HITACHI
Inspire the Next

JP1 Version 12

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

3021-3-D85-10(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 Oracle, see the *Release Notes*.

JP1/Performance Management - Manager (For Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019):

P-2A2C-AACL JP1/Performance Management - Manager version 12-10

The above product includes the following:

P-CC2A2C-5ACL JP1/Performance Management - Manager version 12-00

P-CC2A2C-5RCL JP1/Performance Management - Web Console version 12-10

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

P-812C-AACL JP1/Performance Management - Manager version 12-10

The above product includes the following:

P-CC812C-5ACL JP1/Performance Management - Manager version 12-00

P-CC812C-5RCL JP1/Performance Management - Web Console version 12-10

JP1/Performance Management - Agent Option for Oracle (For Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019):

P-2A2C-ADCL JP1/Performance Management - Agent Option for Oracle version 12-10

The above product includes the following:

P-CC2A2C-AJCL JP1/Performance Management - Base version 12-00

P-CC2A2C-FDCL JP1/Performance Management - Agent Option for Oracle version 12-10

JP1/Performance Management - Agent Option for Oracle (For Linux 6 (x64), Linux 7, Oracle Linux 6 (x64), Oracle Linux 7):

P-812C-ADCL JP1/Performance Management - Agent Option for Oracle version 12-10

The above product includes the following:

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

P-CC812C-FDCL JP1/Performance Management - Agent Option for Oracle version 12-10

JP1/Performance Management - Agent Option for Oracle (For AIX 7.1, AIX 7.2):

P-1M2C-ADCL JP1/Performance Management - Agent Option for Oracle version 12-10

The above product includes the following:

P-CC1M2C-AJCL JP1/Performance Management - Base version 12-00

P-CC1M2C-FDCL JP1/Performance Management - Agent Option for Oracle version 12-10

JP1/Performance Management - Agent Option for Oracle (For Solaris 11 (SPARC)):

P-9D2C-ADCL JP1/Performance Management - Agent Option for Oracle version 12-10

The above product includes the following:

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

P-CC9D2C-FDCL JP1/Performance Management - Agent Option for Oracle version 12-10

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

P-1J2C-ADCL JP1/Performance Management - Agent Option for Oracle version 12-10

The above product includes the following:

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

P-CC1J2C-FDCL JP1/Performance Management - Agent Option for Oracle version 12-10

These products include the parts developed under licenses from other companies.

■ Export restrictions

If you export this product, please check all restrictions (for example, Japan's Foreign Exchange and Foreign Trade Law, and USA export control laws and regulations), and carry out all required procedures.

If you require more information or clarification, please contact your Hitachi sales representative.

■ Trademarks

HITACHI, Job Management Partner 1, and JP1 are either trademarks or registered trademarks of Hitachi, Ltd. in Japan and other countries.

AMD, AMD Opteron, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.

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

IBM, DB2 are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

IBM, DB2 Universal Database are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

IBM, Lotus are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

IBM, WebSphere are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

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 in the United States and other countries.

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Red Hat, and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries. Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.

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

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc., in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

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

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

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

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

■ Microsoft product name abbreviations

This manual uses the following abbreviations for Microsoft product names.

Abbreviation		Full name or meaning
Internet Explorer		Microsoft(R) Internet Explorer(R)
		Windows(R) Internet Explorer(R)
WSFC		Microsoft(R) Windows Server(R) Failover Cluster
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 is sometimes used generically, referring to Windows Server 2012, Windows Server 2016 and Windows Server 2019.

■ 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.

■ Issued

Jan. 2020: 3021-3-D85-10(E)

■ Copyright

Copyright (C) 2019, 2020, Hitachi, Ltd.

Copyright (C) 2019, 2020, Hitachi Solutions, Ltd.

Summary of amendments

The following table lists changes in this manual (3021-3-D85-10(E)) and product changes related to this manual.

Changes	Location
The multitenant container database (CDB) configuration now allows all pluggable databases to be monitored from the root container (`CDB$ROOT`) to which they belong. In response to this, the following reports were added so that PDB information inside the CDB can be monitored: • CDB Audit Size Status • CDB Data File Size Status • CDB Diagnostics Size Status • CDB Status • CDB Tablespace Status	*1.1.7*, *Chapter 5*
Changes were made to the procedure for registering PFM - Agent Option for Oracle.	*2.1.4(1)*, *3.1.4(2)*, *4.3.4(1)*, *4.4.4(1)*
The version of the data model has been changed from 9.1 to 9.2, and the version of the alarm table has been changed from 12.00 to 12.10.	*Chapter 5*, *Appendix J*
The following report was added to monitor the fast recovery area: • Fast Recovery Area Status	*Chapter 5*
The following records were added to collect PDB information from the CDB: • CDB Status (PD_CDCS) • CDB Tablespace (PD_CDTS)	*Chapter 6*
A field for monitoring free space available for the disk group was added to the following record: • ASM Disk Group Interval (PI_PIDG)	*Chapter 6*
A field for monitoring the fast recovery area was added to the following record: • Collection Instance 2 (PD_PCI)	*Chapter 6*

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

Preface

This manual describes the functionality and records of JP1/Performance Management - Agent Option for Oracle.

■ Intended readers

This manual describes JP1/Performance Management. The manual is intended for the following readers:

- Users who wish to design or construct an operation monitoring system.
- Users who wish to define conditions for collecting performance data.
- Users who wish to define reports and alarms.
- Users who wish to use collected performance data to monitor a system.
- Users who wish to consider or take actions for a system based on monitoring results.

Readers are assumed to be familiar with Oracle and the operation of the system being monitored, and to have a knowledge of the OS.

For details about how to design and run systems that use JP1/Performance Management, also see the following manuals:

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

■ Organization of this manual

This manual consists of the following parts, and is a common reference for the following supported OSs: Windows, HP-UX, Solaris, Linux and AIX. Any platform-dependent differences are noted separately in the manual.

Part *1. Overview*

This part provides an overview of JP1/Performance Management - Agent Option for Oracle.

Part *2. Configuration and Operations*

Part 2 describes how to install and set up JP1/Performance Management - Agent Option for Oracle, and how to run the program in a cluster system.

Part *3. Reference*

This part describes the monitoring template, records, and messages of JP1/Performance Management - Agent Option for Oracle.

Part *4. Troubleshooting*

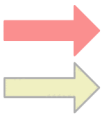
This part describes the actions to be taken for errors that might occur during operation of JP1/Performance Management - Agent Option for Oracle.

■ Conventions: Diagrams

This manual uses the following conventions in diagrams:

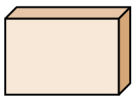
● Computer

● Data flow

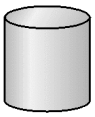


● Processing flow

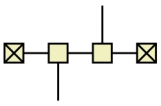
● Program



● File



● Network



● Server

● Failure

● I/O operation

■ Conventions: Fonts and symbols

Font and symbol conventions are classified as:

- General font conventions
- Conventions in syntax explanations

These conventions are described below.

General font conventions

The following table lists the general font conventions:

Font	Convention
Bold	Bold type indicates text on a window, other than the window title. Such text includes menus, menu options, buttons, radio box options, or explanatory labels. For example, bold is used in sentences such as the following: • From the **File** menu, choose **Open**. • Click the **Cancel** button. • In the **Enter name** entry box, type your name.
Italics	Italics are used to indicate a placeholder for some actual text provided by the user or system. Italics are also used for emphasis. For example: • Write the command as follows: `copy` *source-file target-file* • Do *not* delete the configuration file.
`Code font`	A code font indicates 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.`

Examples of coding and messages appear as follows (although there may be some exceptions, such as when coding is included in a diagram):

```
MakeDatabase
...
StoreDatabase temp DB32
```

In examples of coding, an ellipsis (...) indicates that one or more lines of coding are not shown for purposes of brevity.

Conventions in syntax explanations

Syntax definitions appear as follows:

Store**D**atabase [A|B] {C|D|E} (*database-name* ...)

The following table lists the conventions used in syntax explanations.

Example font or symbol	Convention
StoreDatabase	The user should enter code-font characters exactly as shown.
database-name	In actual commands the user must replace the italics by suitable characters.
SD	Bold code-font characters indicate an abbreviation for a command.
A	The underlined characters are the system default when you omit all the items enclosed in brackets. Example: [A|B] indicates that the system uses A if you do not specify either A or B.
|	Only one of the options separated by a vertical bar can be used at one time. Example: A|B|C indicates A, or B, or C.
{ }	One of the items enclosed in braces and separated by a vertical bar must be specified. Example: {C|D|E} indicates that one of the items from C, or D, or E must be specified.
[ ]	The item or items enclosed in brackets are optional. Example: [A] indicates the specification of A or nothing. [B|C] indicates the specification of B or C, or nothing.
...	The item or items preceding the ellipsis (...) can be repeated. To specify multiple items, use a one-byte space to delimit them. Example: A B ... indicates that B can be specified as many times as necessary after A.
( )	The items enclosed by the parentheses are in the range to which | or ... are applied.

■ Conventions: Mathematical expressions

The following table lists conventions used in mathematical expressions:

Symbol	Description
*	Multiplication sign

Symbol	Description
/	Division

■ 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 Oracle 19
1.1 Features of PFM - Agent for Oracle 20
1.1.1 Collecting Oracle performance data 20
1.1.2 Collecting performance data based on its characteristics 21
1.1.3 Saving performance data 21
1.1.4 Notifying users of problems in Oracle operation 21
1.1.5 Easy definition of alarms and reports 22
1.1.6 Operation with a cluster system 22
1.1.7 Operation with a multitenant container database (CDB) configuration 23
1.2 Overview of collection and management of performance data 29
1.3 Example of performance monitoring using PFM - Agent for Oracle 30
1.3.1 Purpose of performance monitoring 30
1.3.2 Determining a baseline 30
1.3.3 Search processing performance 31
1.3.4 Data update processing performance 32
1.3.5 Oracle instance operation monitoring 33
1.3.6 Disk monitoring 33
1.3.7 Monitoring wait events concerning REDO log file 34

Part 2: Configuration and Operations

2 Installation and setup (Windows) 35
2.1 Installation and setup 36
2.1.1 Installation and setup workflow 36
2.1.2 Preparation for installing and setting up PFM - Agent for Oracle 37
2.1.3 Installation procedure 43
2.1.4 Setting up PFM - Agent for Oracle 45
2.2 Setup cancellation and uninstallation 70
2.2.1 Cautionary notes on setup cancellation and uninstallation 70
2.2.2 Procedure for canceling setup 71
2.2.3 Procedure for uninstallation 73

2.3 Changing the system configuration of PFM - Agent for Oracle 75
2.4 Changing the operation of PFM - Agent for Oracle 76
2.4.1 Changing the storage location of performance data 76
2.4.2 Updating an instance environment 79
2.4.3 Cancellation facility for Oracle access during record collection 90
2.4.4 Updating the Store version to 2.0 92
2.5 Backup and restoration 95
2.5.1 Backup 95
2.5.2 Restoration 96
2.6 Online manuals 97
2.6.1 Setting procedure 97
2.6.2 Viewing the manual 98
2.7 Notes regarding operation 99
2.7.1 Notes on operating 99
2.7.2 Notes on monitoring Oracle Database 12c Release 2 or later 100

3 Installation and setup (UNIX) 102
3.1 Installation and setup 103
3.1.1 Installation and setup workflow 103
3.1.2 Preparation for installing and setting up PFM - Agent for Oracle 104
3.1.3 Installation procedure 111
3.1.4 Setting up PFM - Agent for Oracle 114
3.2 Setup cancellation and uninstallation 146
3.2.1 Cautionary notes on setup cancellation and uninstallation 146
3.2.2 Procedure for canceling setup 147
3.2.3 Procedure for uninstallation 149
3.3 Changing the system configuration of PFM - Agent for Oracle 151
3.4 Changing the operation of PFM - Agent for Oracle 152
3.4.1 Changing the storage location of performance data 152
3.4.2 Updating an instance environment 155
3.4.3 Cancellation facility for Oracle access during record collection 170
3.4.4 Updating the Store version to 2.0 171
3.5 Backup and restoration 174
3.5.1 Backup 174
3.5.2 Restoration 175
3.6 Online manuals 176
3.6.1 Setting procedure 176
3.6.2 Viewing the manual 176
3.7 Notes regarding operation 178
3.7.1 Notes on operating 178
3.7.2 Notes on monitoring Oracle Database 12c Release 2 or later 180

4 Operating PFM - Agent for Oracle in a Cluster System 182

4.1 Cluster system overview 183
4.1.1 HA cluster system 183
4.1.2 Load-balancing cluster system 184
4.2 Processing during failover 186
4.2.1 Failover when a failure occurs on PFM - Agent for Oracle host 186
4.2.2 Effects when PFM - Manager stops 186
4.3 Installation and setup (Windows) 188
4.3.1 Installation and setup workflow 188
4.3.2 Preparation for installation and setup 190
4.3.3 Installation procedure 193
4.3.4 Setup procedure 193
4.4 Installation and setup (UNIX) 200
4.4.1 Installation and setup workflow 200
4.4.2 Preparation for installation and setup 202
4.4.3 Installation procedure 204
4.4.4 Setup procedure 205
4.5 Setup cancellation and uninstallation (Windows) 212
4.5.1 Setup cancellation and uninstallation workflow of PFM - Agent for Oracle 212
4.5.2 Setup cancellation procedure 213
4.5.3 Uninstallation procedure 217
4.6 Setup cancellation and uninstallation (UNIX) 219
4.6.1 Setup cancellation and uninstallation workflow of PFM - Agent for Oracle 219
4.6.2 Setup cancellation procedure 220
4.6.3 Uninstallation procedure 225
4.7 Notes on operating PFM - Agent for Oracle in a cluster system 226
4.7.1 Host name in the collected performance data 226
4.8 Changing the system configuration of PFM - Agent for Oracle 227
4.9 Changing the operation of PFM - Agent for Oracle 228
4.9.1 Updating an instance environment 228
4.9.2 Exporting and importing the logical-host environment definition file 229

Part 3: Reference

5 Monitoring template 231

Overview of the monitoring template 232
Format of alarm explanations 233
List of alarms 234
Buffer Cache Usage 235
Buffer Cache Waits 236
Dict. Cache Usage 237
Disk Sorts 238
Free List Waits 239

Full Table Scans 240
Library Cache Usage 241
Redo Log Contention 242
Server Status 243
Tablespace Usage 244
Format of report explanations 245
Organization of report folders 246
List of reports 248
Blocking Locks(9.1) 251
Cache Usage(8.0) 252
Cache Usage Status(Multi-Agent)(8.0) 253
Cache Usage Trend(Multi-Agent)(8.0) 254
CDB Audit Size Status(9.2) 255
CDB Data File Size Status(9.2) 256
CDB Diagnostics Size Status(9.2) 257
CDB Status(9.2) 258
CDB Tablespace Status(9.2) 259
Database Activity Status(8.0) 260
Database Activity Status Detail(8.0) 261
Database Activity Status(Multi-Agent)(8.0) 262
Database Activity Trend(Multi-Agent)(8.0) 263
Database Space Overview(8.0) 264
Database Space Summary(Multi-Agent)(8.0) 265
Database Space Trend(Multi-Agent)(8.0) 266
Datafile I/O Activity Detail(8.0) 267
Datafile I/O Status Detail(Reads)(8.0) 268
Datafile I/O Status Detail(Writes)(8.0) 269
Datafile I/O Status Summary(8.0) 270
Datafile I/O Trend Detail(Reads)(8.0) 271
Datafile I/O Trend Detail(Writes)(8.0) 272
Datafile I/O Trend Summary(8.0) 273
Disk Sorts - Top 10 Sessions(9.1) 274
Error Log(9.1) 275
Fast Recovery Area Status(9.2) 276
Full Table Scans(8.0) 277
I/O Activity - Top 10 Datafiles(9.1) 278
Lock Usage - Top 10 Sessions(9.1) 279
Locked Objects(9.1) 280
Longest Transactions - Top 10 Sessions(9.1) 281
Memory Usage - Top 10 Sessions(9.1) 282
Open Cursors(4.0) 283
Physical I/O - Top 10 Sessions(9.1) 284
Redo Log Buffer Contention(8.0) 285
Server Configuration Status(4.0) 286
Session Detail(9.1) 287
Session Statistics Detail(4.0) 288
SGA Status(9.1) 289

SGA Status Summary(9.1) 290
SQL Text(4.0) 291
System Overview(9.1)(real-time report on the overall status of instance) 292
System Overview(9.1)(real-time report on the general status of instance) 294
Tablespace Status(4.0) 296
Tablespace Status Detail(4.0) 297

6 Records 298
Data model 299
Format of record explanations 300
List of ODBC key fields 303
Summary rules 304
List of data types 306
Field values 307
Fields added only when data is stored in the Store database 310
Fields output when data stored in the Store database is exported 311
Notes on records 312
List of records for PFM - Agent for Oracle 313
Activity Summary (PD_PDAS) 322
ASM Disk (PD_PDDK) 326
ASM Disk Group Interval (PI_PIDG) 331
Backup Async IO (PD_PDBA) 335
Backup Sync IO (PD_PDBS) 338
Block Contention Interval (PI_PIBC) 341
Block Contention Statistics (PD_PDBC) 343
Buffer Pool (PD_PDBP) 345
Cache Summary (PD_PDCS) 348
Cache Summary Interval (PI_PICS) 352
CDB Status (PD_CDCS) 356
CDB Tablespace(PD_CDTS) 360
Circuit (PD_PDCI) 367
Collection Instance 2 (PD_PCI) 369
Collection Tablespace 2(PD_PCTS) 371
Control File (PD_PDCF) 374
Current Sessions Stat Summary (PD_PDS3) 376
Data Dictionary Cache (PD_PDDD) 381
Data Dictionary Cache Interval (PI_PIDD) 383
Data File (PD_PDDF) 385
Data File Interval (PI_PIDF) 394
Database (PD_PDDB) 406
Database Interval (PI_PIDB) 415
Database Object Cache (PD_PDDO) 425
Dispatcher (PD_PDDS) 427
Dispatcher Interval (PI_PIDS) 430
Errorlog Detail (PD_PDEL) 433
GCS Stat Summary (PD_PDGC) 435
GCS Stat Summary Interval (PI_PIGC) 437

Instance (PD_PDI) 439
Instance Availability (PD_PDIA) 441
Latch (PD_PDLA) 444
Latch Interval (PI_PILA) 447
Library Cache (PD_PDLC) 450
Library Cache Interval (PI_PILC) 452
Lock (PD_PDLO) 454
Lock Activity Interval (PI_PIPL) 457
Lock Interval (PI_PILO) 459
Lock Waiters (PD_PDLW) 462
Minimum Database Interval 2 (PI_PMDB) 464
Minimum Data File Interval 2 (PI_PMDF) 465
Minimum Tablespace Interval 2 (PI_PMTS) 469
Multi - Threaded Server (PD_PDMT) 472
Multi - Threaded Server Interval (PI_PIMT) 476
Open Cursor (PD_PDOC) 480
Options Installed (PD_PDO) 482
Parallel Query Server (PD_PDPQ) 483
Parallel Query Server Interval (PI_PIPQ) 486
Parallel Query Statistics (PD_PDPS) 489
Parameter Values (PD_PDP) 490
Process Detail (PD_PDOP) 492
Queue Statistics (PD_PDQU) 494
Resource Limit (PD_PDRL) 496
Rollback Segment (PD_PDRS) 498
Rollback Segment Interval (PI_PIRS) 501
Segment Detail (PD_PDSM) 504
Server Status (PD_STAT) 508
Session Detail (PD_PDS) 510
Session Event (PD_PDEV) 516
Session Event Interval (PI_PIEV) 518
Session I/O Interval (PI_PIIO) 520
Session Stat Summary Interval (PI_PIS2) 522
Session Statistics (PD_PDSS) 527
Session Statistics Summary (PD_PDS2) 529
Session Wait (PD_PDWA) 535
SGA Components (PD_PDSG) 538
Shared Cursor Cache (PD_PDC) 541
Shared Server (PD_PDSH) 545
Shared Server Interval (PI_PISH) 548
Sort Segment (PD_PDSR) 551
Sort Segment Interval (PI_PISR) 554
SQL Text (PD_PDSQ) 557
SQL Text - Performance Based (PD_PDES) 559
SQL*Net Listener (PD_PDNL) 562
SQL*Net Listeners (PD_PDLS) 564
System Event (PD_PDSE) 566

System Event Interval (PI_PISE) 568
System Stat Interval (PI_PIST) 570
System Stat Summary (PD) 572
System Stat Summary Interval (PI) 579
System Statistics (PD_PDST) 587
Table Access (PD_PDTA) 589
Tablespace (PD_PDTS) 591
Tablespace Fragmentation (PD_PDTF) 602
Tablespace Interval (PI_PITS) 607
Transaction (PD_PDTR) 611
Transaction Interval (PI_PITR) 614
Transaction Lock (PD_PDTL) 617
Version (PD_PDV) 619

7 Messages 621
7.1 Message format 622
7.1.1 Format of output messages 622
7.1.2 Format of message explanations 623
7.2 Message output destinations 624
7.3 List of messages output to the Windows event log and syslog 628
7.4 Messages 629

Part 4: Troubleshooting

8 Error Handling Procedures 650
8.1 When an error occurs 651
8.2 Troubleshooting procedures 652
8.2.1 Problems relating to the start and setup of services 652
8.2.2 Problems relating to the collection and management of performance data 653
8.2.3 Other problems 654
8.3 Log information 655
8.3.1 Types of log information 655
8.3.2 List of log files and directories 657
8.4 Required troubleshooting information 660
8.4.1 In Windows 660
8.4.2 In UNIX 664
8.5 Collecting troubleshooting information 669
8.5.1 In Windows 669
8.5.2 In UNIX 671
8.6 Detecting errors in Performance Management 674
8.7 Recovering from errors in Performance Management 675

Appendixes 676
A Estimating System Requirements 677

A.1 Memory requirements 677
A.2 Disk space requirements 677
A.3 Disk space requirements for operation in a cluster system 677
B Kernel Parameters 678
B.1 HP-UX 678
B.2 Solaris 678
B.3 AIX 679
B.4 Linux 679
C List of Identifiers 680
D List of Processes 681
E List of Port Numbers 683
E.1 numbers for PFM - Agent for Oracle 683
E.2 Routing through a firewall 683
F PFM - Agent for Oracle Properties 684
F.1 List of Agent Store service properties 684
F.2 List of Agent Collector service properties 687
G List of Files and Directories 697
G.1 PFM - Agent for Oracle files and directories 697
G.2 List of files and directories in the message catalog storage directory 705
H Migration Procedure and Notes on Migration 708
I Precautions Regarding Permissions 710
I.1 When the sp_inst.sql script is executed 710
I.2 When the sp_drop.sql script is executed 710
I.3 When the mk_user.sql script is executed 711
I.4 When performance information is collected 711
J Version Compatibility 712
K Outputting Action Log Information 713
K.1 Event types output to the action log 713
K.2 Format for saving the action log files 713
K.3 Action log output format 714
K.4 Settings for outputting action logs 719
L Linking with JP1/SLM 722
M About Communication in IPv4 Environments and IPv6 Environments 723
N Version Revisions 724
N.1 Revisions in 12-10 724
N.2 Revisions in 12-00 724
N.3 Revisions in 11-00 725
N.4 Revisions in 10-50 726
N.5 Revisions in 10-00 726
N.6 Revisions in 09-10 727
N.7 Revisions in 09-00 727

O Reference Material for This Manual 729
O.1 Related publications 729
O.2 Conventions Abbreviations for product names 729
O.3 Conventions Acronyms 731
O.4 Conventions: Product names, service IDs, and service keys 732
O.5 Conventions: Installation folder 733
O.6 Conventions: KB, MB, GB, and TB 733
P Glossary 734

Index 741

1 Overview of PFM - Agent for Oracle

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

1.1 Features of PFM - Agent for Oracle

PFM - Agent for Oracle is a program that monitors the performance of an Oracle and collects and manages its performance data.

PFM - Agent for Oracle has the following features:

- Ability to analyze the operating status of Oracle

  PFM - Agent for Oracle provides the information needed for easy analysis of the operating status of Oracle. It does so by collecting and summarizing performance data (such as session statistics) obtained from the Oracle instance being monitored, and then graphically displaying any trends or changes.
- Ability to detect Oracle problems and quickly provide the information needed to identify the cause of a problem

  In the event of a problem, such as a malfunction in a session on the Oracle system being monitored, PFM - Agent for Oracle alerts the user to the problem at an early stage by using email or other means to notify the user. PFM - Agent for Oracle also provides a graphical display of the information needed to identify the cause of a problem.

To use PFM - Agent for Oracle, you must also install the programs PFM - Manager, and PFM - Web Console.

The following subsections summarize the functions of PFM - Agent for Oracle.

1.1.1 Collecting Oracle performance data

PFM - Agent for Oracle enables you to collect performance data, such as statistical information on the current Oracle session on the host being monitored.

Note:

In a UNIX environment, PFM - Agent for Oracle supports SJIS (for Japanese UNIX), EUC (for Japanese UNIX), UTF-8 (for Japanese UNIX and Chinese UNIX) and GB18030 (for Chinese UNIX). In a Windows environment, PFM - Agent for Oracle supports SJIS (for Japanese Windows) and GB18030 (for Simplified-Chinese Windows). In other environment, PFM - Agent for Oracle supports within the scope of 7-bit ASCII characters.

With PFM - Agent for Oracle, you can use the collected performance data as follows:

- To graphically display the operating status of Oracle

  By using PFM - Web Console, you can process and display performance data in a graphical format called a *report*. A report facilitates the analysis of the Oracle operating status.

  There are two types of reports:

  - *Real-time reports*

    A real-time report indicates the current status of an Oracle system being monitored. It is used primarily to check the current status of the system and to detect problems in the system. To display real-time reports, PFM - Agent for Oracle uses current performance data that has just been collected.
  - *Historical reports*

    A historical report indicates the status of an Oracle system being monitored from a selected point of time in the past to the present. It is used primarily to analyze trends in the system. To display a historical report, the system uses performance data that has been stored in a database for PFM - Agent for Oracle.
- As criteria for determining whether a problem has occurred

  You can set PFM - Agent for Oracle to take some action (such as notifying the user) if collected performance data indicates an abnormal condition.

1.1.2 Collecting performance data based on its characteristics

PFM - Agent for Oracle collects performance data in *records*. Each record consists of smaller units called *fields*. Collectively, the records and fields are referred to as the *data model*.

Records are classified into two types according to their characteristics. These record types are predefined in PFM - Agent for Oracle. The user simply uses PFM - Web Console to specify the performance data records to be collected.

PFM - Agent for Oracle supports the following two record types:

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

  For records of the PI record type, the system collects performance data for a specified interval, such as the number of processes executed in one minute. You can use these records to analyze the changes or trends in the system status over time.
- Product Detail record type (referred to hereafter as the *PD record type*)

  For records of the PD record type, the system collects performance data that indicates the system status at a specific point in time, such as detailed information about the currently active processes. You can use these records to obtain the system status at a particular time.

For more information about record types, see *6. Records*.

1.1.3 Saving performance data

Because collected performance data is stored in a special database, you can save performance data up to the current date, and use it to analyze trends (from a selected point in the past to the current date) in the Oracle operating states. This special database is called the *Store database* of PFM - Agent for Oracle. Trends are analyzed using historical reports.

Use PFM - Web Console to select the performance data records to be stored in the Store database. For details about how to select records with PFM - Web Console, see the chapter on the management of operation monitoring data in the *JP1/Performance Management User's Guide*.

1.1.4 Notifying users of problems in Oracle operation

In addition to using performance data collected by PFM - Agent for Oracle to display Oracle Database performance as reports, you can also use it to warn the user of a problem or error occurring during Oracle Database operation.

Suppose that you wish to notify the user by email whenever the percentage of table scans exceeds 10%. You can do this by setting *percentage of table scans that do not use an index exceeds 10%* as the abnormal condition threshold, and setting the system to send an email to the user when this threshold is reached. What the system does when the threshold is reached is called an *action*. The following types of actions are available:

- Sending an email
- Executing a command
- Issuing an SNMP trap
- Issuing a JP1 event

The definition of a threshold or action is called an *alarm*. A table of defined alarms is called an *alarm table*. Once an alarm table is defined, it is associated with PFM - Agent for Oracle. Associating an alarm table with PFM - Agent for

Oracle is called *binding*. Once an alarm table has been bound to PFM - Agent for Oracle, whenever the performance data collected by PFM - Agent for Oracle reaches the threshold defined as an alarm, the event is reported to the user.

By defining alarms and actions, you can detect Oracle problems at an early stage and take appropriate action.

For details about how to set alarms and actions, see the chapter on alarm-based operation monitoring in the *JP1/Performance Management User's Guide*.

1.1.5 Easy definition of alarms and reports

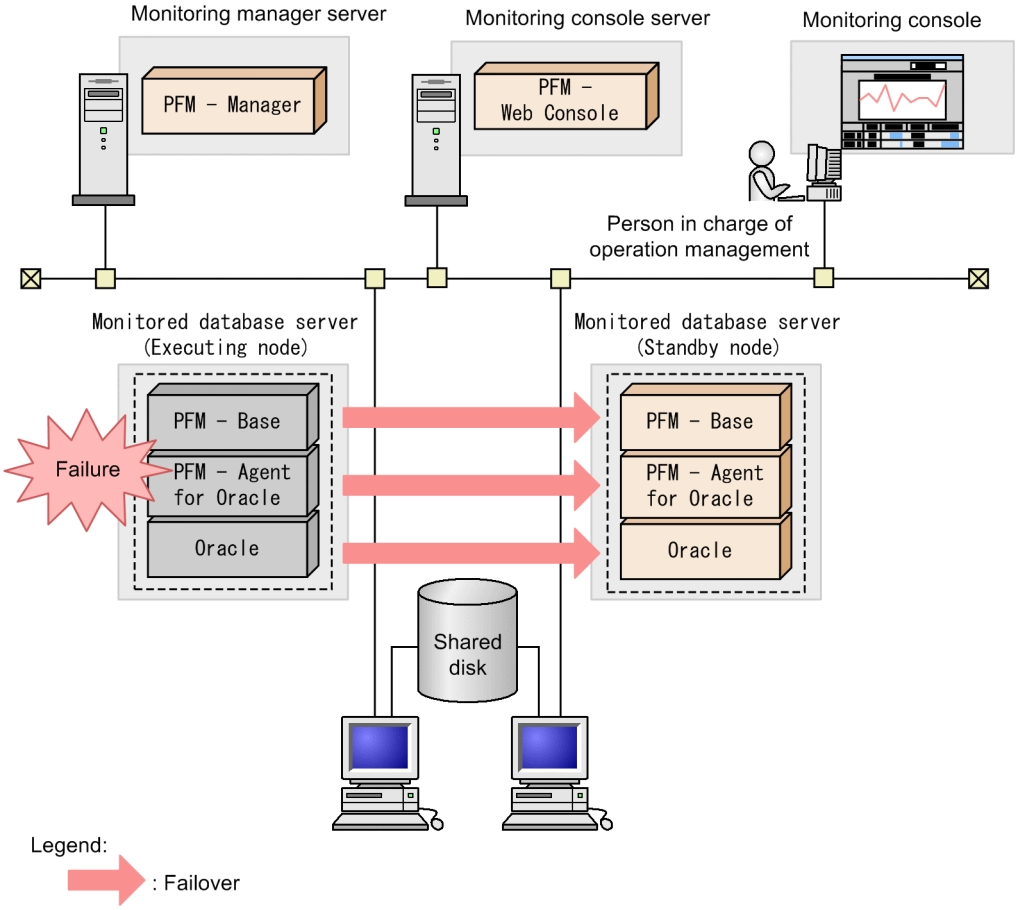
PFM - Agent for Oracle provides a *monitoring template* that contains predefined information necessary for standard reports and alarms. The monitoring template facilitates setup for monitoring the Oracle operating status, because it does not require you to make any complicated definitions. You can also customize the monitoring template as needed for your environment. For details about how to use the monitoring template, see the chapter on creating reports for operation analysis or the chapter on alarm-based operation monitoring in the *JP1/Performance Management User's Guide*. For details about the monitoring template, see *5. Monitoring template*.

1.1.6 Operation with a cluster system

By using a cluster configuration, you can create a highly reliable system that continues to operate even in the event of a system failure. As a result, the programs in Performance Management can continue operation and monitoring 24 hours a day.

The following figure shows an example of operation when a problem occurs on the monitored host in a cluster system.

Figure 1–1: Example of an PFM - Agent for Oracle configuration in an HA cluster system



Set up two environments with the same configuration, and define the environment for normal operation as an *executing node* and the other for failure operation as a *standby node*.

For more details about running the programs in Performance Management on a cluster system, see Part 2, *4. Operating PFM - Agent for Oracle in a Cluster System*.

1.1.7 Operation with a multitenant container database (CDB) configuration

If the monitoring-target program is Oracle Database 12c Release 2 or later, PFM - Agent for Oracle can operate with a multitenant container database (referred to hereafter as the *CDB*) configuration, in addition to a traditional non-multitenant container database (referred to hereafter as the *non-CDB*) configuration.

The following monitoring activities are possible in a CDB configuration environment:

- When the root container (`CDB$ROOT`) or the pluggable databases (referred to hereafter as the *PDBs*) are to be monitored

  Instances of PFM - Agent for Oracle can be created for each root container (`CDB$ROOT`) or PDB for monitoring.

  Use this configuration for monitoring if you want to monitor the operating status of each root container (`CDB$ROOT`) or PDB or if you want to acquire detailed performance data.

  Note that the monitoring of each application container (an application root and application PDBs) is not supported.
- When all PDBs are to be monitored from the root container (`CDB$ROOT`) to which they belong

Instances of PFM - Agent for Oracle for monitoring the root container (CDB$ROOT) can be created to monitor the following performance data:

- The open mode and size of all PDBs belonging to the root container (CDB$ROOT)

  When the size of each PDB is to be monitored, if the maximum storage sizes (MAX_SIZE, MAX_DIAG_SIZE, and MAX_AUDIT_SIZE) of PDBs are either not set or set to *unlimited* in the Oracle Database to be monitored, values corresponding to each maximum storage size and the size of each available free space would be 0, and the percentage of each available free space would be 100. Under this circumstance, field monitoring is not possible. For details, see the information provided in *6. List of records for PFM - Agent for Oracle*.

- The size of the tablespaces of both the root container (CDB$ROOT) and all the open PDBs that belong to it

Use this configuration for monitoring if you want to monitor the root container (CDB$ROOT) so that you can centrally monitor the open mode or other performance data of all the PDBs that belong to it.

In a CDB configuration, the records that are supported on monitoring PDBs and the root container are different from the supported records in a non-CDB configuration. Furthermore, some records and fields collect data from only monitored PDBs and root container while others collect common information via a database instance.

For details, see *6. List of records for PFM - Agent for Oracle* or the descriptions of each record field.

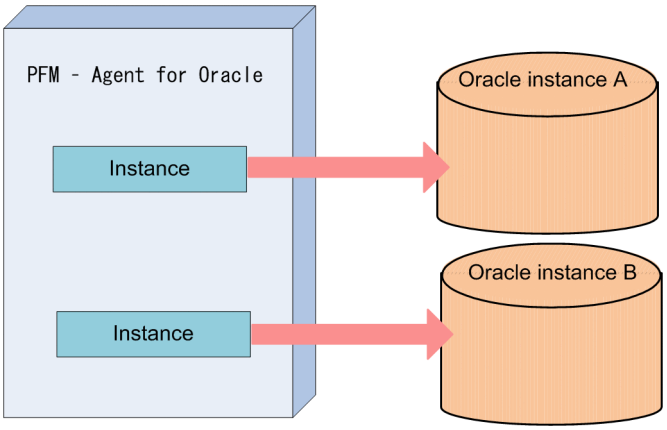
(1) Operating a CDB configuration in single-instance

The following show operations used when monitoring with a non-CDB configuration and with a CDB configuration.

(a) When monitoring a non-multitenant container database (non-CDB)

Instance of PFM - Agent for Oracle for each Oracle instance are created and monitored.

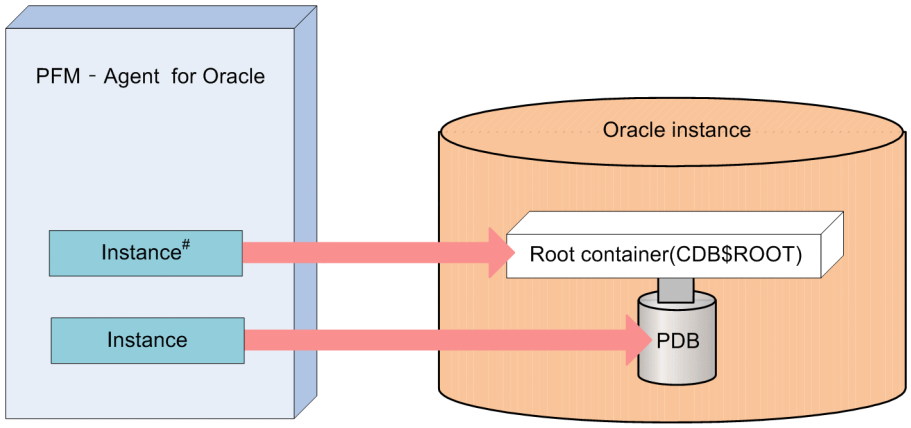
Figure 1–2: Monitoring with a non-CDB configuration



(b) When monitoring the root container (CDB$ROOT) or PDBs

Instance of PFM - Agent for Oracle are created, and a PDB and the root container are monitored.

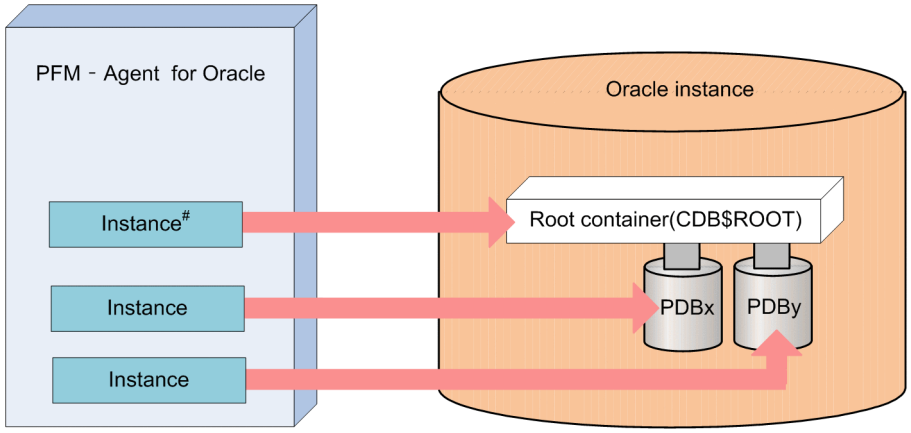
Figure 1–3: Monitoring with a CDB configuration (with a single PDB)



Monitoring the root container (CDB$ROOT) is optional.
Monitor it as needed. For example, you might need to monitor tablespaces of the root container.

Instance of PFM - Agent for Oracle are created, and PDBs and the root container are monitored.

Figure 1–4: Monitoring with a CDB configuration (with multiple PDBs)

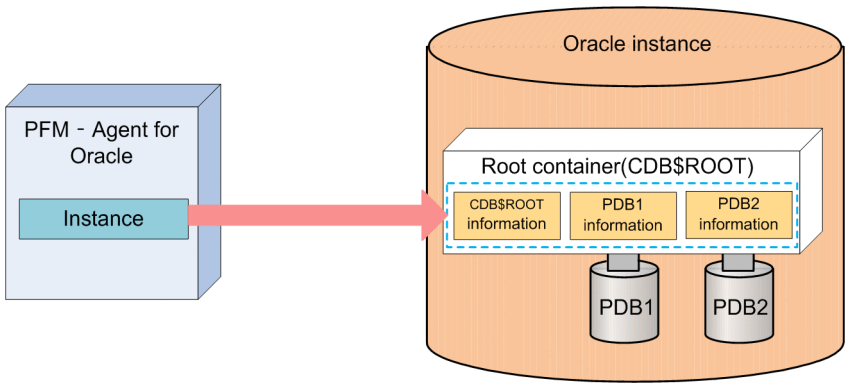


Monitoring the root container (CDB$ROOT) is optional.
Monitor it as needed. For example, you might need to monitor tablespaces of the root container.

(c) When monitoring the root container (CDB$ROOT) to monitor all PDBs belonging to it

Instances of PFM - Agent for Oracle are created for monitoring the root container (`CDB$ROOT`) so that all PDBs belonging to it can also be monitored.

Figure 1–5: Monitoring all PDBs from the root container (CDB$ROOT) to which they belong in a CDB configuration



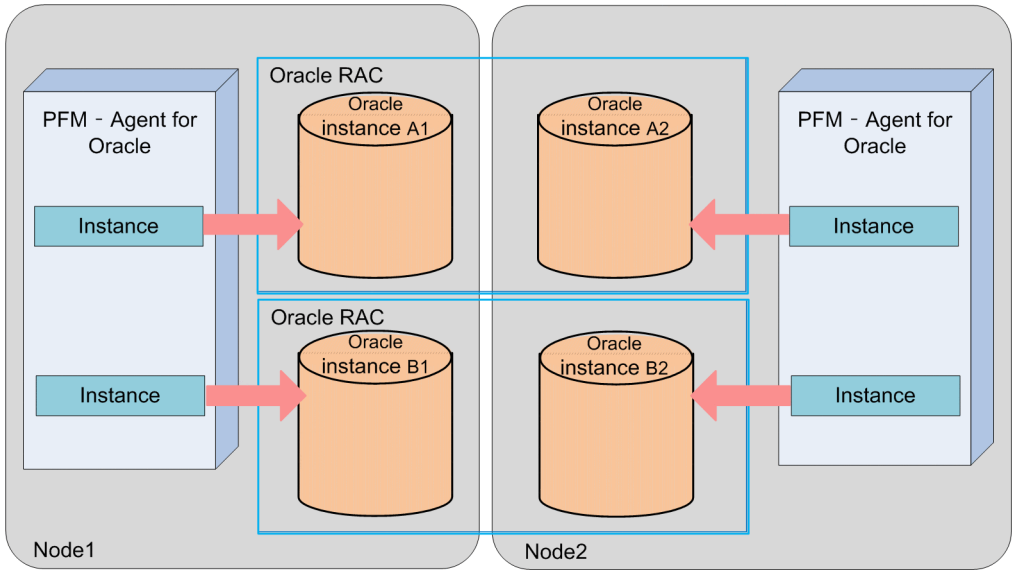
(2) Operating a CDB configuration in an Oracle RAC

The following shows the configurations of PFM - Agent for Oracle and the Oracle database when operating a non-CDB configuration and when operating a CDB configuration in Oracle RAC.

(a) When monitoring a non-multitenant container database (non-CDB)

Instance of PFM - Agent for Oracle for each Oracle instance of each node are created and monitored.

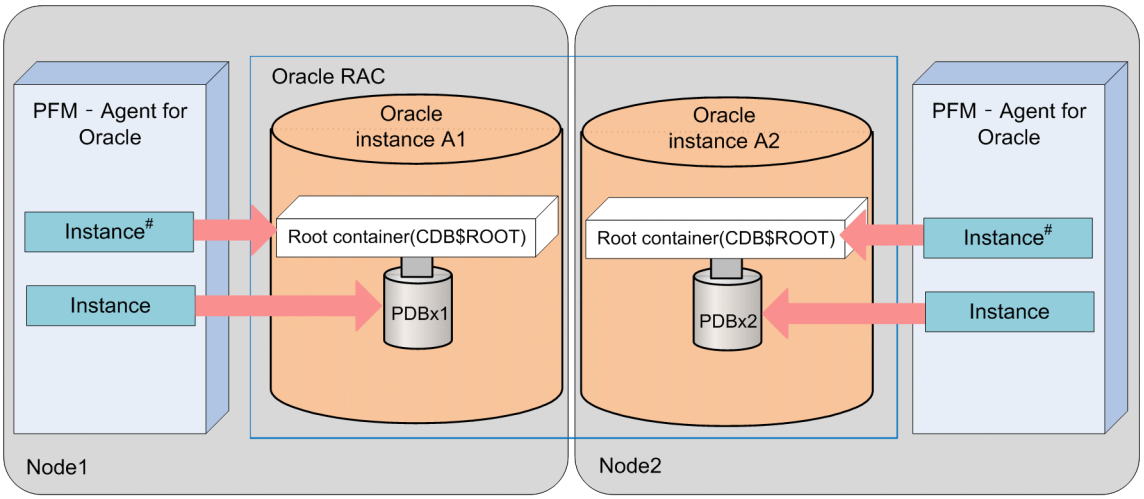
Figure 1–6: Operating a CDB configuration in an Oracle RAC



(b) When monitoring the root container (CDB$ROOT) or PDBs

Instance of PFM - Agent for Oracle are created, and the PDB and the root container of each node are monitored.

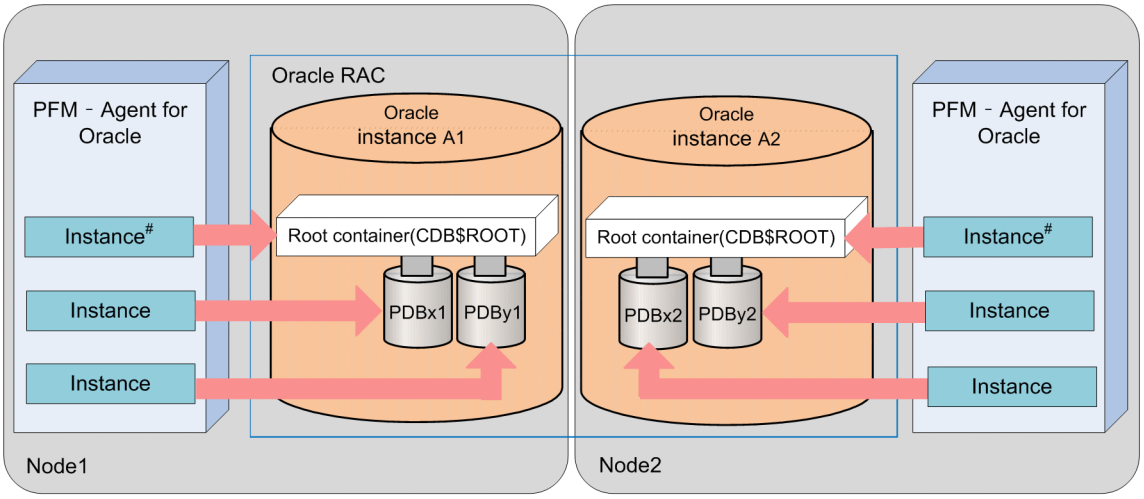
Figure 1–7: Operating a CDB configuration (singletenant) in an Oracle RAC



Monitoring the root container (CDB$ROOT) is optional.
Monitor it as needed. For example, you might need to monitor tablespaces of the root container.

