser 151

C

cluster system 627
cluster system
  installation and setup in 166
  uninstallation and unsetup in 177
command
  description format of 507
  jpcuser 245
  specification format of 507
command syntax descriptions, notations used in 507
common message log 552, 553
communication in IPv4 and IPv6 environments 607
Console Messages report 304
conventions
  abbreviations for products 619
  acronyms 622
  diagrams 7

fonts and symbols 7
KB, MB, GB and TB 624
mathematical expressions 8
version numbers 8
Correlator 627
CPU Per Processor(K) alarm 269
CPU Per Processor(U) alarm 271
CPU - Per Processor Detail (PI_CPUP) record 406
CPU Per Processor Status report 305
CPU Per Processor Usage report 306
CPU Status (Multi-Agent) report 309
CPU Status report 307
CPU Trend (Multi-Agent) report 311
CPU Trend report 310
CPU Usage Summary report 313
CPU Usage - Top 10 Processes report 312

D

database ID 627
data collection in event of failure, preparations for 127
data model 21, 359, 627
data types, list of 366
Device Detail (PI_DEVD) record 411
Device Detail report 314
Device Summary (PI_DEVS) record 415
Device Usage Status (Multi-Agent) report 316
Device Usage Status report 315
diagram conventions 7
directories, list of 592
Disk Busy % alarm 279
disk information, notes regarding collection of 34
disk monitoring
definition examples 116
examples 101
Disk Queue alarm 281
Disk Service Time alarm 262
disk space requirement, estimating 568
DLPAR functionality 372
DR functionality 372
drilldown report 627
field level 292
report level 292

E

Edit Templates menu 203
effects of PFM - Manager stopping, and corrective action 165
error detection 565
error handling procedure 549
error information
to be collected for troubleshooting 558
event file, setting up 158
Event Monitor window 217
Example of collecting user-specific performance data (for a logical host environment) 243
executing node 627
exporting and importing
logical host environment definition file in cluster system 182

F

failover 627
when error occurs on PFM - Agent host 164
field 21, 292, 627
added only when record is recorded in Store database 370
output when data stored in Store database is exported 371
value 367
files, list of 592
File System Detail - Local (PD_FSL) record 419
File System Detail - Remote (PD_FSR) record 423
File System Free(L) alarm 275
File System Free(R) alarm 277
firewall passage direction 574
folder to check for troubleshooting 553
font conventions 7
format
of jpcuser command 245
of user-created data file 247
Free Space Mbytes - Top 10 Local File Systems report 317

G

GB meaning 624

H

HA cluster system 627
historical report 20, 628
Hitachi Compute Blade logical partitioning feature, data that can be collected with virtualization systems that use 44

HP Integrity Virtual Machines (I-VM), data that can be collected on virtualized systems that use 40

I

I/O Activity - Top 10 Processes report 318
I/O Overview report 319
I/O Wait Time alarm 263
identifier, list of 572
information, on terminated processes 375
installation
  OS user permission needed for 126
  preparations required before 124
installation procedure for PFM - Agent for Platform 130
installation procedure in cluster system 170
instance 628
IP address, setting up 124
IPv4 and IPv6 environments, communication in 607
items to be checked before installation in cluster system 168

J

JP1/ITSLM
  linking to 606
JP1/SLM 628
jpcappcvt command 508
jpcconf db define command 144
jpcsto.ini file 145
jpcuser command, format of 245

K

KB meaning 624
Kernel CPU alarm 264
kernel parameter 569

L

LANG environment variable, setting 135
lifetime 628
Linux container, data that can be collected on a virtualized system based on 48
Local File System Detail report 320
log file
  changing size of 139
  to check for troubleshooting 553
Logged Messages (PL_MESS) record 427
logical host 628
logical host environment definition file
  exporting and importing in cluster system 182
log information
  collecting 155
  common message log 552
  notes on collecting 159
  operation status log 552
  overview of collecting 156
  setup for collecting 158
  system log 552
  to be collected for troubleshooting 552
  trace log 552