Instance of PFM - Agent for Oracle are created, and the PDB and the root container of each node are monitored.

Figure 1–8: Operating a CDB configuration (multitenant) in an Oracle RAC

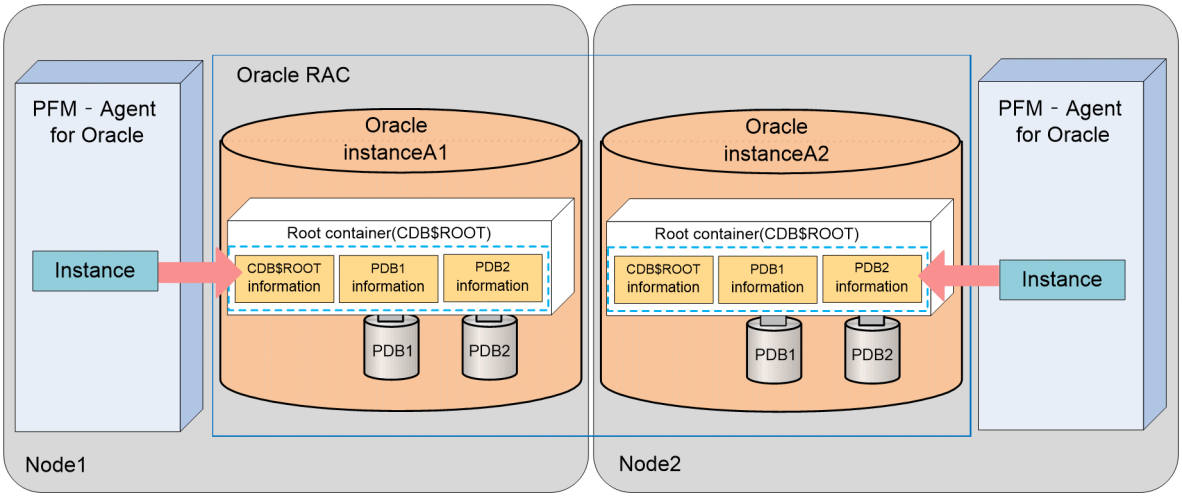


Monitoring the root container (CDB$ROOT) is optional.
Monitor it as needed. For example, you might need to monitor tablespaces of the root container.

(c) When monitoring the root container (CDB$ROOT) to monitor all PDBs belonging to it

Instances of PFM - Agent for Oracle for monitoring the root container (`CDB$ROOT`) are created to monitor the root container (`CDB$ROOT`) of each node as well as the PDBs belonging to it.

Figure 1–9: Monitoring all PDBs from the root container (CDB$ROOT) to which they belong in an Oracle RAC



(3) Note

When monitoring multiple PDBs or root containers (`CDB$ROOT`) in a CDB configuration (multi-tenant), create an instance of PFM - Agent for Oracle that monitors each container. The number of instances that can be created depends on the system configuration, resource status, number of records to be monitored, and collection interval. Please verify the situation thoroughly before operating.

1.2 Overview of collection and management of performance data

The procedures for collecting and managing performance data depend on the record type used to store the performance data. The records for PFM - Agent for Oracle are classified into the following two types:

- PI record type
- PD record type

For details about how to collect and manage performance data, see the following sections:

- Performance data collection procedure

  For details about the performance data collection procedure, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

  For details about the values of the collected performance data, see *6. Records*.
- Performance data management procedure

  For details about the performance data management procedure, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

When you want to select performance data from the records collected and managed by PFM - Agent, you use PFM - Web Console. For details about how to select performance data, see the chapter on the management of operation monitoring data in the *JP1/Performance Management User's Guide*.

1.3 Example of performance monitoring using PFM - Agent for Oracle

Performance monitoring is critical for the setup and management of Oracle server environments. The following explains the purpose of performance monitoring using PFM - Agent for Oracle, and provides an example of performance monitoring.

1.3.1 Purpose of performance monitoring

Performance monitoring using PFM - Agent for Oracle can be used to perform the following:

- Analysis of performance data to discover causes of bottlenecks
- Monitoring to check whether the Oracle server is running properly

During Oracle server operation, specific causes can negatively impact overall Oracle server performance. These causes can include the following:

- Insufficient buffer cache
- Insufficient shared pools
- Insufficient memory for sort operations
- Increase in the "Search All" ratio
- Insufficient segment free lists
- Insufficient disk capacity
- Occurrence of wait events concerning REDO log file

It is very important to make sure that the Oracle server is running properly. In addition to performance standpoints, monitoring such as the following can be used to check whether an Oracle server is running properly.

- Oracle instance operation monitoring

Performance monitoring using PFM - Agent for Oracle can be performed to achieve stable operation for Oracle servers.

Note that the performance monitoring method thresholds are for reference only. The actual thresholds need to be determined through baseline measurement.

The actual items set need to be determined based on the type of operation for the Oracle server.

1.3.2 Determining a baseline

Determining a baseline involves using the results of performance measurement to calculate the line at which no system operation problems are anticipated.

Performance Management products treat baseline values as *thresholds* for system operation monitoring. As such, determining a baseline is important for deciding on a threshold and performing performance monitoring.

Hitachi recommends that you perform the following when you determine a baseline:

- Measuring statuses during peak times, such as by performing testing under heavy operation environment load

- Re-measure baselines when system resources or operation environments are changed, as these can differ significantly by system configuration

1.3.3 Search processing performance

To maintain and improve Oracle Database search processing performance, set the maximum number of tuning items, and monitor Oracle to check whether they remain in the permitted ranges.

The following Oracle items may be monitored to maintain or improve search processing performance:

- Buffer cache usage
- Database data and rollback block contention
- Dictionary caching
- Sorts performed on disk as a ratio of all sorts, for memory and disk I/O usage
- "Search All" ratio
- Library caching

(1) Records and fields related to search processing performance

The following table describes the records and fields related to search processing performance.

Table 1–1: Records and fields related to search processing performance

Record used	Field used	Value measured (example)
PI	Cache Hit %	Buffer cache usage
	Buffer Busy Wait %	Buffer busy wait ratio
	Dict Cache Get Miss %	Ratio of data requests due to cache misses
	Sort Overflow %	Ratio of sorts using temporary segments
	Non-Index Lookups %	Ratio of full-table scans for which caching is not performed
	Lib Cache Miss %	Library cache miss rate This field means the ratio of times the allocated objects in library cache are reloaded. As the value of this field increases, the amount of resources in use also increases.

(2) Monitoring methods

Monitoring buffer cache usage

Buffer cache usage can be monitored by using the *Buffer Cache Usage alarm* provided by the monitoring template.

Monitoring database data and rollback block contention

Database data and rollback block contention can be monitored by using the *Buffer Cache Waits alarm* provided by the monitoring template. Monitoring database data and rollback block contention alongside *Buffer Cache Usage alarm* can be very useful.

When Buffer Cache Usage is at or below its threshold, and Buffer Cache Waits is at or above its threshold, this is often because the buffer cache is insufficient. When the buffer cache is insufficient, disk I/O is performed, which may degrade performance. This problem can be handled by increasing the size of DB_CACHE_SIZE.

Monitoring dictionary caching

Dictionary caching can be monitoring by using the *Dict. Cache Usage alarm* provided by the monitoring template.

Monitoring library caching

Library caching can be monitored by using the *Library Cache Usage alarm* provided by the monitoring template.

When Dict. Cache Usage is at or above its threshold, and Library Cache Usage is at or above its threshold, this is often because the shared pool is insufficient. An insufficient shared pool may degrade search performance. This problem can be handled by increasing `SHARED_POOL_SIZE`#.

#

With Oracle 10g, when the initial parameter `SGA_TARGET` is specified, since the `SGA` configuration parameter is automatically adjusted, no action needs to be performed when thresholds for warning or abnormal conditions are exceeded.

Monitoring sorts performed on disk as a ratio of all sorts, when memory or disk I/O is used

The sorts performed on disk as a ratio of all sorts when memory or disk I/O is used can be monitored using the *Disk Sorts alarm* provided by the monitoring template.

When Disk Sort is at or above its threshold, this is often because the memory for sort operations is insufficient. When the memory for sort operations is insufficient, a disk sort is performed using the `TEMPORARY` segment. This may degrade performance, but the problem can be handled by increasing the value of `SORT_AREA_SIZE`.

Monitoring the "search all" ratio

The "search all" ratio can be monitored using the *Full Table Scans alarm* provided by the monitoring template.

When Full Table Scans is at or above its threshold, this is often because "search all" occurs, degrading search performance. Search performance can be tuned by narrowing down the search targets.

1.3.4 Data update processing performance

Oracle can be monitored to prevent performance degradation for Oracle Database data update processing. The following items can be used in Oracle monitoring to prevent degraded performance for data update processing:

- Buffer cache usage
- Database data and rollback block contention
- Free list contention

(1) Primary fields related to data update processing performance

The following table lists the records and fields related to data update processing performance.

Table 1–2: Records and fields related to data update processing performance

Record used	Field used	Value measured (example)
`PI`	`Cache Hit %`	Buffer cache usage.
	`Buffer Busy Wait %`	Buffer busy wait ratio.
	`Free List Wait Events`	Free list wait events.

(2) Monitoring methods

Monitoring buffer cache usage

Buffer cache usage can be monitored by using the *Buffer Cache Usage alarm* provided by the monitoring template.

Monitoring database data and rollback block contention

Database data and rollback block contention can be monitored by using the *Buffer Cache Waits alarm* provided by the monitoring template. Monitoring database data and rollback block contention alongside buffer cache usage can be very useful.

When Buffer Cache Usage is at or below its threshold, and Buffer Cache Waits is at or above its threshold, this is often because the buffer cache is insufficient. When the buffer cache is insufficient, disk I/O is performed, which may degrade search performance. This problem can be handled by increasing the size of `DB_CACHE_SIZE`.

Monitoring free list contention

The free list ratio can be monitored by using the *Free List Waits alarm* provided by the monitoring template.

When Free List Waits is at or above its threshold, this is often due to insufficient segment free lists. Insufficient free lists may degrade performance for data update processing. This problem can be handled by adding the `FREE LISTS` option to the `STORAGE` operator, to rebuild the table.

1.3.5 Oracle instance operation monitoring

Oracle server operation can be monitored.

The following item can be used for Oracle server operation monitoring:

- Oracle instance operation monitoring

(1) Records and fields related to Oracle instance operation

The following table lists the records and fields related to Oracle instance operation.

Table 1–3: Records and fields related to Oracle instance operation

Record used	Field used	Value measured (example)
`PD_PDIA`	`Availability`	Availability status. Valid values are `0` (stopped) or `1` (running).

(2) Monitoring method

Monitoring Oracle instance operation

Oracle instance operation can be monitored by using the *Server Status alarm* provided by the monitoring template.

If the Availability value is `0`, it is determined that connection cannot be established with the Oracle database. You can remedy this problem by checking the state of the Oracle Database or listener.

1.3.6 Disk monitoring

Changes in disk capacity can be monitored for running Oracle databases to ensure stable Oracle database operation.

The following item can be used for monitoring changes in the disk capacity of a running Oracle database:

- Tablespace capacity

(1) Records and fields related to tablespace

The following table lists the records and fields related to tablespace.

Table 1–4: Records and fields related to tablespace

Record used	Field used	Value measured (example)
PD_PDTS	Free %	Ratio of free space.

(2) Monitoring methods

Monitoring tablespace capacity

Tablespace capacity can be monitored by using the *Tablespace Usage alarm* provided by the monitoring template.

If Tablespace Usage is at or below its threshold, free space is insufficient. After you have identified the tablespace in the PD_PDTS record for which the problem occurred, check the tablespace usage and secure sufficient free space.

1.3.7 Monitoring wait events concerning REDO log file

When the size of REDO log file is small on an Oracle server, wait events may occur. Thus, it is crucial to monitor the frequency of wait events concerning REDO log file:

- Monitoring the frequency of wait events concerning REDO log file

(1) Records and fields related to the frequency of wait events concerning REDO log file

The following table lists the records and fields related to the frequency of wait events concerning REDO log file.

Table 1–5: Records and fields related to wait events concerning REDO log file

Record used	Field used	Value measured (example)
PI	Redo Log Space Requests	Number of times that waiting is required until the disk space is allocated to a REDO log entry because the active redo log file is full.

(2) Monitoring methods

Monitoring wait events concerning REDO log file

Wait events concerning REDO log file can be monitored by using the *Redo Log Contention alarm* provided with the monitoring template.

When Redo Log Contention is at or above its threshold, wait events concerning REDO log file occurs, often because the REDO log file is too small.

This problem can be handled by increasing the size of the REDO log file.

2 Installation and setup (Windows)

This chapter describes the procedures for installing and setting up PFM - Agent for Oracle. For details about how to install and set up an entire Performance Management system, see the chapter on installation and setup for Windows in the *JP1/Performance Management Planning and Configuration Guide*.

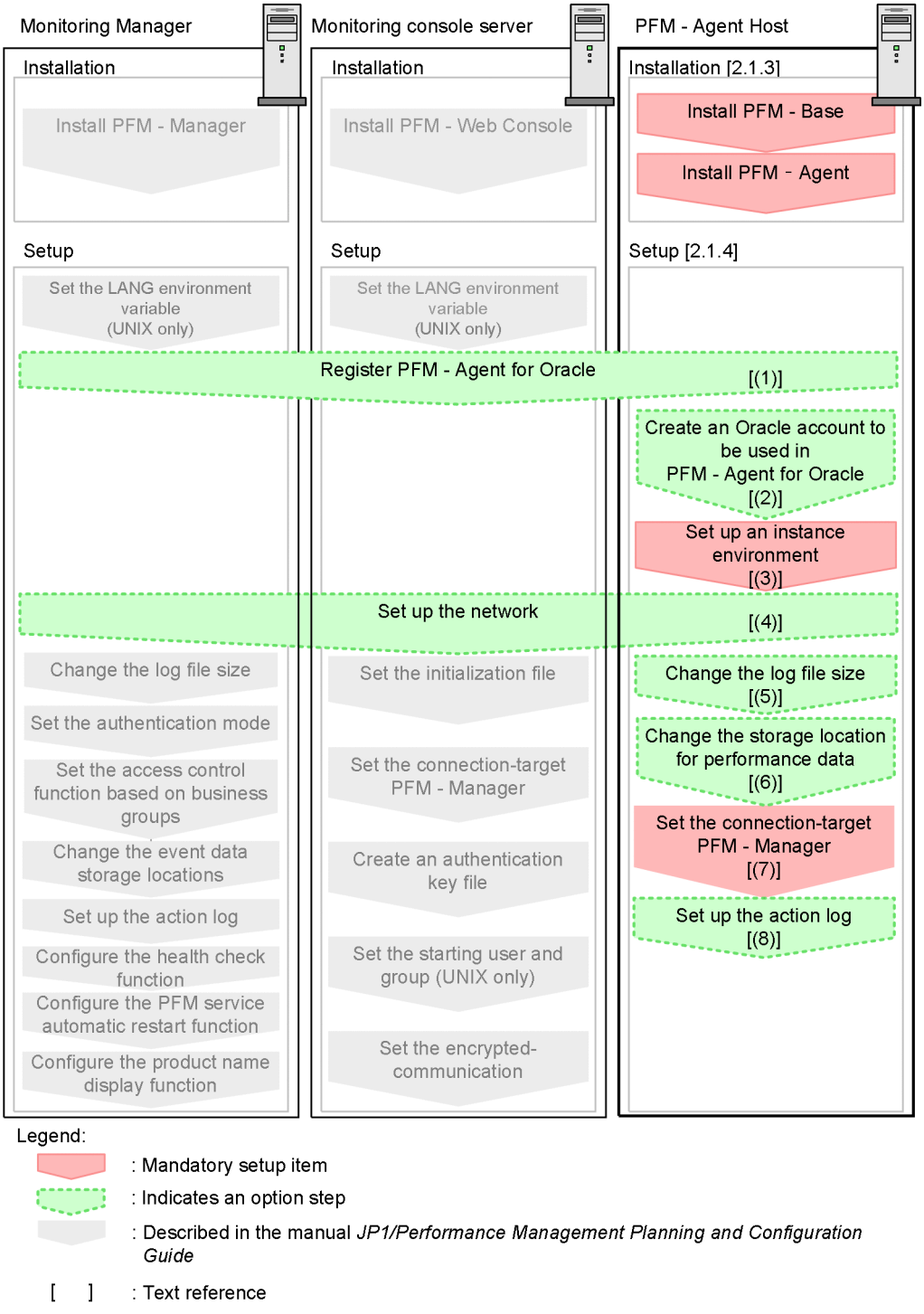
2.1 Installation and setup

This section describes the procedures for installing and setting up PFM - Agent for Oracle.

2.1.1 Installation and setup workflow

The following figure shows the workflow for installing and setting up PFM - Agent for Oracle.

Figure 2–1: Installation and setup workflow (Windows)



For details about the installation and setup procedures for PFM - Manager and PFM - Web Console, see the chapter on 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 chapter on commands in the manual *JP1/Performance Management Reference*.

2.1.2 Preparation for installing and setting up PFM - Agent for Oracle

Check the following before installing and setting up PFM - Agent for Oracle.

(1) OS requirements

PFM - Agent for Oracle runs on the following OSs:

- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019

(2) Network environment settings

The following describes the network environment required to run Performance Management.

(a) IP address settings

The PFM - Agent for Oracle host must be set up in a network environment where IP addresses can be resolved from host names. PFM - Agent for Oracle will not start in an environment where IP addresses cannot be resolved.

PFM - Agent for Oracle can run in an IPv6 environment and dual stack environment in addition to an IPv4 environment. To run PFM - Agent for Oracle in an IPv6 environment, the monitored Oracle database must support IPv6 environments.

You can use the real host name or the alias name as a monitoring host name (a host name used in Performance Management system).

- When using a real 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

  Set up the environment so that an IP address can be resolved from the specified alias name.

For details about the configuration of a monitoring host name, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Use one of the following methods to set the host name and IP address of a host monitored by Performance Management programs:

- Host information settings file for Performance Management (`jpchosts` file)
- `hosts` file
- DNS (Domain Name System)

> **Important**
>
> Notes on setting IP addresses:
>
> - Although the Performance Management programs and PFM - Agent for Oracle can operate in a DNS environment, they do not support host names in FQDN (Fully Qualified Domain Name) format. When you specify an IP address, use the host name returned by the `hostname` command after removing the domain name portion.
> - If you intend to use Performance Management within multiple LAN environments, set the IP addresses in the `jpchosts` file. For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.
> - Performance Management cannot operate on the hosts to which IP addresses are assigned dynamically by DHCP. Make sure that all the hosts on which Performance Management programs are installed are configured with user-specified static IP addresses.

Performance Management supports IPv6 environments as well as IPv4 environments as a network configuration. Therefore, Performance Management can operate even in a network configuration in which both an IPv4 environment and an IPv6 environment are used.

PFM - Agent for Oracle can communicate with PFM - Manager via IPv6. Note that this explanation applies only when the OS of a host on which PFM - Agent for Oracle and PFM - Manager are installed is Windows or Linux.

For details about the scope of communication in an environment with both IPv4 and IPv6, see *M. About Communication in IPv4 Environments and IPv6 Environments*.

When you want to use IPv6 for communication between PFM - Manager and PFM - Agent for Oracle, the settings for using IPv6 must be enabled for both the PFM - Manager host and the PFM - Agent host. In addition, before installing PFM - Agent for Oracle, you need to enable the use of IPv6 on the PFM - Agent host. You have to execute the `jpcconf ipv6 enable` command to enable this setting. If this setting is already enabled, however, you do not need to execute the command. If you want to check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command.

For details about the `jpcconf ipv6 enable` command and `jpcconf ipv6 display` command, see the chapter that describes 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.*

When you use IPv6 for communication between a monitored host and PFM - Agent for Oracle, specify the name of a monitored host where name resolution can be performed.

Communication between PFM - Agent for Oracle and a monitoring target is performed with an IP address that can be resolved. Also, if an IPv4 environment and an IPv6 environment are both used, and communication between PFM - Agent for Oracle and the monitoring target fails with an IP address that can be resolved, the communication is not retried by using another IP address.

For example, if communication fails when IPv4 is used, IPv6 is not used to retry communication. Similarly, if communication fails when IPv6 is used, IPv4 is not used to retry communication. Make sure beforehand that a connection can be established.

(b) Port number settings

The following table shows the default port numbers assigned to the services of Performance Management programs. For other services and programs, available port numbers are automatically assigned each time they are started. If you use Performance Management in a firewall environment, use fixed port numbers. For details about how to set fixed port numbers, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Table 2–1: Default port numbers for Performance Management program services (in Windows)

Service description	Service name	Parameter	Port number	Remarks
Service configuration information management facility	Name Server	`jp1pcnsvr`	22285	The port number used by the Name Server service of PFM - Manager. This port is set up on every Performance Management host.
Service status management facility	Status Server	`jp1pcstatsvr`	22350	The port number used by the Status Server service of PFM - Manager and PFM - Base. This port is set up on the hosts on which PFM - Manager and PFM - Base are installed.
Monitoring console communication facility	View Server	`jp1pcvsvr`	22286	The port number used by the View Server service of PFM - Manager. This port is set up on the hosts on which PFM - Manager is installed.
Web service facility	Web Service	--	20358	The port number used by the Web Service service of PFM - Web Console.
Web container facility	Web Console	--	20359 20360	The port number used by the Web Console service of PFM - Web Console.
JP1/SLM linkage facility	JP1/ITSLM	--	20905	The port number set by JP1/SLM.

Legend:

--: None

Ensure that the network is set up to allow communication using these port numbers, which are used by PFM - Agent for Oracle.

(3) OS user permission required to install PFM - Agent

To install PFM - Agent for Oracle, you must have the following permissions:

[When the UAC function is used]

You must log in to the host to be installed with Administrators permissions or be upgraded to Administrators permissions when you start the installer.

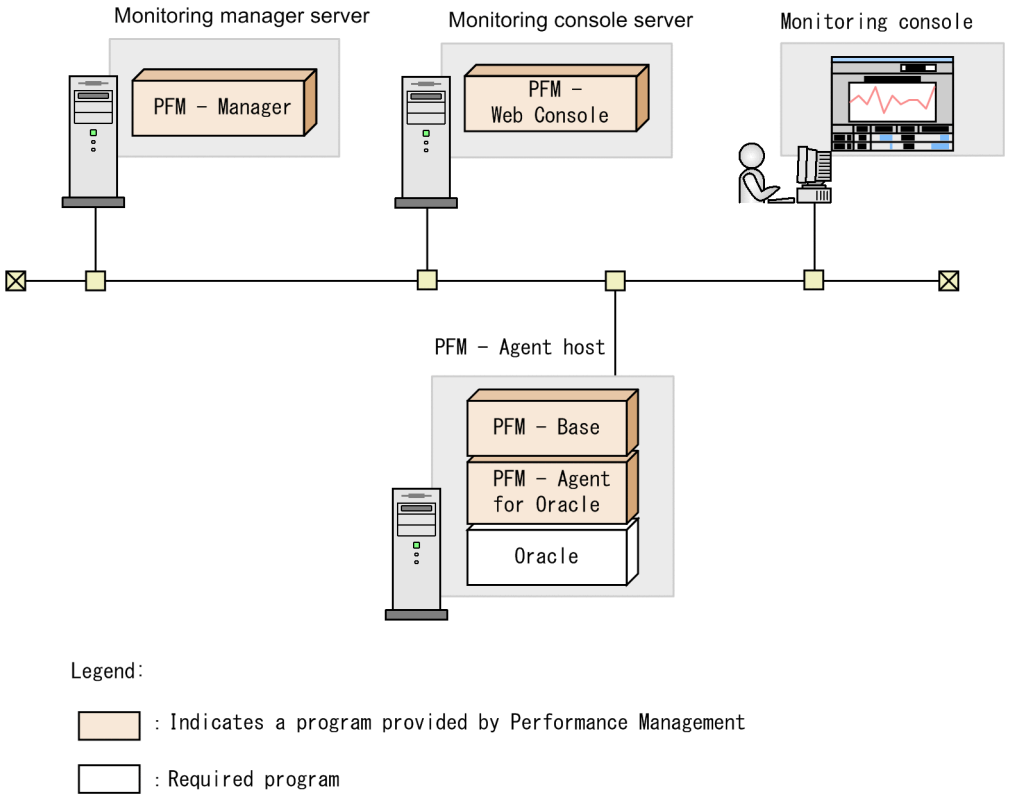
[When the UAC function is not used]

You must log in to the host to be installed with Administrators permissions.

(4) Prerequisite programs

This subsection describes the prerequisite programs required to install PFM - Agent for Oracle. The following figure 2-2 shows the program configuration:

Figure 2–2: Program configuration



(a) Monitoring target programs

The monitoring target programs of PFM - Agent for Oracle are as follows:

- Oracle Database Standard Edition
- Oracle Database Standard Edition One
- Oracle Database Standard Edition 2
- Oracle Database Enterprise Edition

To monitor this program, it must be installed on the same host as PFM - Agent for Oracle.

Note that when a monitoring target program running on a virtualized OS is being monitored, PFM - Agent for Oracle monitors what the monitoring target program can perform on the virtualized OS.

(b) Performance Management programs

Install PFM - Agent for Oracle and PFM - Base on the PFM - Agent host.

PFM - Base is a prerequisite program for PFM - Agent for Oracle. Only one instance of PFM - Base is required, even when multiple instances of PFM - Agent are installed on one host.

Note that you do not need to install PFM - Base if PFM - Manager and PFM - Agent for Oracle are installed on the same host.

To monitor Oracle operation using PFM - Agent for Oracle, PFM - Manager and PFM - Web Console are required.

(5) Installation and setup in a cluster system

When you install and set up PFM - Agent in a cluster system, the prerequisite network environment and program configuration is different from those for a normal system. There are also additional tasks that must be performed on the executing nodes and standby nodes. For details, see *4. Operating PFM - Agent for Oracle in a Cluster System*.

(6) Preparations for collecting data in the event of a failure

If a problem occurs, user mode process dumps, and other data might be required. To obtain these dumps when a problem has occurred, use one of the following methods to set up output of these dumps in advance.

You can use the following registry setting to obtain user mode process dumps of data that you can investigate for troubleshooting when an application program terminates:

```
\\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Loca
lDumps
```

Specify the following registry key values:

- DumpFolder: `REG_EXPAND_SZ` *dump-output-folder-name*
  Permission to write to the output destination folder is needed.
- DumpCount: `REG_DWORD` *Number-of-dumps-to-be-saved*
- DumpType: `REG_DWORD` `2`

> **Important**
>
> - When you set the registry to output user mode process dumps, user mode process dumps are output for not only JP1 programs, but also for other application programs. Make sure you keep this in mind when you specify that user mode process dumps are to be output.
> - When user mode process dumps are output, available disk space can be adversely affected. Make sure that you specify a dump output folder that has enough disk space for the dumps.

(7) Cautionary notes

Note the following when installing and setting up Performance Management.

(a) Precautions regarding 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.

(b) Notes on installing and setting up multiple Performance Management programs on 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:

- When PFM - Manager and PFM - Agent are installed on the same host, PFM - Base is not required. In this case, PFM - Manager is a prerequisite program for PFM - Agent and must be installed before PFM - Agent is installed.

- You cannot install PFM - Base and PFM - Manager on the same host. If you want to install PFM - Manager on a host on which PFM - Base and PFM - Agent are already installed, uninstall all Performance Management programs, and then install PFM - Manager and PFM - Agent in that order. The same applies when you install PFM - Base on a host on which PFM - Manager and PFM - Agent are already installed: you must uninstall all Performance Management programs, and then install PFM - Base and PFM - Agent in that order.
- If you install PFM - Agent on a host on which PFM - Manager is already installed, the connection-target PFM - Manager will be the instance of PFM - Manager on the local host, and cannot change it to that on the remote host. If you want PFM - Agent to connect to PFM - Manager on a remote host, ensure that PFM - Manager is not installed on the host on which you install PFM - Agent.
- If you install PFM - Manager on a host on which PFM - Agent is already installed, the connection-target PFM - Manager is reset to the local host. See the setting results that are output to the common message log.
- If you install PFM - Agent on a host on which PFM - Web Console in already installed, close all the browser windows before you install the program.
- When you perform a new installation of a Performance Management program, the status management facility will be enabled by default. However, if you upgrade from version 07-50 to 08-00 or newer, the settings for the status management function remain the same as they were in the older version. To change the setting of the status management facility, see the chapter on error detection for 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.

(c) Notes on upgrading PFM - Agent for Oracle

Before you perform version upgrade installation, you need to perform the procedure described in *2.2.2(1) (b) Deleting the objects registered in the Oracle Database*. Complete version upgrade installation and then perform the procedure in *2.1.4(3) (b) Registering objects in the Oracle Database*.

For details about notes on upgrading the versions of Performance Management programs, see the section describing the notes on version upgrading in the chapter that explains installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

For details about notes on upgrading the version of PFM - Agent for Oracle, see *H. Migration Procedure and Notes on Migration*.

For details about upgrading, see the *JP1/Performance Management Planning and Configuration Guide*.

If you delete the objects described in *2.2.2(1)(b) Deleting the objects registered in the Oracle Database* after the upgrade installation finishes rather than before the upgrade installation starts, the following messages appear:

"ORA-04043: object LSC_13_PDDB2 does not exist", "ORA-04043: object LSC_13_PIDB2 does not exist", "ORA-04043: object LSC_13_PIDB3 does not exist"

These messages, however, does not indicate a serious problem. Even if this message appears, proceed to register the objects again, see *2.1.4(3)(b) Registering objects in the Oracle Database*.

(d) Other cautionary notes

- When the monitoring-target program is Oracle Database 12c Release 1, you cannot monitor a multitenant container database (CDB) and can monitor a non-CDB.

- If the monitoring-target program is Oracle Database 12c Release 2 or later, in addition to the program in a traditional non-CDB environment, PDBs and the root container (`CDB$ROOT`) in a CDB environment can be monitored. Monitoring application containers (an application root and application PDBs) is not supported.
- When you perform a new installation of PFM - Agent for Oracle in an environment where no other Performance Management program has been installed, make sure that there are no files or folders in the installation folder.
- You may be prompted to restart the system if you attempt to install PFM - Agent for Oracle while another Performance Management program or service is running. Alternatively, you may be prompted to restart the system while Windows Event Viewer or another program that references Performance Management files is running. If this occurs, restart the system as indicated in the message, and complete the installation.
- The installer may be unable to expand the files required for installation if you attempt to install PFM - Agent for Oracle in the following cases:
  - While a Performance Management program or service is running or while another program that references Performance Management files (for example, Windows Event Viewer) is running
  - When there is insufficient disk space
  - When you do not have the required folder permission

  Stop any active Performance Management programs or services or other programs that reference Performance Management files, and then perform the installation again. If the problem is caused by insufficient disk space or a lack of the appropriate folder permissions, fix the problem and then perform the installation again.
- When you install Performance Management programs, check whether the following security-related programs are installed. If they have been installed, take appropriate action according to the explanations below.
  - Security monitoring program

    Stop the security monitoring program or change the settings so that the installation of Performance Management programs will not be interrupted.
  - Virus detection program

    Stop the virus detection program before you install Performance Management programs.

    If a virus detection program is running during the installation of Performance Management programs, the installation processing might slow down, installation might not be executable, or the programs might not be able to be installed correctly.
  - Process monitoring program

    Stop the process monitoring program or change the settings. Also, specify settings that prevent the services or processes of Performance Management and common components from being monitored.

    If the process monitoring program starts or stops these services or processes during the installation of Performance Management programs, installation might fail.
- Do not specify the following folder as the installation-folder. If so, the installation process may fail:

  system-folder`\Program files`
- This software is a Hitachi program product that conforms to the disk copy installation of JP1/ServerConductor/Deployment Manager and Hitachi Compute Systems Manager Deployment Manager Plug-in, or the copy functionality using conversion to image files that is provided by a virtual platform. About the disk copy installation, see notes on installing replicated disks in the manual *JP1/Performance Management Planning and Configuration Guide*.

2.1.3 Installation procedure

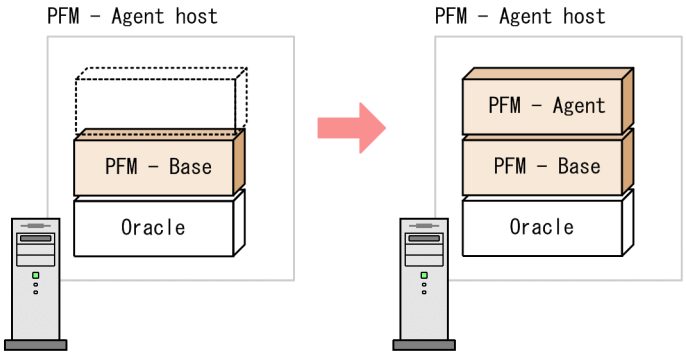
This subsection explains the order in which the component programs of PFM - Agent for Oracle are to be installed, and describes how to install these programs from the supplied medium.

(1) Order of installation

Install PFM - Base, and then PFM - Agent. PFM - Base must be installed on the host before you can install PFM - Agent.

If you want to install PFM - Agent and PFM - Manager on the same host, install PFM - Manager before you install 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 *2.4.4 Updating the Store version to 2.0*.

Multiple instances of PFM - Agent on the same host can be installed in any order.



1. Install PFM - Base.
2. Install PFM - Agent.

(2) How to install the programs

You can install Performance Management programs on a Windows host either by using the supplied medium, or by using JP1/Software Distribution to perform a remote installation. For details about how to use JP1/Software Distribution, see the *Job Management Partner 1/Software Distribution Administrator's Guide Volume 1*, for Windows systems.

Precaution:

If user account control (UAC) functionality is enabled on the OS, the User Account Control dialog box might be displayed during installation. If this dialog box is displayed, click the **Continue** button to continue installation. If you click the **Cancel** button, the installation is canceled.

To install programs from the supplied medium:

1. Log on to the host on which you intend to install the programs as an administrator.
2. If any Performance Management services are running on the local host, stop all of them.

   The services you are going to stop are the Performance Management services running on both the physical and logical hosts. For details about how to stop services, see the chapter that explains startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.
3. Insert the supplied medium into the machine and execute the installer.

   Proceed with installation by following the instructions of the installer that starts.

   The following items, which have been set upon the installation of PFM - Manager or PFM - Base, are displayed for your information:

   - User information
   - Installation folder
   - Program folder

4. Click the **Install** button to start the installation process.

2.1.4 Setting up PFM - Agent for Oracle

This subsection describes how to set up PFM - Agent for Oracle for operation.

Option indicates an item that may or may not be required depending on your operating environment, or an optional item that you can set if you do not wish to use the default.

(1) Register PFM - Agent for Oracle Option

To perform integrated management of PFM - Agent using PFM - Manager and PFM - Web Console, you must register PFM - Agent for Oracle with PFM - Manager and PFM - Web Console.

If PFM - Agent for Oracle is already registered in PFM - Manager and PFM - Web Console, you do not have to follow the procedure described below. If PFM - Agent for Oracle is not registered yet, manually register PFM - Agent for Oracle according to the procedure.

You can determine whether manual registration of PFM - Agent for Oracle is necessary by referring to the conditions described below.

Manually registering PFM - Agent for Oracle in PFM - Manager

When all of the following conditions apply, manually register PFM - Agent for Oracle in PFM - Manager:

- The PFM - Agent for Oracle to be installed is of a product version that is not specified in the *Release Notes* for PFM - Manager.
- PFM - Agent for Oracle is installed on a host other than PFM - Manager.

Manually registering PFM - Agent for Oracle in PFM - Web Console

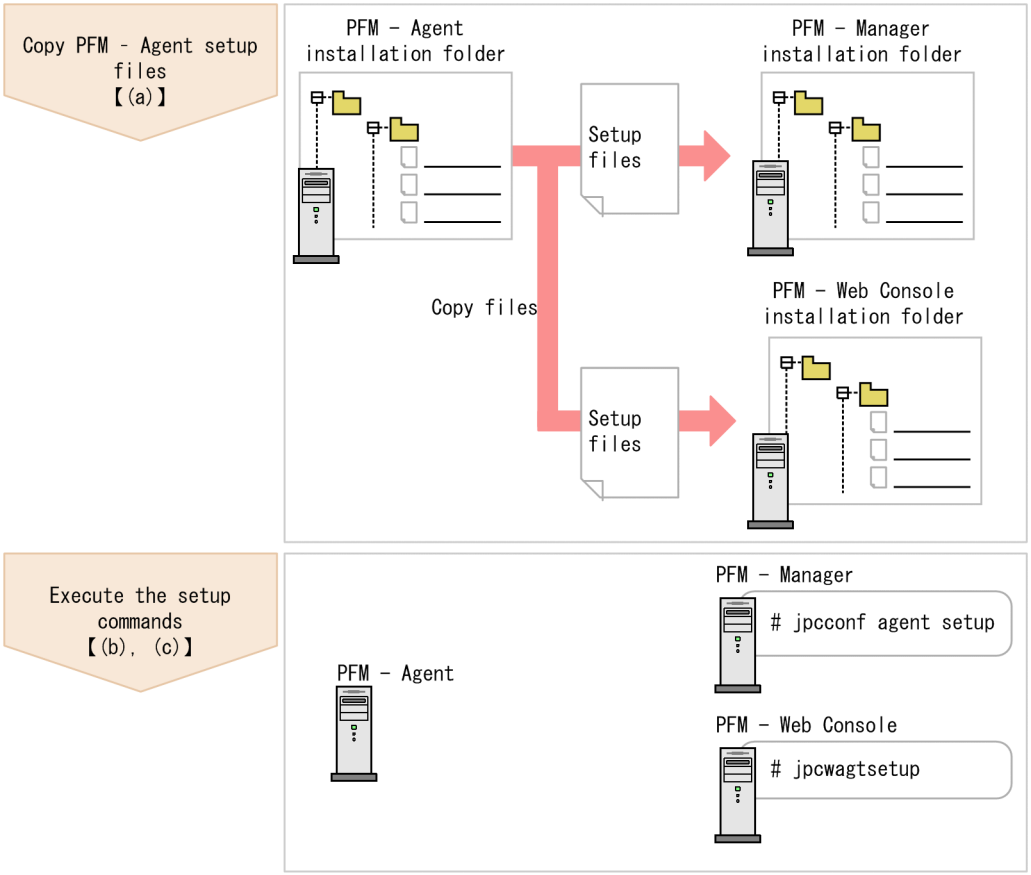
When the following condition applies, manually register PFM - Agent for Oracle in PFM - Web Console:

- The PFM - Agent for Oracle to be installed is of a product version that is not specified in the *Release Notes* for PFM - Web Console.

If, however, the *Release Notes* for PFM - Agent for Oracle state that it is necessary to execute the setup command, execute the setup command.

The following figure shows the flow of PFM - Agent registration.

Figure 2–3: Flow of PFM - Agent for Oracle registration



Legend:

【 】 : Text reference

Notes:

- Register PFM - Agent before setting up an instance environment.
- You do not need to register a new instance of an already registered version of PFM - Agent for Oracle when you add it to the Performance Management system.
- When you have installed different versions of PFM - Agent for Oracle on different hosts, set up the older version before the newer version.
- When you install PFM - Agent on the same host as PFM - Manager, the `jpcconf agent setup` command is executed automatically, and the following message is output to the common message log: KAVE05908-I `New agent setup (`*pfm-agent-service-key*`) ended successfully. (version=`*version*`)`. Check the result of command execution in the log file. If execution was unsuccessful, try it again. For details about how to execute commands, see the chapter on commands in the manual *JP1/Performance Management Reference*.
- Registration of PFM - Agent for Oracle creates the **Oracle** folder in **Reports** window and **Alarms** window of PFM - Web Console. If the **Oracle** file or folder already exists on the **Reports** window, rename the file or folder before registering PFM - Agent for Oracle.

(a) Copy the PFM - Agent for Oracle setup files

Copy the setup files from the host on which you installed PFM - Agent for Oracle to the hosts on which PFM - Manager and PFM - Web Console are installed.

To copy the files:

1. If PFM - Web Console is running, stop it before copying the files.
2. Copy the PFM - Agent for Oracle setup files in binary mode.
   The following table shows the location of the setup files and where they should be copied.

   Table 2–2: Setup files to be copied

PFM - Agent setup file	Destination		
	PFM program name	OS	Destination folder
installation-folder\setup\jpcagtow.EXE	PFM - Manager	Windows	*installation-folder*\setup
installation-folder\setup\jpcagtou.Z		UNIX	/opt/jp1pc/setup/
installation-folder\setup\jpcagtow.EXE	PFM - Web Console	Windows	*installation-folder*\setup
installation-folder\setup\jpcagtou.Z		UNIX	/opt/jp1pcwebcon/setup/

(b) Execute the setup command on the PFM - Manager host

On the PFM - Manager host, execute the following command to set up PFM - Agent for Oracle:

```
jpcconf agent setup -key Oracle
```

Although an example of interactive command execution is shown here, the `jpcconf agent setup` command can be also executed non-interactively. For details about the `jpcconf agent setup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

Notes on executing the command:

If any Performance Management programs or services are still running on the local host when you execute the `jpcconf agent setup` command, an error may occur. If an error occurs, make sure that all Performance Management programs and services have completely stopped, and then execute the `jpcconf agent setup` command again.

You can then delete the PFM - Agent setup files remaining on the PFM - Manager host.

(c) Execute the setup command on the PFM - Web Console host

On the PFM - Web Console host, execute the following command to set up PFM - Agent for Oracle:

```
jpcwagtsetup
```

You can then delete the PFM - Agent setup files remaining on the PFM - Web Console host.

(2) Create an Oracle account to be used in PFM - Agent for Oracle

To monitor an Oracle Database and collect performance data by using PFM - Agent for Oracle, you must perform either of the following setting tasks:

- Set the `sys` account as the account used to monitor the Oracle Database from PFM - Agent for Oracle

- Create a special Oracle account with system privileges, and set it as the account used to monitor the Oracle Database from PFM - Agent for Oracle

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, PDBs and the root container (`CDB$ROOT`) can be monitored. To operate PFM - Agent for Oracle in a CDB configuration, use one of the following Oracle accounts for monitoring:

- When monitoring the root container (`CDB$ROOT`) to monitor all PDBs belonging to it

  If instances of PFM - Agent for Oracle are to be created for monitoring the root container (`CDB$ROOT`) so that both the root container (`CDB$ROOT`) and the PDBs belonging to it can be monitored, perform monitoring by using an administrative account (`sys` or `system`). If you perform monitoring with a common user's account, you cannot collect performance information on PDBs. For details, see *6. List of records for PFM - Agent for Oracle* or descriptions of each record field.
- When monitoring the root container (`CDB$ROOT`) or PDBs

  If instances of PFM - Agent for Oracle are to be created to monitor each root container (`CDB$ROOT`) or PDB, perform monitoring with one of the following users' accounts:

  - If you create a user for monitoring the root container (`CDB$ROOT`)

    A common user with system privileges for `CDB$ROOT` listed in *Table 2-3*

    When the root container (`CDB$ROOT`) that is being monitored, collecting performance data is different by monitored account. For details, see *6. List of records for PFM - Agent for Oracle* or descriptions of each record field.

    This is difference of setting for collecting PDB's performance data. For details, see your Oracle documentation.
  - If you create a user for monitoring PDBs

    A local user with system privileges for the PDBs to be monitored listed in *Table 2-3*

If you create Oracle accounts by executing the `mk_user.sql` script, the following users can be created:

- If the monitoring target is Oracle Database 11g Release 2 or earlier, or Oracle Database 12c Release 1 or later in a non-CDB configuration

  A user with system privileges listed in *Table 2-3*
- If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration
  - A common user with system privileges for `CDB$ROOT` listed in *Table 2-3*
  - A local user with system privileges for the PDBs to be monitored listed in *Table 2-3*

The following table lists the operations that PFM - Agent for Oracle performs for the Oracle Database to collect performance data. The table also lists the system privileges required to perform the operations with the Oracle account.

Table 2–3: Operations for the Oracle Database and required system privileges

Operations that PFM - Agent for Oracle performs for the Oracle Database	System privileges required to perform the operations
• Searching the static data dictionary view • Searching the dynamic performance view • Executing the listener control utility • Acquiring the execution schedule of the selected SQL • Executing a stored package specific to PFM - Agent for Oracle	• CREATE SESSION • CREATE TABLE • CREATE PROCEDURE • SELECT ANY DICTIONARY • SELECT ANY TABLE • INSERT ANY TABLE • DELETE ANY TABLE • UPDATE ANY TABLE

Operations that PFM - Agent for Oracle performs for the Oracle Database	System privileges required to perform the operations
• Searching the static data dictionary view • Searching the dynamic performance view • Executing the listener control utility • Acquiring the execution schedule of the selected SQL • Executing a stored package specific to PFM - Agent for Oracle	• CREATE ANY INDEX • ALTER ANY INDEX • UNLIMITED TABLESPACE (This privilege is not needed when an assignment limit that allows writing to the default tablespace of the account used for monitoring has been set.)

The `sys` account has the system privileges shown in *Table 2-3*. When the `sys` account cannot be used for Oracle Database monitoring due to security requirements, use an Oracle account with the system privileges shown in *Table 2-3*. No functional differences exist between the `sys` account and the Oracle accounts that have the system privileges shown in *Table 2-3*.

When creating an Oracle account for monitoring the Oracle Database, without using the `mk_user.sql` script, grant the appropriate privileges for operations. By granting these privileges, the minimum necessary privileges can be granted to the Oracle account. When granting privileges according to role, do so explicitly (using `GRANT` *privileges*...). For details about the privileges needed for each operation, see *I. Precautions Regarding Permissions*.

The following table lists the information necessary to execute the `mk_user.sql` script and create an Oracle account. Check the information before starting setup operations.

Table 2–4: Information required to create an Oracle account

Item	Description
`Enter username`	Specifies the name of the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `user` parameter of the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. The default is `PFMAGTO`. Note: If you specify an existing account in the database as an account to be used by an instance of PFM - Agent for Oracle, a script error occurs. Always check the account names existing in the database in advance, and specify an account that will be used only for the instance of PFM - Agent for Oracle.
`Enter password`	Specifies the password for the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `BY password` parameter of the `IDENTIFIED` clause in the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item.
`Enter default tablespace`	Specifies the default tablespace used by the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `DEFAULT TABLESPACE` clause in the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item. Note: Do not specify the `SYSTEM` or `INDEX` tablespace as the default tablespace. Before specifying the default tablespace, make sure that no problem occurs when a package for PFM - Agent for Oracle is registered in the tablespace. Alternatively, create an exclusive tablespace for PFM - Agent for Oracle, and then specify the tablespace as the default tablespace.
`Enter default temporary tablespace`	Specifies the default temporary tablespace used by the account to be created.

Item	Description
Enter default temporary tablespace	The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the TEMPORARY TABLESPACE clause in the CREATE USER statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item. Note: Do not specify the SYSTEM, INDEX, or USERS tablespace as the default temporary tablespace. Before specifying the default temporary tablespace, make sure that no problem will occur if the tablespace is used as the default temporary tablespace. Alternatively, create an exclusive tablespace for PFM - Agent for Oracle, and then specify the tablespace as the default temporary tablespace.

Notes:

- Make sure that the value of each item consists of only 7-bit ASCII alphanumeric characters that do not exceed 30 bytes. If the value is longer than 30 bytes or includes a character that is not a 7-bit ASCII alphanumeric character, the script may operate incorrectly.
- Make sure that the value of each parameter is a nonquoted identifier described in the Schema Object Naming Rules. If you specify a value that is not a nonquoted identifier, the script may operate incorrectly. For details about the Schema Object Naming Rules and nonquoted identifiers, see your Oracle documentation.
- If you create a common user, by executing the mk_user.sql script, for monitoring the root container (CDB $ROOT) of Oracle Database 12c Release 2 or later in a CDB configuration, the default tablespace and default temporary tablespace that are specified in the mk_user.sql script must be included in all containers that belong to the CDB. For that reason, before executing the mk_user.sql script, create the default tablespace and default temporary tablespace in all containers that belong to the CDB. If the mk_user.sql script is executed without satisfying the requirements for creating a common user, the creation might fail and the KAVF12506-E message might appear. For details about the requirements for creating a common user, see the Oracle Database documentation.
- If you want to check the details of an account created by mk_user.sql, see DBA_USERS, which is a static dictionary view for the monitoring-target Oracle Database.
- The following example shows how to view the tablespace for the account A4O in the static data dictionary view DBA_USERS. If it is clear from the execution results of this SQL statement that the account has been created in the wrong tablespace, delete the account, and then re-create it using mk_user.sql.

Example:

To check the details of Oracle account A40 in Windows:

1. From the command prompt, use the sys account to connect to SQL*Plus.

   ```
   sqlplus "sys/sys-account-password AS SYSDBA"
   ```

2. Use SQL*Plus to execute the following SQL statement:

   ```
   SQL>select DEFAULT_TABLESPACE, TEMPORARY_TABLESPACE from DBA_USERS
   where USERNAME='A4O';
   ```

3. Check the execution results. For example, you can check the default tablespace from the DEFAULT_TABLESPACE column and the default temporary tablespace from the TEMPORARY_TABLESPACE column.

Note:

The method for connecting to SQL*Plus with the sys account may differ according to the Oracle version. For details, see the Oracle documentation.

Use SYSDBA privileges to connect to the Oracle Database that you want to monitor.

For details about the CREATE USER statement, see your Oracle documentation.

The following procedure shows how to create an Oracle account. Before creating an Oracle account, make sure that the tablespaces and other required resources have been prepared.

To create an Oracle account:

1. Set up an environment where the sqlplus Oracle command can be executed.
   For details about Oracle environment setup, see your Oracle documentation.

2. Navigate to the following folder, which contains mk_user.sql provided by PFM - Agent for Oracle:
   installation-folder\agto\agent\sql

3. Execute the mk_user.sql script for the monitoring-target Oracle Database.
   Example:

   If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-database* @mk_user.sql

   If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

   - If you create a local user for a PDB

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-PDB* @mk_user.sql

   - If you create a common user for the root container (CDB$ROOT)

   sqlplus *Oracle-account*@*net-service-name-for-the-monitoring-target-CDB$ROOT*/*password-for-the-Oracle-account* @mk_user.sql

   Notes:

   - The sqlplus command is provided by Oracle Corporation.
   - The Oracle account with which the mk_user.sql script is executed must be granted the CREATE USER, CREATE SESSION and GRANT ANY PRIVILEGE system privileges before the script is executed.
   - If the SYS account is used to execute the mk_user.sql script, an error may occur unless the AS SYSDBA option is specified.
   - Establish either a SYSDBA connection to the monitoring-target Oracle Database.
     The following shows an example of the mk_user.sql script:
     Example:
     If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

     sqlplus "sys/*password-for-the-sys*@*net-service-name-for-the-monitoring-target-database* AS SYSDBA" @mk_user.sql

     If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

     - If you create a local user for a PDB

     sqlplus "sys/*password-for-the-sys*@*net-service-name-for-the-monitoring-target-PDB* AS SYSDBA" @mk_user.sql

     - If you create a common user for the root container (CDB$ROOT)

     sqlplus "sys/*password-for-the-sys* AS SYSDBA" @mk_user.sql

- When the `mk_user.sql` script is executed, the execution results are output to a spool file. Output is successful only when the current folder is changed to the folder shown in step 2 when the script is executed.

4. Set the parameters that are required to create an Oracle account.

   Enter the values for the items listed in *Table 2-4* as prompted by the command. All items are mandatory. To use the default value displayed for an item, only press the **Enter** key.

When all values have been entered, the Oracle account is created.

Notes:

- Before creating an account, check whether you want to acquire the value of the Explain Plan (`EXPLAIN_PLAN`) field in the SQL Text (`PD_PDSQ`) record for operations on the objects that belong to the `SYS` schema. If you want to do so, use `sys` as the account to be used by PFM - Agent for Oracle. If you use an account other than `sys`, you will no longer be able to acquire the value of that field. If the value of the `EXPLAIN_PLAN` field cannot be acquired, message `Explain Plan Failed` is stored in the field.
- If the account used by PFM - Agent for Oracle has no privileges to access, or fails to reference, an object that belongs to a schema of the user who executed SQL, the following value cannot be acquired:

  The value of the Explain Plan (`EXPLAIN_PLAN`) field in the SQL Text (`PD_PDSQ`) record
- If the value of the `EXPLAIN_PLAN` field cannot be acquired, message `Explain Plan Failed` is stored in the field. If you want to acquire the value of the Explain Plan (`EXPLAIN_PLAN`) field, execute SQL for manipulating the field in the *owner.table-name* format.
- Any Oracle account created using the `mk_user.sql` script is granted `UPDATE ANY TABLE` or another system privilege that can freely manipulate objects of other schemas. Manage such Oracle accounts with special care.
- The following table lists the privileges granted to Oracle accounts and the assignment limits of tablespaces.

Table 2–5: Privileges granted by mk_user.sql to Oracle accounts and the assignment limits of tablespaces

Type	Privileges granted / assignment limits	Description
System privilege	`CREATE SESSION`	Required to establish a session with the monitored Oracle Database.
	`CREATE TABLE`	Required when registering a table needed to monitor the Oracle Database, for the monitored Oracle Database (see the table in *Table 2-10*).
	`CREATE PROCEDURE`	Required when registering a procedure needed to monitor the Oracle Database, for the monitored Oracle Database (see the package in *Table 2-10*).
	`SELECT ANY DICTIONARY`	Required when registering information needed to monitor an Oracle Database in the monitored Oracle Database (see *Table 2-10*) and when collecting information.
	`SELECT ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field in a SQL Text (`PD_PDSQ`) record.
	`INSERT ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field in a SQL Text (`PD_PDSQ`) record.
	`UPDATE ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field in a SQL Text (`PD_PDSQ`) record.
	`DELETE ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field in a SQL Text (`PD_PDSQ`) record.

Type	Privileges granted / assignment limits	Description
System privilege	CREATE ANY INDEX	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
	ALTER ANY INDEX	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
Assignment limits of tablespaces	Unlimited assignment for the default tablespace#	Required when registering information needed to monitor an Oracle Database in the monitored Oracle Database (see *Table 2-10*) and when obtaining the Explain Plan field of the PD_PDSQ record.

#

Any created account is granted a privilege to write to the default tablespace without any limit. To change the size of the tablespace allocated to an account after you have created the account, issue the ALTER USER statement in an environment where the sqlplus Oracle command can be executed. Note that any Oracle account with which you execute the ALTER USER statement must be granted the ALTER USER system privilege.

The following shows an example of changing the size of tablespace allocated to an account.

Example:

```
ALTER USER Oracle-account QUOTA maximum-tablespace-allocation-size ON tabl
espace-name;
```

For details about the ALTER USER statement, see your Oracle documentation.

(3) Set up an instance environment

The following table lists the instance information items that are to be specified. You can set up multiple instance environments and monitoring targets by repeating the procedure for each instance.

- Setting up instance information
- Registering objects in the Oracle Database
- Setting up the Oracle Database

This section describes the procedures for each of the actions.

(a) Set up instance information

You must specify instance information for the Oracle that is to be monitored by the PFM - Agent for Oracle. Specify instance information on the PFM - Agent host.

The following table lists the instance information items that are to be specified. You should check this information before you start the setup procedure. For details about the Oracle instance information items, see your Oracle documentation.

Table 2–6: PFM - Agent for Oracle instance information

Item	Description	Specifiable value	Default
oracle_sid	Monitoring-target Oracle system identifier (the same value as the value of the ORACLE_SID environment variable)	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	The value specified for the -inst option of the jpcconf inst setup command

Item	Description	Specifiable value	Default
oracle_home[#1]	Oracle home folder (the same value as the value of the ORACLE_HOME environment variable)	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	--
oracle_version[#1]	Version number of Oracle Database When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), 12 is set.	A two-digit number. • Oracle 11g: 11 • Oracle 12c or later: 12	11
oracle_user[#2]	An account for monitoring Oracle. For details about accounts that can be specified and the required privileges, see *(2) Create an Oracle account to be used in PFM - Agent for Oracle*.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	sys
oracle_passwd[#2,#3]	A password for the account that was specified in oracle_user	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	--
sqlnet[#1,#4]	Specify Y for any of the following four cases: 1. If using an Oracle RAC configuration For details on the RAC configuration, see the Oracle documentation. 2. If using a PD_PDIA record to monitor the availability of a listener 3. If any Oracle service is running on an account other than the local system account. 4. If the monitoring target is PDB in a CDB configuration of Oracle Database 12c R2 or later. If you specify N for the case 1, 3 or 4, above, Oracle might encounter an error.	{ Y | N }	N
net_service_name[#1]	The net service name of a monitoring-target database. This value is enabled if you specified Y in sqlnet. For details about the net service name of a monitoring-target database, see your Oracle documentation.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Instance name (the value of oracle_sid)
listener_home[#1]	Specify the environment variable ORACLE_HOME of the Oracle component containing the listener you want to monitor.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs	Instance name (the value of oracle_home)

Item	Description	Specifiable value	Default
listener_home[#1]	Specify environment variable GRID_HOME in the case of Oracle RAC environment.	• The following symbols: , < >	Instance name (the value of oracle_home)
listener_name	Specify the name of one listener that you want to monitor using the PDLS record. If a PDLS record is not being used to monitor the listener, specify the default listener name "LISTENER". This is because even though the specified value is not used, a blank is not permitted.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Default listener name (LISTENER)
retry_time	The retry interval for re-establishing connection in the event of an authentication error during establishment of a connection with the Oracle system. If an authentication error occurs after the specified period has passed, the PFM - Agent for Oracle service stops. When the value is 0, the PFM - Agent for Oracle service stops without retrying to re-establish connection in the event of an authentication error. This item is enabled when startup_always is N. The specification of this item is ignored when startup_always is Y.	0 to 600 (seconds)	0
log_path[#5]	The absolute path name of the folder for storing agent log information	A character string of 245 or fewer bytes that does not include the following characters: • Tabs • The following symbols: / : , ; * ? " < > | Notes: • You can specify the path to a folder under the installation folder only when the default folder is set. • You cannot specify the path to a folder that is used as the output destination of another instance.	*installation-folder*\agto\agent*instance-name*\log
log_size[#5]	The maximum size of one agent log file[#6]	1 to 32 (in megabytes). The recommended value is 16 or greater.	16
timeout[#5, #7]	The timeout period for Oracle access during a query.	0, or 10 to 3600 (in seconds). When 0 is specified, timeout monitoring is not performed. When a value from 1 to 9 is specified, it is changed to 10 at runtime. For details about timeouts, see *2.4.3 Cancellation facility for Oracle access during record collection*.	0
sql_option[#5, #8]	When Y is specified, information about the following items[#8] is not collected for PI_PIDB and PD_PDTS records, and the value specified by 0 or numeric_10 is set.	{ Y | N }	N

Item	Description	Specifiable value	Default
numeric_10[#9]	When sql_option is set to Y, the value specified is set for items for which information is not collected. If sql_option is set to N, this specification is disregarded.	0 to 99999. Note that if the value specified exceeds the maximum value for the data format of the set field (32767 for short and 65535 for ushort), the maximum value for the data format is set.[#10]	0
startup_always	PFM - Agent for Oracle may stop due to an Oracle connection error, such as when the monitored Oracle is still starting up when PFM - Agent for Oracle starts up. If Y is specified, start processing continues even if a connection error occurs. If N is specified, start processing will stop if an error occurs.	{Y|N}	Y
Store Version[#11]	Specify the version of the Store database to be used. For details about the version of the Store database, see *2.4.4 Updating the Store version to 2.0*.	{1.0|2.0}	2.0
localtemp_option[#12]	Option for switching the display of the free space of the locally managed temporary tablespace of PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTF, PD_PDTS, and PD_PCTS records. If Y is specified, display the size of the free space. If N is specified, display the size of the unallocated space.	{Y|N}	N
undospace_option[#13]	Option for switching how the value displayed as the free space in the UNDO tablespace of the PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTS, PD_PCTS and PD_CDTS records is determined. If N is specified, the size of the unallocated space is displayed. If Y is specified, the size of the free space is displayed.	{Y|N}	Y
nls_lang[#14]	Option for specifying the character encoding used for communication between PFM - Agent for Oracle and Oracle Database.	Character code set: • In Japanese Windows: {AMERICAN_AMERICA.US7ASCII|AMERICAN_AMERICA.JA16SJISTILDE} • In Simplified-Chinese Windows: {AMERICAN_AMERICA.US7ASCII|AMERICAN_AMERICA.ZHS32GB18030} • Other case: AMERICAN_AMERICA.US7ASCII	AMERICAN_AMERICA.US7ASCII

Legend:

--: None

#1

For monitoring of the Oracle database, the 32-bit Oracle Client has been the required product in PFM - Agent for Oracle 10-50 or earlier. However, in PFM - Agent for Oracle 11-00 or later, the Oracle Client 64-bit library is used for monitoring.

The setting method for version 10-50 or earlier cannot be used to connect to an Oracle Database. Therefore, you must set up PFM - Agent for Oracle version 11-00 or later on the assumption that Oracle Client 64-bit is used.

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
`oracle_home`	Specify the Oracle home folder for Oracle Client 32-bit.	Specify the Oracle home folder for Oracle Database.
`oracle_version`	Specify the version of Oracle Client 32-bit.	Specify the version of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.
`sqlnet`	Specify `Y`	Specify a value according to the conditions described in *Table 2-6*.
`net_service_name`	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.	If you specify `Y` for `sqlnet`, specify this item.Specify the name of the net service that you set in the Oracle database to be monitored and that can connect to the Oracle database.

Notes:

- PFM - Agent for Oracle 11-00 or later uses the Oracle Client 64-bit library included with the Oracle database, so you do not need to install the 64-bit Oracle Client.
- To upgrade PFM - Agent for Oracle 10-50 or earlier to 11-00 or later, instance information is subject to change, and must therefore be updated before starting the PFM - Agent for Oracle service.

  Note that Oracle Client 32-bit is no longer needed, and can be uninstalled if it is not used by any product other than PFM - Agent for Oracle.
- If you specify `ORACLE_HOME` for Oracle Client 32-bit and start PFM - Agent for Oracle, the KAVF12020-E message appears.

#2

PFM - Agent for Oracle runs using Oracle password authentication.

#3

If the expiration date is set on `oracle_passwd`, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors, perform either of the following procedures before the password is expired:

- Unset the expiration date of the password
- After updating password, execute the `jpcconf inst setup` command to update `oracle_passwd`.
- Note that the Oracle default profile is applied to the Oracle account created by `mk_user.sql`.

#4

Specify whether to use Oracle network services.

- If `Y` is specified:

  PFM - Agent for Oracle connects to Oracle via a listener that is made up of Oracle network services.

  In this case, you must set the Oracle network service definitions (such as tnsnames.ora and listener.ora).

  When monitoring Oracle instances in an Oracle RAC configuration, set up the PFM - Agent for Oracle so that it monitors Oracle instances on each node. For details about how to set up, see the Oracle documentation.

When using a CDB configuration of Oracle Database 12c Release 2 or later, set this setting in case of monitoring each PDB. In case of monitoring the root container (CDB$ROOT), this setting is not required.

Store the tnsnames.ora file in the following directory. If you store the tnsnames.ora file in any other location, PFM - Agent for Oracle will encounter an Oracle connection error.

`oracle_home\network\admin`

- If `N` is specified:

  PFM - Agent for Oracle connects to the local database without using the Oracle network services.

#5

When PFM - Agent for Oracle is upgraded from a version earlier than 08-00, the default values are set.

#6

A maximum of 4 agent log files are collected for one instance. Before specifying the `log_size` value, make sure that the value satisfies the following condition (this condition also applies when `log_path` is set to the default):

```
Amount of free space on the drive containing the folder specified in log_p
ath (MB) > log_size * 4
```

If the free disk space is insufficient, agent log cannot be output. For details about the agent log, see *8.3 Log information*.

#7

Set the timeout value according to the time needed to collect records during heavy load (peak time).

#8

To obtain each piece of Oracle segment-related information, PFM - Agent for Oracle searches Oracle's static data dictionary views `DBA_SEGMENTS`. If a large number of segments (more than hundreds of thousands) exist for Oracle, information collection requires a significant amount of time. As such, when a large number of segments exist, and the information listed in the following table no longer needs to be collected, set the `sql_option` to `Y` during operation.

Table 2–7: Record names and the values specified for numeric_10 (setting up instance information)

Record name	PFM - View name	Value specified for numeric_10
`PD_PDTS`	`Segments`	Enabled
	`Extents`	Enabled
`PI_PIDB`	`DB Files %`	Enabled
	`Log Files %`	Enabled
	`NextAlloc Fails`	Enabled
	`Tablespaces`	Enabled
	`Rollback Segments`	Enabled
	`Rollback Segments Trans`	Enabled
	`Blocks`	Enabled
	`Segments`	Enabled
	`Extents`	Enabled
	`Free Mbytes`	Enabled
	`Overextended`	Enabled

Record name	PFM - View name	Value specified for numeric_10
PI_PIDB	High Max Extent	Enabled
	Datafiles	Enabled
	Mbytes	Enabled
	Free Extents	Enabled
	Free%	Enabled
	Free Change	Enabled
	Write%	Enabled
	Write/sec	Enabled
	Redo Files	Enabled
	Links	Enabled
	Links Logged On	Enabled
	Links In Tran	Enabled
	Links Open Cursors	Enabled
	Used Change	Enabled
	Used Mbytes	Enabled
	Rollback Segments Hit%	Enabled
	Sort Segments	Enabled
	Sorting Users	Enabled
	Physical Blocks Read	Always set to 0 because it is a delta item.
	Physical Blocks Written	Always set to 0 because it is a delta item.
	Physical Reads	Always set to 0 because it is a delta item.
	Physical Writes	Always set to 0 because it is a delta item.

#9

When displayed in PFM - Web Console, this item indicates whether the values set in each field in #8 are values collected from the Oracle Database, or fixed values.

#10

If the field format for each record is `float` or `double`, since the data is a floating-point number, it may be rounded depending on the specified value.

Example:

When `numeric_10` is set to `32767`, it may be displayed as `32760`.

#11

You can specify the `Store Version` item only when setting up a new instance environment. You cannot specify this item when updating an existing instance environment.

#12

When `localtemp_option` is set to `N`, collect free space of the locally managed temporary tablespace and information about the extents from v$temp_space_header of the dynamic performance view. The displayed values of the size of free space are the size of the unallocated space. Since the allocated space is not freed until the temporary tablespace is reconstructed or recreated, the displayed values of free space do not increase until the space is freed.

When localtemp_option is set to Y, collect free space of the locally managed temporary tablespace and information about the extents from v$sort_segment or v$temp_extent_pool of the dynamic performance view. The displayed values of the size of free space are calculated from the size of the used space.

When issuing query to v$temp_extent_pool view, the Oracle instance goes to sleep. Since this may have effect on the performance of the Oracle instance, you need adequate consideration before localtemp_option is set to Y. For details, see your Oracle documentation.

The following records use v$temp_extent_pool view:

- Data File (PD_PDDF)
- Data File Interval (PI_PIDF)

#13

When undospace_option is set to N, the size of the unallocated space is collected as the amount of free space in the UNDO tablespace. Space in the UNDO tablespace that becomes available because its retention period has expired is treated as allocated space until it is released.

When undospace_option is set to Y, the size of the free space is collected as the amount of free space in the UNDO tablespace. Space of the UNDO tablespace that becomes available because its retention period has expired is included in the size of the free space.

The following table shows the fields whose values change depending on the specification of the undospace_option:

Table 2–8: The fields whose values change depending on the specification of the undospace_option

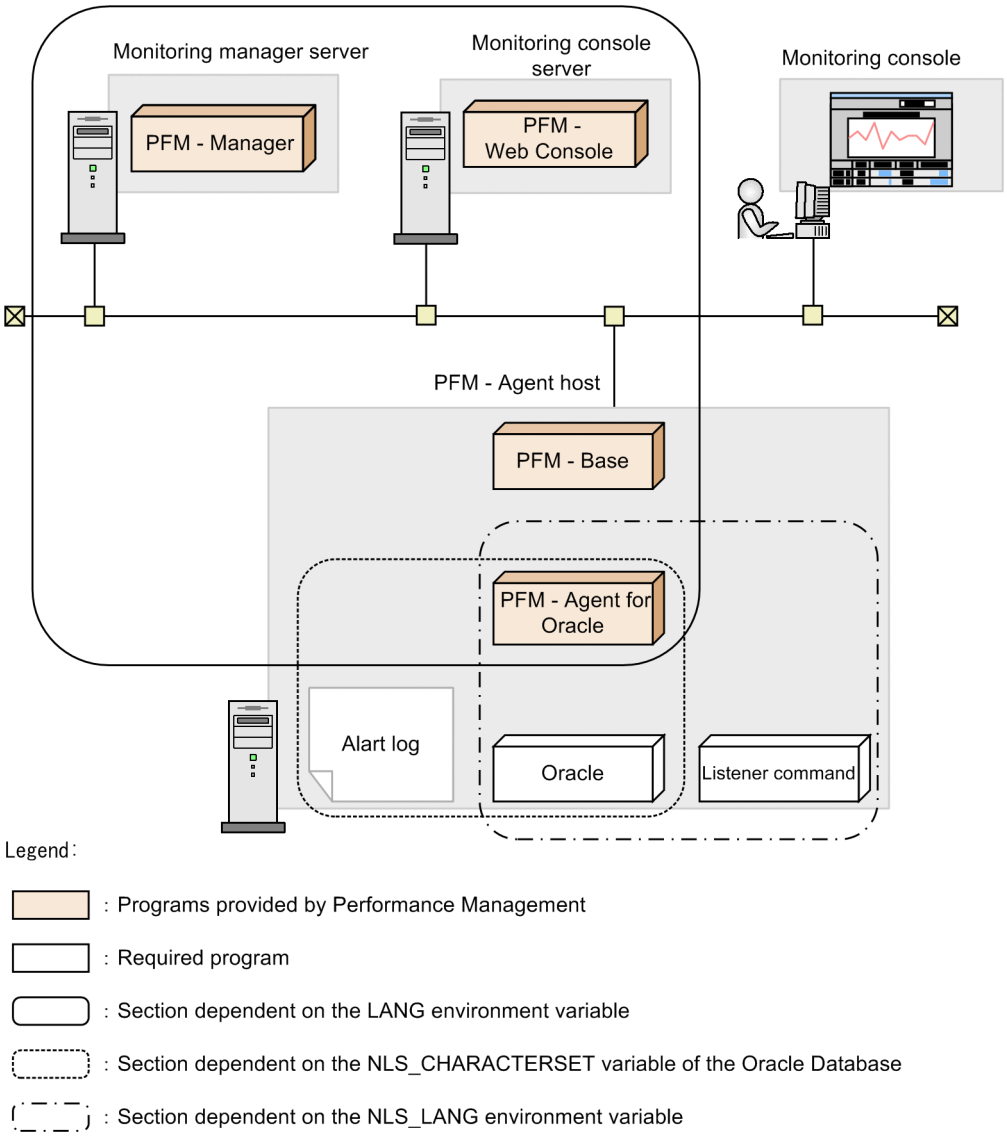
Record name	Field name
CDB Tablespace (PD_CDTS)	Free Mbytes
	Free %
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Data File (PD_PDDF)	Free %
	Free Mbytes
	Used Mbytes
Data File Interval (PI_PIDF)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Database (PD_PDDB)	Free %
	Free Mbytes
	Used Mbytes
Database Interval (PI_PIDB)	Free %
	Free Change
	Free Mbytes

Record name	Field name
Database Interval (PI_PIDB)	Used Change
	Used Mbytes
Tablespace (PD_PDTS)	Free %
	Free Mbytes
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Collection Tablespace 2 (PD_PCTS)	Free Mbytes

#14

The following figure shows the relationship among data, language environment of the OS, and instance information handled by PFM - Agent for Oracle.

Figure 2–4: Relationship between data and the setting values



PFM - Agent for Oracle can collect performance data in SJIS (in Japanese Windows) and GB18030 (in Simplified-Chinese Windows) format as well as 7-bit ASCII format.

Acceptable values for the nls_lang instance information depend on the combination of the language environment of the OS and the database character set of the monitored Oracle. The following table lists acceptable values for the nls_lang instance information for each combination.

OS language of PFM - Agent for Oracle installed	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Japanese	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
	JA16SJIS	
	other	AMERICAN_AMERICA.US7ASCII#
Simplified-Chinese	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	AL32UTF8	
	other	AMERICAN_AMERICA.US7ASCII#
other language	notdependent	AMERICAN_AMERICA.US7ASCII#

#

Performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

For any other combination, performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

When you specify an invalid character code set for the nls_lang instance information, the message KAVF12302-W with errcode 12705 is output, and the connection with Oracle will fail.

In the following cases as well, unreadable characters might occur in the performance data:

1. The Oracle column length is exceeded.

   If you store data that exceeds the Oracle column length, the last character might become unreadable. If you use PFM - Agent for Oracle to collect data in Oracle and that data contains unreadable characters, the last character of the performance data will be unreadable.

2. The field size of PFM - Agent for Oracle is exceeded.

   PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Record name	Field name	Filed size (unit: bytes)
Database Object Cache(PD_PDDO)	Object Name	100
Errorlog Detail(PD_PDEL)	Message	512
Parameter Values(PD_PDP)	Value	512
SQL Text(PD_PDSQ)	Explain Plan	30000
	SQL Text	30000
SQL Text - Performance Based(PD_PDES)	SQL Text	10000
Table Access(PD_PDTA)	Object Name	100

Notes:

- The PFM - Agent for Oracle services can be started only when an instance environment has been set up.

- When you use the `jpcconf inst setup` command to create an instance environment, the command terminates normally even if an instance name that does not exist in Oracle is specified. However, if you then start record collection, message KAVF12401-W is output to the common message log, and you cannot connect to the monitored Oracle. If this problem occurs, check whether you specified the correct instance name, and re-execute the `jpcconf inst setup` command with the correct instance name specified.
- Do not use PFM - Agent for Oracle and PFM - RM for Oracle to monitor the same Oracle instance. If so, the normal operation of PFM - Agent for Oracle is not guaranteed.

An instance environment is created by using the `jpcconf inst setup` command. The following procedure shows how to create an instance environment.

To create an instance environment:

1. Execute the `jpcconf inst setup` command with a service key and instance name specified.

   For example, when you want to create the instance environment for the PFM - Agent for Oracle instance named `SDC`, use the following command line:

   ```
   jpcconf inst setup -key Oracle -inst SDC
   ```

   Note that you cannot use `sql` as an instance name.

2. Although an example of interactive command execution is shown here, the `jpcconf inst setup` command can be also executed non-interactively. For details about the `jpcconf inst setup` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

3. Set up Oracle instance information for PFM - Agent for Oracle.

4. Enter the values for the items listed in *Table 2-6* as prompted by the command. All items are mandatory. To use the default value displayed for an item, only press the **Enter** key.

When all values have been entered, the instance environment is created. If you want to change the instance information, re-execute the `jpcconf inst setup` command to update the instance environment. For details about updating an instance environment, see *2.4.2 Updating an instance environment*.

The following describes the created instance environment.

- Folder configuration of the instance environment

  The instance environment is set up in the following folders:

  For a physical host: *installation-folder*`\agto`

  For a logical host: *environment-folder*[#]`\jp1pc\agto`

  #

  The environment folder is a folder on the shared disk specified when the logical host was created.

  The following table describes the folder configuration of the created instance environment.

  Table 2–9: Folder configuration of the instance environment

Folder and file			Description
`agent`	*instance-name*	`jpcagt.ini`	Agent Collector service startup initialization file
		`jpcagt.ini.model`[#]	Model file for the Agent Collector service startup initialization file
		`log`	Storage folder for log files

Folder and file			Description
store	*instance-name*	jpcsto.ini	Agent Store service startup initialization file
		jpcsto.ini.model#	Model file for the Agent Store service startup initialization file
		*.DAT	Data model definition file
		dump	Export destination folder
		import	Standard database import destination folder (for Store version 2.0)
		backup	Backup destination folder
		log	Storage folder for log files
		partial	Standard database partial backup destination folder (for Store version 2.0)
		STPD	Performance data storage destination folder for records of the PD record type (for Store version 2.0)
		STPI	Performance data storage destination folder for records of the PI record type (for Store version 2.0)

#

This file is used to reset all values to the initial values set when the instance environment was created.

- Service ID for the instance environment

  The service for the instance environment has the following format:

  - Agent Collector service:
  - OA*instance-number instance-name* [*host-name*]
  - Agent Store service:

    OS*instance-number instance-name* [*host-name*]

  In PFM - Agent for Oracle, the instance name specified in the `jpcconf inst setup` command is displayed.

  For example, if you execute the command with host name host1 and instance name `SDC`, the service names will be as follows:

  - Agent Collector service:

    `OA1SDC` [host1]
  - Agent Store service:

    `OS1SDC` [host1]

  For details about the service ID, see the naming rules described in Appendix in the *JP1/Performance Management Planning and Configuration Guide*.

- Windows service names in the instance environment

  The Windows service names in the instance environment are as follows:

  - Agent Collector service:
  - PFM - Agent for Oracle *instance-name* [*logical-host-name*]
  - Agent Store service:
  - PFM - Agent Store for Oracle *instance-name* [*logical-host-name*]

  For example, when the logical host name is `lhost` and the instance name is `SDC`, the service name will be:

- Agent Collector service:
  PFM - Agent for Oracle SDC [lhost]
- Agent Store service:
- PFM - Agent Store for Oracle SDC [lhost]

For details about Windows service names, see the naming rules described in Appendix in the *JP1/Performance Management Planning and Configuration Guide*.

For details about the Windows service names for logical host operation, in the *JP1/Performance Management User's Guide*, see the chapters explaining setup and operation for cluster systems.

(b) Registering objects in the Oracle Database

To use PFM - Agent for Oracle to monitor an Oracle Database, you must register the objects provided by PFM - Agent for Oracle in the Oracle Database. The objects are registered by using an SQL script provided by PFM - Agent for Oracle. The following procedure shows how to execute the SQL script. Note that the procedure is used only once for each account with which the Oracle Database instance is to be monitored.

If Oracle Database 12c Release 2 or later is to be monitored in a CDB configuration, register the objects only when instances of PFM - Agent for Oracle for monitoring PDBs have been created. If instances of PFM - Agent for Oracle for monitoring the root container (`CDB$ROOT`) have been created, there is no need to register the objects. Similarly, there is no need to register the objects when PDBs are to be monitored from the root container (`CDB$ROOT`) to which they belong, because, in this case, instances of PFM - Agent for Oracle are created for the root container (`CDB$ROOT`).

Monitoring target	Object registration necessary?
Root container (`CDB$ROOT`) Note that the monitoring target is the root container (`CDB$ROOT`) when all PDBs are monitored from the root container (`CDB$ROOT`) to which they belong.	No.
PDB	Yes. Execute `sp_inst.sql` on the PDBs to be monitored. When the `sys` account is to be used for monitoring, execute `sp_inst.sql` for each PDB to be monitored.

1. Set up an environment where the `sqlplus` Oracle command can be executed.
   For details about Oracle environment setup, see your Oracle documentation.
2. Navigate to the following directory, which contains the `sp_inst.sql` file provided by PFM - Agent for Oracle:
3. *installation-folder*`\agto\agent\sql`
4. Execute the `sp_inst.sql` script for the Oracle Database that you want to monitor.
   Connect to the Oracle Database by using the account specified by `oracle_user` in the instance information, and then execute the `sp_inst.sql` script.
5. The `sp_inst.sql` script will register with Oracle the objects (procedures for monitoring and tables for operation) PFM - Agent for Oracle needs to perform Oracle monitoring.
   Note:
   Do not use the `sp_inst_seg2.sql` script.
   Example:

If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

`sqlplus` *Oracle-account*`/`*password-for-the-Oracle-account*`@`*net-service-name-for-the-monitoring-target-database* `@sp_inst.sql`

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

`sqlplus` *Oracle-account*`/`*password-for-the-Oracle-account*`@`*net-service-name-for-the-monitoring-target-PDB* `@sp_inst.sql`

- The `sqlplus` command is provided by Oracle Corporation.
- Specify the `oracle_user` value as the Oracle account. The objects are created in the database with the Oracle account used here. You must specify the same Oracle account when setup of the instance environment is canceled.
- When you use a SYS user for the Oracle account, executing the `sp_inst.sql` script without specifying the `AS SYSDBA` option may result in an error. If an error occurs, execute the script with the `AS SYSDBA` option specified.

When the above command is executed, the table and packages shown in the following table are created.

Table 2–10: Table and packages to be created

Table	Package
`LSC_13_PLAN_TABLE`#	LSC_13_PDAS, LSC_13_PICS, LSC_13_73_PDDB, LSC_13_PDDB2, LSC_13_PDI, LSC_13_717273_PDMT, LSC_13_PDS3, LSC_13_73_PIDB, LSC_13_PIDB2, LSC_13_PIDB3

#

`LSC_13_PLAN_TABLE` is only used during collection of the SQL Text (`PD_PDSQ`) record. Therefore, when you collect the SQL Text (`PD_PDSQ`) record, make sure that at least 5 megabytes of free space is allocated to the default tablespace.

(c) Setting up the Oracle Database

To use the records provided by PFM - Agent for Oracle to collect the performance data items listed in the following table, you must set the `TIMED_STATISTICS` Oracle Database initialization parameter to `TRUE`.

Table 2–11: Items that can be collected only when TIMED_STATISTICS=TRUE is set

Record	Field	Remarks
ASM Disk (`PD_PDDK`)	Read Time (`READ_TIME`)	--
	Write Time (`WRITE_TIME`)	--
Block Contention Interval (`PI_PIBC`)	--	Entire record
Block Contention Statistics (`PD_PDBC`)	--	Entire record
Data File Interval (`PI_PIDF`)	Write Time (`WRITE_TIME`)	--
Session Detail (`PD_PDS`)	Avg Wait (`AVERAGE_WAIT`)	--
	Avg Wait String (`AVERAGE_WAIT_STRING`)	--
	Time Waited (`TIME_WAITED`)	--
	Time Waited String (`TIME_WAITED_STRING`)	--
Session Event (`PD_PDEV`)	Avg Wait (`AVERAGE_WAIT`)	--
	Avg Wait String (`AVERAGE_WAIT_STRING`)	--

Record	Field	Remarks
Session Event (PD_PDEV)	Time Waited (TIME_WAITED)	--
	Time Waited String (TIME_WAITED_STRING)	--
Session Event Interval (PI_PIEV)	Avg Wait (AVERAGE_WAIT)	--
	Avg Wait String (AVERAGE_WAIT_STRING)	--
	Time Waited (TIME_WAITED)	--
	Time Waited String (TIME_WAITED_STRING)	--
Session Stat Summary Interval (PI_PIS2)	Statement CPU (STATEMENT_CPU)	--
Session Statistics Summary (PD_PDS2)	Statement CPU (STATEMENT_CPU)	--
System Stat Summary (PD)	Session CPU Usage (SESSION_CPU_USAGE)	--
System Stat Summary Interval (PI)	Session CPU Usage (SESSION_CPU_USAGE)	--
Session Wait (PD_PDWA)	Wait Time (WAIT_TIME)	--
	Wait Time String (WAIT_TIME_STRING)	--
System Event (PD_PDSE)	Avg Wait (AVERAGE_WAIT)	--
	Time Waited (TIME_WAITED)	--
System Event Interval (PI_PISE)	Avg Wait (AVERAGE_WAIT)	--
	Time Waited (TIME_WAITED)	--

Legend:

--: None

Notes:

- If you modify the initialization parameters file, you must restart the instance's database.
- Starting from Oracle 9i, the server parameters file is supported to store Oracle parameter information.A value change you make in the server parameters file may take precedence over a change made to the initialization parameters file.
- Setting the TIMED_STATISTICS initialization parameter to TRUE may have adverse effects on the performance of the Oracle Database. If you plan to use this setting, you should first evaluate the possible effects. For details, see your Oracle documentation.

(4) Specifying network settings Option

You must specify the network settings according to the configuration in which Performance Management will be used.

The following are the two network setting items:

- IP addresses

  Set the IP addresses when using Performance Management in a network environment where multiple LANs are connected. You can set multiple IP addresses by defining the host names and IP addresses in the jpchosts file. Use the same jpchosts file throughout the Performance Management system.

  For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.
- Port numbers

Specify the port numbers to be used by Performance Management. To avoid conflicts, use the same port numbers and service names across the Performance Management system.

For details about setting port numbers, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

(5) Change the size of log files Option

Performance Management outputs its operating status to a set of proprietary log files called a *common message log*. Perform this setting if you want to change the default file size.

For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

(6) Change the storage location of performance data Option

Perform this setting if you want to change the folders where the database of performance data managed by PFM - Agent for Oracle is saved, backed up, or exported.

By default, performance data is saved in the following locations:

- Save destination folder: *installation-folder*`\agto\store\`*instance-name*`\`
- Backup destination folder: *installation-folder*`\agto\store\`*instance-name*`\backup\`
- Partial backup destination folder#: *installation-folder*`\agto\store\`*instance-name*`\partial`
- Export destination folder: *installation-folder*`\agto\store\`*instance-name*`\dump\`
- Import destination folder#: *installation-folder*`\agto\store\`*instance-name*`\import\`

Note:

For the default save destination for logical host operation, replace *installation-folder* with *environment-folder* `\jp1pc`.

#:

This folder can be set only when the Store version is 2.0.

For details, see *2.4.1 Changing the storage location of performance data*.

(7) Set the connection-target PFM - Manager of PFM - Agent for Oracle

On each host on which a PFM - Agent for Oracle program is installed, set the connection-target PFM - Manager program that is to manage the PFM - Agent for Oracle. Use the `jpcconf mgrhost define` command to set the connection-target PFM - Manager.

Notes:

- When multiple instances of PFM - Agent are installed on a single host, you can specify only one PFM - Manager as their connection target. You cannot have a different instance of PFM - Manager as the connection target for each PFM - Agent.
- When PFM - Agent for Oracle is installed on the same host as PFM - Manager, the connection-target PFM - Manager will be the PFM - Manager on the local host. You cannot change this.

To set the connection-target PFM - Manager:

1. Stop all Performance Management programs and services.

Stop all active Performance Management programs and services on the host before beginning setup. For details about how to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

If any Performance Management programs or services are running when you execute the `jpcconf mgrhost define` command, you will be prompted by a message to stop the programs or services.

2. Execute the `jpcconf mgrhost define` command with the host name of the PFM - Manager host you wish to use specified as the connection target.

   For example, if the PFM - Manager you wish to use as the connection target resides on host `host01`, execute the command as follows:

   ```
   jpcconf mgrhost define -host host01
   ```

   Although an example of interactive command execution is shown here, the `jpcconf mgrhost define` command can be also executed non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(8) Setting up the action log Option

This setting is required to output action logs when alarms occur. 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 *K. Outputting Action Log Information*.

2.2 Setup cancellation and uninstallation

This section describes the procedures for uninstalling and canceling setup of PFM - Agent for Oracle.

2.2.1 Cautionary notes on setup cancellation and uninstallation

Note the following when uninstalling and canceling setup of PFM - Agent for Oracle.

(1) Note on OS user permission required to uninstall PFM - Agent for Oracle

Before you uninstall PFM - Agent, make sure that you have logged on with an account that belongs to the Administrators group.

(2) Note on network configuration

When you uninstall a Performance Management program, the port numbers defined in the `services` file will remain in the file.

(3) Notes on programs

- If you uninstall a Performance Management program while another Performance Management program or service or other program that references Performance Management files (for example, Windows Event Viewer) is running, some files or folders may remain in the system. In this case, manually delete everything under the installation folder.
- If you uninstall a Performance Management program while another Performance Management program or service or other program that references Performance Management files (for example, Windows Event Viewer) is running, you may be prompted to restart the system. If you are prompted to restart the system, restart the system to complete the uninstallation process.
- If both PFM - Base and PFM - Agent are installed on a host, you cannot uninstall PFM - Base without first uninstalling PFM - Agent. In this case, uninstall PFM - Agent and then PFM - Base, in that order. The same applies when both PFM - Manager and PFM - Agent are installed on a host. You will be unable to uninstall PFM - Manager without first uninstalling PFM - Agent. In this case, uninstall PFM - Agent and then PFM - Manager, in that order.

(4) Notes on services

- Uninstalling PFM - Agent does not delete the information about the service from the list that appears when you execute the `jpctool service list` command. To delete this information, use the `jpctool service delete` command.
- For details about deleting the information about the service, see the section about deleting the service in the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Note that when you want to update the PFM - Web Console host to reflect the deletion of service information, you need to execute the `jpctool service sync` command to synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host.

(5) Other notes

- When uninstalling a Performance Management program from a host on which PFM - Web Console is installed, close all browser windows before you uninstall the program.

- Before you start uninstallation, use the `jpcconf inst setup` command or PFM - Web Console to check the agent log output folder. If this folder has been changed from the default folder (*installation-folder*`\agto\store\`*instance-name*`\log\`), the agent log files remain after uninstallation. Manually delete these files after uninstallation.

2.2.2 Procedure for canceling setup

This subsection describes how to cancel setup of PFM - Agent for Oracle.

(1) Canceling setup of an instance environment

Canceling setup of an instance environment involves the tasks listed below. To cancel setup of multiple instance environments, you must repeat the procedure for each environment.

- Deleting an instance environment
- Deleting the objects registered in the Oracle Database

The following describes the above tasks.

(a) Deleting an instance environment

Check the instance name and delete the instance environment. Deletion of an instance environment is performed from a PFM - Agent host.

To check the instance name, use the `jpcconf inst list` command. To delete an instance environment that has been created, use the `jpcconf inst unsetup` command.

To delete an instance environment:

1. Check the instance name.

   Execute the `jpcconf inst list` command specified with the service key that indicates PFM - Agent for Oracle. The following shows the command format:

   ```
   jpcconf inst list -key Oracle
   ```

   For example, if the instance name is `SDC`, the command displays `SDC`.

2. Stop all active PFM - Agent services in the instance environment.

   For details about how to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Delete the instance environment.

   Execute the `jpcconf inst unsetup` command specified with the service key that indicates PFM - Agent for Oracle and the instance name.

   For example, if the instance name is `SDC`, use the following command line:

   ```
   jpcconf inst unsetup -key Oracle -inst SDC
   ```

   If execution is successful, the folders created as the instance environment are deleted, as well as the service IDs and Windows services.

Note:

Canceling setup of an instance environment does not delete the service information that is displayed with the `jpctool service list` command. Use the `jpctool service delete` command to delete service information.

If you want to update the PFM - Web Console host to reflect the deletion of instance environment, you need to execute the `jpctool service sync` command to synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host.

The following shows sample conditions.

- Instance name: `SDC`
- Host name: `host03`
- Service ID of the Agent Collector service: `OA1SDC[host03]`
- Service ID of the Agent Store service: `OS1SDC[host03]`

```
jpctool service delete -id O?1SDC[host03] -host host03
```

For details about the command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(b) Deleting the objects registered in the Oracle Database

This subsection describes the procedure for deleting the table and packages that were created in the Oracle Database being monitored. To execute this procedure, you must use the same Oracle account that you used when you registered the objects in the Oracle Database. Note that this procedure must be used only once for each account that is used to monitor the Oracle Database instance.

To delete the objects registered in the Oracle Database:

1. Set up an environment where the Oracle `sqlplus` command can be executed.

   For details about Oracle environment setup, see your Oracle documentation.

2. Navigate to the following folder that contains the `sp_drop.sql` script provided by the PFM - Agent for Oracle:

   ```
   installation-folder\agto\agent\sql
   ```

3. Execute the `sp_drop.sql` script on the Oracle Database being monitored.

   PFM - Agent for Oracle deletes monitoring procedures, work tables, and other objects that are required to monitor Oracle from Oracle.

   Example:

   If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

   `sqlplus` *Oracle-account*`/`*password-for-the-Oracle-account*`@`*net-service-name-of-the-monitoring-target-database* `@sp_drop.sql`

   If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

   `sqlplus` *Oracle-account*`/`*password-for-the-Oracle-account*`@`*net-service-name-for-the-monitoring-target-PDB* `@sp_drop.sql`

   - `sqlplus` is a command provided by Oracle Corporation.
   - *Oracle-account* is the same Oracle account that was used to register the objects in the database.

For Oracle 10g or later, LSC_13_PLAN_TABLE is placed in DBA_RECYCLEBIN and is not deleted completely. If you want to delete LSC_13_PLAN_TABLE completely, execute the `PURGE TABLE LSC_13_PLAN_TABLE`; command.