M

Major Page Faults - Top 10 Processes report 321
Master Manager 628
Master Store 628
mathematical expressions, conventions 8
MB meaning 624
memory information, notes regarding collection of 33
memory monitoring
  definition examples 114
  examples 98
Memory Paging (8.4) report 323
Memory Paging report 322
Memory Paging Status (8.4) report 325
Memory Paging Status (Multi-Agent) (8.4) report 327
Memory Paging Status (Multi-Agent) report 326
Memory Paging Status report 324
memory requirement, estimating 568
message 510, 519
  format of 511
  format of explanation 512
  format of output 511
  output destination of 513
Micro-Partitioning function 374
monitoring conditions for processes, prerequisite for specifying 4,096-byte 127
Monitoring Process Detail report 328
monitoring target, distinguishing letter case of 215
monitoring template 22, 94, 258, 628
  overview of 94
multi-instance record 628

N

Name Server 629
network environment setup 124

network information, notes regarding collection of 34
Network Interface Detail (PI_NIND) record 429
Network Interface Detail report 329
Network Interface Summary (PI_NINS) record 435
Network Interface Summary report 330, 331
network monitoring
definition examples 118
examples 104
Network Overview report 332
Network Rcvd/sec alarm 283
network setup 139
Network Status (Multi-Agent) report 334
Network Status report 333
NFS Activity Overview report 335
NFS Load Trend report 336
NFS Usage Status (Multi-Agent) report 338
NFS Usage Status report 337
non-interactive (commands) 629
note
collecting same records from multiple hosts in same machine (only in AIX) 184
continuity of performance data before and after failover 184
notes
collecting information about operating status of processes 219
collecting user-specific performance data 183
operating PFM - Agent for Platform in cluster system 183
specifying settings for collecting information about operating status of applications and processes 183

O

ODBC key field 629
ODBC key fields, list of 363
operating status information
of application 28
of process 28
operating status of processes
command-based settings for collecting information about 212
differences in functionality from previous versions 193
example of responding to alarm while collecting information about 217
overview of collecting information about 193
records used to collect information about 194
settings for collecting information about 193
settings in Agents tree for collecting information about 195
settings in Services tree for collecting information about 206
operation in cluster system 161
operation information
to be collected for troubleshooting 557
operation status log 552
OS log information to be collected for troubleshooting 554

P

Pagescans alarm 265
Paging Trend (Multi-Agent) (8.4) report 340
Paging Trend (Multi-Agent) report 339
PD 438
PD_APP 395
PD_APP2 403
PD_APPD 379
PD_APS 388
PD_APSI 381
PD_FSL 419
PD_FSR 423
PD_PDI 447
PD_PDS 454
PD_PGM 457
PD_TERM 475
PD_UPD 478
PD_UPDB 480
PD_USER 499
PD record type 21, 629
performance data 629
changing storage destination of 139
changing storage location of 144
collection methods 24
from multiple hosts on which PFM products are not installed, examples of collecting 108
management methods 24
performance data information, to be collected for troubleshooting 558
performance information
about resources that are logically partitioned 374
notes on, when modifying system resources 372
Performance Management 629
error handling procedure 550
system error recovery 566
Performance Management information, to be collected for troubleshooting 555

Performance Management programs
multiple, installing on same host 128
multiple, setting up on same host 128
performance monitoring
definition examples 111
examples 96
objective of 24
overview of 24
system resources in need of 25
performance monitoring on virtualized systems
objectives of 29
system resources most in need of 30
periodically executing user commands, functionality for 232
PFM - Agent 629
PFM - Agent for Platform
changing operation of 144
changing system configuration of 143
configuration in cluster system 162
features of 20
flow of installing in cluster system 166
flow of setting up in cluster system 166
flow of uninstalling 177
flow of unsetting up in cluster system 177
installation procedure for 130
installation workflow 122
installing 121
installing in cluster system 166
overview of 19
points to note before installing 128
points to note before uninstalling 141
port numbers of 574
registering, in PFM - Manager and PFM - Web Console 137
setting up 121
setting up in cluster system 166
setting up PFM - Manager for connection destination of 140
setup file 138
setup workflow 122
uninstallation procedure for 142
uninstalling in cluster system 177
unsetting up in cluster system 177
PFM - Agent for Platform in virtualized system
examples of using (determining whether additional physical resources are needed) 32
examples of using (determining whether to change virtualized system settings) 31
installing 30
notes regarding usage of 32
setting up 30
PFM - Agent for Platform operation method
changing in cluster system 182
PFM - Base 629
PFM - Manager 630
releasing connection to 142
PFM - Manager name 630
PFM - View name 630
PFM - Web Console 630
setting up 159, 192
physical host 630
PI 460
PI_CPUP 406
PI_DEVD 411
PI_DEVS 415
PI_NIND 429
PI_NINS 435
PI_UPI 483
PI_UPIB 495
PI_WGRP 502
PI_XUI1 486
PI_XUI2 491
PI_XUI3 492
PI_XUI4 493
PI_XUI5 494
PI record type 21, 630
PL_MESS 427
PL record type 21, 630
port number
of PFM - Agent for Platform 574
setting up 125
port numbers, list of 574
PowerVM, data that can be collected with virtualized systems that use 38
prerequisite program 126
Process Alive alarm 286
Process Detail (PD) record 438
Process Detail Interval (PD_PDI) record 447
Process Detail report 341
Process End alarm 285
processes, list of 573
Process Existence alarm 288
process information, notes regarding collection of 34
processing at failover 164
process monitoring