Note that if the Oracle account is `sys`, `LSC_13_PLAN_TABLE` is not stored in `DBA_RECYCLEBIN`. Therefore, you do not need to execute the `PURGE TABLE LSC_13_PLAN_TABLE;` command.

4. Reset the value of the `TIMED_STATISTICS` Oracle initialization parameter.

   If the value of the `TIMED_STATISTICS` Oracle initialization parameter has been changed in order to collect records of PFM - Agent for Oracle, reset the value, if necessary.

(2) Deleting an Oracle account used in PFM - Agent for Oracle

Oracle accounts used in PFM - Agent for Oracle are authorized to change the objects of other schemas freely in order to monitor the Oracle Database. For this reason, unnecessary Oracle accounts must be deleted. If the tablespaces that were used by a deleted account are unnecessary, also delete the tablespaces.

(a) Deleting an Oracle account

To delete an Oracle account, issue the `DROP USER` statement in an environment where the `sqlplus` Oracle command can be executed. Before issuing the statement, make sure that your Oracle account has the `DROP USER` system privilege.

To delete an Oracle account:

1. Issue the `DROP USER` statement.

   Example:

   ```
   DROP USER Oracle-account CASCADE;
   ```

   If you add the `CASCADE` option, you can also delete the objects owned by the account.

For details about the `DROP USER` statement, see your Oracle documentation.

(b) Deleting the tablespaces used by a deleted Oracle account

When an Oracle account is deleted, the tablespaces used by the Oracle account become unnecessary. To delete these tablespaces, issue the `DROP TABLESPACE` statement in an environment where the `sqlplus` Oracle command can be executed. Before issuing the statement, make sure that your Oracle account has the `DROP TABLESPACE` system privilege.

To delete tablespaces:

1. Issue the `DROP TABLESPACE` statement.

For details about the `DROP TABLESPACE` statement, see your Oracle documentation.

2.2.3 Procedure for uninstallation

To uninstall PFM - Agent for Oracle:

1. On the host from which you want to uninstall PFM - Agent for Oracle, log on as a member of the Administrators group.

2. Stop all Performance Management programs and services on the local host.

Display the service information and check whether any services are running. For details about how to display service information and to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

Stop all Performance Management programs and services running on the local host. This includes services running on physical and logical hosts.

3. Select the Performance Management program you want to uninstall.

   In Windows **Control Panel**, choose **Programs and Features**#, and then select the Performance Management program you want to uninstall.

   #:

   This name might differ depending on the Windows version.

4. Click **Remove**, and then click **OK**.

   The program you selected is uninstalled.

Precaution:

If user account control (UAC) functionality is enabled on the OS, the User Account Control dialog box might be displayed during uninstallation. If this dialog box is displayed, click the **Continue** button to continue uninstallation. If you click the **Cancel** button, the uninstallation is canceled.

2.3 Changing the system configuration of PFM - Agent for Oracle

You may need to change the PFM - Agent for Oracle system configuration because of a change in the network configuration or host name of the monitored system.

When you change the PFM - Agent for Oracle system configuration, you must also change the settings for PFM - Manager and PFM - Web Console. For details about how to change the Performance Management system configuration, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

For some kinds of PFM - Agent, changing the physical host name or alias name requires additional tasks specific to the PFM - Agent. However, PFM - Agent for Oracle does not require such additional specific tasks.

2.4 Changing the operation of PFM - Agent for Oracle

In some circumstances, such as when changes are made to the way in which collected operation monitoring data is utilized, you may need to change how PFM - Agent for Oracle operates.

For details about changing the operation method across the entire Performance Management system, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

2.4.1 Changing the storage location of performance data

Performance data collected by PFM - Agent for Oracle is managed in the Store database of the Remote Monitor Store service of PFM - Agent for Oracle. 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 folders to manage the collected performance data. These folders can be changed by using the `jpcconf db define` command.

Use the `jpcconf db define` command with -move option if you want to copy the performance data to the new storage location of the Store database.

For details about the `jpcconf db define` command, see the manual *JP1/Performance Management Reference*.

- Save destination folder
- Backup destination folder
- Partial backup destination folder#
- Export destination folder
- Import destination folder#

#

This folder 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 2–12: Options of the command that changes the performance data storage location

Description	Label name	Specifiable values (Store 1.0)	Specifiable values (Store 2.0)	Default value#
Save destination folder	`sd`	1 to 127 bytes absolute path name	1 to 214 bytes absolute path name	*installation-folder*`\agto\store\`*instance-name*
Backup destination folder	`bd`	1 to 127 bytes absolute path name	1 to 211 bytes absolute path name	*installation-folder*`\agto\store\`*instance-name*`\backup`
Partial backup destination folder	`pbd`	--	1 to 214 bytes absolute path name	*installation-folder*`\agto\store\`*instance-name*`\partial`

Description	Label name	Specifiable values (Store 1.0)	Specifiable values (Store 2.0)	Default value#
Maximum backup generation number	bs	1 to 9	1 to 9	5
Export destination folder	dd	1 to 127 bytes absolute path name	1 to 127 bytes absolute path name	*installation-folder*\agto\store*instance-name*\dump
Import destination folder	id	--	1 to 222 bytes absolute path name	*installation-folder*\agto\store*instance-name*\import

Legend:

--: This item cannot be set.

#

For the default save destination for logical host operation, replace *installation-folder* with *environment-folder* \jp1pc.

(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) Setting items in jpcsto.ini

The following table lists the label names for which information can be edited in the jpcsto.ini file, and other information such as the values that can be specified.

Table 2–13: Performance-data storage location settings (under [Data Section] in jpcsto.ini)

Description	Label name	Specifiable values (Store 1.0) #1	Default value#2
Save destination folder	Store Dir#3	1 to 127 bytes absolute path name	*installation-folder*\agto\store*instance-name*
Backup destination folder	Backup Dir#3	1 to 127 bytes absolute path name	*installation-folder*\agto\store*instance-name*\backup
Maximum backup generation number	Backup Save	1 to 9	5
Export destination folder	Dump Dir#3	1 to 127 bytes absolute path name	*installation-folder*\agto\store*instance-name*\dump

#1

- The folder name must be an absolute path name or a relative path name from the default Store database folder (*installation-folder*\agto\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

For the default save destination for logical host operation, replace *installation-folder* with *environment-folder* \jp1pc.

#3

You cannot specify the same folders for Store Dir, Backup Dir, and Dump Dir.

(b) Before editing the jpcsto.ini file

- When changing the Store database folder, make sure that the folder after the change has already been created.
- When the Store database folder is changed, performance data collected before the change can no longer be used. If the performance data collected before the change is needed, carry 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 folder 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 folder.

(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 folder 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
        :
```

   Notes:
   - 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 folder (*installation-folder*\agto\store*instance-name*) for the Store database of the Agent Store service. To change the storage destination, specify a relative path from the storage destination folder, or specify an absolute path.

- The jpcsto.ini file also describes definition information in addition to the database storage destination folder. Therefore, do not change the values other than those in the [Data Section] section. If you change the values other than those in the [Data Section] section, Performance Management might 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 folder, the performance data files are not deleted from the previous folder. If these files are no longer necessary, delete only the following files:

- All files with the .DB extension
- All files with the .IDX extension

2.4.2 Updating an instance environment

To update an instance environment, check the name of the instance that you want to update, and change the instance information. The instance information is set from a PFM - Agent host.

Before you change an information item, check the following table. For details about Oracle instance information, see your Oracle documentation.

Table 2–14: PFM - Agent for Oracle instance information

Item	Description	Specifiable value	Default
`oracle_sid`	The value of this item can be updated. ID of the monitored Oracle system (same value as the `ORACLE_SID` environment variable).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value
`oracle_home`[#1]	The value of this item can be updated. Oracle home folder (same value as the `ORACLE_HOME` environment variable).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value
`oracle_version`[#1]	The value of this item can be updated. Version number of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.	A two-digit number. • Oracle 11g: 11 • Oracle 12c or later: 12	Previous value
`oracle_user`[#2]	The value of this item can be updated. Account for monitoring Oracle (for details about accounts that can be specified, and the required privileges, see *2.1.4(2)*	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value

Item	Description	Specifiable value	Default
oracle_user[#2]	*Create an Oracle account to be used in PFM - Agent for Oracle*).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
oracle_passwd[#3]	The value of this item can be updated. Specify the password for the account specified in oracle_user.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
sqlnet[#1, #4]	The value of this item can be updated. Specify Y for any of the following four cases: 1. If using an Oracle RAC configurationFor details on the RAC configuration, see the Oracle documentation. 2. If using a PD_PDIA record to monitor the availability of a listener 3. If any Oracle service is running on an account other than the local system account. 4. If the monitoring target is PDB in a CDB configuration of Oracle Database 12c R2 or later. If you specify N for the case 1, 3 or 4, above, Oracle might encounter an error.	{ Y | N }	Previous value
net_service_name[#1]	The value of this item can be updated. Net service name of the monitored database. This value is enabled if you specified Y in sqlnet. For details about the net service name of the monitored database, see the Oracle documentation.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
listener_home[#1]	The value of this item can be updated. Specify the environment variable ORACLE_HOME of the Oracle component containing the listener you want to monitor.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value

Item	Description	Specifiable value	Default
listener_home[#1]	Specify environment variable GRID_HOME in the case of Oracle RAC environment.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
listener_name	The value of this item can be updated. Specify the name of one listener that you want to monitor using the PDLS record. If a PDLS record is not being used to monitor the listener, specify the default listener name "LISTENER". This is because even though the specified value is not used, a blank is not permitted.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
retry_time	The value of this item can be updated. The number of seconds for which reconnection is attempted when an authentication error is output during Oracle connection. If an authentication error occurs after the specified period has passed, the PFM - Agent for Oracle service stops. When the value is 0, the PFM - Agent for Oracle service stops without retrying to re-establish connection in the event of an authentication error. This item is enabled when startup_always is N. The specification of this item is ignored when startup_always is Y.	0 to 600 (in seconds).	Previous value
log_path	The value of this item can be updated. Specify the absolute path name of the agent log output folder.	A character string of 245 or fewer bytes that does not include the following characters: • Tabs • The following symbols: / : , ; * ? " < > | Notes: • You can specify the path to a folder under the installation folder only when the default folder is set. • You cannot specify the path to a folder that is used as the output destination of another instance.	Previous value
log_size	The value of this item can be updated.	1 to 32 (in kilobytes). 16 or a greater value is recommended.	Previous value

Item	Description	Specifiable value	Default
log_size	Specify the maximum size of each agent log file.	1 to 32 (in kilobytes). 16 or a greater value is recommended.	Previous value
timeout	The value of this item can be updated. The timeout period for Oracle access during a query.	0, or 10 to 3600 (in seconds). When 0 is specified, timeout monitoring is not performed. When a value from 1 to 9 is specified, it is changed to 10 at runtime.	Previous value
sql_option[#5]	The value of this item can be updated. When Y is specified, information about the following items[#5] is not collected for PI_PIDB and PD_PDTS records, and the value specified by 0 or numeric_10 is set.	{ Y | N }	Previous value
numeric_10	The value of this item can be updated. When sql_option is set to Y, the value specified is set for items for which information is not collected. If sql_option is set to N, this specification is disregarded.	0 to 99999. Note that if the value specified exceeds the maximum value for the data format of the set field (32767 for short and 65535 for ushort), the maximum value for the data format is set.[#6]	Previous value
startup_always	The value of this item can be updated. PFM - Agent for Oracle may stop due to an Oracle connection error, such as when the monitored Oracle is still starting up when PFM - Agent for Oracle starts up. If Y is specified, start processing continues even if a connection error occurs. If N is specified, start processing will stop if an error occurs.	{ Y | N }	Previous value
localtemp_option[#7]	The value of this item can be updated. Option for switching the display of the free space of the locally managed temporary tablespace of PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTF, PD_PDTS, and PD_PCTS records. If Y is specified, display the size of the free space. If N is specified, display the size of the unallocated space.	{ Y | N }	Previous value

Item	Description	Specifiable value	Default
undospace_option[#8]	The value of this item can be updated. Option for switching how the value displayed as the free space in the UNDO tablespace of the PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTS, PD_PCTS, and PD_CDTS records is determined. If N is specified, the size of the unallocated space is displayed. If Y is specified, the size of the free space is displayed.	{ Y | N }	Previous value
nls_lang[#9]	The value of this item can be updated. Option for specifying the character encoding used for communication between PFM - Agent for Oracle and Oracle Database.	Character code set: • In Japanese Windows: {AMERICAN_AMERICA.US7ASCII | AMERICAN_AMERICA.JA16SJISTILDE} • In Simplified-Chinese Windows: {AMERICAN_AMERICA.US7ASCII | AMERICAN_AMERICA.ZHS32GB18030} • Other case: AMERICAN_AMERICA.US7ASCII	Previous value

Legend:

--: None

#1

For monitoring of the Oracle database, the 32-bit Oracle Client has been the required product in PFM - Agent for Oracle 10-50 or earlier. However, in PFM - Agent for Oracle 11-00 or later, the Oracle Client 64-bit library is used for monitoring.

The setting method for version 10-50 or earlier cannot be used to connect to an Oracle Database. Therefore, you must set up PFM - Agent for Oracle version 11-00 or later on the assumption that Oracle Client 64-bit is used.

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
oracle_home	Specify the Oracle home folder for Oracle Client 32-bit.	Specify the Oracle home folder for Oracle Database.
oracle_version	Specify the version of Oracle Client 32-bit.	Specify the version of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), 12 is set.
sqlnet	Specify Y	Specify a value according to the conditions described in *Table 2-14*.
net_service_name	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.	If you specify Y for sqlnet, specify this item. Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Database.

Notes:

- PFM - Agent for Oracle 11-00 or later uses the Oracle Client 64-bit library included with the Oracle database, so you do not need to install the 64-bit Oracle Client.
- To upgrade PFM - Agent for Oracle 10-50 or earlier to 11-00 or later, instance information is subject to change, and must therefore be updated before starting the PFM - Agent for Oracle service.

  Note that Oracle Client 32-bit is no longer needed, and can be uninstalled if it is not used by any product other than PFM - Agent for Oracle.
- If you specify `ORACLE_HOME` for Oracle Client 32-bit and start PFM - Agent for Oracle, the KAVF12020-E message appears.

#2

To change an account for monitoring Oracle, use the following procedure:

1. Delete the objects created by the account you want to change.
2. Register new objects as the new account.

Performance data is not deleted when a account is changed.

For details about how to delete objects, see *2.2.2(1)(b) Deleting the objects registered in the Oracle Database*. For details about how to register objects, see *2.1.4(3)(b) Registering objects in the Oracle Database*.

#3

If the expiration date is set on `oracle_passwd`, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors, perform either of the following procedures before the password is expired:

- Unset the expiration date of the `password`
- After updating password, execute the `jpcconf inst setup` command to update `oracle_passwd`.

Note that the Oracle DEFAULT profile is applied to the Oracle account created by `mk_user.sql`.

#4

Specify whether to use Oracle network services.

- If `Y` is specified:

  PFM - Agent for Oracle connects to Oracle via a listener that is made up of Oracle network services.

  In this case, you must set the Oracle network service definitions (such as tnsnames.ora and listener.ora).

  When monitoring Oracle Database instances in an Oracle RAC configuration, set up the PFM - Agent for Oracle so that it monitors Oracle Database instances on each node.

  When using a CDB configuration of Oracle Database 12c Release 2 or later, set this setting in case of monitoring each PDB. In case of monitoring the root container (CDB$ROOT), this setting is not required.

  For details about how to set up, see the Oracle documentation.

  Store the tnsnames.ora file in the following directory.

`oracle_home\network\admin`

If tnsnames.ora is located on other folder, PFM - Agent for Oracle cannot connect to Oracle.

- If `N` is specified:

  PFM - Agent for Oracle connects to the local database without using the Oracle network services.

#5

To obtain each piece of Oracle segment-related information, PFM - Agent for Oracle searches Oracle's static data dictionary views `DBA_SEGMENTS`. If a large number of segments (more than hundreds of thousands) exist for Oracle, information collection requires a significant amount of time. As such, when a large number of segments

exist, and the information listed in the following table no longer needs to be collected, set the `sql_option` to `Y` during operation.

Table 2–15: Record names and the values specified for numeric_10 (updating instance information)

Record Name	PFM - View name	Value specified for numeric_10
PD_PDTS	Segments	Enabled
	Extents	Enabled
PI_PIDB	DB Files %	Enabled
	Log Files %	Enabled
	NextAlloc Fails	Enabled
	Tablespaces	Enabled
	Rollback Segments	Enabled
	Rollback Segments Trans	Enabled
	Blocks	Enabled
	Segments	Enabled
	Extents	Enabled
	Free Mbytes	Enabled
	Overextended	Enabled
	High Max Extent	Enabled
	Datafiles	Enabled
	Mbytes	Enabled
	Free Extents	Enabled
	Free%	Enabled
	Free Change	Enabled
	Write%	Enabled
	Write/sec	Enabled
	Redo Files	Enabled
	Links	Enabled
	Links Logged On	Enabled
	Links In Tran	Enabled
	Links Open Cursors	Enabled
	Used Change	Enabled
	Used Mbytes	Enabled
	Rollback Segments Hit%	Enabled
	Sort Segments	Enabled
	Sorting Users	Enabled
	Physical Blocks Read	Always set to 0 because it is a delta item.

Record Name	PFM - View name	Value specified for numeric_10
PI_PIDB	Physical Blocks Written	Always set to 0 because it is a delta item.
	Physical Reads	Always set to 0 because it is a delta item.
	Physical Writes	Always set to 0 because it is a delta item.

#6

If the field format for each record is float or double, since the data is a floating-point number, it may be rounded depending on the specified value.

Example:

When numeric_10 is set to 32767, it may be displayed as 32760.

#7

When localtemp_option is set to N, collect free space of the locally managed temporary tablespace and information about the extents from v$temp_space_header of the dynamic performance view. The displayed values of the size of free space are the size of the unallocated space. Since the allocated space is not freed until the temporary tablespace is reconstructed or recreated, the displayed values of free space do not increase until the space is freed.

When localtemp_option is set to Y, collect free space of the locally managed temporary tablespace and information about the extents from v$sort_segment or v$temp_extent_pool of the dynamic performance view. The displayed values of the size of free space are calculated from the size of the used space.

When issuing query to v$temp_extent_pool view, the Oracle instance goes to sleep. Since this may have effect on the performance of the Oracle instance, you need adequate consideration before localtemp_option is set to Y. For details, see your Oracle documentation.

The following records use v$temp_extent_pool view:

- Data File (PD_PDDF)
- Data File Interval (PI_PIDF)

#8

When undospace_option is set to N, the size of the unallocated space is collected as the amount of free space in the UNDO tablespace. Space in the UNDO tablespace that becomes available because its retention period has expired is treated as allocated space until it is released.

When undospace_option is set to Y, the size of the free space is collected as the amount of free space in the UNDO tablespace. Space of the UNDO tablespace that becomes available because its retention period has expired is included in the size of the free space.

The following table shows the fields whose values change depending on the specification of the undospace_option:

Table 2–16: The fields whose values change depending on the specification of the undospace_option

Record name	Field name
CDB Tablespace (PD_CDTS)	Free Mbytes
	Free %
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Data File (PD_PDDF)	Free %

Record name	Field name
Data File (PD_PDDF)	Free Mbytes
	Used Mbytes
Data File Interval (PI_PIDF)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Database (PD_PDDB)	Free %
	Free Mbytes
	Used Mbytes
Database Interval (PI_PIDB)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Tablespace (PD_PDTS)	Free %
	Free Mbytes
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Collection Tablespace 2 (PD_PCTS)	Free Mbytes

Use the `jpcconf inst list` command to check the instance name. To update an instance environment, use the `jpcconf inst setup` command.

Updating an instance environment involves the steps described below. To update multiple instance environments, repeat the procedure for each instance environment.

To update an instance environment:

1. Find the instance name.
   Execute the `jpcconf inst list` command specified with the service key that indicates PFM - Agent for Oracle.

   ```
   jpcconf inst list -key Oracle
   ```

   If the specified instance name is `SDC`, the command displays `SDC`.

2. If the PFM - Agent for Oracle service is active in the instance environment that is to be updated, stop the service.
   For details about stopping services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

If the service is still active in the instance environment that is to be updated when you execute the `jpcconf inst setup` command, a confirmation message is displayed to enable you to stop the service. If you stop the service, update processing resumes; if you do not stop the service, update processing is canceled.

3. Execute the `jpcconf inst setup` command specified with the service key that indicates PFM - Agent for Oracle and the instance name.

   For example, if you are updating the instance environment for the PFM - Agent for Oracle with instance name `SDC`, execute the following command:

   ```
   jpcconf inst setup -key Oracle -inst SDC
   ```

4. Update the instance information for Oracle.

   Enter the information shown in *Table 2-14* in accordance with the command's instructions. The current settings are displayed (except for the value of `oracle_passwd`). To use the displayed value, press the **Enter** key. When you have finished entering information, the instance environment is updated.

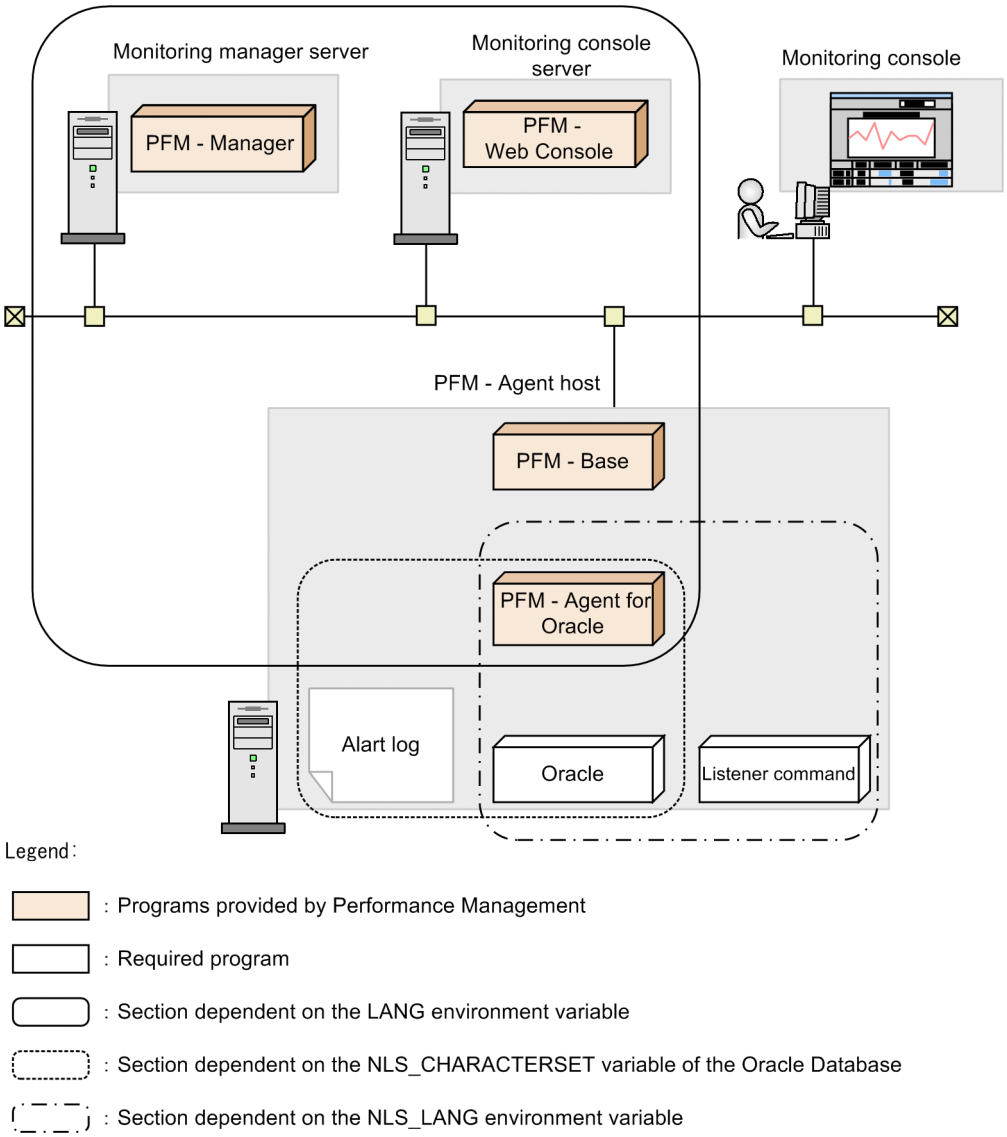
5. Restart the service in the updated instance environment.

   For details about starting services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

#9

The following figure shows the relationship among data, language environment of the OS, and instance information handled by PFM - Agent for Oracle.

Figure 2–5: Relationship between data and the setting values



PFM - Agent for Oracle can collect performance data in SJIS (in Japanese Windows) and GB18030 (in Simplified-Chinese Windows) format as well as 7-bit ASCII format.

Acceptable values for the `nls_lang` instance information depend on the combination of the language environment of the OS and the database character set of the monitored Oracle. The following table lists acceptable values for the `nls_lang` instance information for each combination.

OS language of PFM - Agent for Oracle installed	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Japanese	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
	JA16SJIS	
	other	AMERICAN_AMERICA.US7ASCII#
Simplified-Chinese	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	AL32UTF8	
	other	AMERICAN_AMERICA.US7ASCII#

OS language of PFM - Agent for Oracle installed	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
other language	notdependent	AMERICAN_AMERICA.US7ASCII#

#

Performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

For any other combination, performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

When you specify an invalid character code set for the `nls_lang` instance information, the message `KAVF12302-W` with `errcode 12705` is output, and the connection with Oracle will fail.

In the following cases as well, unreadable characters might occur in the performance data:

1. The Oracle column length is exceeded.
2. If you store data that exceeds the Oracle column length, the last character might become unreadable. If you use PFM - Agent for Oracle to collect data in Oracle and that data contains unreadable characters, the last character of the performance data will be unreadable.
3. The field size of PFM - Agent for Oracle is exceeded.

   PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Record name	Field name	Filed size (unit: bytes)
Database Object Cache(PD_PDDO)	Object Name	100
Errorlog Detail(PD_PDEL)	Message	512
Parameter Values(PD_PDP)	Value	512
SQL Text(PD_PDSQ)	Explain Plan	30000
	SQL Text	30000
SQL Text - Performance Based(PD_PDES)	SQL Text	10000
Table Access(PD_PDTA)	Object Name	100

Note:

If you want to change an item that cannot be updated, delete the instance environment and then re-create it.

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

2.4.3 Cancellation facility for Oracle access during record collection

The maximum time for accessing Oracle can be set as a timeout value, for the time it takes to collect 1 record.

When record data is collected while Oracle and the machine are experiencing heavy load, it may take a significant amount of time to perform record collection, depending on the amount of data collected. In this case, PFM - Agent for Oracle requests may impact Oracle operation. As such, a timeout value can be set to cancel requests from PFM - Agent for Oracle to Oracle, to prevent impact on Oracle operation.

Record collection is performed in the following order for each record:

1. Oracle is accessed

2. Data is written to the Store database

However, when a timeout value is set, and a timeout occurs during Oracle access, collection for that record is canceled.

This facility is implemented using the `OCIBreak` function of the Oracle OCI (Oracle Call Interface).

The timeout value can be set as follows:

- During instance environment setup by using the `jpcconf inst setup` command
- By changing the `TIMEOUT` property for the Agent Collector service, in the PFM - Web Console GUI

The values that can be set are `0`, or anything from 10 to 3,600 (in seconds). If `0` is specified, this facility is not used. Out-of-range values will be disregarded. `0` is set as the default.

The following table lists the values that can be entered as timeout values, for each setting method.

Table 2–17: Possible timeout values

Setting method	Value input				
	-1 or less	0	1 to 9	10 to 3,600	3,601 or more
Setting/update from the `jpcconf inst setup` command	Input error (cannot be input)	Yes	Yes, but replaced on restart	Yes	Input error (cannot be input)
Change from PFM - Web Console	Yes, but not updated	Yes	Yes, but not updated	Yes	Yes, but not updated

Legend:

Yes: Can be input.

Yes, but replaced on restart: Can be input, but replaced with `10` when PFM - Agent for Oracle is restarted. A `KAVF12630-W` message is output to the common message log.

Yes, but not updated: Can be input, but cannot be updated. A `KAVF12630-W` message is output to the common message log.

> **Important**
>
> Set the timeout value according to the time needed to collect records during heavy load (peak time).

For details about the timeout values set by the `jpcconf inst setup` command, see *2.1.4(3) Set up an instance environment*.

Note that the following records are not subject to cancellation:

- Instance Availability (`PD_PDIA`)
- Server Status (`PD_STAT`)
- SQL*Net Listener (`PD_PDNL`)
- SQL*Net Listeners (`PD_PDLS`)

When a timeout occurs, the following message is output to the common log (`agtoinf0x.log`) of the agent log.

```
KAVF12636-I
The cancellation of the record collection (record-name) by the time-out was
accepted.
```

Note that when records are not collected due to cancellation, a KAVF12401-W message is output to the common message log.

When a collection of multiple records, such as historical data, is performed at the same time, even if a timeout occurs for a single record, collection of other records is not performed.

Important

Since PFM - Agent for Oracle uses the OCI (Oracle Call Interface), the time required for actual cancellation depends on the processing time for OCIBreak(). As such, cancellation may not happen instantly.

In the following cases, a record collection is not canceled even when a cancellation request occurs due to a timeout:

- When a timeout occurs while a collection sequence is being moved to be written to the Store database.
- Cancellation occurs due to timeout while Oracle access is terminating, in which case the KAVF12636-I message is output to the log file, but record collection is performed normally.

2.4.4 Updating the Store version to 2.0

The Store database comes in two versions, 1.0 and 2.0. 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 Oracle version 08-10 or later in an environment with PFM - Base or PFM - Manager version 08-10 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 describes the installation conditions, whether Store 2.0 is available for the given conditions, and the procedure for enabling Store 2.0.

Table 2–18: 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		
08-00 or earlier	Overwrite installation	Not supported	Upgrade PFM - Base or PFM - Manager to version 08-10 and then execute the setup command.
	New installation		
08-11 or later	Overwrite installation	An existing instance is supported after setup	Execute the setup command.
		A new instance is supported	Use the jpcconf inst setup command to set up when the instance is configured.

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		
08-11 or later	New installation	Supported	Use the jpcconf inst setup command to set up when the instance is configured.

(1) Setup of Store 2.0

This subsection describes how to set up Store version 2.0 when you update the Store database.

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 folder settings.

   When the Store version is updated to 2.0, the Agent Store service might fail to start with the same folder settings that were available in Store 1.0. For this reason, you must review the settings of the folders used by the Agent Store service. You can use the jpcconf db define command to view and change the settings of these folders.

   The maximum length of the names of the save destination folder, backup destination folder, and other folders used by the Store database in Store 2.0 differs from the maximum length in Store 1.0. In particular, be careful when the folder 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 folder names in Store 2.0. If the absolute path name is longer than the maximum, change the settings of each folder 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
   ```

   To update the Store version to 2.0, execute the `jpcconf db vrset -ver 2.0` command. You must execute this command for each Agent instance.

   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

This subsection provides notes about updating.

(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 to Store version 1.0, use the `jpctool db dump` command to output Store version 2.0 information.

2.5 Backup and restoration

This section explains how to back up and restore PFM - Agent for Oracle.

In preparation for the system failure due to errors, back up the configuration files. When you change the system configuration (such as setting up PFM - Agent for Oracle), back up the configuration files.

For details about how to back up and restore the whole Performance Management system, see the chapter on backup and restoration in the *JP1/Performance Management User's Guide*.

2.5.1 Backup

When you back up the configuration files, you back up those files in any measure (such as copying the files). When you back up those configuration files, make sure that the PFM - Agent for Oracle service is inactive before you back up the files.

Important

When you back up the configuration files, record the product version number of PFM - Agent for Oracle. For details about the product version number, see the *Release Notes*.

Table below shows the backup target files for PFM - Agent for Oracle:

For details about other files, see the section presenting a list of files to be backed up for PFM - Agent (in Windows) in the *JP1/Performance Management User's Guide*.

Table 2–19: Backup target files for PFM - Agent for Oracle (for a Windows physical host)

File name	Description
Installation-folder\agto\agent*.ini files	Configuration file for Agent Collector service
Installation-folder\agto\agent*instance-name*#*.ini files	
Installation-folder\agto\store*.ini files	Configuration file for Agent Store service
Installation-folder\agto\store*instance-name*#*.ini files	

Table 2–20: Backup target files for PFM - Agent for Oracle (for a Windows logical host)

File name	Description
Installation-folder\agto\agent*.ini files	Configuration file for Agent Collector service
Environment-folder#\jp1pc\agto\agent*instance-name**.ini files	
Installation-folder\agto\store*.ini files	Configuration file for Agent Store service
Environment-folder#\jp1pc\agto\store*instance-name**.ini files	

#

Environment-folder is the folder that is created on the shared disk when setting up the logical host.

2.5.2 Restoration

Make sure that the following prerequisite conditions are met before you restore the configuration information of PFM - Agent for Oracle. After confirming that the conditions have been met, copy the backup files to the original location, and overwrite the existing configuration files on the host with the backup configuration files.

Prerequisites:

- PFM - Agent for Oracle has been already installed.
- The PFM - Agent for Oracle service is inactive.

Important

When you restore the configuration files for PFM - Agent for Oracle, the product version number of the PFM - Agent for Oracle in the backup environment must be the same as that of the PFM - Agent for Oracle in the restoration target environment. For details about the product version number, 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 Oracle 10-00 is restored to PFM - Agent for Oracle 10-00.

Cases for which restoration cannot be performed:

- Settings information backed up from PFM - Agent for Oracle 09-00 is restored to PFM - Agent for Oracle 10-00.

- Settings information backed up from PFM - Agent for Oracle 09-00 is restored to PFM - Agent for Oracle 09-00-05.

2.6 Online manuals

The standard manual supplied medium accompanying the Performance Management program product contains a manual that you can copy to the host on which PFM - Web Console is installed and then view in a Web browser. Make sure that you copy the manual to the executing and standby nodes when PFM - Web Console runs in a cluster system.

2.6.1 Setting procedure

(1) See the manual from the help menu bar of PFM - Web Console

1. Register PFM - Agent with PFM - Web Console, following the PFM - Web Console setup procedure (Follow the procedure to register an additional PFM - Agent).
2. On the host on which PFM - Web Console is installed, create a directory to copy the manual to.
   - Windows: *Web-Console-installation-folder*\doc*language-cord**help-ID-of-PFM - Agent*
   - UNIX: /opt/jp1pcwebcon/doc/*language-cord*/*help-ID-of-PFM - Agent*
3. For details about the help ID of PFM - Agent, see *C. List of Identifiers*.
4. From the manual supplied medium, copy all the manual files to the root of the directory you created.

   HTML manual:

   Windows: all the htm files and FIGURE folder from the *applicable-drive* \MAN\3021*material number* (such as 03004A0D)

   UNIX: all the htm files and FIGURE directory from /*mount-point-directory-for-the-supplied-medium*/MAN/3021/*material number* (such as 03004A0D)

   PDF manual:

   Windows: the PDF file from *applicable-drive*\MAN\3021*material number* (such as 03004A0D)

   UNIX: the PDF file from /*mount-point-directory-for-the-supplied-medium*/MAN/3021/*material number* (such as 03004A0D)
5. Make sure you copy the INDEX.HTM files (for the HTML manual) or the PDF files (for the PDF manual) to the root of the created directory.
6. Restart PFM - Web Console.

(2) See the manual from the hard disk

Execute the setup.exe command on supplied medium to install the manual, or copy htm files, PDF files, and GIF files to any folders or directories. For HTML manual, the folder or directory organization must be:

```
html (storage folder or directory of the htm files and the PDF files)
 └ GRAPHICS (storage folder or directory of GIF files)
```

2.6.2 Viewing the manual

To view the manual:

1. In the menu bar of the PFM - Web Console main window, click Help. A help selection window appears.
2. Click the manual name, or click [PDF] after the manual name.
   Clicking the manual name displays the manual in HTML format. Clicking [PDF] displays the manual in PDF format.

Notes on the display of characters in a Web browser:

In Windows, when you display the online manual from the **Start** menu, the HTML manual might be displayed in the Web browser that is already open.

2.7 Notes regarding operation

2.7.1 Notes on operating

- Before you start PFM - Agent for Oracle, you need to complete installation and setup the Oracle Database and then start the Oracle Database. When the Oracle Database is not running, performance data except for Instance Availability (PD_PDIA) and Server Status (PD_STAT) is not collected. Also, if an Oracle instance or container is mounted, performance data is not collected.
- After you stop PFM - Agent for Oracle, you can change constitution or perform maintenance of Oracle Database instance or container.
- When PFM - Agent for Oracle collects the following records obtained dynamic performance v$ views (such as V $ACCESS, V$OPEN_CURSOR), an error (such as ORA-00600, ORA-00602) might be output from the Oracle Database and it might fail to collect the performance data. In this case, when the records are collected at next time, it collects the performance data again.
  - Activity Summary (PD_PDAS) record
  - Open Cursor (PD_PDOC) record
  - Session Detail (PD_PDS) record
- When the port to use in Agent Collector service has been already used, fail to bind to port at the time of Agent Collector service start#. And, when startup_always is set to Y, Agent Collector service is started. In this case, you can not monitor correctly. You stop the service, and confirm port conflict, later start the service.

  1. Stop the service of PFM - Agent for Oracle.
     ```
     jpcspm stop -key Oracle
     ```
     Stop PFM - Agent for Oracle service which started normally.
     ```
     jpcspm stop -key Oracle -kill immediate
     ```
     Stop PFM - Agent for Oracle service which failed to bind to port.
  2. Confirm port conflict to use in PFM - Agent for Oracle.
     Confirm whether other service do not use a fixed port by `jpcconf port define` command.
  3. Start the service of PFM - Agent for Oracle.
     ```
     jpcspm start -key Oracle [-inst <instance name>]
     ```

  # When failed to bind to port, the message KAVE00100-E, KAVE00107-E, KAVE00160-E are output to common message log.
- If you do not stop PFM - Agent for Oracle before you stop the monitored Oracle Database instance or container, you may not be able to stop the Oracle Database by using the NORMAL shutdown procedure. To stop the Oracle Database, you need to stop PFM - Agent for Oracle beforehand or else use the IMMEDIATE shutdown procedure.
- When you fail in monitoring using `sys` account, either set the following 1 and 2 together, or confirm 3 setting.

  1. Create a password file.
  2. Set "exclusive" as the value of the "remote_login_passwordfile" initialization parameter.
  3. Set "true" as the value of the "o7_dictionary_accessibility" initialization parameter.
- If the expiration date is set on oracle_passwd, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors, perform either of the following procedures before the password is expired:

  - Unset the expiration date of the password

- After updating password, execute the `jpcconf inst setup` command to update oracle_passwd.

  Note that the Oracle default profile is applied to the Oracle account created by `mk_user.sql`.

  For details about the above settings, see the manual for Oracle Database.

- When a connection error occurred while PFM - Agent for Oracle is running, an error is logged to the sqlnet.log file is stored in the directory specified by Oracle. When you use Oracle Database 11g, or Oracle Database 12c or later without setting `ORACLE_BASE`, the sqlnet.log is stored in the directory created under the "/" directory. Before you start PFM - Agent for Oracle, the sqlnet.log can also be stored in the directory specified by the `ORACLE_BASE` environment variable.
- PFM - Agent for Oracle that monitors performance data for each an instance of the node cannot monitor Oracle RAC One Node configuration has dynamically changed Oracle SIDs.
- When specifying an execution module in the Windows WOW64 system directory (SysWOW64) for an alarm action definition, see the chapter on notes on creating alarms in the *JP1/Performance Management Planning and Configuration Guide*.
- 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 Oracle, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.
- Do not specify a folder as the Performance Management folder if it has a Windows Resource Protection function (WRP) applied,(such as a specified Store database folder or the backup folder). If you do specify such a folder, you might not be able to update or delete files or folders, and the product might not work properly.
- If you restart the machine by using the reboot command while the PFM - Agent for Oracle service is running, the message shown below might be output to the common message log, indicating that the Agent Collector service has ended abnormally. This message, however, does not actually indicate the occurrence of an error because the Agent Collector service has in fact ended successfully.

  KAVF12004-E Agent Collector stopped abnormally.
- When you perform a new installation of this product, the first icon displayed might be replaced immediately by another icon. This problem concerns only what icon is displayed. Product operation is not affected.

2.7.2 Notes on monitoring Oracle Database 12c Release 2 or later

(1) Notes on monitoring all version of Oracle Database 12c Release 2 or later

- PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Record name	Field name
Circuit (PD_PDCI)	User
Database Object Cache (PD_PDDO)	Owner
Latch (PD_PDLA)	OS User
	User
Latch Interval (PI_PILA)	OS User
	User

Record name	Field name
Lock (PD_PDLO)	User
Lock Interval (PI_PILO)	User
Lock Waiters (PD_PDLW)	Holding User
	Waiting User
Open Cursor (PD_PDOC)	User
Resource Limit (PD_PDRL)	Resource Name
Segment Detail (PD_PDSM)	Owner
	Segment Name
Session Detail (PD_PDS)	Module
	OS User
	Schema Name
	User
Session Event (PD_PDEV)	User
Session Event Interval (PI_PIEV)	User
Session I/O Interval (PI_PIIO)	User
Session Stat Summary Interval (PI_PIS2)	User
Session Statistics (PD_PDSS)	User
Session Statistics Summary (PD_PDS2)	User
Session Wait (PD_PDWA)	User
SQL Text - Performance Based (PD_PDES)	Parsing User
Table Access (PD_PDTA)	Owner
	User
Transaction (PD_PDTR)	User
Transaction Interval (PI_PITR)	User
Transaction Lock (PD_PDTL)	Object Name
	Owner
	User

- If you set the INSTANCE_MODE Oracle Database initialization parameter to READ-MOSTLY or READ-ONLY, PFM - Agent for Oracle cannot collect the performance information.
- When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), and set to `oracle_version` instance information is `12`. If set a value invalid, the following message appears:

  KAVE05924-W An invalid character string was entered.

(2) Notes on monitoring verison of Oracle Database 18c or later

When set the `OPTIMIZER_IGNORE_HINTS` initialization parameter in enable, it may take an amount of time to perform record collection. In this case, consider that disable the initialization parameter.

3

Installation and setup (UNIX)

This chapter describes the procedures for installing and setting up PFM - Agent for Oracle. For details about how to install and set up an entire Performance Management system, see the chapter on installation and setup for UNIX in the *JP1/Performance Management Planning and Configuration Guide*.

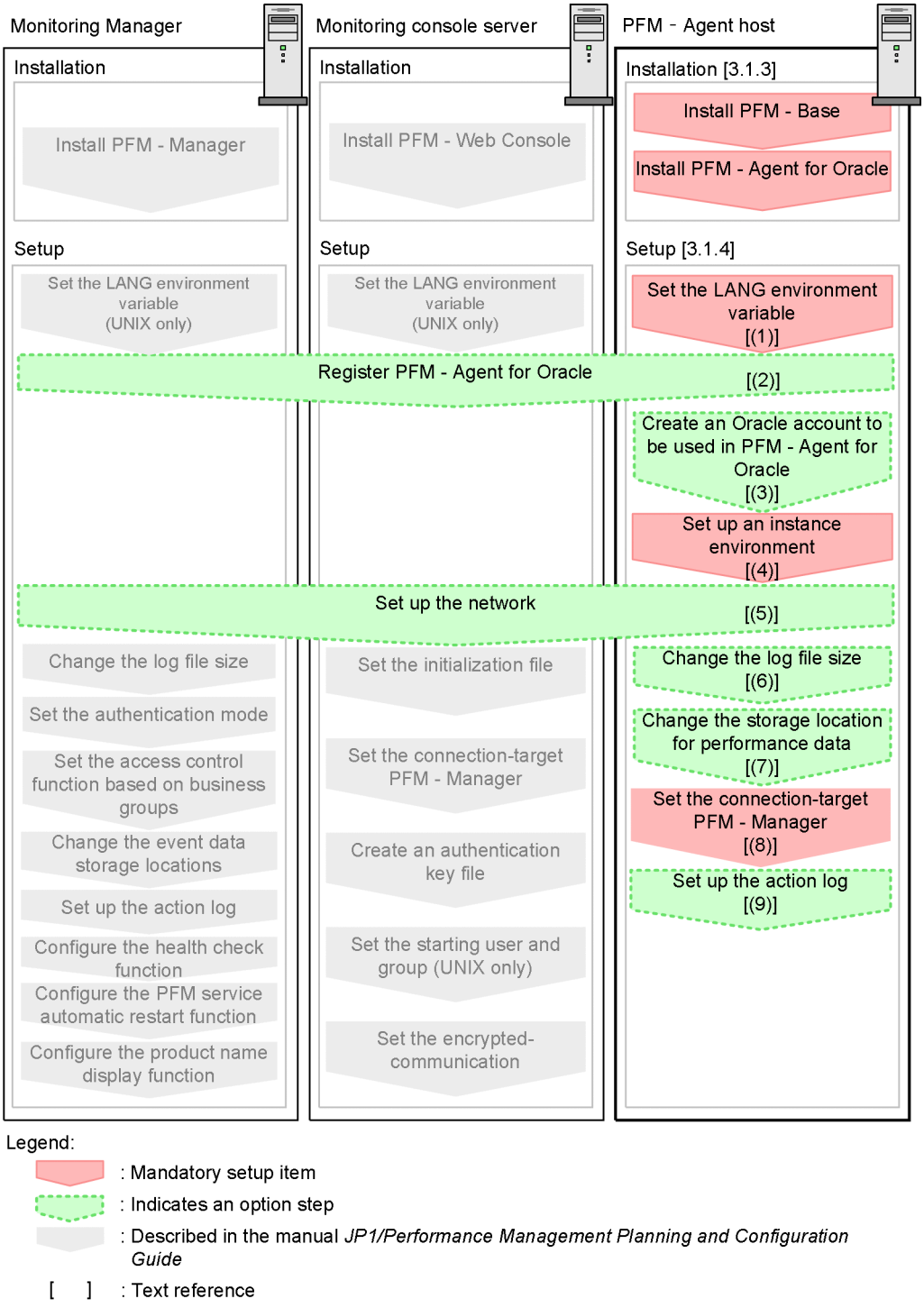
3.1 Installation and setup

This section describes the procedures for installing and setting up PFM - Agent for Oracle.

3.1.1 Installation and setup workflow

The following figure shows the workflow for installing and setting up PFM - Agent for Oracle.

Figure 3–1: Installation and setup workflow (UNIX)



For details about the installation and setup procedures for PFM - Manager and PFM - Web Console, see the chapter on 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 chapter on commands in the manual *JP1/Performance Management Reference*.

3.1.2 Preparation for installing and setting up PFM - Agent for Oracle

Check the following before installing and setting up PFM - Agent for Oracle.

(1) OS requirements

PFM - Agent for Oracle runs on the following OSs:

- HP-UX
- Solaris
- AIX
- Linux

(2) Network environment settings

The following describes the network environment required to run Performance Management.

(a) IP addresssettings

The PFM - Agent for Oracle host must be set up in a network environment where IP addresses can be resolved from host names. PFM - Agent for Oracle will not start in an environment where IP addresses cannot be resolved.

PFM - Agent for Oracle can run in an IPv6 environment and dual stack environment in addition to an IPv4 environment. To run PFM - Agent for Oracle in an IPv6 environment, the monitored Oracle database must support IPv6 environments.

You can use the real host name or the alias name as a monitoring host name (a host name used in Performance Management system).

- When using a real 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

  Set up the environment so that an IP address can be resolved from the specified alias name.

For details about the configuration of a monitoring host name, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Use one of the following methods to set the host name and IP address of a host monitored by Performance Management programs:

- Host information settings file for Performance Management (`jpchosts` file)
- `hosts` file
- DNS (Domain Name System)

Notes on setting IP addresses:

- Although the Performance Management programs and PFM - Agent for Oracle can operate in a DNS environment, they do not support host names in FQDN (Fully Qualified Domain Name) format. When you specify an IP address, use the host name returned by the `hostname` command after removing the domain name portion.
- If you intend to use Performance Management within multiple LAN environments, set the IP addresses in the `jpchosts` file. For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.
- Performance Management cannot operate on the hosts to which IP addresses are assigned dynamically by DHCP. Make sure that all the hosts on which Performance Management programs are installed are configured with user-specified static IP addresses.

Performance Management supports IPv6 environments as well as IPv4 environments as a network configuration. Therefore, Performance Management can operate even in a network configuration in which both an IPv4 environment and an IPv6 environment are used.

PFM - Agent for Oracle can communicate with PFM - Manager via IPv6. Note that this explanation applies only when the OS of a host on which PFM - Agent for Oracle and PFM - Manager are installed is Windows or Linux.

For details about the scope of communication in an environment with both IPv4 and IPv6, see *M. About Communication in IPv4 Environments and IPv6 Environments*.

When you want to use IPv6 for communication between PFM - Manager and PFM - Agent for Oracle, the settings for using IPv6 must be enabled for both the PFM - Manager host and the PFM - Agent host. In addition, before installing PFM - Agent for Oracle, you need to enable the use of IPv6 on the PFM - Agent host. You have to execute the `jpcconf ipv6 enable` command to enable this setting. If this setting is already enabled, however, you do not need to execute the command. If you want to check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command.

For details about the `jpcconf ipv6 enable` command and `jpcconf ipv6 display` command, see the chapter that describes 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.*

When you use IPv6 for communication between a monitored host and PFM - Agent for Oracle, specify the name of a monitored host where name resolution can be performed.

Communication between PFM - Agent for Oracle and a monitoring target is performed with an IP address that can be resolved. Also, if an IPv4 environment and an IPv6 environment are both used, and communication between PFM - Agent for Oracle and the monitoring target fails with an IP address that can be resolved, the communication is not retried by using another IP address.

For example, if communication fails when IPv4 is used, IPv6 is not used to retry communication. Similarly, if communication fails when IPv6 is used, IPv4 is not used to retry communication. Make sure beforehand that a connection can be established.

(b) Port number settings

The following table shows the default port numbers assigned to the services of Performance Management programs. For other services and programs, available port numbers are automatically assigned each time they are started. If you use Performance Management in a firewall environment, use fixed port numbers. For details about how to set fixed port numbers, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Table 3–1: Default port numbers for Performance Management program services (in UNIX)

Service description	Service name	Parameter	Port number	Remarks
Service configuration information management facility	Name Server	`jp1pcnsvr`	22285	The port number used by the Name Server service of PFM - Manager. This port is set up on every Performance Management host.
Service status management facility	Status Server	`jp1pcstatsvr`	22350	The port number used by the Status Server service of PFM - Manager and PFM - Base. This port is set up on the hosts on which PFM - Manager and PFM - Base are installed.
Monitoring console communication facility	View Server	`jp1pcvsvr`	22286	The port number used by the View Server service of PFM - Manager. This port is set up on the hosts on which PFM - Manager is installed.
Web service facility	Web Service	--	20358	The port number used by the Web Service service of PFM - Web Console.
Web container facility	Web Console	--	20359 20360	The port number used by the Web Console service of PFM - Web Console.
JP1/SLM linkage facility	JP1/ITSLM	--	20905	The port number set by JP1/SLM.

Legend:

--: None

Ensure that the network is set up to allow communication using these port numbers, which are used by PFM - Agent for Oracle.

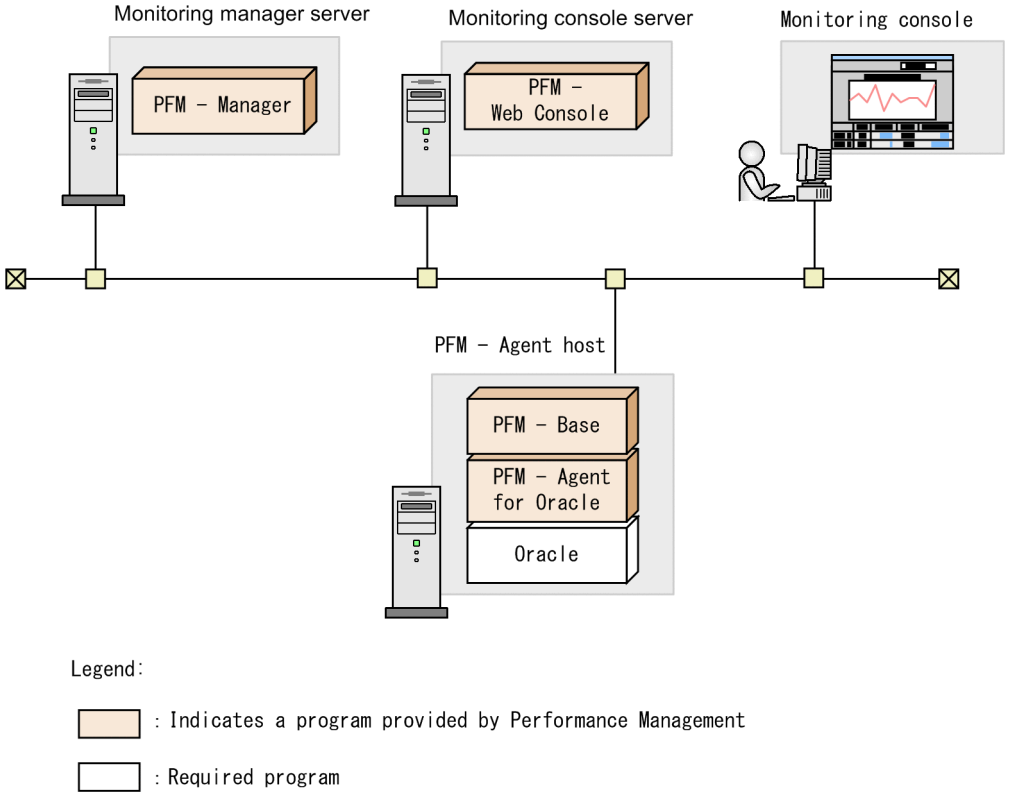
(3) OS user permission required to install PFM - Agent

To install PFM - Agent for Oracle, use an account that belongs to the Administrators group.

(4) Prerequisite programs

This subsection describes the prerequisite programs required to install PFM - Agent for Oracle. The following figure 3-2 shows the program configuration:

Figure 3–2: Program configuration



(a) Monitoring target programs

The monitoring target programs of PFM - Agent for Oracle are as follows:

- Oracle Database Standard Edition
- Oracle Database Standard Edition One
- Oracle Database Standard Edition 2
- Oracle Database Enterprise Edition

To monitor this program, it must be installed on the same host as PFM - Agent for Oracle.

Note that when a monitoring target program running on a virtualized OS is being monitored, PFM - Agent for Oracle monitors what the monitoring target program can perform on the virtualized OS.

(b) Performance Management programs

Install PFM - Agent and PFM - Base on the PFM - Agent host.

PFM - Base is a prerequisite program for PFM - Agent. Only one instance of PFM - Base is required, even when multiple instances of PFM - Agent are installed on one host.

Note that you do not need to install PFM - Base if PFM - Manager and PFM - Agent are installed on the same host.

To monitor Oracle operation using PFM - Agent for Oracle, PFM - Manager and PFM - Web Console are required.

(5) Installation and setup in a cluster system

When you install and set up PFM - Agent in a cluster system, the prerequisite network environment and program configuration is different from those for a normal system. There are also additional tasks that must be performed on the executing nodes and standby nodes. For details, see *4. Operating PFM - Agent for Oracle in a Cluster System*.

(6) Preparations for collecting data in the event of a failure

When a failure occurs, you might need to use the core dump files to investigate the cause. Whether the core dump files are output depends on the user environment settings, so confirm in advance that the settings are as follows:

Settings for the size of the core dump files

The maximum size of the core dump files is limited by the size setting (`ulimit -c`) for the root user's core dump files. Set the scripts as the follows:

```
ulimit -c unlimited
```

If this setting violates the security policies of your computer, state this script setting in a comment line as follows:

```
# ulimit -c unlimited
```

Important

After the above is stated in a comment line, core dump files might not be output when they should be output, such as when a segment fault or bus error occurs in a process. If this is the case, investigation might be impossible.

Settings for the core dump-related kernel parameter (for Linux only)

If you used the Linux kernel parameter (`kernel.core_pattern`) to change the default output destination and names of core dump files, the core dump files might not be able to be acquired. Therefore, we recommend that you do not change the settings for the Linux kernel parameter (`kernel.core_pattern`).

(7) Cautionary notes

Note the following when installing and setting up Performance Management.

(a) Precautions regarding 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.

(b) Notes on installing and setting up multiple Performance Management programs on 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:

- When PFM - Manager and PFM - Agent are installed on the same host, PFM - Base is not required. In this case, PFM - Manager is a prerequisite program for PFM - Agent and must be installed before PFM - Agent is installed.
- You cannot install PFM - Base and PFM - Manager on the same host. If you want to install PFM - Manager on a host on which PFM - Base and PFM - Agent are already installed, uninstall all Performance Management programs, and then install PFM - Manager and PFM - Agent in that order. The same applies when you install PFM - Base on

a host on which PFM - Manager and PFM - Agent are already installed: you must uninstall all Performance Management programs, and then install PFM - Base and PFM - Agent in that order.

- If you install PFM - Agent on a host on which PFM - Manager is already installed, the connection-target PFM - Manager will be the instance of PFM - Manager on the local host, and cannot change it to that on the remote host. If you want PFM - Agent to connect to PFM - Manager on a remote host, ensure that PFM - Manager is not installed on the host on which you install PFM - Agent.
- If you install PFM - Manager on a host on which PFM - Agent is already installed, the connection-target PFM - Manager is reset to the local host. See the setting results that are output to the common message log.
- If you install PFM - Agent on a host on which PFM - Web Console is already installed, close all the browser windows before you install the program.
- When you perform a new installation of a Performance Management program, the status management facility will be enabled by default. However, if you upgrade from version 07-50 to 08-00 or newer, the settings for the status management function remain the same as they were in the older version. To change the setting of the status management facility, see the chapter on error detection for 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.

(c) Notes on upgrading PFM - Agent for Oracle

Before you perform version upgrade installation, you need to perform the procedure described in *3.2.2(1)(b) Deleting the objects registered in the Oracle Database*. Complete version upgrade installation and then perform the procedure in *3.1.4(4)(b) Registering objects in the Oracle Database*.

For details about notes on upgrading the versions of Performance Management programs, see the section describing the notes on version upgrading in the chapter that explains installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

For details about notes on upgrading the version of PFM - Agent for Oracle, see *H. Migration Procedure and Notes on Migration*.

For details about upgrading, see the *JP1/Performance Management Planning and Configuration Guide*.

If you delete the objects after the upgrade installation finishes rather than before the upgrade installation starts, the following message appears:

ORA-04043: object LSC_13_PDDB2 does not exist, ORA-04043: object LSC_13_PIDB2 does not exist, ORA-04043: object LSC_13_PIDB3 does not exist, see *3.2.2(1)(b) Deleting the objects registered in the Oracle Database*. This message, however, does not indicate a serious problem. Even if this message appears, proceed to register the objects again, see *3.1.4(4)(b) Registering objects in the Oracle Database*.

(d) Other cautionary notes

- When the monitoring-target program is Oracle Database 12c Release 1, you cannot monitor a multitenant container database (CDB) and can monitor a non-CDB.
- When the monitoring-target program is Oracle Database 12c Release 2 or later, in addition to the program in a traditional non-CDB environment, PDBs and the root container (`CDB$ROOT`) in a CDB environment can be monitored. Monitoring application containers (an application root and application PDBs) is not supported.

- When you install Performance Management programs, check whether the following security-related programs are installed. If they have been installed, take appropriate action according to the following explanation.
  - Security monitoring program

    Stop the security monitoring program or change the settings so that the installation of Performance Management programs will not be interrupted.
  - Virus detection program

    Stop the virus detection program before you install Performance Management programs.

    If a virus detection program is running during the installation of Performance Management programs, the installation processing might slow down, installation might not be executable, or the programs might not be able to be installed correctly.
  - Process monitoring program

    Stop the process monitoring program or change the settings. Also, specify settings that prevent the services or processes of Performance Management and common components from being monitored.

    If the process monitoring program starts or stops these services or processes during the installation of Performance Management programs, installation might fail.
- When you perform a new installation of PFM - Agent for Oracle in an environment where no other Performance Management program has been installed, make sure that there are no files or folders in the installation folder.
- If installation fails and `Install failed` is displayed on the installation status bar as a result, acquire the `/etc/.hitachi/.hitachi.log` file. Note that this log file is overwritten every time installation is performed. If you want to keep its contents, you will have to back it up. For the default file name for the installation log file, see *8.4.2(2) Information about Performance Management*.
- If you install Performance Management programs in a directory linked to another directory, some files and directories may remain when the programs are uninstalled. If files and directories remain, delete them manually. When Performance Management programs are installed in the link-target directory, any existing files and directories whose names are the same as the names of the files and directories being installed will be overwritten.
- If the `/opt/jp1pc/setup` directory contains the setup file for PFM - Agent for Oracle, additional setup for a new instance of PFM - Agent for Oracle is performed. When additional setup of PFM - Agent for Oracle is successful, message `KAVE05908-I New agent setup (`*pfm-agent-service-key*`) ended successfully` is output to the common message log. Check whether this message has been output.
- In Linux, this software is a Hitachi program product that conforms to the disk copy installation of JP1/ServerConductor/Deployment Manager and Hitachi Compute Systems Manager Deployment Manager Plug-in, or the copy functionality using conversion to image files that is provided by a virtual platform. About the disk copy installation, see notes on installing replicated disks in the manual *JP1/Performance Management Planning and Configuration Guide*.
- In Solaris or AIX, monitoring Oracle Database 11g Release 2 or later means that the Oracle Client 32-bit library is used. In this case, you have to install one of the following products on the machine and build an environment for it:
  - Oracle Database 11g Release 2 Client (32-bit) or later

  If you start PFM - Agent for Oracle with the `ORACLE_HOME` of Oracle Database 11g Release 2 or later,or Oracle Client 64-bit specified as the Oracle instance information item `oracle_home`, the following messages are output to the common message log:

  KAVF12011-E Initialization of interprocess communication failed.

  KAVF12021-E Error occurred by function function-name. (en=error-code, arg1=argument-1, arg2=argument-2, arg3=argument-3)

  Note:

  PFM - Agent for Oracle does not support Oracle Instant Client. Select either Administrator or Runtime as the Oracle Client installation type. If you have installed Instant Client on an `ORACLE_HOME` by mistake, do not

install an Oracle Client on the same ORACLE_HOME with Administrator or Runtime selected as the installation type.

You have to install the Oracle Client on another ORACLE_HOME, or delete the

Instant Client before installing the Oracle Client.

3.1.3 Installation procedure

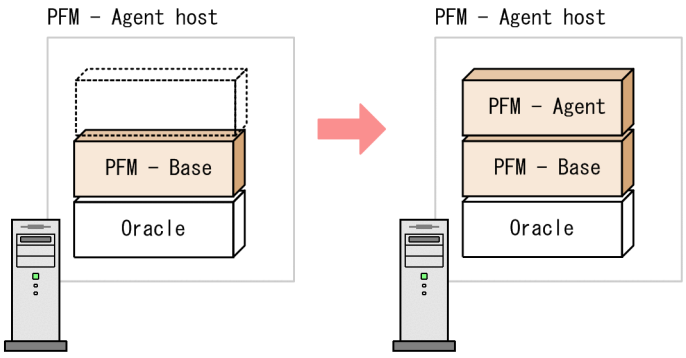
This subsection explains the order in which the component programs of PFM - Agent for Oracle are to be installed, and describes how to install these programs from the supplied medium.

(1) Order of installation

Install PFM - Base, and then PFM - Agent. PFM - Base must be installed on the host before you can install PFM - Agent.

If you want to install PFM - Agent and PFM - Manager on the same host, install PFM - Manager before you install 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 *3.4.4 Updating the Store version to 2.0*.

Multiple instances of PFM - Agent on the same host can be installed in any order.



(2) How to install the programs

You can install Performance Management programs on a UNIX host either by using the supplied medium, or by using JP1/Software Distribution to perform a remote installation. For details about how to use JP1/Software Distribution, see the applicable one of the following manuals:

- *Job Management Partner 1/Software Distribution Manager Description and Administrator's Guide*
- *Job Management Partner 1/Software Distribution SubManager Description and Administrator's Guide*, for UNIX systems
- *Job Management Partner 1/Software Distribution Client Description and User's Guide*, for UNIX systems

To install programs from the supplied medium:

(a) HP-UX

1. As the superuser, log on to the host on which Performance Management programs are to be installed, or use the `su` command to change the user to superuser.

2. If any Performance Management services are running on the local host, stop all of them.

   The services you are going to stop are the Performance Management services running on both the physical and logical hosts. For details about how to stop services, see the chapter that explains startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the supplied medium into the machine.

4. Execute the `mount` command to mount the supplied medium.

   The following is an example of the command executed to mount the supplied medium on the mount directory:

   ```
   /usr/sbin/mount -F cdfs -r device-special-file-name mount-directory
   ```

   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 Performance Management programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space bar to select programs.

8. After installation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

(b) Solaris (SPARC)

1. As the superuser, log on to the host on which Performance Management programs are to be installed, or use the `su` command to change the user to superuser.

2. If any Performance Management services are running on the local host, stop all of them.

   The services you are going to stop are the Performance Management services running on both the physical and logical hosts. For details about how to stop services, see the chapter that explains startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the supplied medium into the machine.

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 Performance Management programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space bar 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 mount-directory
```

The specified command differs depending on the environment used.

(c) AIX

1. As the superuser, log on to the host on which Performance Management programs are to be installed, or use the `su` command to change the user to superuser.

2. If any Performance Management services are running on the local host, stop all of them.

   The services you are going to stop are the Performance Management services running on both the physical and logical hosts. For details about how to stop services, see the chapter that explains startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.

3. Insert the supplied medium into the machine.

4. Execute the `mount` command to mount the supplied medium.

   The following is an example of the command executed to mount the supplied medium on the mount directory:

   ```
   /usr/sbin/mount -r -v cdrfs device-special-file-name mount-directory
   ```

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 Performance Management programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space bar to select programs.

8. After installation is correctly finished, enter `Q`.

   The initial window of Hitachi PP Installer reappears.

(d) Linux

1. Make sure that the installation directory for Performance Management programs (`/opt/jp1pc/*`) actually exists.

2. As the superuser, log on to the host on which Performance Management programs are to be installed, or use the `su` command to change the user to superuser.

3. If any Performance Management services are running on the local host, stop all of them.

   The services you are going to stop are the Performance Management services running on both the physical and logical hosts. For details about how to stop services, see the chapter that explains startup and termination of Performance Management in the *JP1/Performance Management User's Guide*.

4. Insert the supplied medium into the machine.

5. Execute the following command to start Hitachi PP Installer:#

   ```
   mount-directory/X64LIN/SETUP mount-directory
   ```

   The underlined parts, which indicate the device special file name and mount-point directory name, vary depending on the environment being used.

   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 Performance Management programs you wish to install, and enter `I`.

   The selected programs are installed. Note that you can move the cursor and use the space bar 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 device-special-file-name mount-directory
```

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.

3.1.4 Setting up PFM - Agent for Oracle

This subsection describes how to set up PFM - Agent for Oracle for operation.

〈 Option 〉 indicates an item that may or may not be required depending on your operating environment, or an optional item that you can set if you do not wish to use the default.

(1) Set the LANG environment variable

The following table lists the `LANG` environment variable values supported by PFM - Agent for Oracle. If you set a language other than the ones indicated in the table below (such as German, French, Spanish, Korean, or Russian), `C` is assumed as the value of the `LANG` environment variable.

Before you specify the `LANG` environment variable, make sure that the language environment you want to set has been installed and set up correctly. If you fail to check, characters may not be displayed correctly or definition data may be overwritten.

Notes:

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.

Table 3–2: LANG environment variables that can be used in PFM - Agent for Oracle

OS	Language and code		LANG value
HP-UX#	Japanese	Shift-JIS	• ja_JP.SJIS • japanese
		EUC	• ja_JP.eucJP • japanese.euc
		UTF-8	• ja_JP.utf8
	English		• C
	Simplified-Chinese	UTF-8	• zh_CN.utf8
		GB18030	• zh_CN.gb18030
Solaris#	Japanese	Shift-JIS	• ja_JP.PCK
		EUC	• ja • japanese • ja_JP.eucJP
		UTF-8	• ja_JP.UTF-8
	English		• C
	Simplified-Chinese	UTF-8	• zh_CN.UTF-8 • zh_CN.UTF-8@pinyin • zh_CN.UTF-8@radical • zh_CN.UTF-8@stroke • zh.UTF-8
		GB18030	• zh_CN.GB18030 • zh_CN.GB18030@pinyin • zh_CN.GB18030@radical • zh_CN.GB18030@stroke
AIX#	Japanese	Shift-JIS	• Ja_JP • Ja_JP.IBM-932
		EUC	• ja_JP • ja_JP.IBM-eucJP
		UTF-8	• JA_JP • JA_JP.UTF-8
	English		• C
	Simplified-Chinese	UTF-8	• ZH_CN • ZH_CN.UTF-8
		GB18030	• Zh_CN • Zh_CN.GB18030
Linux#	Japanese	Shift-JIS	nothing

OS	Language and code		LANG value
Linux#	Japanese	EUC	nothing
		UTF-8	• ja_JP.UTF-8 • ja_JP.utf8
	English		• C
	Simplified-Chinese	UTF-8	• zh_CN.utf8 • zh_CN.UTF-8
		GB18030	• zh_CN.gb18030

#

The ASCII 7-bit character set is supported, except for the following language types:

- Japanese
- English
- Simplified-Chinese

(2) Register PFM - Agent for Oracle Option

To perform integrated management of PFM - Agent using PFM - Manager and PFM - Web Console, you must register PFM - Agent for Oracle with PFM - Manager and PFM - Web Console.

If PFM - Agent for Oracle is already registered in PFM - Manager and PFM - Web Console, you do not have to follow the procedure described below. If PFM - Agent for Oracle is not registered yet, manually register PFM - Agent for Oracle according to the procedure.

You can determine whether manual registration of PFM - Agent for Oracle is necessary by referring to the conditions described below.

Manually registering PFM - Agent for Oracle in PFM - Manager

When all of the following conditions apply, manually register PFM - Agent for Oracle in PFM - Manager:

- The PFM - Agent for Oracle to be installed is of a product version that is not specified in the *Release Notes* for PFM - Manager.
- PFM - Agent for Oracle is installed on a host other than PFM - Manager.

Manually registering PFM - Agent for Oracle in PFM - Web Console

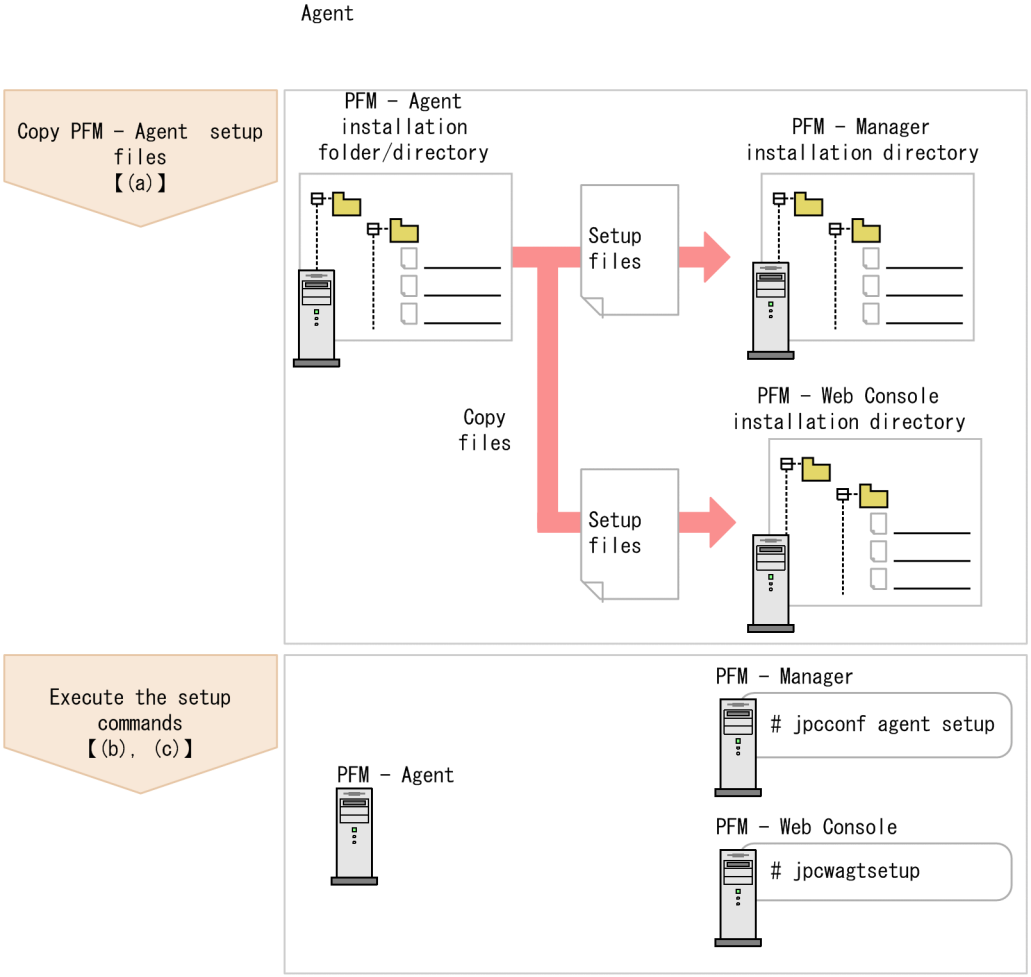
When the following condition applies, manually register PFM - Agent for Oracle in PFM - Web Console:

- The PFM - Agent for Oracle to be installed is of a product version that is not specified in the *Release Notes* for PFM - Web Console.

If, however, the *Release Notes* for PFM - Agent for Oracle state that it is necessary to execute the setup command, execute the setup command.

The following figure shows the flow of PFM - Agent registration.

Figure 3–3: Flow of PFM - Agent for Oracle registration



Legend:

【 】 : Text reference

Notes:

- Register PFM - Agent before setting up an instance environment.
- You do not need to register a new instance of an already registered version of PFM - Agent for Oracle when you add it to the Performance Management system.
- When you have installed different versions of PFM - Agent for Oracle on different hosts, set up the older version before the newer version.
- When you install PFM - Agent on the same host as PFM - Manager, the `jpcconf agent setup` command is executed automatically, and the following message is output to the common message log: `KAVE05908-I New agent setup (`*pfm-agent-service-key*`) ended successfully. (version=`*version*`)`. Check the result of command execution in the log file. If execution was unsuccessful, try it again. For details about how to execute commands, see the chapter on commands in the manual *JP1/Performance Management Reference*.
- Registration of PFM - Agent for Oracle creates the **Oracle** folder in **Reports** window and **Alarms** window of PFM - Web Console. If the **Oracle** file or folder already exists on the **Reports** window, rename the file or folder before registering PFM - Agent for Oracle.

(a) Copy the PFM - Agent for Oracle setup files

Copy the setup files from the host on which you installed PFM - Agent for Oracle to the hosts on which PFM - Manager and PFM - Web Console are installed.

To copy the files:

1. If PFM - Web Console is running, stop it before copying the files.
2. Copy the PFM - Agent setup files in binary mode.
   The following table shows the location of the setup files and where they should be copied.

   Table 3–3: Setup files to be copied

PFM - Agent for Oracle setup file	Destination		
	PFM program name	OS	Destination folder
/opt/jp1pc/setup/jpcagtow.EXE	PFM - Manager	Windows	*installation-folder*\setup
/opt/jp1pc/setup/jpcagtou.Z		UNIX	/opt/jp1pc/setup/
/opt/jp1pc/setup/jpcagtow.EXE	PFM - Web Console	Windows	*installation-folder*\setup
/opt/jp1pc/setup/jpcagtou.Z		UNIX	/opt/jp1pcwebcon/setup/

(b) Execute the setup command on the PFM - Manager host

On the PFM - Manager host, execute the following command to set up PFM - Agent for Oracle:

```
jpcconf agent setup -key Oracle
```

Although an example of interactive command execution is shown here, the `jpcconf agent setup` command can be also executed non-interactively. For details about the `jpcconf agent setup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

Notes on executing the command:

If any Performance Management programs or services are still running on the local host when you execute the `jpcconf agent setup` command, an error may occur. If an error occurs, make sure that all Performance Management programs and services have completely stopped, and then execute the `jpcconf agent setup` command again.

You can then delete the PFM - Agent setup files remaining on the PFM - Manager host.

(c) Execute the setup command on the PFM - Web Console host

On the PFM - Web Console host, execute the following command to set up PFM - Agent for Oracle:

```
jpcwagtsetup
```

You can then delete the PFM - Agent setup files remaining on the PFM - Web Console host.

(3) Create an Oracle account to be used in PFM - Agent for Oracle

To monitor an Oracle Database and collect performance data by using PFM - Agent for Oracle, you must perform either of the following setting tasks:

- Set the `sys` account as the account used to monitor the Oracle Database from PFM - Agent for Oracle
- Create a special Oracle account with system privileges, and set it as the account used to monitor the Oracle Database from PFM - Agent for Oracle

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, PDBs and the root container (`CDB$ROOT`) can be monitored. To operate PFM - Agent for Oracle in a CDB configuration, use one of the following Oracle accounts for monitoring:

- When monitoring the root container (`CDB$ROOT`) to monitor all PDBs belonging to it

  If instances of PFM - Agent for Oracle are to be created for monitoring the root container (`CDB$ROOT`) so that both the root container (`CDB$ROOT`) and the PDBs belonging to it can be monitored, perform monitoring by using an administrative account (`sys` or `system`). If you perform monitoring with a common user's account, you cannot collect performance information on PDBs. For details, see *6. List of records for PFM - Agent for Oracle* or descriptions of each record field.
- When monitoring the root container (`CDB$ROOT`) or PDBs

  If instances of PFM - Agent for Oracle are to be created to monitor each root container (`CDB$ROOT`) or PDB, perform monitoring with one of the following users' accounts:
  - If you create a user for monitoring the root container (`CDB$ROOT`)

    A common user with system privileges for `CDB$ROOT` listed in *Table 3-4*

    When the root container (`CDB$ROOT`) that is being monitored, collecting performance data is different by monitored account. For details, see *6. List of records for PFM - Agent for Oracle* or descriptions of each record field.

    This is difference of setting for collecting PDB's performance data. For details, see your Oracle documentation.
  - If you create a user for monitoring PDBs

    A local user with system privileges for the PDBs to be monitored listed in *Table 3-4*

If you create Oracle accounts by executing the `mk_user.sql` script, the following users can be created:

- If the monitoring target is Oracle Database 11g Release 2 or earlier, or Oracle Database 12c Release 1 or later in a non-CDB configuration

  A user with system privileges listed in *Table 3-4*
- If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration
  - A common user with system privileges for `CDB$ROOT` listed in *Table 3-4*
  - A local user with system privileges for the PDBs to be monitored listed in *Table 3-4*

The following table lists the operations that PFM - Agent for Oracle performs for the Oracle Database to collect performance data. The table also lists the system privileges required to perform the operations with the Oracle account.

Table 3–4: Operations for the Oracle Database and required system privileges

Operations that PFM - Agent for Oracle performs for the Oracle Database	System privileges required to perform the operations
• Searching the static data dictionary view • Searching the dynamic performance view	• CREATE SESSION • CREATE TABLE

Operations that PFM - Agent for Oracle performs for the Oracle Database	System privileges required to perform the operations
• Executing the listener control utility • Acquiring the execution schedule of the selected SQL • Executing a stored package specific to PFM - Agent for Oracle	• CREATE PROCEDURE • SELECT ANY DICTIONARY • SELECT ANY TABLE • INSERT ANY TABLE • DELETE ANY TABLE • UPDATE ANY TABLE • CREATE ANY INDEX • ALTER ANY INDEX • UNLIMITED TABLESPACE (This privilege is not needed when an assignment limit that allows writing to the default tablespace of the account used for monitoring has been set.)

The `sys` account has the system privileges shown in *Table 3-4*. When the `sys` account cannot be used for Oracle Database monitoring due to security requirements, use an Oracle account with the system privileges shown in *Table 3-4*.

To use an Oracle account that can use PFM - Agent for Oracle without use of the `mk_user.sql` script, grant the privileges necessary for operations. By granting these privileges, the minimum necessary privileges can be granted to the Oracle account. When granting privileges according to role, do so explicitly (using `GRANT` *privileges*...). For details about the privileges needed for each operation, see *I. Precautions Regarding Permissions*.

The following table lists the information needed to create an Oracle account by executing the `mk_user.sql` script. Check the information before starting setup operations.

Table 3–5: Information required to create an Oracle account

Item	Description
`Enter username`	Specifies the name of the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `user` parameter of the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. The default is `PFMAGTO`. Note: If you specify an existing account in the database as an account to be used by an instance of PFM - Agent for Oracle, a script error occurs. Always check the account names existing in the database in advance, and specify an account that will be used only for the instance of PFM - Agent for Oracle.
`Enter password`	Specifies the password for the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `BY password` parameter of the `IDENTIFIED` clause in the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item.
`Enter default tablespace`	Specifies the default tablespace used by the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the `DEFAULT TABLESPACE` clause in the `CREATE USER` statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item. Note: Do not specify the `SYSTEM` or `INDEX` tablespace as the default tablespace. Before specifying the default tablespace, make sure that no problem occurs when a package for PFM - Agent for Oracle is registered in the tablespace. Alternatively,

Item	Description
Enter default tablespace	create an exclusive tablespace for PFM - Agent for Oracle, and then specify the tablespace as the default tablespace.
Enter default temporary tablespace	Specifies the default temporary tablespace used by the account to be created. The value you specify must consist of 7-bit ASCII alphanumeric characters, must not exceed 30 bytes, and must be specifiable in the TEMPORARY TABLESPACE clause in the CREATE USER statement. Note that a character string that begins with a number and includes an alphabetic character cannot be specified. This is a mandatory item. Note: Do not specify the SYSTEM, INDEX, or USERS tablespace as the default temporary tablespace. Before specifying the default temporary tablespace, make sure that no problem will occur if the tablespace is used as the default temporary tablespace. Alternatively, create an exclusive tablespace for PFM - Agent for Oracle, and then specify the tablespace as the default temporary tablespace.

Notes:

- Make sure that the value of each item consists of only 7-bit ASCII alphanumeric characters that do not exceed 30 bytes. If the value is longer than 30 bytes or includes a character that is not a 7-bit ASCII alphanumeric character, the script may operate incorrectly.
- Make sure that the value of each parameter is a nonquoted identifier described in the Schema Object Naming Rules. If you specify a value that is not a nonquoted identifier, the script may operate incorrectly. For details about the Schema Object Naming Rules and nonquoted identifiers, see your Oracle documentation.
- If you create a common user, by executing the mk_user.sql script, for monitoring the root container (CDB $ROOT) of Oracle Database 12c Release 2 or later in a CDB configuration, the default tablespace and default temporary tablespace that are specified in the mk_user.sql script must be included in all containers that belong to the CDB. For that reason, before executing the mk_user.sql script, create the default tablespace and default temporary tablespace in all containers that belong to the CDB. If the mk_user.sql script is executed without satisfying the requirements for creating a common user, the creation might fail and the KAVF12506-E message might appear. For details about the requirements for creating a common user, see the Oracle Database documentation.
- If you want to check the details of an account created by mk_user.sql, see DBA_USERS, which is a static dictionary view for the monitoring-target Oracle Database.

  The following example shows how to view the tablespace for the account A4O in the static data dictionary view DBA_USERS. If it is clear from the execution results of this SQL statement that the account has been created in the wrong tablespace, delete the account, and then re-create it using mk_user.sql.

Example:

To check the details of Oracle account A4O in UNIX:

1. From the command prompt, use the sys account to connect to SQL*Plus.

   sqlplus "sys/*sys-account-password* AS SYSDBA"

2. Use SQL*Plus to execute the following SQL statement:

   ```
   SQL>select DEFAULT_TABLESPACE, TEMPORARY_TABLESPACE from DBA_USERS
   where USERNAME='A4O';
   ```

3. Check the execution results. For example, you can check the default tablespace from the DEFAULT_TABLESPACE column and the default temporary tablespace from the TEMPORARY_TABLESPACE column.

Note:

The method for connecting to SQL*Plus with the sys account may differ according to the Oracle version. For details, see the Oracle documentation.

Use SYSDBA privileges to connect to the Oracle Database that you want to monitor.

For details about the CREATE USER statement, see your Oracle documentation.

The following procedure shows how to create an Oracle account. Before creating an Oracle account, make sure that the tablespaces and other required resources have been prepared.

To create an Oracle account:

1. Set up an environment where the sqlplus Oracle command can be executed.

   For details about Oracle environment setup, see your Oracle documentation.

2. Navigate to the following folder, which contains mk_user.sql provided by PFM - Agent for Oracle:

   ```
   /opt/jp1pc/agto/agent/sql
   ```

3. Execute the mk_user.sql script for the monitoring-target Oracle Database.

   Example:

   If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-database* @mk_user.sql

   If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

   - If you create a local user for a PDB

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-PDB* @mk_user.sql

   - If you create a common user for the root container (CDB$ROOT)

   sqlplus *Oracle-account*/*password-for-the-Oracle-account* @mk_user.sql

   Notes:

   - The sqlplus command is provided by Oracle Corporation.
   - The Oracle account with which the mk_user.sql script is executed must be granted the CREATE USER, CREATE SESSION and GRANT ANY PRIVILEGE system privileges before the script is executed.
   - If the SYS account is used to execute mk_user.sql script, an error may occur unless the AS SYSDBA option is specified.
   - Establish either a SYSDBA connection to the monitoring-target Oracle Database.

     The following shows an example of the mk_user.sql script:

     Example:

     If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

     sqlplus "sys/*password-for-the-sys*@*net-service-name-for-the-monitoring-target-database* AS SYSDBA" @mk_user.sql

     If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

     - If you create a local user for a PDB

sqlplus "sys/*password-for-the-sys*@*net-service-name-for-the-monitoring-target-PDB* AS SYSDBA" @mk_user.sql

- If you create a common user for the root container (CDB$ROOT)

sqlplus "sys/*password-for-the-sys* AS SYSDBA" @mk_user.sql

- When the mk_user.sql script is executed, the execution results are output to a spool file. Note that the spool file cannot be created in the following cases:
- • The current directory has not been changed to the folder shown in step 2 when the mk_user.sql script is executed.
- • A user without root privileges uses SQL*Plus to execute the mk_user.sql script.

4. Set the parameters that are required to create an Oracle account.

   Enter the values for the items listed in *Table 3-5* as prompted by the command. All items are mandatory. To use the default value displayed for an item, only press the **Enter** key.

When all values have been entered, the Oracle account is created.

Notes:

- Before creating an account, check whether you want to acquire the value of the Explain Plan (EXPLAIN_PLAN) field in the SQL Text (PD_PDSQ) record for operations on the objects that belong to the SYS schema. If you want to do so, use sys as the account to be used by PFM - Agent for Oracle. If you use an account other than sys, you will no longer be able to acquire the value of that field. If the value of the EXPLAIN_PLAN field cannot be acquired, message Explain Plan Failed is stored in the field.
- If the account used by PFM - Agent for Oracle has no privileges to access, or fails to reference, an object that belongs to a schema of the user who executed SQL, the following value cannot be acquired:

  The value of the Explain Plan (EXPLAIN_PLAN) field in the SQL Text (PD_PDSQ) record
- If the value of the EXPLAIN_PLAN field cannot be acquired, message Explain Plan Failed is stored in the field. If you want to acquire the value of the Explain Plan (EXPLAIN_PLAN) field, execute the SQL for manipulating the field in the *owner.table-name* format.
- Any Oracle account created using the mk_user.sql script is granted UPDATE ANY TABLE or another system privilege that can freely manipulate objects of other schemas. Manage such Oracle accounts with special care.

  The following table lists the privileges granted to Oracle accounts and the assignment limits of tablespaces.

Table 3–6: Privileges granted by mk_user.sql to Oracle accounts and the assignment limits of tablespaces

Type	Privileges granted / assignment limits	Description
System privilege	CREATE SESSION	Required to establish a session with the monitored Oracle Database.
	CREATE TABLE	Required when registering a table needed to monitor the Oracle Database, for the monitored Oracle Database (see the table in *Table 3-14*).
	CREATE PROCEDURE	Required when registering a procedure needed to monitor the Oracle Database, for the monitored Oracle Database (see the package in *Table 3-14*).
	SELECT ANY DICTIONARY	Required when registering information needed to monitor the Oracle Database in the monitored Oracle Database (see *Table 3-14*) and when collecting information.
	SELECT ANY TABLE	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.

Type	Privileges granted / assignment limits	Description
System privilege	INSERT ANY TABLE	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
	UPDATE ANY TABLE	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
	DELETE ANY TABLE	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
	CREATE ANY INDEX	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
	ALTER ANY INDEX	Required to obtain the Explain Plan (EXPLAIN_PLAN) field in a SQL Text (PD_PDSQ) record.
Assignment limits of tablespaces	Unlimited assignment for the default tablespace#	Required when registering information needed to monitor the Oracle Database in the monitored Oracle Database (see *Table 3-14*) and when obtaining the Explain Plan field of the PD_PDSQ record.

#

Any created account is granted a privilege to write to the default tablespace without any limit. To change the size of the tablespace allocated to an account after you have created the account, issue the ALTER USER statement in an environment where the sqlplus Oracle command can be executed. Note that any Oracle account with which you execute the ALTER USER statement must be granted the ALTER USER system privilege.

The following shows an example of changing the size of tablespace allocated to an account.

Example:

```
ALTER USER Oracle-account QUOTA maximum-tablespace-allocation-size ON t
ablespace-name;
```

For details about the ALTER USER statement, see your Oracle documentation.

(4) Set up an instance environment

The following table lists the instance information items that are to be specified. You can set up multiple instance environments by repeating the procedure for each instance.

- Setting up instance information
- Registering objects in the Oracle Database
- Setting up the Oracle Database

This section describes the procedures for each of the actions.

(a) Set up instance information

You must specify instance information for the Oracle that is to be monitored by the PFM - Agent for Oracle. Specify instance information on the PFM - Agent host.

The following table lists the instance information items that are to be specified. You should check this information before you start the setup procedure. For details about the Oracle instance information items, see your Oracle documentation.