definition examples 119
examples 105
Process Monitoring Settings > Application Definition Templates window 203
Process Monitoring Settings > Create a New Application Definition > Add from monitoring-target-type window 200
Process Monitoring Settings > Create a New Application Definition > Input Using Application Definition Template window 205
Process Monitoring Settings > Create a New Application Definition Template window 204
Process Monitoring Settings > Create a New Application Definition window 196
Process Monitoring Settings window 196
processor information, notes regarding collection of 33
processor monitoring
definition examples 111
examples 96
processor performance based on results output by sar command in AIX environment, monitoring 82
Process Overview report 343
Process Summary (PD_PDS) record 454
Process Summary Status report 344
Process Trend report 345
Product Detail record type 630
Product Interval record type 630
Product Log record type 631
program, order in which to install 130
Program Summary (PD_PGM) record 457
property 575
Agent Collector service 578
Agent Store service 575

R

real-time report 20, 631
record 21, 292, 358, 631
Application Process Detail (PD_APPD) 379
Application Process Interval (PD_APSI) 381
Application Process Overview (PD_APS) 388
Application Summary (PD_APP) 395
Application Summary Extension (PD_APP2) 403
CPU - Per Processor Detail (PI_CPUP) 406
Device Detail (PI_DEVD) 411
Device Summary (PI_DEVS) 415
File System Detail - Local (PD_FSL) 419
File System Detail - Remote (PD_FSR) 423
format of explanation 360
generation results when data cannot be collected374
Logged Messages (PL_MESS) 427
Network Interface Detail (PI_NIND) 429
Network Interface Summary (PI_NINS) 435
notes on 372
PD record type 21
PI record type 21
PL record type 21
Process Detail (PD) 438
Process Detail Interval (PD_PDI) 447
Process Summary (PD_PDS) 454
Program Summary (PD_PGM) 457
single-instance 631
System Summary Overview (PI) 460
Terminal Summary (PD_TERM) 475
User Data Detail (PD_UPD) 478
User Data Detail - Extended (PD_UPDB) 480
User Data Interval (PI_UPI) 483
User Data Interval - Expanded 1 (PI_XUI1) 486
User Data Interval - Expanded 2 (PI_XUI2) 491
User Data Interval - Expanded 3 (PI_XUI3) 492
User Data Interval - Expanded 4 (PI_XUI4) 493
User Data Interval - Expanded 5 (PI_XUI5) 494
User Data Interval - Extended (PI_UPIB) 495
User Summary (PD_USER) 499
Workgroup Summary (PI_WGRP) 502
records, list of 377
Remote File System Detail report 346
report 20, 631
Application Process Count 300
Application Process Status 301
Application Status 299
Avg Service Time Status - Top 10 Devices 303
Avg Service Time - Top 10 Devices 302
Console Messages 304
CPU Per Processor Status 305
CPU Per Processor Usage 306
CPU Status 307
CPU Status (Multi-Agent) 309
CPU Trend 310
CPU Trend (Multi-Agent) 311
CPU Usage Summary 313
CPU Usage - Top 10 Processes 312
Device Detail 314
Device Usage Status 315
Device Usage Status (Multi-Agent) 316
format of explanation 292

Free Space Mbytes - Top 10 Local File Systems 317
historical 20
I/O Activity - Top 10 Processes 318
I/O Overview 319
Local File System Detail 320
Major Page Faults - Top 10 Processes 321
Memory Paging 322
Memory Paging (8.4) 323
Memory Paging Status 324
Memory Paging Status (8.4) 325
Memory Paging Status (Multi-Agent) 326
Memory Paging Status (Multi-Agent) (8.4) 327
Monitoring Process Detail 328
Network Interface Detail 329
Network Interface Summary 330, 331
Network Overview 332
Network Status 333
Network Status (Multi-Agent) 334
NFS Activity Overview 335
NFS Load Trend 336
NFS Usage Status 337
NFS Usage Status (Multi-Agent) 338
organization of directory 293
Paging Trend (Multi-Agent) 339
Paging Trend (Multi-Agent) (8.4) 340
Process Detail 341
Process Overview 343
Process Summary Status 344
Process Trend 345
real-time 20
Remote File System Detail 346
Space Usage - Top 10 Local File Systems 347
Space Usage - Top 10 Remote File Systems 348
System Overview 349, 351
System Overview(8.4) 353
System Utilization Status 355
Workload Status 356
Workload Status (Multi-Agent) 357
reports, list of 296
required OS 124
restoration 149, 150
Run Queue alarm 266

S

service ID 631
setup 135
setup command 138
setup procedure in cluster system 170
single-instance record 631
Space Usage - Top 10 Local File Systems report 347
Space Usage - Top 10 Remote File Systems report 348
stand-alone mode 631
standby node 631
status management function 565, 632
Store database 21, 632
updating version to 2.0 146
summarization rules 364
Swap Outs alarm 267
symbol conventions 7
syslog, list of messages output to 518
system log 552
System Overview (8.4) report 353
System Overview report 349, 351
system requirement, estimating 568
System Summary Overview (PI) record 460
System Utilization Status report 355

T

TB meaning 624
Terminal Summary (PD_TERM) record 475
trace log 552, 553
Trap Generator 632
troubleshooting 551
other information to be collected for 563
procedures for collecting UNIX-related data for 564
UNIX--related data to be collected for 554

U

uninstallation 141
uninstallation procedure in cluster system 181
unsetup procedure in cluster system 178
upgrading
notes on 595
procedure for 595
used port information, example of collecting 106
user command 231
User CPU alarm 268
user-created data 231
checking debug log to determine if correct 251
user-created data file, format of 247
user data file 231
User Data Interval - Expanded 1 (PI_XUI1) record 486
User Data Interval - Expanded 2 (PI_XUI2) record 491

User Data Interval - Expanded 3 (PI_XUI3) record 492
User Data Interval - Expanded 4 (PI_XUI4) record 493
User Data Interval - Expanded 5 (PI_XUI5) record 494
user defined record 632
user-defined record 26
collection 185
overview of collection of 26
user-specific performance data 28
example of collecting for physical host environment 241
overview of functionality for collecting 231
settings for collecting 231, 237
User Summary (PD_USER) record 499

V

version compatibility 597
version number conventions 8
version upgrade, notes on 128
virtualized system
advantages of 29
functionality and collected performance data 34
installing PFM - Agent for Platform in 30
notes regarding collection of disk information on 34
notes regarding collection of memory information on 33
notes regarding collection of network information on 34
notes regarding collection of process information on 34
notes regarding collection of processor information on 33
objectives of performance monitoring on 29
overview of operations on 29
range of data collected for each 52
setting up PFM - Agent for Platform in 30
system resources most in need of performance monitoring on 30
VMware/KVM, data that can be collected with virtualized systems that use 42
vPars functionality 372

W

workgroup 632
workgroup file
specification example of 190
specifying 187
workgroup information 26
examples of alarm operation when collecting 192
settings for collecting 186
setting workgroup file for collecting 186
Workgroup Summary (PI_WGRP) record 502
Workload Status (Multi-Agent) report 357
Workload Status report 356
WPAR, data that can be collected with virtualized systems that use 45

Z

zone, data that can be collected with virtualized systems that use 35

Hitachi, Ltd.

6-6, Marunouchi 1-chome, Chiyoda-ku, Tokyo, 100–8280 Japan