Table 3–7: PFM - Agent for Oracle instance information

Item	Description	Specifiable value	Default
oracle_sid	Monitoring-target Oracle system identifier (the same value as the value of the ORACLE_SID environment variable)	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	The value specified for the -inst option of the jpcconf inst setup command
oracle_home[#1]	Oracle home folder used by PFM - Agent for Oracle (the same value as the value of the ORACLE_HOME environment variable)	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	--
oracle_version[#1]	Version number of Oracle Database used by PFM - Agent for Oracle When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), 12 is set.	A two-digit number. • Oracle 10g: 10 • Oracle 11g: 11 • Oracle 12c or later: 12	In HP-UX: 10 In OSs other than HP-UX: 11
oracle_user[#2]	An account for monitoring Oracle For details about accounts that can be specified and the required privileges, see *(3) Create an Oracle account to be used in PFM - Agent for Oracle*.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	sys
oracle_passwd[#2, #3]	A password for the account that was specified in oracle_user	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	--
sqlnet[#1,#4]	Specify Y for any of the following three cases: 1. If using an Oracle RAC configuration For details on the RAC configuration, see the Oracle documentation. 2. If using a PD_PDIA record to monitor the availability of a listener. 3. If the monitoring target is PDB in a CDB configuration	{ Y | N }	N

Item	Description	Specifiable value	Default
sqlnet[#1,#4]	of Oracle Database 12c R2 or later. If you specify N for the case 1 or 3, above, Oracle might encounter an error.	{ Y | N }	N
net_service_name[#1]	The net service name of a monitoring-target database. This value is enabled if you specified Y in sqlnet. For details about the net service name of a monitoring-target database, see your Oracle documentation.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Instance name (the value of oracle_sid)
listener_home[#1]	Specify the environment variable ORACLE_HOME of the Oracle component containing the listener you want to monitor. Specify environment variable GRID_HOME in the case of Oracle RAC environment.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Instance name (the value of oracle_home)
listener_name	Specify the name of one listener that you want to monitor using the PDLS record. If a PDLS record is not being used to monitor the listener, specify the default listener name "LISTENER". This is because even though the specified value is not used, a blank is not permitted.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Default listener name (LISTENER)
log_path[#5]	The full path name of the directory for storing agent log information	A character string of 245 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < > Notes: • You can specify the path to a directory under the installation directory only when the default directory is set. • You cannot specify the path to a directory that is used as the output destination of another instance.	/opt/jp1pc/agto/agent/*instance-name*/log
log_size[#5]	The maximum size of one agent log file[#6]	1 to 32 (in megabytes). The recommended value is 16 or greater.	16

Item	Description	Specifiable value	Default
timeout[#5,#7]	The timeout period for Oracle access during a query.	0, or 10 to 3600 (in seconds). When 0 is specified, timeout monitoring is not performed. When a value from 1 to 9 is specified, it is changed to 10 at runtime. For details about timeouts, see *3.4.3 Cancellation facility for Oracle access during record collection*.	0
sql_option[#5,#8]	When Y is specified, information about the following items[#8] is not collected for PI_PIDB and PD_PDTS records, and the value specified by 0 or numeric_10 is set.	{ Y | N }	N
numeric_10[#9]	When sql_option is set to Y, the value specified is set for items for which information is not collected. If sql_option is set to N, this specification is disregarded.	0 to 99999. Note that if the value specified exceeds the maximum value for the data format of the set field (32767 for short and 65535 for ushort), the maximum value for the data format is set.[#10]	0
startup_always	PFM - Agent for Oracle may stop due to an Oracle connection error, such as when the monitored Oracle is still starting up when PFM - Agent for Oracle starts up. If Y is specified, start processing continues even if a connection error occurs. If N is specified, start processing will stop if an error occurs.	{ Y | N }	Y
Store Version[#11]	Specify the version of the Store database to be used. For details about the version of the Store database, see 3.4.4 Updating the Store version to 2.0.	{ 1.0 | 2.0 }	2.0
localtemp_option[#12]	Option for switching the display of the free space of the locally managed temporary tablespace of PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTF, PD_PDTS, and PD_PCTS records. If Y is specified, display the size of the free space. If N is specified, display the size of the unallocated space.	{ Y | N }	N
nls_lang[#13]	Option for specifying the character encoding used for communication between PFM -	Character code set: Specifiable values vary depending on the combination of the OS LANG	AMERICAN_AMERICA.US7ASCII

Item	Description	Specifiable value	Default
nls_lang[#13]	Agent for Oracle and Oracle Database.	and NLS_CHARACTER_SET of Oracle Database. For details, see *Table 3-9*.	AMERICAN_AMERICA.US7ASCII
undospace_option[#14]	Option for switching how the value displayed as the free space in the UNDO tablespace of the PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTS, PD_PCTS, and PD_CDTS records is determined. If N is specified, the size of the unallocated space is displayed. If Y is specified, the size of the free space is displayed.	{ Y | N }	Y

Legend:

--: None

#1

Depending on the OS, the required settings differ. The following table shows the OS and how to specify the settings.

OS	Method for specifying settings
Linux	Specify settings according to the explanation in the following section *Settings in Linux*.
AIX	Specify settings according to the explanation in the following section *Settings in AIX or Solaris*.
HP-UX(IPF)	Specify the settings according to the explanation in *Table 3-7*.
Solaris	Specify settings according to the explanation in the following section *Settings in AIX or Solaris*.

Setting in Linux

For monitoring of the Oracle database versions in a Linux environment, the 32-bit Oracle Client has been the required product in PFM - Agent for Oracle 10-50 or earlier. However, in PFM - Agent for Oracle 11-00 or later, the Oracle Client 64-bit library is used for monitoring.

Because instance information settings have been changed in version 11-00 or later as shown in the following table, make sure that the information is set up correctly:

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
oracle_home	Specify the Oracle home folder for Oracle Client 32-bit.	Specify the Oracle home folder for Oracle Database.
oracle_version	Specify the version of Oracle Client 32-bit.	Specify the version of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), 12 is set.
sqlnet	Specify Y	Specify a value according to the conditions described in *Table 3-7*.
net_service_name	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.	If you specify Y for sqlnet, specify this item. Specify the name of the net service that you set in the Oracle database to be monitored and that can connect to the Oracle database.

Notes:

- PFM - Agent for Oracle 11-00 or later uses the Oracle Client 64-bit library included with the Oracle database, so you do not need to install the 64-bit Oracle Client.
- To upgrade PFM - Agent for Oracle 10-50 or earlier to 11-00 or later, instance information is subject to change, and must therefore be updated before starting the PFM - Agent for Oracle service.

  Note that Oracle Client 32-bit is no longer needed, and can be uninstalled if it is not used by any product other than PFM - Agent for Oracle.
- If you specify `ORACLE_HOME` for Oracle Client 32-bit and start PFM - Agent for Oracle, the `KAVF12011-E` and `KAVF12021-E` message appear.

Setting in AIX or Solaris

For monitoring of the Oracle database versions in an AIX environment or a Solaris environment, the 32-bit Oracle Client has been the required product.

The following table shows the instance information for which you need to specify the environment settings, and the settings themselves.

Item	Description
`oracle_home`	Specify the Oracle home folder for Oracle Client 32-bit.
`oracle_version`	Specify the version of Oracle Client 32-bit. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.
`sqlnet`	Specify `Y`
`net_service_name`	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.
`listener_name`	Specify the Oracle home folder for the monitoring-target Oracle Database. In the case of Oracle RAC environment, specify the home of Oracle Grid Infrastructure.

Notes:

- If you specify ORACLE_HOME from the Oracle database or ORACLE_HOME from the 64-bit Oracle Client for the instance information item `oracle_home` to start PFM - Agent for Oracle while the installation of the 32-bit Oracle Client and the creation of the environment are not complete, the `KAVF12011-E` and `KAVF12021-E` messages appear.
- You need to install the 32-bit Oracle Client by specifying Administrator or run time for the installation type. Instant Client is not supported.

#2

PFM - Agent for Oracle runs using Oracle password authentication.

#3

If the expiration date is set on `oracle_passwd`, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors perform either of the following procedures before the password is expired:

- Unset the expiration date of the password
- After updating password, execute the `jpcconf inst setup` command to update `oracle_passwd`.

Note that the Oracle default profile is applied to the Oracle account created by `mk_user.sql`.

#4

Specify whether to use Oracle network services.

- If `Y` is specified:

PFM - Agent for Oracle connects to Oracle via a listener that is made up of Oracle network services.

In this case, you must set the Oracle network service definitions (such as tnsnames.ora and listener.ora).

When monitoring Oracle Database instances in an Oracle RAC configuration, set up the PFM - Agent for Oracle so that it monitors Oracle Database instances on each node.

When using a CDB configuration of Oracle Database 12c Release 2 or later, set this setting in case of monitoring each PDB. In case of monitoring the root container (CDB$ROOT), this setting is not required.

For details about how to set up, see the Oracle documentation.

Store the tnsnames.ora file in the following directory.

`oracle_home/network/admin`

If you store the tnsnames.ora file in any other location, must therefore set to `TNS_ADMIN` environment variable in starting user of PFM - Agent for Oracle before starting the PFM - Agent for Oracle service. For details about the `TNS_ADMIN` environment variable, see the Oracle documentation.

- If `N` is specified:

  PFM - Agent for Oracle connects to the local database without using the Oracle network services.

#5

When PFM - Agent for Oracle is upgraded from a version earlier than 08-00, the default values are set.

#6

A maximum of 4 agent log files are collected for one instance. Before specifying the `log_size` value, make sure that the value satisfies the following condition (this condition also applies when `log_path` is set to the default):

```
Amount of free space on the disk containing the directory specified in log
_path (MB) > log_size * 4
```

If the free disk space is insufficient, agent log cannot be output. For details about the agent log, see *8.3 Log information*.

#7

Set the timeout value according to the time needed to collect records during heavy load (peak time).

#8

To obtain each piece of Oracle segment-related information, PFM - Agent for Oracle searches Oracle's static data dictionary views `DBA_SEGMENTS`. If a large number of segments (more than hundreds of thousands) exist for Oracle, information collection requires a significant amount of time. As such, when a large number of segments exist, and the information listed in the following table no longer needs to be collected, set the `sql_option` to `Y` during operation.

Table 3–8: Record names and the values specified for numeric_10 (setting up instance information)

Record name	PFM - View name	Value specified for numeric_10
`PD_PDTS`	`Segments`	Enabled
	`Extents`	Enabled
`PI_PIDB`	`DB Files %`	Enabled
	`Log Files %`	Enabled
	`NextAlloc Fails`	Enabled
	`Tablespaces`	Enabled
	`Rollback Segments`	Enabled
	`Rollback Segments Trans`	Enabled

Record name	PFM - View name	Value specified for numeric_10
PI_PIDB	Blocks	Enabled
	Segments	Enabled
	Extents	Enabled
	Free Mbytes	Enabled
	Overextended	Enabled
	High Max Extent	Enabled
	Datafiles	Enabled
	Mbytes	Enabled
	Free Extents	Enabled
	Free%	Enabled
	Free Change	Enabled
	Write%	Enabled
	Write/sec	Enabled
	Redo Files	Enabled
	Links	Enabled
	Links Logged On	Enabled
	Links In Tran	Enabled
	Links Open Cursors	Enabled
	Used Change	Enabled
	Used Mbytes	Enabled
	Rollback Segments Hit%	Enabled
	Sort Segments	Enabled
	Sorting Users	Enabled
	Physical Blocks Read	Always set to 0 because it is a delta item.
	Physical Blocks Written	Always set to 0 because it is a delta item.
	Physical Reads	Always set to 0 because it is a delta item.
	Physical Writes	Always set to 0 because it is a delta item.

#9

When displayed in PFM - Web Console, this item indicates whether the values set in each field in #8 are values collected from the Oracle Database, or fixed values.

#10

If the field format for each record is `float` or `double`, since the data is a floating-point number, it may be rounded depending on the specified value.

Example:

When `numeric_10` is set to `32767`, it may be displayed as `32760`.

#11

You can specify the Store Version item only when setting up a new instance environment. You cannot specify this item when updating an existing instance environment.

#12

When `localtemp_option` is set to `N`, collect free space of the locally managed temporary tablespace and information about the extents from v$temp_space_header of the dynamic performance view. The displayed values of the size of free space are the size of the unallocated space. Since the allocated space is not freed until the temporary tablespace is reconstructed or recreated, the displayed values of free space do not increase until the space is freed.

When `localtemp_option` is set to `Y`, collect free space of the locally managed temporary tablespace and information about the extents from v$sort_segment or v$temp_extent_pool of the dynamic performance view. The displayed values of the size of free space are calculated from the size of the used space.

When issuing query to v$temp_extent_pool view, the Oracle instance goes to sleep. Since this may have effect on the performance of the Oracle instance, you need adequate consideration before `localtemp_option` is set to `Y`. For details, see your Oracle documentation.

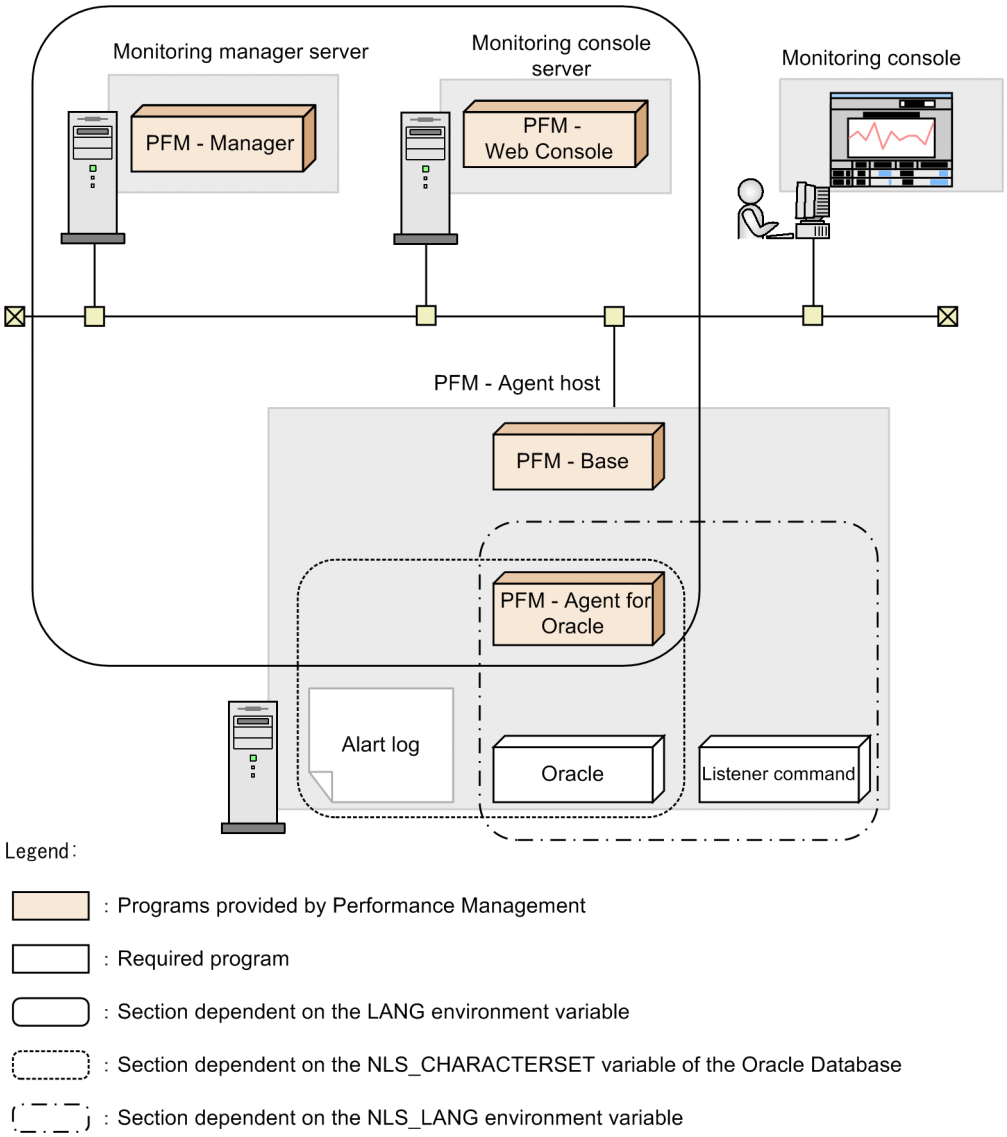
The following records use `v$temp_extent_pool` view:

- Data File (PD_PDDF)
- Data File Interval (PI_PIDF)

#13

The following figure shows the relationship among data, environment variable, and instance information handled by PFM - Agent for Oracle.

Figure 3–4: Relationship between data and the setting values



PFM - Agent for Oracle can collect performance data in SJIS format (for Japanese environment), EUC format (for Japanese environment), UTF-8 format (for Japanese and Chinese environment) and GB18030 format (for Chinese environment) as well as 7-bit ASCII format. To collect performance data in a format other than 7-bit ASCII format, you need to set the `nls_lang` instance information of PFM - Agent for Oracle.

Acceptable values for the `nls_lang` instance information depend on the combination of the language environment of the OS and the database character set of the monitored Oracle. The following table lists acceptable values for the `nls_lang` instance information for each combination.

Table 3–9: Relationship between the combination of the OS language environment and the database character set of the monitored Oracle and the nls_lang settings

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Linux	`ja_JP.UTF-8`, `ja_JP.utf8`	`AL32UTF8`	`AMERICAN_AMERICA.AL32UTF8` or `AMERICAN_AMERICA.US7ASCII`#
	`zh_CN.UTF-8`, `zh_CN.utf8`		

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Linux	zh_CN.gb18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
AIX	Ja_JP, Ja_JP.IBM-932	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja_JP, ja_JP.IBM-eucJP	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	JA_JP, JA_JP.UTF-8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	ZH_CN, ZH_CN.UTF-8		
	Zh_CN, Zh_CN.GB18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
HP-UX	ja_JP.SJIS, japanese	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja_JP.eucJP, japanese.euc	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	ja_JP.utf8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	zh_CN.utf8		
	zh_CN.gb18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
Solaris	ja_JP.PCK	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja, Japanese, ja_JP.eucJP	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	ja_JP.UTF-8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	zh_CN.UTF-8, zh_CN.UTF-8@pinyin, zh_CN.UTF-8@radical, zh_CN.UTF-8@stroke, zh.UTF-8		
	zh_CN.GB18030, zh_CN.GB18030@pinyin, zh_CN.GB18030@radical,	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Solaris	zh_CN.GB18030@stroke	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#

#

Performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

Notes:

For any other combination, performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

When you specify an invalid character code set for the nls_lang instance information, the message KAVF12302-W with errcode 12705 is output, and the connection with Oracle will fail.

In the following cases as well, unreadable characters might occur in the performance data:

1. The Oracle column length is exceeded.

   If you store data that exceeds the Oracle column length, the last character might become unreadable. If you use PFM - Agent for Oracle to collect data in Oracle and that data contains unreadable characters, the last character of the performance data will be unreadable.

2. The field size of PFM - Agent for Oracle is exceeded.

   PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

   Table 3–10: Applicable fields (Unreadable characters caused by cases where a field size of PFM - Agent for Oracle is exceeded)

Record name	Field name	Field size (Unit: bytes)
Database Object Cache (PD_PDDO)	Object Name	100
Errorlog Detail (PD_PDEL)	Message	512
Parameter Values (PD_PDP)	Value	512
SQL Text (PD_PDSQ)	Explain Plan	30,000
	SQL Text	30,000
SQL Text - Performance Based (PD_PDES)	SQL Text	10,000
Table Access (PD_PDTA)	Object	100

3. Unreadable or lack of characters in different between database character set of the Oracle and nls_lang instance variable of the PFM - Agent for Oracle.

   A data of 2 bytes on the basis of Oracle may be collected at 3 bytes when you set AMERICAN_AMERICA.AL32UTF8 in nls_lang and a database character set for the monitoring is not UTF-8. Therefore, if performance data takes from ORACLE that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Table 3–11: Applicable fields (Unreadable or lack of characters caused by cases where the database character set differs from nls_lang)

Record name	Field name	Field size (Unit: bytes)
Circuit (PD_PDCI)	User	30
Database Object Cache (PD_PDDO)	Object Name	100
	Owner	64
CDB Tablespace (PD_CDTS)	Tablespace Name	30
Collection Instance 2 (PD_PCI)	Recovery File Dest	513
Collection Tablespace 2 (PD_PCTS)	Tablespace Name	30
Data File (PD_PDDF)	File Name	513
	Tablespace Name	30
Data File Interval (PI_PIDF)	File Name	513
	Tablespace Name	30
Database (PD_PDDB)	DB Name	9
Database Interval (PI_PIDB)	DB Name	9
Instance (PD_PDI)	Host	30
Latch (PD_PDLA)	OS User	30
	Program	48
	User	30
Latch Interval (PI_PILA)	OS User	30
	Program	48
	User	30
Lock (PD_PDLO)	Program	48
	User	30
Lock Interval (PI_PILO)	Program	48
	User	30
Lock Waiters (PD_PDLW)	Holding User	30
	Waiting User	30
Minimum Database Interval 2 (PI_PMDB)	DB Name	9
Minimum Data File Interval 2 (PI_PMDF)	File Name	513
Minimum Tablespace Interval 2 (PI_PMTS)	Tablespace Name	30
Open Cursor (PD_PDOC)	Program	48
	SQL Text	60
Parameter Values (PD_PDP)	Value	512
Process Detail (PD_PDOP)	Program	48

Record name	Field name	Field size (Unit: bytes)
Process Detail (PD_PDOP)	User	15
Rollback Segment (PD_PDRS)	Tablespace Name	30
Rollback Segment Interval (PI_PIRS)	Tablespace Name	30
Segment Detail (PD_PDSM)	Owner	30
	Segment Name	81
	Tablespace Name	30
Session Detail (PD_PDS)	Machine	64
	Module	48
	OS User	30
	Program	64
	Schema Name	30
	User	30
Session Event (PD_PDEV)	Program	64
	User	30
Session Event Interval (PI_PIEV)	Program	64
	User	30
Session I/O Interval (PI_PIIO)	User	30
Session Statistics Summary (PD_PDS2)	Program	48
	User	30
Session Stat Summary Interval (PI_PIS2)	Program	48
	User	30
Session Statistics (PD_PDSS)	Program	48
	User	30
Session Wait (PD_PDWA)	Program	48
	User	30
Shared Cursor Cache (PD_PDC)	SQL Text	1,000
Sort Segment (PD_PDSR)	Tablespace Name	31
Sort Segment Interval (PI_PISR)	Tablespace Name	31
SQL Text (PD_PDSQ)	Explain Plan	30,000
	SQL Text	30,000
SQL Text - Performance Based (PD_PDES)	Parsing User	30
	SQL Text	10,000
Table Access (PD_PDTA)	Object	100
	Owner	64
	Program	48

Record name	Field name	Field size (Unit: bytes)
Table Access (PD_PDTA)	User	30
Tablespace Fragmentation (PD_PDTF)	Tablespace Name	30
Tablespace Interval (PI_PITS)	Tablespace Name	30
Tablespace (PD_PDTS)	Tablespace Name	30
Transaction (PD_PDTR)	User	30
Transaction Interval (PI_PITR)	User	30
Transaction Lock (PD_PDTL)	Object Name	30
	Owner	30
	User	30

#14

When undospace_option is set to N, the size of the unallocated space is collected as the amount of free space in the UNDO tablespace. Space in the UNDO tablespace that becomes available because its retention period has expired is treated as allocated space until it is released.

When undospace_option is set to Y, the size of the free space is collected as the amount of free space in the UNDO tablespace. Space of the UNDO tablespace that becomes available because its retention period has expired is included in the size of the free space.

The following table shows the fields whose values change depending on the specification of the undospace_option:

Table 3–12: the fields whose values change depending on the specification of the undospace_option

Record name	Field name
CDB Tablespace (PD_CDTS)	Free Mbytes
	Free %
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Data File(PD_PDDF)	Free %
	Free Mbytes
	Used Mbytes
Data File Interval (PI_PIDF)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Database(PD_PDDB)	Free %
	Free Mbytes
	Used Mbytes

Record name	Field name
Database Interval (PI_PIDB)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Tablespace (PD_PDTS)	Free %
	Free Mbytes
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Collection Tablespace 2 (PD_PCTS)	Free Mbytes

Notes:

- The PFM - Agent for Oracle services can be started only when an instance environment has been set up.
- When you use the `jpcconf inst setup` command to create an instance environment, the command terminates normally even if an instance name that does not exist in Oracle is specified. However, if you then start record collection, message KAVF12401-W is output to the common message log, and you cannot connect to the monitored Oracle. If this problem occurs, check whether you specified the correct instance name, and re-execute the `jpcconf inst setup` command with the correct instance name specified.
- Do not use PFM - Agent for Oracle and PFM - RM for Oracle to monitor the same Oracle instance. If so, the normal operation of PFM - Agent for Oracle is not guaranteed.

An instance environment is created by using the `jpcconf inst setup` command. The following procedure shows how to create an instance environment.

To create an instance environment:

1. Execute the `jpcconf inst setup` command with a service key and instance name specified.

   For example, when you want to create the instance environment for the PFM - Agent for Oracle instance named `SDC`, use the following command line:

   ```
   jpcconf inst setup -key Oracle -inst SDC
   ```

   Note that you cannot use `sql` as an instance name.

   Although an example of interactive command execution is shown here, the jpcconf inst setup command can be also executed non-interactively. For details about the `jpcconf inst setup` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

2. Set up Oracle instance information.

   Enter the values for the items listed in *Table 3-7* as prompted by the command. All items are mandatory. To use the default value displayed for an item, only press the **Enter** key.

When all values have been entered, the instance environment is created. If you want to change the instance information, re-execute the `jpcconf inst setup` command to update the instance environment. For details about updating an instance environment, see *3.4.2 Updating an instance environment*.

The following describes the created instance environment.

- Directory configuration of the instance environment
  The instance environment is set up in the following directory:
  For a physical host: `/opt/jp1pc/agto`
  For a logical host: *environment-directory*[#]`/jp1pc/agto`
  #
  The environment directory is a directory on the shared disk specified when the logical host was created.
  The following table describes the directory configuration of the created instance environment.

  Table 3–13: Directory configuration of the instance environment

Directory and file			Description
`agent`	*instance-name*	`jpcagt.ini`	Agent Collector service startup initialization file
		`jpcagt.ini.model`[#]	Model file for the Agent Collector service startup initialization file
		`log`	Storage directory for log files
`store`	*instance-name*	`jpcsto.ini`	Agent Store service startup initialization file
		`jpcsto.ini.model`[#]	Model file for the Agent Store service startup initialization file
		`*.DAT`	Data model definition file
		`dump`	Export destination directory
		`import`	Standard database import destination directory (for Store version 2.0)
		`backup`	Backup destination directory
		`log`	Storage directory for log files
		`partial`	Standard database partial backup destination directory (for Store version 2.0)
		`STPD`	Performance data storage destination directory for records of the PD record type(for Store version 2.0)
		`STPI`	Performance data storage destination directory for records of the PI record type (for Store version 2.0)

  #
  This file is used to reset all values to the initial values set when the instance environment was created.

- Service ID for the instance environment
  The service for the instance environment has the following format:
  - Agent Collector service:
    `OA`*instance-number instance-name* [*host-name*]
  - Agent Store service:
    `OS`*instance-number instance-name* [*host-name*]

  In PFM - Agent for Oracle, the instance name specified in the `jpcconf inst setup` command is displayed.
  For example, if you execute the command with host name host1 and instance name `SDC`, the service IDs will be as follows:
  - Agent Collector service:

OA1SDC [host1]

- Agent Store service:
  OS1SDC [host1]

For details about the service ID, see the naming rules described in Appendix in the *JP1/Performance Management Planning and Configuration Guide*.

(b) Registering objects in the Oracle Database

To use PFM - Agent for Oracle to monitor an Oracle Database, you must register the objects provided by PFM - Agent for Oracle in the Oracle Database. The objects are registered by using an SQL script provided by PFM - Agent for Oracle. The following procedure shows how to execute the SQL script. Note that the procedure is used only once for each account with which the Oracle Database instance is to be monitored.

If Oracle Database 12c Release 2 or later is to be monitored in a CDB configuration, register the objects only when instances of PFM - Agent for Oracle for monitoring PDBs have been created. If instances of PFM - Agent for Oracle for monitoring the root container (CDB$ROOT) have been created, there is no need to register the objects. Similarly, there is no need to register the objects when PDBs are to be monitored from the root container (CDB$ROOT) to which they belong, because, in this case, instances of PFM - Agent for Oracle are created for the root container (CDB$ROOT).

Monitoring target	Object registration necessary?
Root container (CDB$ROOT) Note that the monitoring target is the root container (CDB$ROOT) when all PDBs are monitored from the root container (CDB$ROOT) to which they belong.	No.
PDB	Yes. Execute sp_inst.sql on the PDBs to be monitored. When the sys account is to be used for monitoring, execute sp_inst.sql for each PDB to be monitored.

1. Set up an environment where the sqlplus Oracle command can be executed.
   For details about Oracle environment setup, see your Oracle documentation.
2. Navigate to the following directory, which contains the sp_inst.sql file provided by PFM - Agent for Oracle:

```
/opt/jp1pc/agto/agent/sql
```

3. Execute the sp_inst.sql script for the Oracle Database that you want to monitor.
   Connect to the Oracle Database by using the account specified by oracle_user in the instance information, and then execute the sp_inst.sql script.
   The sp_inst.sql script will register with Oracle the objects (procedures for monitoring and tables for operation) PFM - Agent for Oracle needs to perform Oracle monitoring.
   Note:
   Do not use the sp_inst_seg2.sql script.
   Example:
   If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration
   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-database* @sp_inst.sql

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

`sqlplus` *Oracle-account*`/`*password-for-the-Oracle-account*`@`*net-service-name-for-the-monitoring-target-PDB* `@sp_inst.sql`

- The `sqlplus` command is provided by Oracle Corporation.
- Specify the `oracle_user` value as the Oracle account. The objects are created in the database with the Oracle account used here. You must specify the same Oracle account when setup of the instance environment is canceled.
- When you use a SYS user for the Oracle account, executing the `sp_inst.sql` script without specifying the `AS SYSDBA` option may result in an error. If an error occurs, execute the script with the `AS SYSDBA` option specified.

When the above command is executed, the table and packages shown in the following table are created.

Table 3–14: Table and packages to be created

Table	Package
`LSC_13_PLAN_TABLE`#	LSC_13_PDAS, LSC_13_PICS, LSC_13_73_PDDB, LSC_13_PDDB2, LSC_13_PDI, LSC_13_717273_PDMT, LSC_13_PDS3, LSC_13_73_PIDB, LSC_13_PIDB2, LSC_13_PIDB3

#

`LSC_13_PLAN_TABLE` is only used during collection of the SQL Text (`PD_PDSQ`) record. Therefore, when you collect the SQL Text (`PD_PDSQ`) record, make sure that at least 5 megabytes of free space is allocated to the default tablespace.

(c) Setting up the Oracle Database

To use the records provided by PFM - Agent for Oracle to collect the performance data items listed in the following table, you must set the `TIMED_STATISTICS` Oracle Database initialization parameter to `TRUE`.

Table 3–15: Items that can be collected only when TIMED_STATISTICS=TRUE is set

Record	Field	Remarks
`ASM Disk (PD_PDDK)`	`Read Time (READ_TIME)`	--
	`Write Time (WRITE_TIME)`	--
`Block Contention Interval (PI_PIBC)`	--	Entire record
`Block Contention Statistics (PD_PDBC)`	--	Entire record
`Data File Interval (PI_PIDF)`	`Write Time (WRITE_TIME)`	--
`Session Detail (PD_PDS)`	`Avg Wait (AVERAGE_WAIT)`	--
	`Avg Wait String (AVERAGE_WAIT_STRING)`	--
	`Time Waited (TIME_WAITED)`	--
	`Time Waited String (TIME_WAITED_STRING)`	--
`Session Event (PD_PDEV)`	`Avg Wait (AVERAGE_WAIT)`	--
	`Avg Wait String (AVERAGE_WAIT_STRING)`	--
	`Time Waited (TIME_WAITED)`	--

Record	Field	Remarks
Session Event (PD_PDEV)	Time Waited String (TIME_WAITED_STRING)	--
Session Event Interval (PI_PIEV)	Avg Wait (AVERAGE_WAIT)	--
	Avg Wait String (AVERAGE_WAIT_STRING)	--
	Time Waited (TIME_WAITED)	--
	Time Waited String (TIME_WAITED_STRING)	--
Session Stat Summary Interval (PI_PIS2)	Statement CPU (STATEMENT_CPU)	--
Session Statistics Summary (PD_PDS2)	Statement CPU (STATEMENT_CPU)	--
System Stat Summary (PD)	Session CPU Usage (SESSION_CPU_USAGE)	--
System Stat Summary Interval (PI)	Session CPU Usage (SESSION_CPU_USAGE)	--
Session Wait (PD_PDWA)	Wait Time (WAIT_TIME)	--
	Wait Time String (WAIT_TIME_STRING)	--
System Event (PD_PDSE)	Avg Wait (AVERAGE_WAIT)	--
	Time Waited (TIME_WAITED)	--
System Event Interval (PI_PISE)	Avg Wait(AVERAGE_WAIT)	--
	Time Waited (TIME_WAITED)	--

Legend:

--: None

Notes:

- If you modify the initialization parameters file, you must restart the instance's database.
- Starting from Oracle 9i, the server parameters file is supported to store Oracle parameter information.A value change you make in the server parameters file may take precedence over a change made to the initialization parameters file.
- Setting the TIMED_STATISTICS initialization parameter to TRUE may have adverse effects on the performance of the Oracle Database. If you plan to use this setting, you should first evaluate the possible effects. For details, see your Oracle documentation.

(5) Specifying network settings Option

You must specify the network settings according to the configuration in which Performance Management is used.

The following are the two network setting items:

- IP addresses

  Set the IP addresses when using Performance Management in a network environment where multiple LANs are connected. You can set multiple IP addresses by defining the host names and IP addresses in the jpchosts file. Use the same jpchosts file throughout the Performance Management system.

For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

- Port numbers

  Specify the port numbers to be used by Performance Management. To avoid conflicts, use the same port numbers and service names across the Performance Management system.

  For details about setting port numbers, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

(6) Change the size of log files Option

Performance Management outputs its operating status to a set of proprietary log files called a *common message log*. Perform this setting if you want to change the default file size.

For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

(7) Change the storage location of performance data Option

Perform this setting if you want to change the folders where the database of performance data managed by PFM - Agent for Oracle is saved, backed up, or exported.

By default, performance data is saved in the following locations:

- Save destination directory: `/opt/jp1pc/agto/store/`*instance-name*`/`
- Backup destination directory: `/opt/jp1pc/agto/store/`*instance-name*`/backup/`
- Partial backup destination directory#: `/opt/jp1pc/agto/store/`*instance-name*`/partial`
- Export destination directory: `/opt/jp1pc/agto/store/`*instance-name*`/dump/`
- Import destination directory#: `/opt/jp1pc/agto/store/`*instance-name*`/import`

Note:

For the default save destination for logical host operation, replace `/opt/jp1pc` with *environment-directory*`/jp1pc`.

#

This folder can be set only when the Store version is 2.0.

For details, see *3.4.1 Changing the storage location of performance data*.

(8) Set the connection-target PFM - Manager of PFM - Agent for Oracle

On each host on which a PFM - Agent program is installed, set the connection-target PFM - Manager program that is to manage the PFM - Agent. Use the `jpcconf mgrhost define` command to set the connection-target PFM - Manager.

Notes:

- When multiple instances of PFM - Agent are installed on a single host, you can specify only one PFM - Manager as their connection target. You cannot have a different instance of PFM - Manager as the connection target for each PFM - Agent.
- When PFM - Agent is installed on the same host as PFM - Manager, the connection-target PFM - Manager will be the PFM - Manager on the local host. You cannot change this.

To set the connection-target PFM - Manager:

1. Stop all Performance Management programs and services.

   Stop all active Performance Management programs and services on the host before beginning setup. For details about how to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

   If any Performance Management programs or services are running when you execute the `jpcconf mgrhost define` command, you will be prompted by a message to stop the programs or services.

2. Execute the `jpcconf mgrhost define` command with the host name of the PFM - Manager host you wish to use specified as the connection target.

   For example, if the PFM - Manager you wish to use as the connection target resides on host `host01`, execute the command as follows:

   ```
   jpcconf mgrhost define -host host01
   ```

Although an example of interactive command execution is shown here, the `jpcconf mgrhost define` command can be also executed non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(9) Setting up the action log Option

This setting is required to output action logs when alarms occur. 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 *K. Outputting Action Log Information*.

3.2 Setup cancellation and uninstallation

This section describes the procedures for uninstalling and canceling setup of PFM - Agent for Oracle.

3.2.1 Cautionary notes on setup cancellation and uninstallation

Note the following when uninstalling and canceling setup of PFM - Agent for Oracle.

(1) Note on OS user permission required to uninstall PFM - Agent for Oracle

Before you uninstall PFM - Agent, make sure that you have logged on with an account that belongs to the Administrators group.

(2) Note on network configuration

When you uninstall a Performance Management program, the port numbers defined in the `services` file will remain in the file.

(3) Notes on programs

- If you uninstall a Performance Management program while another Performance Management program or service or other program that references Performance Management files is running, some files or folders may remain in the system. In this case, manually delete everything under the installation folder.
- If both PFM - Base and PFM - Agent are installed on a host, you cannot uninstall PFM - Base without first uninstalling PFM - Agent. In this case, uninstall PFM - Agent and then PFM - Base, in that order. The same applies when both PFM - Manager and PFM - Agent are installed on a host. You will be unable to uninstall PFM - Manager without first uninstalling PFM - Agent. In this case, uninstall PFM - Agent and then PFM - Manager, in that order.

(4) Notes on services

- Uninstalling PFM - Agent does not delete the information about the service from the list that appears when you execute the `jpctool service list` command. To delete this information, use the `jpctool service delete` command.
- For details about deleting the information about the service, see the section about deleting the service in the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

Note that when you want to update the PFM - Web Console host to reflect the deletion of service information, you need to execute the `jpctool service sync` command to synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host.

(5) Other notes

- Before you start uninstallation, use the `jpcconf inst setup` command or PFM - Web Console to check the agent log output folder. If this folder has been changed from the default folder (`/opt/jp1pc/agto/agent/`*instance-name*`/log`), the agent log files remain after uninstallation. Manually delete these files after uninstallation.
- When uninstalling a Performance Management program from a host on which PFM - Web Console is installed, close all browser windows before you uninstall the program.

3.2.2 Procedure for canceling setup

This subsection describes how to cancel setup of PFM - Agent for Oracle.

(1) Canceling setup of an instance environment

Canceling setup of an instance environment involves the tasks listed below. To cancel setup of multiple instance environments, you must repeat the procedure for each environment.

- Deleting an instance environment
- Deleting the objects registered in the Oracle Database

The following describes the above tasks.

(a) Deleting an instance environment

Check the instance name and delete the instance environment. Deletion of an instance environment is performed from a PFM - Agent host.

To check the instance name, use the `jpcconf inst list` command. To delete an instance environment that has been created, use the `jpcconf inst unsetup` command.

To delete an instance environment:

1. Check the instance name.

   Execute the `jpcconf inst list` command specified with the service key that indicates PFM - Agent for Oracle. The following shows the command format:

   ```
   jpcconf inst list -key Oracle
   ```

   For example, if the instance name is `SDC`, the command displays `SDC`.

2. Stop all active PFM - Agent services in the instance environment.

   For details about how to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Delete the instance environment.

   Execute the `jpcconf inst unsetup` command specified with the service key that indicates PFM - Agent for Oracle and the instance name.

   For example, if the instance name is `SDC`, use the following command line:

   ```
   jpcconf inst unsetup -key Oracle -inst SDC
   ```

   If execution is successful, the folders created as the instance environment are deleted, as well as the service IDs.

Note:

Canceling setup of an instance environment does not delete the service information that is displayed with the `jpctool service list` command. Use the `jpctool service delete` command to delete service information.

If you want to update the PFM - Web Console host to reflect the deletion of instance environment, you need to execute the `jpctool service sync` command to synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host.

The following shows sample conditions and a command line applicable for the conditions.

- Instance name: SDC
- Host name: host03
- Service ID of the Agent Collector service: OA1SDC[host03]
- Service ID of the Agent Store service: OS1SDC[host03]

```
jpctool service delete -id O?1SDC[host03] -host host03
```

For details about the command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(b) Deleting the objects registered in the Oracle Database

This subsection describes the procedure for deleting the table and packages that were created in the Oracle Database being monitored. To execute this procedure, you must use the same Oracle account that you used when you registered the objects in the Oracle Database. Note that this procedure must be used only once for each account that is used to monitor the Oracle Database instance.

Note:

If you execute the procedure described below when no objects have been registered in the Oracle Database, an Oracle error message is displayed during execution of the sp_drop.sql script.

To delete the objects registered in the Oracle Database:

1. Set up an environment where the Oracle sqlplus command can be executed.

   For details about Oracle environment setup, see your Oracle documentation.

2. Navigate to the following folder that contains the sp_drop.sql script provided by the PFM - Agent for Oracle:

   ```
   /opt/jp1pc/agto/agent/sql
   ```

3. Execute the sp_drop.sql script on the Oracle Database being monitored.

   PFM - Agent for Oracle deletes monitoring procedures, work tables, and other objects that are required to monitor Oracle from Oracle.

   Example:

   If the monitoring target is Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 or later in a non-CDB configuration

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-of-the-monitoring-target-database* @sp_drop.sql

   If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration

   sqlplus *Oracle-account*/*password-for-the-Oracle-account*@*net-service-name-for-the-monitoring-target-PDB* @sp_drop.sql

   - sqlplus is a command provided by Oracle Corporation.
   - *Oracle-account* is the same Oracle account that was used to register the objects in the database.

   For Oracle 10g or later, LSC_13_PLAN_TABLE is placed in DBA_RECYCLEBIN and is not deleted completely. If you want to delete LSC_13_PLAN_TABLE completely, execute the PURGE TABLE LSC_13_PLAN_TABLE; command.

   Note that if the Oracle account is sys, LSC_13_PLAN_TABLE is not stored in DBA_RECYCLEBIN. Therefore, you do not need to execute the PURGE TABLE LSC_13_PLAN_TABLE; command.

4. Reset the value of the TIMED_STATISTICS Oracle initialization parameter.

   If the value of the TIMED_STATISTICS Oracle initialization parameter has been changed in order to collect records of PFM - Agent for Oracle, reset the value, if necessary.

(2) Deleting an Oracle account used in PFM - Agent for Oracle

Oracle accounts used in PFM - Agent for Oracle are authorized to change the objects of other schemas freely in order to monitor the Oracle Database. For this reason, unnecessary Oracle accounts must be deleted. If the tablespaces that were used by a deleted account are unnecessary, also delete the tablespaces.

(a) Deleting an Oracle account

To delete an Oracle account, issue the DROP USER statement in an environment where the sqlplus Oracle command can be executed. Before issuing the statement, make sure that your Oracle account has the DROP USER system privilege.

To delete an Oracle account:

1. Issue the DROP USER statement.

   Example:

   ```
   DROP USER Oracle-account CASCADE;
   ```

   If you add the CASCADE option, you can also delete the objects owned by the account.

For details about the DROP USER statement, see your Oracle documentation.

(b) Deleting the tablespaces used by a deleted Oracle account

When an Oracle account is deleted, the tablespaces used by the Oracle account become unnecessary. To delete these tablespaces, issue the DROP TABLESPACE statement in an environment where the sqlplus Oracle command can be executed. Before issuing the statement, make sure that your Oracle account has the DROP TABLESPACE system privilege.

To delete tablespaces:

1. Issue the DROP TABLESPACE statement.

For details about the DROP TABLESPACE statement, see your Oracle documentation.

3.2.3 Procedure for uninstallation

To uninstall PFM - Agent for Oracle:

1. On the host from which you want to uninstall a Performance Management program, log in as a superuser. Alternatively, use the su command to become a superuser.

2. On the local host, stop all Performance Management programs and services.

   Display the service information and check whether any services are running. Stop all Performance Management programs and services running on the local host. This includes services running on physical and logical hosts. For details about how to display service information and to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

3. Execute the following command to start Hitachi Program Product Installer:

In Linux:

```
/etc/hitachi_x64setup
```

In other than Linux:

```
/etc/hitachi_setup
```

Hitachi Program Product Installer starts, and displays the initial screen.

4. On the initial screen, enter **D**.

   A list of programs that can be uninstalled appears.

5. Select the Performance Management program you want to uninstall, and enter **D**.

   The selected program is uninstalled. To select the program, position the cursor at the program and press the space bar.

6. When uninstallation has terminated normally, enter **Q**.

   The initial screen of Hitachi Program Product Installer re-appears.

3.3 Changing the system configuration of PFM - Agent for Oracle

You may need to change the PFM - Agent for Oracle system configuration because of a change in the network configuration or host name of the monitored system.

When you change the PFM - Agent for Oracle system configuration, you must also change the settings for PFM - Manager and PFM - Web Console. For details about how to change the Performance Management system configuration, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

For some kinds of PFM - Agent, changing the physical host name or alias name requires additional tasks specific to the PFM - Agent. However, PFM - Agent for Oracle does not require such additional specific tasks.

3.4 Changing the operation of PFM - Agent for Oracle

In some circumstances, such as when changes are made to the way in which collected operation monitoring data is utilized, you may need to change how PFM - Agent for Oracle operates. This section explains how to change the operation method of PFM - Agent for Oracle.

For details about changing the operation method across the entire Performance Management system, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

3.4.1 Changing the storage location of performance data

Performance data collected by PFM - Agent for Oracle is managed in the Store database of the Agent Store service of PFM - Agent for Oracle. 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 folders to manage the collected performance data. These folders can be changed by using the `jpcconf db define` command.

Use the `jpcconf db define` command with -move option if you want to copy the performance data to the new storage location of the Store database.

For details about the `jpcconf db define` command, see the manual *JP1/Performance Management Reference*.

- Save destination folder
- Backup destination folder
- Partial backup destination folder#
- Export destination folder
- Import destination folder#

#

This folder 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 3–16: Options of the command that changes the performance data storage location

Description	Label name	Specifiable values (Store 1.0)	Specifiable values (Store 2.0)	Default value#
Save destination directory	`sd`	1 to 127 bytes absolute path name	1 to 222 bytes absolute path name	`/opt/jp1pc/agto/store/` *instance-name*
Backup destination directory	`bd`	1 to 127 bytes absolute path name	1 to 219 bytes absolute path name	`/opt/jp1pc/agto/store/` *instance-name*`/backup`
Partial backup destination directory	`pbd`	--	1 to 222 bytes absolute path name	`/opt/jp1pc/agto/store/` *instance-name*`/partial`

Description	Label name	Specifiable values (Store 1.0)	Specifiable values (Store 2.0)	Default value#
Maximum backup generation number	bs	1 to 9	1 to 9	5
Export destination directory	dd	1 to 127 bytes absolute path name	1 to 127 bytes absolute path name	/opt/jp1pc/agto/store/*instance-name*/dump
Import destination directory	id	--	1 to 222 bytes absolute path name	/opt/jp1pc/agto/store/*instance-name*/import

Legend:

--: This item cannot be set.

#

For the default save destination for logical host operation, replace /opt/jp1pc with *environment-folder*/jp1pc.

(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) Setting items in jpcsto.ini

The following table lists the label names for which information can be edited in the jpcsto.ini file, and other information such as the values that can be specified.

Table 3–17: Performance-data storage location settings (under [Data Section] in jpcsto.ini)

Description	Label name	Specifiable values (Store 1.0) #1	Default value#2
Save destination directory	Store Dir#3	1 to 127 bytes absolute path name	/opt/jp1pc/agto/store/*instance-name*
Backup destination directory (Full backup)	Backup Dir#3	1 to 127 bytes absolute path name	/opt/jp1pc/agto/store/*instance-name*/backup
Maximum backup generation number	Backup Save	1 to 9	5
Export destination directory	Dump Dir#3	1 to 127 bytes absolute path name	/opt/jp1pc/agto/store/*instance-name*/dump

#1

- The directory name must be an absolute path name or a relative path name from the default Store database directory (/opt/jp1pc/agto/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

For the default save destination for logical host operation, replace /opt/jp1pc with *environment-folder*/jp1pc.

#3

You cannot specify the same folders for Store Dir, Backup Dir, and Dump Dir.

(b) Before editing the jpcsto.ini file

- When changing the Store database folder, make sure that the folder after the change has already been created.
- When the Store database folder is changed, performance data collected before the change can no longer be used. If the performance data collected before the change is needed, carry 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 folder 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 folder.

(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 folder 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
        :
```

   Notes:

   - 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 folder (/opt/jp1pc/agto/store/*instance-name*) for the Store database of the Agent Store service. To change the storage destination, specify a relative path from the storage destination folder, or specify an absolute path.
   - The jpcsto.ini file also describes definition information in addition to the database storage destination folder. Therefore, do not change the values other than those in the [Data Section] section. If you change the values other than those in the [Data Section] section, Performance Management might 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 folder, the performance data files are not deleted from the previous folder. If these files are no longer necessary, delete only the following files:

- All files with the `.DB` extension
- All files with the `.IDX` extension

3.4.2 Updating an instance environment

To update an instance environment, check the name of the instance that you want to update, and change the instance information. The instance information is set from a PFM - Agent host.

Before you change an information item, check the following table. For details about Oracle instance information, see your Oracle documentation.

Table 3–18: PFM - Agent for Oracle instance information

Item	Description	Specifiable value	Default
`oracle_sid`	The value of this item can be updated. ID of the monitored Oracle system (same value as the `ORACLE_SID` environment variable).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value
`oracle_home`[#1]	The value of this item can be updated. Oracle home folder of Oracle Database (same value as the `ORACLE_HOME` environment variable).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value
`oracle_version`[#1]	The value of this item can be updated. Version number of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.	A two-digit number. • Oracle 10g: 10 • Oracle 11g: 11 • Oracle 12c or later: 12	Previous value
`oracle_user`[#2]	The value of this item can be updated. Account for monitoring Oracle (for details about accounts that can be specified, and the required privileges, see *3.1.4(3) Create an Oracle account to be used in PFM - Agent for Oracle*).	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: `, < >`	Previous value

Item	Description	Specifiable value	Default
oracle_passwd[#3]	The value of this item can be updated. Specify the password for the account specified in oracle_user.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
sqlnet[#1,#4]	The value of this item can be updated. Specify Y for any of the following three cases: 1. If using an Oracle RAC configurationFor details on the RAC configuration, see the Oracle documentation. 2. If using a PD_PDIA record to monitor the availability of a listener. 3. If the monitoring target is PDB in a CDB configuration of Oracle Database 12c R2 or later. If you specify N for the case 1 or 3, above, Oracle might encounter an error.	{ Y | N }	Previous value
net_service_name[#1]	The value of this item can be updated. Net service name of the monitored database. This value is enabled if you specified Y in sqlnet. For details about the net service name of the monitored database, see the Oracle documentation.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
listener_home[#1]	The value of this item can be updated. Specify the environment variable ORACLE_HOME of the Oracle component containing the listener you want to monitor. Specify environment variable GRID_HOME in the case of Oracle RAC environment.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
listener_name	The value of this item can be updated. Specify the name of one listener that you want to monitor using the PDLS record. If a PDLS record is not being used to monitor the listener, specify the	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value

Item	Description	Specifiable value	Default
listener_name	default listener name "LISTENER". This is because even though the specified value is not used, a blank is not permitted.	A character string of 255 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < >	Previous value
log_path[#5]	The value of this item can be updated. Specify the absolute path name of the agent log output folder.	A character string of 245 or fewer bytes that does not include the following characters: • Spaces • Tabs • The following symbols: , < > Notes: • You can specify the path to a folder under the installation folder only when the default folder is set. • You cannot specify the path to a folder that is used as the output destination of another instance.	Previous value
log_size	The value of this item can be updated. Specify the maximum size of each agent log file.	1 to 32 (in kilobytes). 16 or a greater value is recommended.	Previous value
timeout	The value of this item can be updated. The timeout period for Oracle access during a query.	0, or 10 to 3600 (in seconds). When 0 is specified, timeout monitoring is not performed. When a value from 1 to 9 is specified, it is changed to 10 at runtime.	Previous value
sql_option[#6]	The value of this item can be updated. When Y is specified, information about the following items[#6] is not collected for PI_PIDB and PD_PDTS records, and the value specified by 0 or numeric_10 is set.	{ Y | N }	Previous value
numeric_10	The value of this item can be updated. When sql_option is set to Y, the value specified is set for items for which information is not collected. If sql_option is set to N, this specification is disregarded.	0 to 99999. Note that if the value specified exceeds the maximum value for the data format of the set field (32767 for short and 65535 for ushort), the maximum value for the data format is set.[#7]	Previous value
startup_always	The value of this item can be updated. PFM - Agent for Oracle may stop due to an Oracle connection error, such as when the monitored Oracle is still starting up when	{ Y | N }	Previous value

Item	Description	Specifiable value	Default
startup_always	PFM - Agent for Oracle starts up. If Y is specified, start processing continues even if a connection error occurs. If N is specified, start processing will stop if an error occurs.	{ Y | N }	Previous value
localtemp_option[#8]	The value of this item can be updated. Option for switching the display of the free space of the locally managed temporary tablespace of PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTF, PD_PDTS, and PD_PCTS records. If Y is specified, display the size of the free space. If N is specified, display the size of the unallocated space.	{ Y | N }	Previous value
nls_lang[#9]	The value of this item can be updated. Option for specifying the character encoding used for communication between PFM - Agent for Oracle and Oracle Database.	Character code set: Specifiable values vary depending on the combination of the OS LANG and NLS_CHARACTER_SET of Oracle Database. For details, see *Table 3-20*.	Previous value
undospace_option[#10]	The value of this item can be updated. Option for switching how the value displayed as the free space in the UNDO tablespace of the PD_PDDB, PI_PIDB, PD_PDDF, PI_PIDF, PD_PDTS, PD_PCTS, and PD_CDTS records is determined. If N is specified, the size of the unallocated space is displayed. If Y is specified, the size of the free space is displayed.	{ Y | N }	Previous value

Legend:

--: None

#1

Depending on the OS, the required settings differ. The following table shows the OS and how to specify the settings.

OS	Method for specifying settings
Linux	Specify settings according to the explanation in the following section *Settings in Linux*.
AIX	Specify settings according to the explanation in the following section *Settings in AIX or Solaris*.

OS	Method for specifying settings
HP-UX(IPF)	Specify the settings according to the explanation in *Table 3-18*.
Solaris	Specify settings according to the explanation in the following section *Settings in AIX or Solaris*.

Setting in Linux

For monitoring of the Oracle database in a Linux environment, the 32-bit Oracle Client has been the required product in PFM - Agent for Oracle 10-50 or earlier. However, in PFM - Agent for Oracle 11-00 or later, the Oracle Client 64-bit library is used for monitoring.

Because instance information settings have been changed in version 11-00 or later as shown in the following table, make sure that the information is set up correctly:

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
`oracle_home`	Specify the Oracle home folder for Oracle Client 32-bit.	Specify the Oracle home folder for Oracle Database.
`oracle_version`	Specify the version of Oracle Client 32-bit.	Specify the version of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.
`sqlnet`	Specify `Y`	Specify a value according to the conditions described in *Table 3-18*.
`net_service_name`	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.	If you specify `Y` for sqlnet, specify this item. Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Database.

Notes:

- PFM - Agent for Oracle 11-00 or later uses the Oracle Client 64-bit library included with the Oracle database, so you do not need to install the 64-bit Oracle Client.
- To upgrade PFM - Agent for Oracle 10-50 or earlier to 11-00 or later, instance information is subject to change, and must therefore be updated before starting the PFM - Agent for Oracle service.

  Note that Oracle Client 32-bit is no longer needed, and can be uninstalled if it is not used by any product other than PFM - Agent for Oracle.
- If you specify `ORACLE_HOME` for Oracle Client 32-bit and start PFM - Agent for Oracle, the KAVF12011-E and KAVF12021-E message appear.

Setting in AIX or Solaris

For monitoring of the Oracle database in an AIX environment or a Solaris environment, the 32-bit Oracle Client has been the required product.

The following table shows the instance information for which you need to specify the environment settings, and the settings themselves.

Item	Description
`oracle_home`	Specify the Oracle home folder for Oracle Client 32-bit.
`oracle_version`	Specify the version of Oracle Client 32-bit. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.
`sqlnet`	Specify `Y`
`net_service_name`	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.

Item	Description
listener_name	Specify the Oracle home folder for the monitoring-target Oracle Database. In the case of Oracle RAC environment, specify the home of Oracle Grid Infrastructure.

Notes:

- If you specify ORACLE_HOME from the Oracle database or ORACLE_HOME from the 64-bit Oracle Client for the instance information item oracle_home to start PFM - Agent for Oracle while the installation of the 32-bit Oracle Client and the creation of the environment are not complete, the KAVF12011-E and KAVF12021-E messages appear.
- You need to install the 32-bit Oracle Client by specifying Administrator or run time for the installation type. Instant Client is not supported.

#2

To change an account for monitoring Oracle, use the following procedure:

1. Delete the objects created by the account you want to change.
2. Register new objects as the new account.

Performance data is not deleted when an account is changed.

For details about how to delete objects, see *3.2.2(1)(b) Deleting the objects registered in the Oracle Database*. For details about how to register objects, see *3.1.4(4)(b) Registering objects in the Oracle Database*.

#3

If the expiration date is set on oracle_passwd, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors, perform either of the following procedures before the password is expired:

- Unset the expiration date of the password
- After updating password, execute the jpcconf inst setup command to update oracle_passwd.

Note that the Oracle DEFAULT profile is applied to the Oracle account created by mk_user.sql.

#4

Specify whether to use Oracle network services.

- If Y is specified:

  PFM - Agent for Oracle connects to Oracle via a listener that is made up of Oracle network services.

  In this case, you must set the Oracle network service definitions (such as tnsnames.ora and listener.ora).

  When monitoring Oracle Database instances in an Oracle RAC configuration, set up the PFM - Agent for Oracle so that it monitors Oracle Database instances on each node.

  When using a CDB configuration of Oracle Database 12c Release 2 or later, set this setting in case of monitoring each PDB. In case of monitoring the root container (CDB$ROOT), this setting is not required.

  For details about how to set up, see the Oracle documentation.

  Store the tnsnames.ora file in the following directory.

```
oracle_home/network/admin
```

If you store the tnsnames.ora file in any other location, must therefore set to TNS_ADMIN environment variable in starting user of PFM - Agent for Oracle before starting the PFM - Agent for Oracle service.

- If N is specified:

  PFM - Agent for Oracle connects to the local database without using the Oracle network services.

#5

Manually save the old path information in a file as history data, since the information is not saved automatically. You may need to acquire the agent log information from the old directory if a problem occurs.

#6

To obtain each piece of Oracle segment-related information, PFM - Agent for Oracle searches Oracle's static data dictionary views `DBA_SEGMENTS`. If a large number of segments (more than hundreds of thousands) exist for Oracle, information collection requires a significant amount of time. As such, when a large number of segments exist, and the information listed in the following table no longer needs to be collected, set the `sql_option` to `Y` during operation.

Table 3–19: Record names and the values specified for numeric_10 (updating instance information)

Record Name	PFM - View name	Value specified for numeric_10
PD_PDTS	Segments	Enabled
	Extents	Enabled
PI_PIDB	DB Files %	Enabled
	Log Files %	Enabled
	NextAlloc Fails	Enabled
	Tablespaces	Enabled
	Rollback Segments	Enabled
	Rollback Segments Trans	Enabled
	Blocks	Enabled
	Segments	Enabled
	Extents	Enabled
	Free Mbytes	Enabled
	Overextended	Enabled
	High Max Extent	Enabled
	Datafiles	Enabled
	Mbytes	Enabled
	Free Extents	Enabled
	Free%	Enabled
	Free Change	Enabled
	Write%	Enabled
	Write/sec	Enabled
	Redo Files	Enabled
	Links	Enabled
	Links Logged On	Enabled
	Links In Tran	Enabled
	Links Open Cursors	Enabled

Record Name	PFM - View name	Value specified for numeric_10
PI_PIDB	Used Change	Enabled
	Used Mbytes	Enabled
	Rollback Segments Hit%	Enabled
	Sort Segments	Enabled
	Sorting Users	Enabled
	Physical Blocks Read	Always set to 0 because it is a delta item.
	Physical Blocks Written	Always set to 0 because it is a delta item.
	Physical Reads	Always set to 0 because it is a delta item.
	Physical Writes	Always set to 0 because it is a delta item.

#7

If the field format for each record is float or double, since the data is a floating-point number, it may be rounded depending on the specified value.

Example:

When numeric_10 is set to 32767, it may be displayed as 32760.

#8

When localtemp_option is set to N, collect free space of the locally managed temporary tablespace and information about the extents from v$temp_space_header of the dynamic performance view. The displayed values of the size of free space are the size of the unallocated space. Since the allocated space is not freed until the temporary tablespace is reconstructed or recreated, the displayed values of free space do not increase until the space is freed.

When localtemp_option is set to Y, collect free space of the locally managed temporary tablespace and information about the extents from v$sort_segment or v$temp_extent_pool of the dynamic performance view. The displayed values of the size of free space are calculated from the size of the used space.

When issuing query to v$temp_extent_pool view, the Oracle instance goes to sleep. Since this may have effect on the performance of the Oracle instance, you need adequate consideration before localtemp_option is set to Y. For details, see your Oracle documentation.

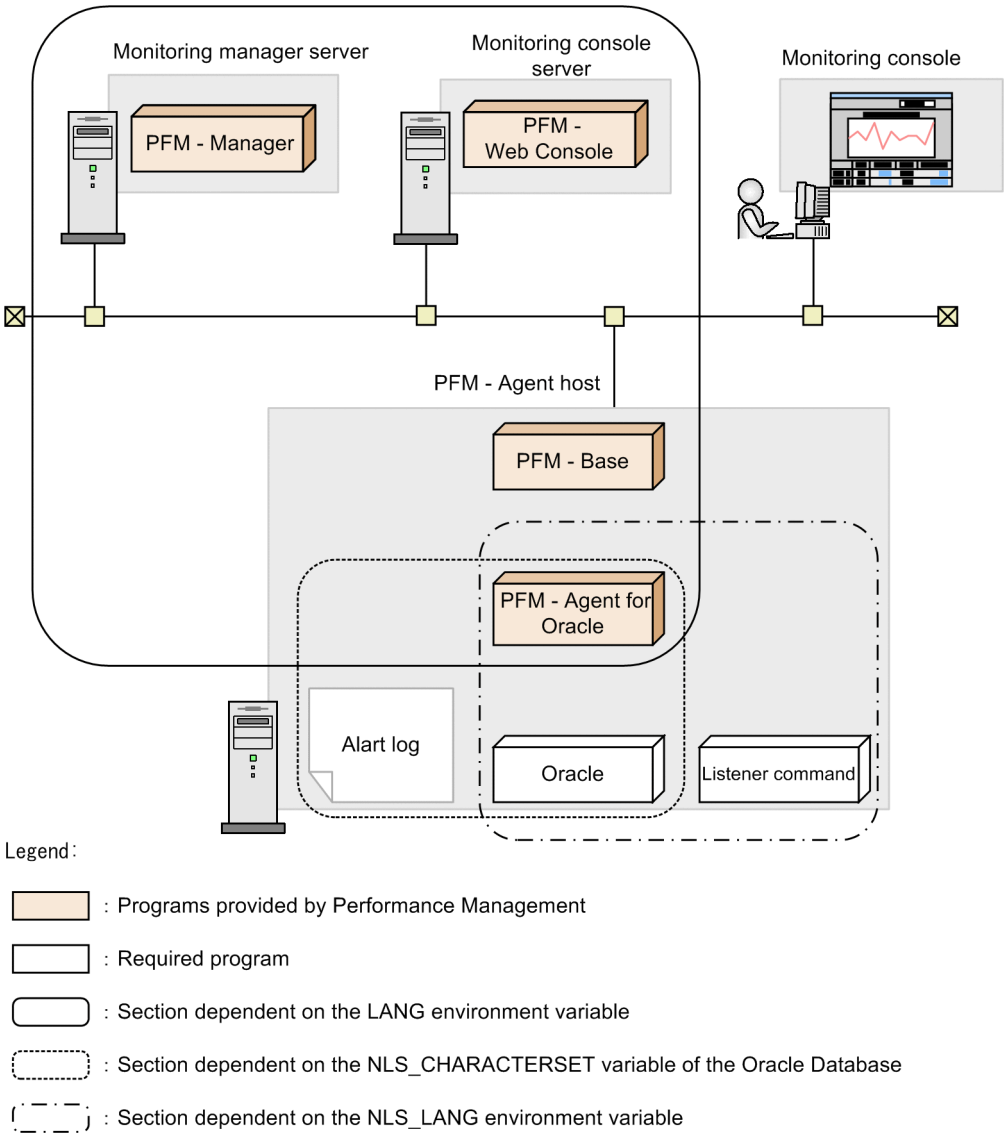
The following records use v$temp_extent_pool view:

- Data File (PD_PDDF)
- Data File Interval (PI_PIDF)

#9

The following figure shows the relationship among data, environment variable, and instance information handled by PFM - Agent for Oracle.

Figure 3–5: Relationship between data and the setting values



PFM - Agent for Oracle can collect performance data in SJIS format (for Japanese environment), EUC format (for Japanese environment), UTF-8 format (for Japanese and Chinese environment) and GB18030 format (for Chinese environment) as well as 7-bit ASCII format.

To collect performance data in a format other than 7-bit ASCII format, you need to set the `nls_lang` instance information of PFM - Agent for Oracle.

Acceptable values for the `nls_lang` instance information depend on the combination of the language environment of the OS and the database character set of the monitored Oracle. The following table lists acceptable values for the `nls_lang` instance information for each combination.

Table 3–20: Relationship between the combination of the OS language environment and the database character set of the monitored Oracle and the nls_lang settings

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Linux	`ja_JP.UTF-8`, `ja_JP.utf8`	`AL32UTF8`	`AMERICAN_AMERICA.AL32UTF8` or `AMERICAN_AMERICA.US7ASCII`#
	`zh_CN.UTF-8`, `zh_CN.utf8`		

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Linux	zh_CN.gb18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
AIX	Ja_JP, Ja_JP.IBM-932	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja_JP, ja_JP.IBM-eucJP	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	JA_JP, JA_JP.UTF-8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	ZH_CN, ZH_CN.UTF-8		
	Zh_CN, Zh_CN.GB18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
HP-UX	ja_JP.SJIS, japanese	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja_JP.eucJP, japanese.euc	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	ja_JP.utf8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	zh_CN.utf8		
	zh_CN.gb18030	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#
Solaris	ja_JP.PCK	JA16SJISTILDE	AMERICAN_AMERICA.JA16SJISTILDE or AMERICAN_AMERICA.US7ASCII#
		JA16SJIS	
	ja, Japanese, ja_JP.eucJP	JAEUCTILDE	AMERICAN_AMERICA.JA16EUCTILDE or AMERICAN_AMERICA.US7ASCII#
		JAEUC	
	ja_JP.UTF-8	AL32UTF8	AMERICAN_AMERICA.AL32UTF8 or AMERICAN_AMERICA.US7ASCII#
	zh_CN.UTF-8, zh_CN.UTF-8@pinyin, zh_CN.UTF-8@radical, zh_CN.UTF-8@stroke, zh.UTF-8		
	zh_CN.GB18030, zh_CN.GB18030@pinyin, zh_CN.GB18030@radical,	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#

OS	OS language (LANG)	NLS_CHARACTERSET of the monitored Oracle (Database character set)	nls_lang instance information (this item)
Solaris	zh_CN.GB18030@stroke	ZHS16GBK	AMERICAN_AMERICA.ZHS32GB18030 or AMERICAN_AMERICA.US7ASCII#
	Other than the above	notdependent	AMERICAN_AMERICA.US7ASCII#

#

Performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

Note:

For any other combination, performance data is collected within the scope of 7-bit ASCII characters, likely resulting in unreadable characters.

For details about LANG settings when UTF-8 is used, see *3.1.4(1) Set the LANG environment variable*.

When you specify an invalid character code set for the nls_lang instance information, the message KAVF12302-W with errcode 12705 is output, and connection with Oracle will fail.

In the following cases as well, unreadable characters might occur in the performance data:

1. The Oracle column length is exceeded.

   If you store data that exceeds the Oracle column length, the last character might become unreadable. If you use PFM - Agent for Oracle to collect data in Oracle that contains unreadable characters, the last character of the performance data will be unreadable.

2. The field size of PFM - Agent for Oracle is exceeded.

   PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

   Table 3–21: The applicable fields (Unreadable characters caused by cases where a field size of PFM - Agent for Oracle is exceeded)

Record name	Field name	Field size (Unit: bytes)
Database Object Cache(PD_PDDO)	Object Name	100
Errorlog Detail(PD_PDEL)	Message	512
Parameter Values(PD_PDP)	Value	512
SQL Text(PD_PDSQ)	Explain Plan	30,000
	SQL Text	30,000
SQL Text - Performance Based(PD_PDES)	SQL Text	10,000
Table Access(PD_PDTA)	Object	100

3. Unreadable or lack of characters in different between database character set of the Oracle and nls_lang instance variable of the PFM - Agent for Oracle.

   A data of 2 bytes on the basis of Oracle may be collected at 3 bytes when you set AMERICAN_AMERICA.AL32UTF8 in nls_lang and a database character set for the monitoring is not UTF-8. Therefore, if performance data takes from ORACLE that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Table 3–22: The applicable fields (Unreadable or lack of characters caused by cases where the database character set differs from nls_lang)

Record name	Field name	Field size (Unit: bytes)
Circuit(PD_PDCI)	User	30
Database Object Cache(PD_PDDO)	Object Name	100
	Owner	64
CDB Tablespace (PD_CDTS)	Tablespace Name	30
Collection Instance 2 (PD_PCI)	Recovery File Dest	513
Collection Tablespace 2(PD_PCTS)	Tablespace Name	30
Data File(PD_PDDF)	File Name	513
	Tablespace Name	30
Data File Interval(PI_PIDF)	File Name	513
	Tablespace Name	30
Database(PD_PDDB)	DB Name	9
Database Interval(PI_PIDB)	DB Name	9
Instance(PD_PDI)	Host	30
Latch(PD_PDLA)	OS User	30
	Program	48
	User	30
Latch Interval(PI_PILA)	OS User	30
	Program	48
	User	30
Lock(PD_PDLO)	Program	48
	User	30
Lock Interval(PI_PILO)	Program	48
	User	30
Lock Waiters(PD_PDLW)	Holding User	30
	Waiting User	30
Minimum Database Interval 2(PI_PMDB)	DB Name	9
Minimum Data File Interval 2(PI_PMDF)	File Name	513
Minimum Tablespace Interval 2(PI_PMTS)	Tablespace Name	30
Open Cursor(PD_PDOC)	Program	48
	SQL Text	60
Parameter Values(PD_PDP)	Value	512
Process Detail(PD_PDOP)	Program	48

Record name	Field name	Field size (Unit: bytes)
Process Detail(PD_PDOP)	User	15
Rollback Segment(PD_PDRS)	Tablespace Name	30
Rollback Segment Interval(PI_PIRS)	Tablespace Name	30
Segment Detail(PD_PDSM)	Owner	30
	Segment Name	81
	Tablespace Name	30
Session Detail(PD_PDS)	Machine	64
	Module	48
	OS User	30
	Program	64
	Schema Name	30
	User	30
Session Event(PD_PDEV)	Program	64
	User	30
Session Event Interval(PI_PIEV)	Program	64
	User	30
Session I/O Interval(PI_PIIO)	User	30
Session Statistics Summary(PD_PDS2)	Program	48
	User	30
Session Stat Summary Interval(PI_PIS2)	Program	48
	User	30
Session Statistics(PD_PDSS)	Program	48
	User	30
Session Wait(PD_PDWA)	Program	48
	User	30
Shared Cursor Cache(PD_PDC)	SQL Text	1,000
Sort Segment(PD_PDSR)	Tablespace Name	31
Sort Segment Interval(PI_PISR)	Tablespace Name	31
SQL Text(PD_PDSQ)	Explain Plan	30,000
	SQL Text	30,000
SQL Text - Performance Based(PD_PDES)	Parsing User	30
	SQL Text	10,000
Table Access(PD_PDTA)	Object	100
	Owner	64

Record name	Field name	Field size (Unit: bytes)
Table Access(PD_PDTA)	Program	48
	User	30
Tablespace Fragmentation(PD_PDTF)	Tablespace Name	30
Tablespace Interval(PI_PITS)	Tablespace Name	30
Tablespace(PD_PDTS)	Tablespace Name	30
Transaction(PD_PDTR)	User	30
Transaction Interval(PI_PITR)	User	30
Transaction Lock(PD_PDTL)	Object Name	30
	Owner	30
	User	30

#10

When undospace_option is set to N, the size of the unallocated space is collected as the amount of free space in the UNDO tablespace. Space in the UNDO tablespace that becomes available because its retention period has expired is treated as allocated space until it is released.

When undospace_option is set to Y, the size of the free space is collected as the amount of free space in the UNDO tablespace. Space of the UNDO tablespace that becomes available because its retention period has expired is included in the size of the free space.

The following table shows the fields whose values change depending on the specification of the undospace_option:

Table 3–23: The fields whose values change depending on the specification of the undospace_option

Record name	Field name
CDB Tablespace (PD_CDTS)	Free Mbytes
	Free %
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Data File(PD_PDDF)	Free %
	Free Mbytes
	Used Mbytes
Data File Interval (PI_PIDF)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Database(PD_PDDB)	Free %

Record name	Field name
Database(PD_PDDB)	Free Mbytes
	Used Mbytes
Database Interval (PI_PIDB)	Free %
	Free Change
	Free Mbytes
	Used Change
	Used Mbytes
Tablespace(PD_PDTS)	Free %
	Free Mbytes
	Used Mbytes
	Max Extend Free %
	Max Extend Free Mbytes
Collection Tablespace 2 (PD_PCTS)	Free Mbytes

Use the `jpcconf inst list` command to check the instance name. To update an instance environment, use the `jpcconf inst setup` command.

Updating an instance environment involves the steps described below. To update multiple instance environments, repeat the procedure for each instance environment.

To update an instance environment:

1. Find the instance name.
   Execute the `jpcconf inst list` command specified with the service key that indicates PFM - Agent for Oracle.

   ```
   jpcconf inst list -key Oracle
   ```

   If the specified instance name is `SDC`, the command displays `SDC`.

2. If the PFM - Agent for Oracle service is active in the instance environment that is to be updated, stop the service.
   For details about stopping services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.
   If the service is still active in the instance environment that is to be updated when you execute the `jpcconf inst setup` command, a confirmation message is displayed to enable you to stop the service. If you stop the service, update processing resumes; if you do not stop the service, update processing is canceled.

3. Execute the `jpcconf inst setup` command specified with the service key that indicates PFM - Agent for Oracle and the instance name.
   For example, if you are updating the instance environment for the PFM - Agent for Oracle with instance name `SDC`, execute the following command:

   ```
   jpcconf inst setup -key Oracle -inst SDC
   ```

4. Update the instance information for Oracle.

Enter the information shown in *Table 3-18* in accordance with the command's instructions. The current settings are displayed (except for the value of `oracle_passwd`). To use the displayed value, press the **Enter** key. When you have finished entering information, the instance environment is updated.

5. Restart the service in the updated instance environment.

   For details about starting services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

Note:

If you want to change an item that cannot be updated, delete the instance environment and then re-create it.

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

3.4.3 Cancellation facility for Oracle access during record collection

The maximum time for accessing Oracle can be set as a timeout value, for the time it takes to collect 1 record.

When record data is collected while Oracle and the machine are experiencing heavy load, it may take a significant amount of time to perform record collection, depending on the amount of data collected. In this case, PFM - Agent for Oracle requests may impact Oracle operation. As such, a timeout value can be set to cancel requests from PFM - Agent for Oracle to Oracle, to prevent impact on Oracle operation.

Record collection is performed in the following order for each record:

1. Oracle is accessed
2. Data is written to the Store database

However, when a timeout value is set, and a timeout occurs during Oracle access, collection for that record is canceled.

This facility is implemented using the `OCIBreak` function of the Oracle OCI (Oracle Call Interface).

The timeout value can be set as follows:

- During instance environment setup by using the `jpcconf inst setup` command
- By changing the `TIMEOUT` property for the Agent Collector service, in the PFM - Web Console GUI

The values that can be set are `0`, or anything from 10 to 3,600 (in seconds). If `0` is specified, this facility is not used. Out-of-range values will be disregarded. `0` is set as the default.

The following table lists the values that can be entered as timeout values, for each setting method.

Table 3–24: Possible timeout values

Setting method	Value input				
	-1 or less	0	1 to 9	10 to 3,600	3,601 or more
Setting/update from the `jpcconf inst setup` command	Input error (cannot be input)	Yes	Yes, but replaced on restart	Yes	Input error (cannot be input)
Change from PFM - Web Console	Yes, but not updated	Yes	Yes, but not updated	Yes	Yes, but not updated

Legend:

Yes: Can be input.

Yes, but replaced on restart: Can be input, but replaced with `10` when PFM - Agent for Oracle is restarted. A `KAVF12630-W` message is output to the common message log.

Yes, but not updated: Can be input, but cannot be updated. A `KAVF12630-W` message is output to the common message log.

Note:

Set the timeout value according to the time needed to collect records during heavy load (peak time).

For details about the timeout values set by the `jpcconf inst setup` command, see *2.1.4(3) Set up an instance environment*.

Note that the following records are not subject to cancellation:

- Instance Availability (`PD_PDIA`)
- Server Status(PD_STAT)
- SQL*Net Listener(PD_PDNL)
- SQL*Net Listeners(PD_PDLS)

When a timeout occurs, the following message is output to the common log (`agtoinf0x.log`) of the agent log.

```
KAVF12636-I
The cancellation of the record collection (record-name) by the time-out
was accepted.
```

Note that when records are not collected due to cancellation, a `KAVF12401-W` message is output to the common message log.

When a collection of multiple records, such as historical data, is performed at the same time, even if a timeout occurs for a single record, collection of other records is not performed.

Note:

Since PFM - Agent for Oracle uses the OCI (Oracle Call Interface), the time required for actual cancellation depends on the processing time for `OCIBreak()`. As such, cancellation may not happen instantly.

In the following cases, a record collection is not canceled even when a cancellation request occurs due to a timeout:

- When a timeout occurs while a collection sequence is being moved to be written to the Store database.
- Cancellation occurs due to timeout while Oracle access is terminating, in which case the `KAVF12636-I` message is output to the log file, but record collection is performed normally.

3.4.4 Updating the Store version to 2.0

The Store database comes in two versions, 1.0 and 2.0. 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 Oracle version 08-10 or later in an environment with PFM - Base or PFM - Manager version 08-10 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 describes the installation conditions, whether Store 2.0 is available for the given conditions, and the procedure for enabling Store 2.0.

Table 3–25: 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		
08-00 or earlier	Overwrite installation	Not supported	Upgrade PFM - Base or PFM - Manager to version 08-10 and then execute the setup command.
	New installation		
08-10 or later	Overwrite installation	An existing instance is supported after setup	Execute the setup command.
		A new instance is supported	Use the jpcconf inst setup command to set up when the instance is configured.
	New installation	Supported	Use the jpcconf inst setup command to set up when the instance is configured.

(1) Setup of Store 2.0

This subsection describes how to set up Store version 2.0 when you update the Store database.

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 folder settings.

   When the Store version is updated to 2.0, the Agent Store service might fail to start with the same folder settings that were available in Store 1.0. For this reason, you must review the settings of the folders used by the Agent Store service. You can use the jpcconf db define command to view and change the settings of these folders.

   The maximum length of the names of the save destination folder, backup destination folder, and other folders used by the Store database in Store 2.0 differs from the maximum length in Store 1.0. In particular, be careful when the folder 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 folder names in Store 2.0. If the absolute path name is longer than the maximum, change the settings of each folder 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`

To update the Store version to 2.0, execute the jpcconf db vrset -ver 2.0 command. You must execute this command for each Agent instance.

For details about the `jpcconf db vrset` command, see the chapter on commands in 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 chapter on commands in the manual *JP1/Performance Management Reference*.

(3) Notes

This subsection provides notes about updating.

(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 to Store version 1.0, use the `jpctool db dump` command to output Store version 2.0 information.

3.5 Backup and restoration

This section explains how to back up and restore PFM - Agent for Oracle.

In preparation for the system failure due to errors, back up the configuration files. When you change the system configuration (such as setting up PFM - Agent for Oracle), back up the configuration files.

For details about how to back up and restore the whole Performance Management system, see the chapter on backup and restoration in the *JP1/Performance Management User's Guide*.

3.5.1 Backup

When you back up the configuration files, you back up those files in any measure (such as copying the files). When you back up those configuration files, make sure that the PFM - Agent for Oracle service is inactive before you back up the files.

Important

When you back up the configuration files, record the product version number of PFM - Agent for Oracle. For details about the product version number, see the *Release Notes*.

Table below shows the backup target files for PFM - Agent for Oracle:

For details about other files, see the section presenting a list of files to be backed up for PFM - Agent (in UNIX) in the *JP1/Performance Management User's Guide*.

Table 3–26: Backup target files for PFM - Agent for Oracle (for a UNIX physical host)

File name	Description
`/opt/jp1pc/agto/agent/*.ini` files	Configuration file for Agent Collector service
`/opt/jp1pc/agto/agent/`*instance-name*`/*.ini` files	
`/opt/jp1pc/agto/store/*.ini` files	Configuration file for Agent Store service
`/opt/jp1pc/agto/store/`*instance-name*`/*.ini` files	

Table 3–27: Backup target files for PFM - Agent for Oracle (for a UNIX logical host)

File name	Description
`/opt/jp1pc/agto/agent/*.ini` files	Configuration file for Agent Collector service
/*Environment-directory*#`/jp1pc/agto/agent/`*instance-name*`/*.ini` files	
`/opt/jp1pc/agto/store/*.ini` files	Configuration file for Agent Store service
/*Environment-directory*#`/jp1pc/agto/store/`*instance-name*`/*.ini` files	

#

Environment-directory is the directory that is created on the shared disk when setting up the logical host.

3.5.2 Restoration

Make sure that the following prerequisite conditions are met before you restore the configuration information of PFM - Agent for Oracle. After confirming that the conditions have been met, copy the backup files to the original location, and overwrite the existing configuration files on the host with the backup configuration files.

Prerequisites:

- PFM - Agent for Oracle has been already installed.
- The PFM - Agent for Oracle service is inactive.

> **Important**
>
> When you restore the configuration files for PFM - Agent for Oracle, the product version number of the PFM - Agent for Oracle in the backup environment must be the same as that of the PFM - Agent for Oracle in the restoration target environment. For details about the product version number, 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 Oracle 10-00 is restored to PFM - Agent for Oracle 10-00.
>
> **Cases for which restoration cannot be performed:**
>
> - Settings information backed up from PFM - Agent for Oracle 09-00 is restored to PFM - Agent for Oracle 10-00.
>
> - Settings information backed up from PFM - Agent for Oracle 09-00 is restored to PFM - Agent for Oracle 09-00-05.

3.6 Online manuals

The standard manual supplied medium accompanying the Performance Management program product contains a manual that you can copy to the host on which PFM - Web Console is installed and then view in a Web browser. Make sure that you copy the manual to the executing and standby nodes when PFM - Web Console runs in a cluster system.

3.6.1 Setting procedure

(1) See the manual from the help menu bar of PFM - Web Console

1. Register PFM - Agent with PFM - Web Console, following the PFM - Web Console setup procedure (Follow the procedure to register an additional PFM - Agent).

2. On the host on which PFM - Web Console is installed, create a directory to copy the manual to.
   - Windows: *Web-Console-installation-folder*\doc*language-cord**help-ID-of-PFM - Agent*
   - UNIX: /opt/jp1pcwebcon/doc/*language-cord*/*help-ID-of-PFM - Agent*

   For details about the help ID of PFM - Agent, see *C. List of Identifiers*.

3. From the manual supplied medium, copy all the manual files to the root of the directory you created.

   HTML manual:

   Windows: all the htm files and FIGURE folder from the *applicable-drive* \MAN\3021*material number* (such as 03004A0D)

   UNIX: all the htm files and FIGURE directory from /*mount-point-directory-for-the-supplied-medium*/MAN/3021/*material number* (such as 03004A0D)

   PDF manual:

   Windows: the PDF file from *applicable-drive*\MAN\3021*material number* (such as 03004A0D)

   UNIX: the PDF file from /*mount-point-directory-for-the-supplied-medium*/MAN/3021/*material number* (such as 03004A0D)

   Make sure you copy the index.htm files (for the HTML manual) or the PDF files (for the PDF manual) to the root of the created directory. For details about how to copy manual files, see *readme.txt* on the manuals media.

4. Restart PFM - Web Console.

(2) See the manual from the hard disk

Execute the setup.exe command on supplied medium to install the manual, or copy htm files, PDF files, and GIF files to any folders or directories. For HTML manual, the folder or directory organization must be:

```
html (storage folder or directory of the htm files and the PDF files)
 └ FIGURE (storage folder or directory of GIF files)
```

3.6.2 Viewing the manual

To view the manual:

1. In the menu bar of the PFM - Web Console main window, click Help. A help selection window appears.

2. Click the manual name, or click [PDF] after the manual name.
   Clicking the manual name displays the manual in HTML format. Clicking [PDF] displays the manual in PDF format.

Notes on the display of characters in a Web browser:

In Windows, when you display the online manual from the **Start** menu, the HTML manual might be displayed in the Web browser that is already open.

3.7 Notes regarding operation

3.7.1 Notes on operating

- Before you start PFM - Agent for Oracle, you need to complete installation and setup the Oracle Database and then start the Oracle Database. When the Oracle Database is not running, performance data except for Instance Availability (PD_PDIA) and Server Status (PD_STAT) is not collected. Also, if an Oracle instance or container is mounted, performance data is not collected.
- After you stop PFM - Agent for Oracle, you can change constitution or perform maintenance of Oracle Database instance or container.
- When PFM - Agent for Oracle collects the following records obtained dynamic performance v$ views (such as V $ACCESS, V$OPEN_CURSOR), an error (such as ORA-00600, ORA-00602) might be output from the Oracle Database and it might fail to collect the performance data. In this case, when the records are collected at next time, it collects the performance data again.
  - Activity Summary (PD_PDAS)
  - Open Cursor (PD_PDOC)
  - Session Detail (PD_PDS)
  - Table Access (PD_PDTA)
- When the port to use in Agent Collector service has been already used, fail to bind to port at the time of Agent Collector service start#. And, when startup_always is set to Y, Agent Collector service is started. In this case, you can not monitor correctly. You stop the service, and confirm port conflict, later start the service.

  1. Stop the service of PFM - Agent for Oracle.
     `jpcspm stop -key Oracle`
     Stop PFM - Agent for Oracle service which started normally.
     `jpcspm stop -key Oracle -kill immediate`
     Stop PFM - Agent for Oracle service which failed to bind to port.
  2. Confirm port conflict to use in PFM - Agent for Oracle.
     Confirm whether other service do not use a fixed port by `jpcconf port define` command.
  3. Start the service of PFM - Agent for Oracle.
     `jpcspm start -key Oracle [-inst <instance name>]`

  # When failed to bind to port, the message KAVE00100-E, KAVE00107-E, KAVE00160-E are output to common message log.
- If you do not stop PFM - Agent for Oracle before you stop the monitored Oracle Database instance or container, you may not be able to stop the Oracle Database by using the NORMAL shutdown procedure. To stop the Oracle Database, you need to stop PFM - Agent for Oracle beforehand or else use the IMMEDIATE shutdown procedure.
- When you fail in monitoring using `sys` account, either set the following 1 and 2 together, or confirm 3 setting.
  Create a password file.
  1. Set "exclusive" as the value of the "remote_login_passwordfile" initialization parameter.
  2. Set "true" as the value of the "o7_dictionary_accessibility" initialization parameter.
- If the expiration date is set on oracle_passwd, once the password is out of date connections to Oracle fail so that PFM - Agent for Oracle cannot collect the performance information. In order to avoid connection errors, perform either of the following procedures before the password is expired:

- Unset the expiration date of the password

- After updating password, execute the `jpcconf inst setup` command to update oracle_passwd.

Note that the Oracle default profile is applied to the Oracle account created by `mk_user.sql`.

For details about the above settings, see the manual for Oracle Database.

- When a connection error occurred while PFM - Agent for Oracle is running, an error is logged to the sqlnet.log file is stored in the directory specified by Oracle. When you use Oracle Database 11g or later without setting `ORACLE_BASE`, the sqlnet.log is stored in the directory created under the "/" directory. Before you start PFM - Agent for Oracle, the sqlnet.log can also be stored in the directory specified by the `ORACLE_BASE` environment variable.
- In Solaris or AIX, PFM - Agent for Oracle does not reference the PATH environment variable. For this reason, no issue will arise even when a setting that prioritizes the `PATH` environment variable specified for the Oracle Database over the `PATH` environment variable specified for the Oracle Client 32-bit environment is specified.
- PFM - Agent for Oracle that monitors performance data for each an instance of the node cannot monitor Oracle RAC One Node configuration has dynamically changed Oracle SIDs.
- 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 Oracle, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.
- In Linux, if you restart the machine by using the reboot command while the PFM - Agent for Oracle service is running, the message shown below might be output to the common message log, indicating that the Agent Collector service has ended abnormally. This message, however, does not actually indicate the occurrence of an error because the Agent Collector service has in fact ended successfully.

  KAVF12004-E Agent Collector stopped abnormally.
- If you restart the machine by using the reboot command while the PFM - Agent for Oracle service is running, the message shown below might be output to the common message log, indicating that the Agent Collector service has ended abnormally. This message, however, does not actually indicate the occurrence of an error because the Agent Collector service has in fact ended successfully.

  KAVF12004-E Agent Collector stopped abnormally.
- In Linux, when it starts by Run Level 3 and 5 of the operating system regulations, the auto-start of the Performance Management service is effective. When it starts by Run Level 0 and 6 of the operating system regulations, the auto-stop of the Performance Management service is effective. Neither the auto-start nor the auto-stop operates when starting by Run Level 1 and 2.
- In AIX, you cannot create a PFM - Agent for Oracle instance for the file system for which cio is specified as the mount option. Before creating an instance, check the mount status of the file system. If you create an instance for the file system for which the cio option is specified, the following messages appear when the instance starts:
  - Common message log

    KAVF12020-E While executing _ai_ipc_open() function MMF_attach() failed.

    KAVF12020-E While executing _ai_ipc_open() function MMF_create() failed.

    KAVF12021-E Error occurred by function waitIpcInitializeSignal.

    (en=error-code, arg1=argument-1, arg2=argument-2, arg3=argument-3)
  - Agent log (error log)

    KAVF12600-E An error occurred in OS function mmap. (rc=22)
- In HP-UX, PFM - Agent for Oracle does not support operation in high reliability mode.

3.7.2 Notes on monitoring Oracle Database 12c Release 2 or later

(1) Notes on monitoring all version of Oracle Database 12c Release 2 or later

- PFM - Agent for Oracle collects performance data of the record field size from Oracle. Therefore, if Oracle contains data that exceeds the field size, the last character of the performance data might be unreadable. The following table lists the applicable fields:

Record name	Field name
Circuit (PD_PDCI)	User
Database Object Cache (PD_PDDO)	Owner
Latch (PD_PDLA)	OS User
	User
Latch Interval (PI_PILA)	OS User
	User
Lock (PD_PDLO)	User
Lock Interval (PI_PILO)	User
Lock Waiters (PD_PDLW)	Holding User
	Waiting User
Open Cursor (PD_PDOC)	User
Resource Limit (PD_PDRL)	Resource Name
Segment Detail (PD_PDSM)	Owner
	Segment Name
Session Detail (PD_PDS)	Module
	OS User
	Schema Name
	User
Session Event (PD_PDEV)	User
Session Event Interval (PI_PIEV)	User
Session I/O Interval (PI_PIIO)	User
Session Stat Summary Interval (PI_PIS2)	User
Session Statistics (PD_PDSS)	User
Session Statistics Summary (PD_PDS2)	User
Session Wait (PD_PDWA)	User
SQL Text - Performance Based (PD_PDES)	Parsing User
Table Access (PD_PDTA)	Owner
	User
Transaction (PD_PDTR)	User

Record name	Field name
Transaction Interval (PI_PITR)	User
Transaction Lock (PD_PDTL)	Object Name
	Owner
	User

- If you set the INSTANCE_MODE Oracle Database initialization parameter to READ-MOSTLY or READ-ONLY, PFM - Agent for Oracle cannot collect the performance information.
- When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), and set to `oracle_version` instance information is `12`. If set a value invalid, the following message appears:

  KAVE05924-W An invalid character string was entered.

(2) Notes on monitoring verison of Oracle Database 18c or later

When set the `OPTIMIZER_IGNORE_HINTS` initialization parameter in enable, it may take an amount of time to perform record collection. In this case, consider that disable the initialization parameter.

4 Operating PFM - Agent for Oracle in a Cluster System

This chapter describes the procedures for installing and setting up PFM - Agent for Oracle for use in a cluster system. This chapter also describes the flow of processing when you use PFM - Agent for Oracle in a cluster system.

4.1 Cluster system overview

A *cluster system* is a system in which multiple server systems are linked and operated as a single system. The Oracle Database that is a monitoring target program of PFM - Agent for Oracle can operate in the following cluster systems:

- An HA (High Availability) cluster system configured for Oracle
- Oracle Real Application Clusters or Oracle Parallel Server

This section describes the configuration you use for operating PFM - Agent for Oracle in a cluster system. For an overview of cluster systems and information about the system configuration when you use a Performance Management system in a cluster system, see the chapter on operation in a cluster system in the *JP1/Performance Management User's Guide*.

The term *cluster system* when used alone in this chapter refers to an HA cluster system.

Oracle Database 12c Release 2 or later in a CDB configuration that is a monitoring target of PFM - Agent for Oracle can operate in a manner similar to a cluster system in a non-CDB configuration.

4.1.1 HA cluster system

(1) Oracle configuration in an HA cluster system

You can improve availability of an Oracle Database (single-instance Oracle) by using it in an HA cluster system, which can perform a failover if a failure occurs.

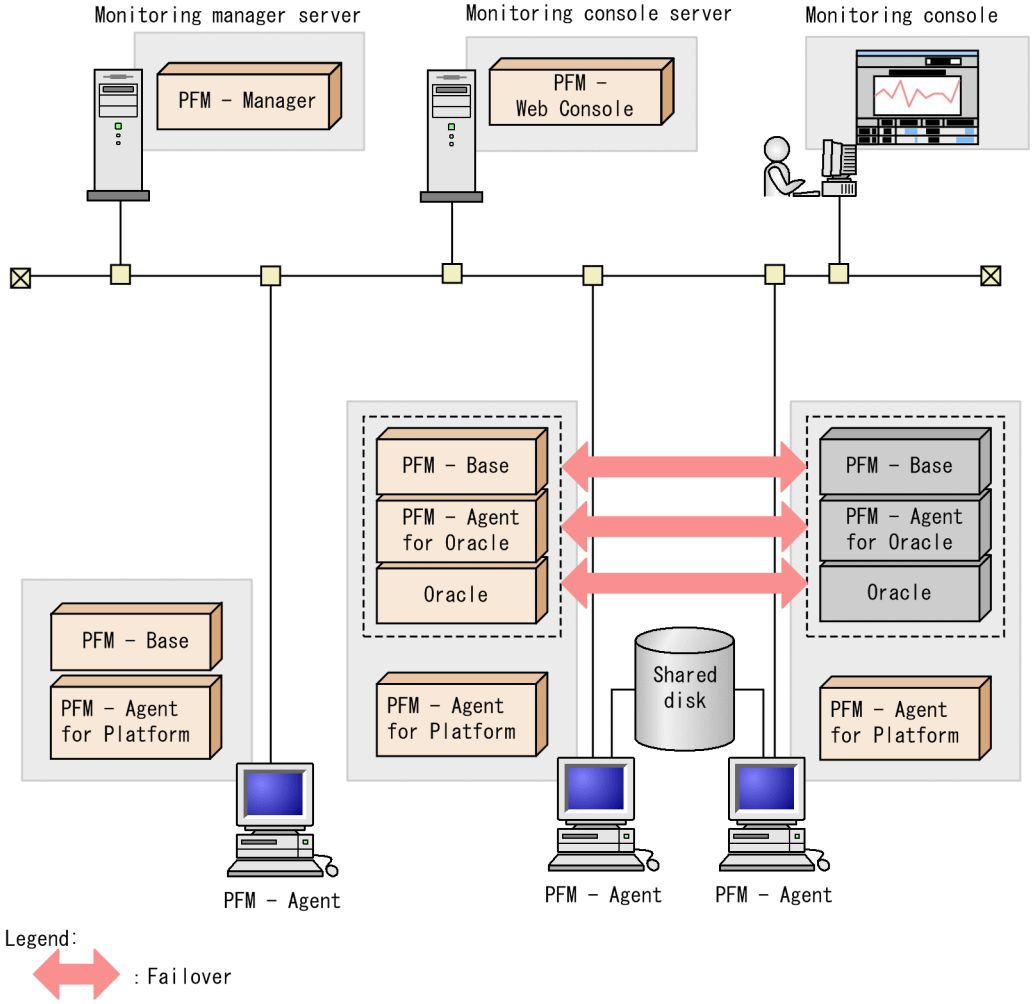
Typically, when you operate Oracle in an HA cluster system, you construct an environment that permits execution of the same Oracle instance on both an executing node and a standby node. The configuration you use stores a single set of Oracle data (data files, configuration files, log files, etc.) on a shared disk. To control Oracle, you usually use solution products for controlling Oracle from the cluster software. For example, you use Oracle Fail Safe with Windows WSFC. For UNIX, you use products provided by various cluster software vendors.

There are also cases when the configuration or method of operating Oracle in a cluster system depends on the system.

(2) PFM - Agent for Oracle configuration in an HA cluster system

PFM - Agent for Oracle can operate in an HA cluster system and can monitor Oracle in a cluster configuration. Figure 4-1 shows a configuration for operating PFM - Agent for Oracle in an HA cluster system.

Figure 4–1: Example of an PFM - Agent for Oracle configuration in an HA cluster system



As Figure 4-1 shows, PFM - Agent for Oracle operates in a cluster system and monitors Oracle. When a failure occurs, failover applies to PFM - Agent for Oracle at the same time it applies to Oracle, allowing PFM - Agent for Oracle to continue monitoring Oracle.

PFM - Agent for Oracle also stores definition information about the shared disk and continues operating when a failover occurs. When there are multiple Performance Management programs on a single logical host, all programs use the same shared directories.

When you monitor multiple Oracle, you can install PFM - Agent for Oracle on separate logical hosts so that each PFM - Agent for Oracle can operate and perform a failover independently.

4.1.2 Load-balancing cluster system

(1) Configuration of Oracle in a load-balancing cluster system (Oracle Real Application Clusters)

Oracle Real Application Clusters (or Oracle Parallel Server) is a system consisting of multiple nodes running Oracle that function as a single Oracle system that processes a single database. The data is stored on a shared disk and is shared by all nodes.

Distribution of workload to multiple nodes improves scalability and fault tolerance.

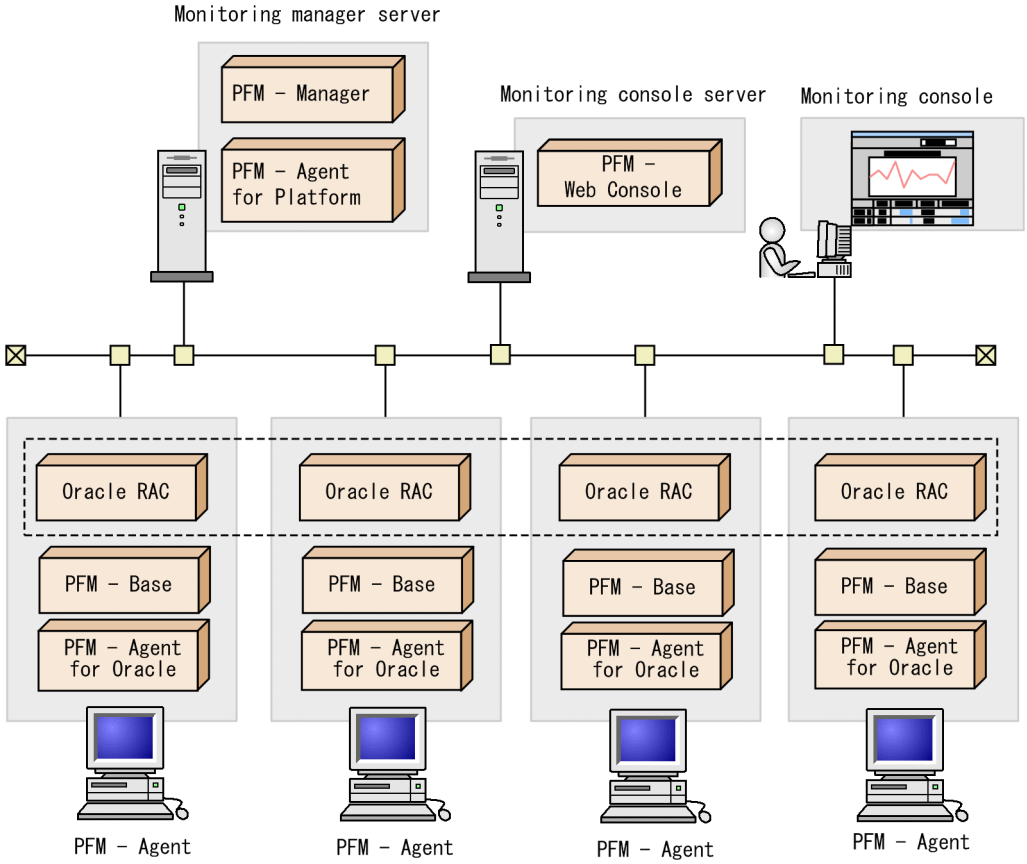
Although applications see the set of nodes as a single Oracle system, each node runs an Oracle system with a unique instance name. For example, a database might be run by Oracle instances `SID=ora1` at `node1` and `SID=ora2` at `node2`. Applications use a global database name to access the database via Oracle Net Services.

For details about Oracle Real Application Clusters (or Oracle Parallel Server), see your Oracle documentation.

(2) Configuration of PFM - Agent for Oracle if Oracle is in a load-balancing cluster system configuration

To operate PFM - Agent for Oracle in a non-cluster system if Oracle is operating in a load-balancing cluster system, use a configuration such as that shown in the following figure.

Figure 4–2: Example configuration of PFM - Agent for Oracle in a load-balancing cluster system



An Oracle system with a unique instance name is run on each node. PFM - Agent for Oracle monitors the Oracle instance on each node.

As with a single-node system, set up PFM - Agent for Oracle and configure it to monitor each node's Oracle Real Application Clusters instance. In this case, do not register PFM - Agent for Oracle in the cluster software.

Note:

To operate PFM - Agent for Oracle in a load-balancing cluster system and monitor the Oracle Real Application Cluster (or the Oracle Parallel Server), handle PFM - Agent for Oracle as you would in a system with many single nodes. That is, you can operate PFM - Agent for Oracle in the same way as in a typical non-cluster system.

4.2 Processing during failover

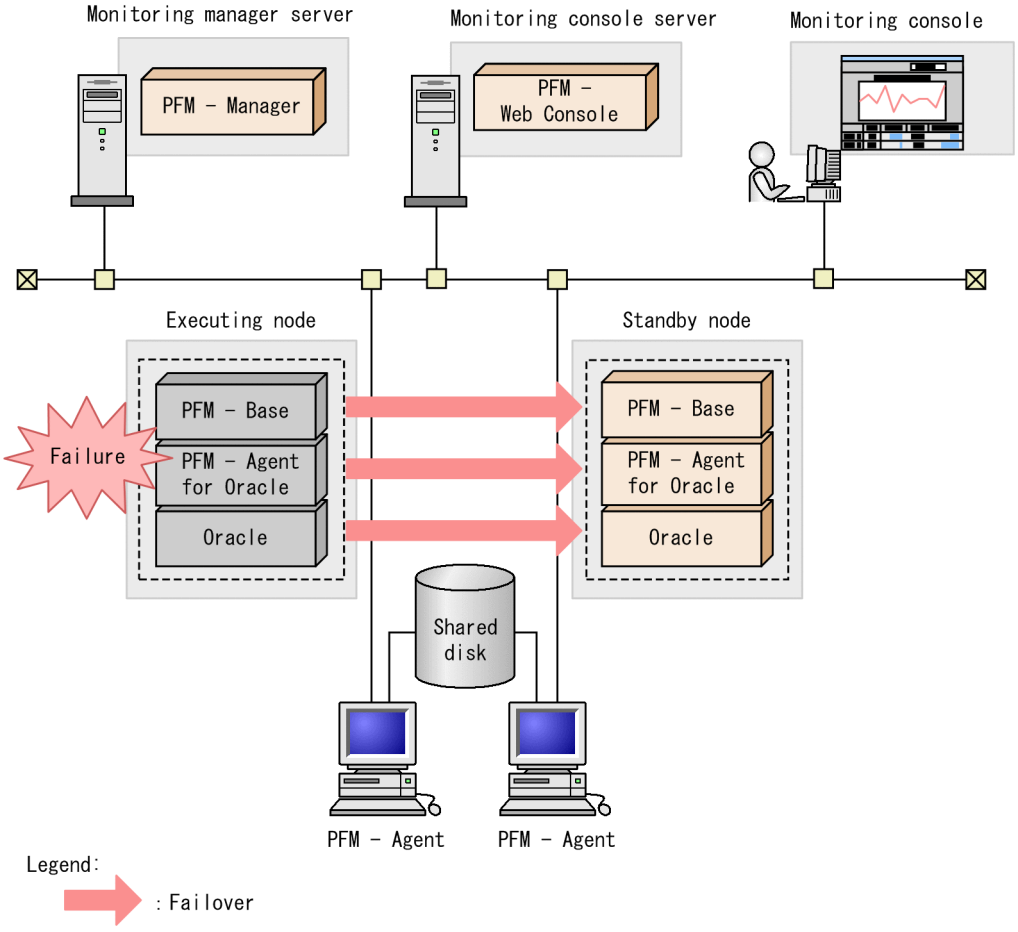
When a failure occurs on the executing host, processing moves to the standby host.

This section describes failover processing in the event of a failure in PFM - Agent for Oracle. This section also describes the effect that PFM - Manager failures have on PFM - Agent for Oracle.

4.2.1 Failover when a failure occurs on PFM - Agent for Oracle host

Figure 4-3 shows the processing when failover occurs on a PFM - Agent for Oracle host.

Figure 4–3: Processing when a PFM - Agent host performs failover



When you attempt to operate PFM - Web Console during failover of PFM - Agent for Oracle, PFM - Web Console displays the message `There was no answer(-6)`. In such a case, wait for the failover to be completed before performing the PFM - Web Console operation.

Once PFM - Agent for Oracle failover has been completed, you will be connected to the PFM - Agent for Oracle that was started on the failover target node and you will again be able to operate PFM - Web Console.

4.2.2 Effects when PFM - Manager stops

PFM - Manager affects the entire Performance Management system when it stops.

PFM - Manager centrally manages the agent information of the PFM - Agent for Oracle agents that are running on all nodes. It also controls notification of alarm events in the event a threshold value is exceeded during performance monitoring by PFM - Agent for Oracle as well as execution of actions based on alarm events. Accordingly, when PFM - Manager stops, the entire Performance Management system is affected, as described in Table 4-1.

Table 4–1: Effect on PFM - Agent for Oracle when PFM - Manager stops

Program name	Effects	Solution
PFM - Agent for Oracle	If PFM - Manager stops while PFM - Agent for Oracle is running, PFM - Agent for Oracle: • Continues collecting performance data. • Retains alarm events for each alarm definition and retries until PFM - Manager is recovered if it cannot send the alarm events to PFM - Manager. When the number of retained alarm events exceeds 3, the oldest alarm event is overwritten. If PFM - Agent for Oracle is stopped, all the alarm events that have been retained are deleted. • Resets, when PFM - Manager is restarted, the alarm statuses that have already been reported to PFM - Manager. The alarm statuses are then initialized after checking with PFM - Agent for Oracle. • Takes a long time to stop if you attempt to stop it, because PFM - Manager cannot be notified.	Start PFM - Manager. You can continue to run any PFM - Agent for Oracle that is currently running. Check the common log after PFM - Manager is recovered because alarms may not be reported exactly as expected. (KAVE00024-I)

Consider the effects of stopping PFM - Manager when considering how to operate Performance Management. There are times when you have to stop PFM - Manager in order to change the configuration or perform maintenance work. Hitachi recommends that you schedule maintenance work for times that will have the least impact on operations.

4.3 Installation and setup (Windows)

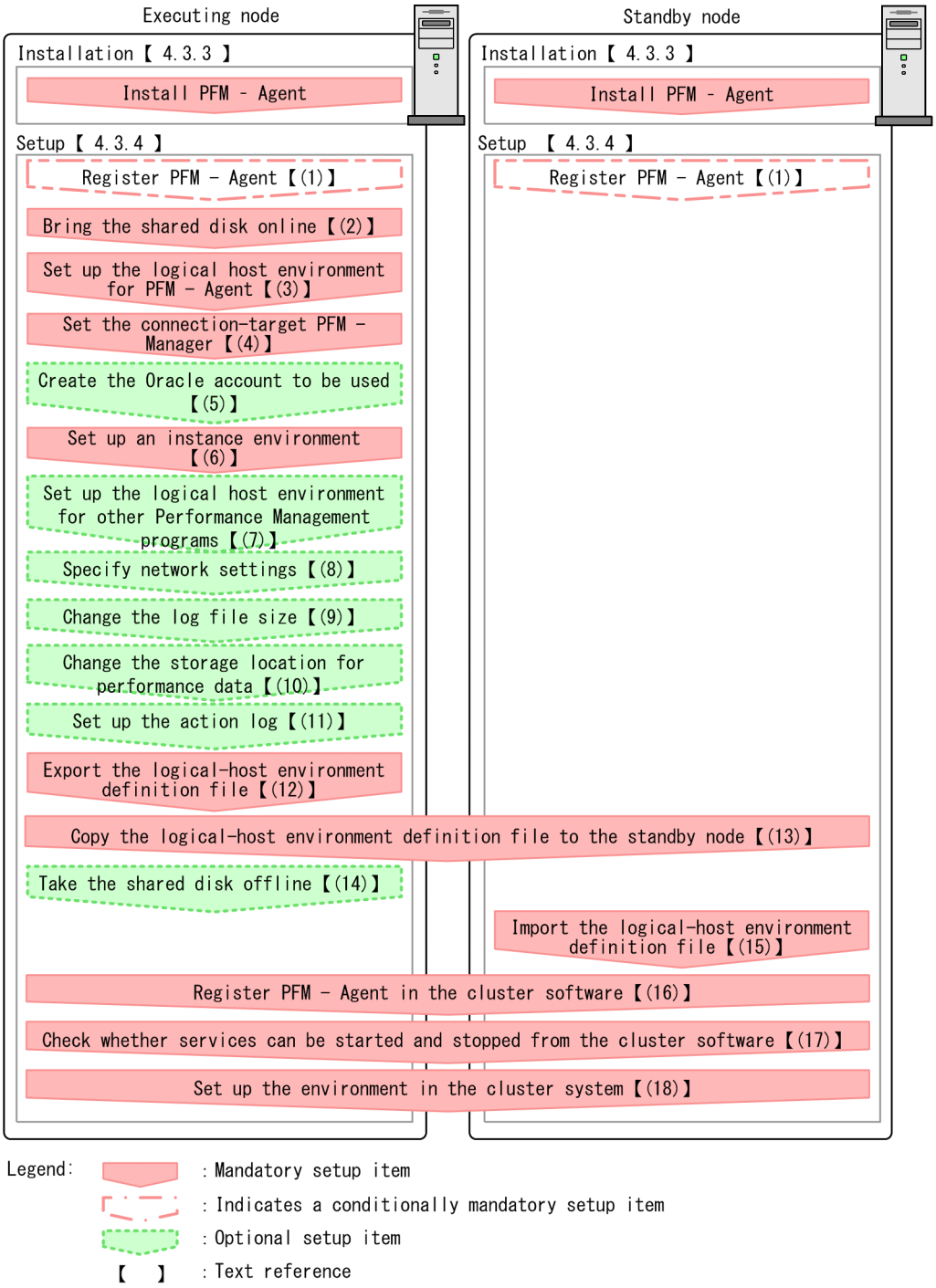
The following figure shows the workflow for installing and setting up PFM - Agent for Oracle to run in a cluster system.

For details about how to install and set up PFM - Manager, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.3.1 Installation and setup workflow

The following figure shows the workflow for installing and setting up PFM - Agent for Oracle to run on a logical host in a cluster system.

Figure 4–4: Workflow for installing and setting up PFM - Agent for Oracle to run on a logical host in a cluster system (Windows)



Note:

The definitions of PFM - Agent in a physical host environment cannot be inherited by setting up PFM - Agent in a logical host environment. For logical and physical host environments, a new environment is created when an instance environment is set up.

Note that you, 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 in accordance with the 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 by 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. Non-interactive commands are useful in the following cases:

- If you want to regularly change the passwords used to connect with monitoring targets
- If you want to improve operational efficiency when adding multiple monitoring targets

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

4.3.2 Preparation for installation and setup

This subsection describes the prerequisites for installation and setup, and provides cautionary notes and other information you should know before installing and setting up PFM - Agent for Oracle.

(1) Prerequisites

The following are the prerequisites for running PFM - Agent for Oracle in a cluster system.

(a) Cluster system

Make sure that the following conditions are satisfied:

- The cluster system is controlled by cluster software.
- The cluster software is able to start and stop PFM - Agent for Oracle on a logical host. Additionally, PFM - Agent for Oracle is able to fail over at the same time when the monitored Oracle fails over.
- Both the executing and standby systems are configured to suppress error reporting to Microsoft.

  When an application error occurs, a dialog box giving you the option of reporting the problem to Microsoft appears. Because the dialog box can interfere with failover, you must disable error reporting. If the nodes have not been set up to disable error reporting, take the following steps.

In Windows Server 2012

1. Choose **Control Panel** > **System and Security** > **Action Center** > **Maintenance**.
2. In **Check for solutions to unreported problems**, click **Settings**.
3. In the **Windows Error Reporting Configuration** dialog box, choose **I don't want to participate, and don't ask me again**.
4. Click the **OK** button.

In Windows Server 2016 or later

1. Right-click the Windows **Start** menu and then choose **Run** from the displayed menu.
2. Enter `gpedit.msc`, and then click the **OK** button.

   The Local Group Policy Editor appears.
3. Click **Computer Configuration**, **Administrative Templates**, **Windows Components**, and then **Windows Error Reporting**.
4. In the right pane, right-click **Disable Windows Error Reporting**, and then from the displayed menu, choose **Edit**.

   The setting window appears.

5. In the setting window, select the **Enabled** check box.
6. Click the **OK** button.

Notes:

- Failover may not take place if Dr. Watson encounters an application error and displays a message box. For this reason, you must disable error notification that displays a message box. For details about the procedure, see the documentation for your OS. Note that disabling error notification may affect data collection when an application error occurs.
- When an application error occurs in Windows, a dialog box giving you the option of reporting the problem to Microsoft appears. Because the dialog box can interfere with failover, you must disable error reporting. For details about disabling error reporting, see the documentation for the OS.

(b) Shared disk

Make sure that the following conditions are satisfied:

- Each logical host has a shared disk that the standby node can inherit from the executing node.
- The shared disk is physically connected to each node via a Fibre Channel, SCSI, or similar connection.

  Performance Management does not support the use of network drives or disks replicated over the network as the shared disk.
- If a failover is requested while a process is accessing the shared disk, the cluster software can take the shared disk offline and force a failover.
- Each instance of Performance Management programs on the same logical host uses the same directory on the shared disk.

  Note that you can change the location of the Store database to another directory on the shared disk.

(c) 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, which the standby node inherits from the executing node.
- Logical host names and logical IP addresses are set in the `hosts` file and on the name server.
- For DNS operation, host names in FQDN format cannot be used. For the logical host name, use the host name with the domain name portion removed.
- Physical host names and logical host names are unique within the system.

Notes:

- Do not specify the physical host name (the host name displayed by the `hostname` command) as the name of the logical host. If you do so, normal communication may not be possible.
- Logical host names can consist of 1 to 32 alphanumeric characters. A logical host name cannot include space characters or any of the following characters:

  `/ \ : ; * ? ' " < > | & = , .`
- You cannot specify `localhost`, an IP address, or a host name beginning with a hyphen (`-`) as a logical host name.

(d) Settings when IPv6 used

Performance Management supports IPv6 environments as well as IPv4 environments as a network configuration. Therefore, Performance Management can operate even in a network configuration in which both an IPv4 environment and an IPv6 environment are used.

PFM - Agent for Oracle can communicate with PFM - Manager via IPv6. Note that this explanation applies only when the OS of a host on which PFM - Agent for Oracle and PFM - Manager are installed is Windows or Linux.

For details about the scope of communication in an environment with both IPv4 and IPv6, see *M. About Communication in IPv4 Environments and IPv6 Environments*.

When you want to use IPv6 for communication between PFM - Manager and PFM - Agent for Oracle, the settings for using IPv6 must be enabled for both the PFM - Manager host and the PFM - Agent host. In addition, before installing PFM - Agent for Oracle, you need to enable the use of IPv6 on the PFM - Agent host. You have to execute the `jpcconf ipv6 enable` command to enable this setting. If this setting is already enabled, however, you do not need to execute the command. If you want to check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command.

For details about the `jpcconf ipv6 enable` command and `jpcconf ipv6 display` command, see the chapter that describes 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.*

When you use IPv6 for communication between a monitored host and PFM - Agent for Oracle, specify the name of a monitored host where name resolution can be performed.

Communication between PFM - Agent for Oracle and a monitoring target is performed with an IP address that can be resolved. Also, if an IPv4 environment and an IPv6 environment are both used, and communication between PFM - Agent for Oracle and the monitoring target fails with an IP address that can be resolved, the communication is not retried by using another IP address.

For example, if communication fails when IPv4 is used, IPv6 is not used to retry communication. Similarly, if communication fails when IPv6 is used, IPv4 is not used to retry communication. Make sure beforehand that a connection can be established.

(2) Information required to set up PFM - Agent for Oracle to run on a logical host

When you set up PFM - Agent for Oracle to run in a logical host environment, you must specify the information shown in the following table, in addition to the environment information required for PFM - Agent for Oracle setup.

Table 4–2: Information required to set up PFM - Agent for Oracle to run on a logical host

Item	Example
The same logical host name as for Oracle	`jp1-halora`
The same logical IP address as for Oracle	`172.16.92.100`
Shared disk	`S:\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 how much shared disk capacity is needed, see *A. Estimating System Requirements*.

(3) Cautionary notes on failing over a logical host with PFM - Agent for Oracle

In a system configuration in which PFM - Agent for Oracle runs on a logical host, consider whether you want the entire logical host to fail over when an error occurs in PFM - Agent for Oracle.

If the entire logical host is failed over when an error occurs in PFM - Agent for Oracle, business applications on the same logical host will also be failed over. Failover of these applications may affect any business operations that are in progress.

We recommend that you use one of the following cluster software settings so that errors on PFM - Agent for Oracle do not affect the operations of Oracle:

- Do not monitor operation of PFM - Agent for Oracle
- Do not allow detection of errors in PFM - Agent for Oracle to result in failover

(4) Cautionary notes on upgrading when Oracle runs on a logical host

When you upgrade PFM - Agent for Oracle that runs on a logical host, you need to bring the shared disk on either the executing node or the standby node online.

4.3.3 Installation procedure

Install PFM - Agent for Oracle on the executing node and the standby node.

> **Important**
>
> You must install PFM - Agent for Oracle on a local disk. Do not install it on a shared disk.

The installation procedure is the same as the installation procedure in a non-cluster system. For details about the installation procedure, see *2.1.3 Installation procedure*.

4.3.4 Setup procedure

This subsection describes how to set up Performance Management to run in a cluster system.

Setup must be performed first on the executing node and then on the standby node.

Executing indicates a task to be performed on the executing node. Standby indicates a task to be performed on the standby node. Option indicates a setup item that is required depending on the environment or an optional setup item used when the default is to be changed.

(1) Register PFM - Agent for Oracle Executing Standby Option

To perform integrated management of PFM - Agent for Oracle using PFM - Manager and PFM - Web Console, you must register PFM - Agent for Oracle with PFM - Manager and PFM - Web Console.

The conditions and procedure for registering PFM - Agent for Oracle are the same as when a cluster system is not used. For details about the conditions and procedure for registering PFM - Agent for Oracle, see *2.1.4(1) Register PFM - Agent for Oracle*.

(2) Bring the shared disk online Executing

Make sure that the shared disk is online. If the shared disk is not online, use the cluster software or the volume manager to bring it online.

(3) Set up the logical host environment for PFM - Agent Executing

Execute the `jpcconf ha setup` command to set up the logical host environment. When you execute the command, the necessary files are copied to the shared disk, the logical host definition is set up, and the logical host environment is created.

Note:

Before you execute the command, stop all Performance Management programs and services throughout the Performance Management system. For details about how to stop services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

To set up the logical host environment:

1. Execute the `jpc conf ha setup` command to create the logical host environment for PFM - Agent for Oracle.

   Execute the command as follows:

   ```
   jpcconf ha setup -key Oracle -lhost jp1-halora -d S:\jp1
   ```

   Use the `-lhost` option to specify the logical host name. In the example above, the logical host name is `jp1-halora`. For DNS operation, specify a logical host name with the domain name portion removed.

   Specify the name of the shared disk directory as the environment directory name of the `-d` option. For example, if you specify `-d S:\jp1`, the directory `S:\jp1\jp1pc` is created, and the files for the logical host environment are created in that directory.

   Notes:

   - PFM - Agent for Oracle runs on file systems. When the database is built on RAW devices or ASM, the shared disk must be based on a file system.
   - For the environment directory name, do not specify a path containing a parenthesis ((,)).

   If the path contains a parenthesis, the logical host environment will be created successfully, but PFM - Agent for Oracle startup may fail.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Confirm that the logical host environment you created has been set up correctly.

(4) Set the connection-target PFM - Manager Executing

Execute the `jpcconf mgrhost define` command to specify which PFM - Manager is to manage PFM - Agent for Oracle.

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-halora
```

Use the `-host` option to specify the host name of the connection-target PFM - Manager. If that PFM - Manager is running in a logical host environment, specify the logical host name of the PFM - Manager in the `-host` option. In the example above, the logical host name of PFM - Manager is `jp1-hal`.

Use the `-lhost` option to specify the logical host name of PFM - Agent for Oracle. In the example above, the logical host name of PFM - Agent for Oracle is `jp1-halora`.

Although an example of interactive command execution is shown here, the `jpcconf mgrhost define` command can be also executed non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(5) Create an Oracle account to be used Executing Option

Create an Oracle account that has specific system privileges to collect performance data for the Oracle Database monitored by PFM - Agent for Oracle.

For details about how to create an Oracle account, see *2.1.4(2) Create an Oracle account to be used in PFM - Agent for Oracle*.

This setup task is unnecessary when you use the `sys` account.

(6) Set up an instance environment Executing

Execute the `jpcconf inst setup` command to set up an instance environment for PFM - Agent for Oracle.

The setup procedure is the same as the setup procedure in a non-cluster system, except that, in a cluster system, you must specify the logical host name in the `-lhost` option when executing the `jpcconf inst setup` command.

In a cluster system, the `jpcconf inst setup` command is executed in the following format:

```
jpcconf inst setup -key Oracle -lhost logical-host-name -inst instance-name
```

Although an example of interactive command execution is shown here, the `jpcconf inst setup` command can be also executed non-interactively. For details about the `jpcconf inst setup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

As the agent log output folder (the value of `log_path`), specify the path of a folder that is on the shared disk.

For details about other settings and procedures, see *2.1.4(3) Set up an instance environment*.

(7) Set up the logical host environment for other Performance Management programs Executing Option

At this point, set up any other Performance Management programs, such as PFM - Manager or PFM - Agent, on the same logical host.

For details about how to set up these products, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*, and the chapter on operation in a cluster system in the manuals for the applicable version of PFM - Agent.

(8) Specify network settings Executing Option

This setup task is necessary only when you change the network settings to match the configuration in which Performance Management is used.

The following are the two network setting items:

- IP addresses

  To specify the IP address used by Performance Management in a network environment where multiple LANs are connected, directly edit the contents of the `jpchosts` file.

  After editing the file, copy it from the executing node to the standby node.

  For details about how to set IP addresses, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide.*

- Port numbers

  If Performance Management programs will communicate with each other through a firewall, use the `jpcconf port` command to set the port numbers.

  For details about how to set port numbers, see the chapter on installation and setup and the chapters on setup and operation in a cluster system in the *JP1/Performance Management Planning and Configuration Guide.*

(9) Change the size of log files Executing Option

Performance Management outputs its operating status to a set of proprietary log files called a *common message log*. Perform this setting only if you want to change the default file size.

For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide.*

(10) Change the storage location of performance data Executing Option

Perform this setting only if you want to change the folders where the database of performance data managed by PFM - Agent for Oracle is saved, backed up, exported, or imported.

For details, see *2.4.1 Changing the storage location of performance data*.

(11) Setting up the action log Executing Option

This setting is required to output action log when alarm occur. 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 *K. Outputting Action Log Information*.

(12) Export the logical-host environment definition file Executing

Once you have created the logical host environment for PFM - Agent for Oracle, export the environment definition as a file. In the export process, a file containing the collective definition information for all Performance Management programs set up on that logical host is output. Export the environment definition only after you have set up all additional Performance Management programs on the logical host.

To export the environment definition of the logical host:

1. Execute the `jpcconf ha export` command to export the environment definition of the logical host.

The definition information for the logical host environment you created is output to an export file. You can give this file any name you wish.

For example, execute the command as follows to output the logical host environment definition to the file `lhostexp.txt`:

```
jpcconf ha export -f lhostexp.txt
```

Although an example of interactive command execution is shown here, the `jpcconf ha export` command can be also executed non-interactively. For details about the `jpcconf ha export` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(13) Copy the logical-host environment definition file to the standby node

Executing Standby

Copy the file you exported in *(12) Export the logical-host environment definition file* from the executing node to the standby node.

(14) Take the shared disk offline Executing Option

Use the cluster software or the volume manager to take the shared disk offline. Note that if you intend to continue using the shared disk, you do not need to take it offline at this point.

(15) Import the logical-host environment definition file Standby

On the standby node, import the exported file you copied from the executing node.

Use the `jpcconf ha import` command to set up the environment definition for the Performance Management programs of the logical host that you created on the executing node so that they run on the standby node. If more than one Performance Management program was installed on the logical host, the definition information for all of the programs is imported in batch form.

Note that the shared disk does not need to be online when you execute this command.

To import the logical-host environment definition file:

1. Execute the `jpcconf ha import` command to import the environment definition of the logical host.

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha import` command can be also executed non-interactively. For details about the `jpcconf ha import` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

   When you execute the command, the settings on the standby node are changed to reflect the environment described in the export file. This sets up the standby node to run PFM - Agent for Oracle as a logical host.

   If you used the `jpcconf port` command to assign fixed port numbers during setup, the same port numbers will take effect on the standby node.

2. Execute the `jpcconf ha list` command to check whether the logical host is set up correctly.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Check whether the same output is displayed as when you executed `jpcconf ha list` on the executing node.

(16) Register PFM - Agent for Oracle in the cluster software Executing Standby

If you intend to use Performance Management programs in a logical host environment, make sure that the programs are registered in the cluster software. Also, set up the environment so that the Performance Management programs are started and stopped based on instructions from the cluster software.

Note:

Before you can register PFM - Agent for Oracle, you must register Oracle Database resource.

For details about how to register PFM - Agent for Oracle in the cluster software, see your cluster software documentation.

For the dependency settings specified if PFM - Agent for Oracle is on the PFM - Manager logical host, see the chapter on cluster system setup and operation in the manual *JP1/Performance Management User's Guide*.

This subsection describes how to register PFM - Agent for Oracle in your cluster software, using the settings for Windows WSFC as an example.

Register the following services of PFM - Agent for Oracle in the cluster software:

Table 4–3: PFM - Agent for Oracle services to be registered in the cluster software

No.	Name	Service name	Dependencies
1	PFM - Agent Store for Oracle *instance-name* [*LHOST*]	`JP1PCAGT_OS_`*instance-name* [*LHOST*]	IP address resource[#1] Physical disk resource[#2] Oracle Database resource
2	PFM - Agent for Oracle *instance-name* [*LHOST*]	`JP1PCAGT_OA_`*instance-name* [*LHOST*]	Cluster resource in No. 1
3	PFM - Action Handler [*LHOST*]	`JP1PCMGR_PH` [*LHOST*]	IP address resource[#1] Physical disk resource[#2]

#1

IP address resource defined in the Oracle cluster environment

#2

Shared disk resource

Replace *LHOST* with the logical host name. For example, if the instance name is `SDC1` and the logical host name is `jp1-halora`, the display name of the service will be `PFM - Agent Store for Oracle SDC1 [jp1-halora]`, and the service name will be `JP1PCAGT_OS_SDC1 [jp1-halora]`.

When using WSFC as your cluster software, register the services as WSFC resources. Set up each resource according to the instructions below.

- Register the service with **Resource type** set to *General-Purpose service*.
- Set **Dependency** as shown in Table 4-3.
  In any of the following cases, for PFM - Agent Store for Oracle *instance-name* [*LHOST*], specify a dependency with Oracle TNS Listener:
  - When you want to monitor alarms or resources of the listener
  - When there is an Oracle service that is running with an account other than the local system account
  - When you specify `Y` for the instance item `sqlnet` in the instance environment settings

If you do not specify a dependency with Oracle TNS Listener in these cases, an error might occur in Oracle.

- On the *Policies* page of the **Properties** dialog box, choose how you want WSFC to behave when an error occurs in a Performance Management program.

  For example, if you want WSFC to fail over the node when PFM - Agent for Oracle fails, perform the following settings:

  - Select the **If resource fails, attempt restart on current node** radio button.
  - Select the **If restart is unsuccessful, fail over all resources in this Role** check box.
  - In principle, set **Maximum restarts in the specified period:** to 3.

  #

  Usually, specify 3 as the *Threshold* for the restart retry count.

Note:

The cluster software is responsible for starting and stopping the services that have been registered in it. Set **Startup type** for those services to **Manual** to prevent them from starting automatically when the OS starts. When you set up a service using the `jpcconf ha setup` command, **Startup type** for the service will be set to **Manual**. Also, do not use the following command to forcibly stop a service registered in the cluster software:

`jpcspm stop -key all -lhost` *`logical-host-name`* `-kill immediate`

(17) Check whether services can be started and stopped from the cluster software Executing Standby

Check whether the cluster software is operating correctly by using it to issue start and stop requests to Performance Management programs on each node.

(18) Set up the environment in the cluster system Executing Standby

After setting up the Performance Management programs, use PFM - Web Console to set up the environment for the programs. You will then be able to display reports on the operating status of monitoring targets, and notify users whenever a problem occurs.

For details about setting up the environment for Performance Management programs, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.4 Installation and setup (UNIX)

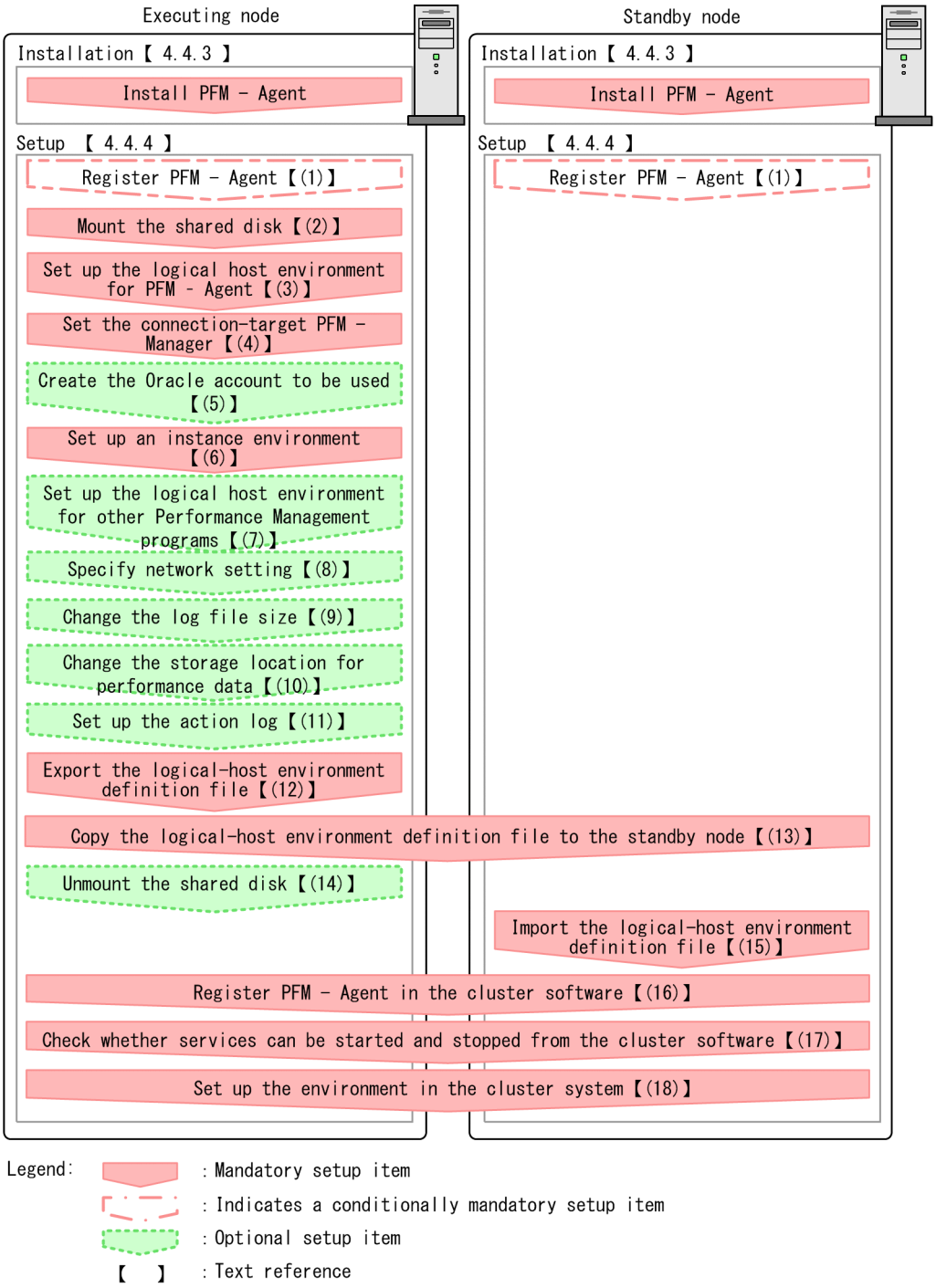
This section describes how to install and set up PFM - Agent for Oracle in a cluster system.

For details about how to install and set up PFM - Manager, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.4.1 Installation and setup workflow

The following figure shows the workflow for installing and setting up PFM - Agent for Oracle to run on a logical host in a cluster system.

Figure 4–5: Workflow for installing and setting up PFM - Agent for Oracle to run on a logical host in a cluster system (UNIX)



Note:

When you set up PFM - Agent in a logical host environment, it does not inherit definition information from any instances of PFM - Agent that may be in the physical host environment. In both physical and logical host environments, a new environment is created only when an instance environment is set up.

Note that you can select whether to execute a setup command requiring user entry interactively or non-interactively.

If you execute a setup command interactively, you need to enter a value in accordance with command directives.

If you execute a setup command non-interactively, user entry is not required because the operator entry required during command execution can be replaced by the specification of options or definition files. Also, batch processing or remote execution can automate setup operations to reduce administrator workload and operating costs. Non-interactive commands are useful in the following cases:

- You want to regularly change the password to be used for connection with the monitoring target.
- You want to improve operational efficiency when adding multiple monitoring targets.

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

4.4.2 Preparation for installation and setup

This subsection describes the prerequisites for installation and setup, and provides cautionary notes and other information you should know before installing and setting up PFM - Agent for Oracle in a cluster system.

(1) Prerequisites

The following are the prerequisites for running PFM - Agent for Oracle in a cluster system.

(a) Cluster system

Make sure that the following conditions are satisfied:

- The cluster system is controlled by cluster software.
- The cluster software is able to start and stop PFM - Agent for Oracle on a logical host. Additionally, PFM - Agent for Oracle is able to fail over at the same time when the monitored Oracle fails over.

(b) Shared disk

Make sure that the following conditions are satisfied:

- Each logical host has a shared disk that the standby node can inherit from the active node.
- The shared disk is physically connected to each node via a Fibre Channel, SCSI, or similar connection.

  Performance Management does not support the use of network drives or disks replicated over the network as the shared disk.
- If a failover is requested while a process is accessing the shared disk, the cluster software can unmount the shared disk and force a failover.
- Each instance of Performance Management programs on the same logical host uses the same directory on the shared disk.

  Note that you can change the location of the Store database to another directory on the shared disk.

(c) 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, which the standby node inherits from the executing node.
- Logical host names and logical IP addresses are set in the `hosts` file and on the name server.
- For DNS operation, host names in FQDN format cannot be used. For the logical host name, use the host name with the domain name portion removed.

- Physical host names and logical host names are unique within the system.

Notes:

- Do not specify the physical host name (the host name displayed by the `uname -n` command) as the name of the logical host. If you do so, normal communication may not be possible.
- Logical host names can consist of 1 to 32 alphanumeric characters. A logical host name cannot include space characters or any of the following characters:
  `/ \ : ; * ? ' " < > | & = , .`
- You cannot specify `localhost`, an IP address, or a host name beginning with a hyphen (-) as a logical host name.

(d) Settings when IPv6 used

Performance Management supports IPv6 environments as well as IPv4 environments as a network configuration. Therefore, Performance Management can operate even in a network configuration in which both an IPv4 environment and an IPv6 environment are used.

PFM - Agent for Oracle can communicate with PFM - Manager via IPv6. Note that this explanation applies only when the OS of a host on which PFM - Agent for Oracle and PFM - Manager are installed is Windows or Linux.

For details about the scope of communication in an environment with both IPv4 and IPv6, see *M. About Communication in IPv4 Environments and IPv6 Environments*.

When you want to use IPv6 for communication between PFM - Manager and PFM - Agent for Oracle, the settings for using IPv6 must be enabled for both the PFM - Manager host and the PFM - Agent host. In addition, before installing PFM - Agent for Oracle, you need to enable the use of IPv6 on the PFM - Agent host. You have to execute the `jpcconf ipv6 enable` command to enable this setting. If this setting is already enabled, however, you do not need to execute the command. If you want to check whether the use of IPv6 is enabled, execute the `jpcconf ipv6 display` command. Execute the `jpcconf ipv6 enable` command separately on the executing node and on the standby node.

For details about the `jpcconf ipv6 enable` command and `jpcconf ipv6 display` command, see the chapter that describes 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.*

When you use IPv6 for communication between a monitored host and PFM - Agent for Oracle, specify the name of a monitored host where name resolution can be performed.

Communication between PFM - Agent for Oracle and a monitoring target is performed with an IP address that can be resolved. Also, if an IPv4 environment and an IPv6 environment are both used, and communication between PFM - Agent for Oracle and the monitoring target fails with an IP address that can be resolved, the communication is not retried by using another IP address.

For example, if communication fails when IPv4 is used, IPv6 is not used to retry communication. Similarly, if communication fails when IPv6 is used, IPv4 is not used to retry communication. Make sure beforehand that a connection can be established.

(2) Information required to set up PFM - Agent for Oracle to run on a logical host

When you set up PFM - Agent for Oracle to run in a logical host environment, you must specify the information shown in the following table, in addition to the environment information required for PFM - Agent for Oracle setup.

Table 4–4: Information required to set up PFM - Agent for Oracle to run on a logical host

Item	Example
The same logical host name as for Oracle	`jp1-halora`
The same logical IP address as for Oracle	`172.16.92.100`
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 how much shared disk capacity is needed, see *A. Estimating System Requirements*.

(3) Cautionary notes on failing over a logical host with PFM - Agent for Oracle

In a system configuration in which PFM - Agent for Oracle runs on a logical host, consider whether you want the entire logical host to fail over when an error occurs in PFM - Agent for Oracle.

If the entire logical host is failed over when an error occurs in PFM - Agent for Oracle, business applications on the logical host will also be failed over. Failover of these applications may affect any business operations that are in progress.

- We recommend that you use one of the following cluster software settings so that errors on PFM - Agent for Oracle do not affect the operations of Oracle:
- Do not monitor operation of PFM - Agent for Oracle
- Do not allow detection of errors in PFM - Agent for Oracle to result in failover

(4) Cautionary notes on upgrading when Oracle runs on a logical host

When you upgrade PFM - Agent for Oracle that runs on a logical host, you need to bring the shared disk on either the executing node or the standby node online.

4.4.3 Installation procedure

Install PFM - Agent for Oracle on the executing node and the standby node.

> **Important**
>
> You must install PFM - Agent for Oracle on a local disk. Do not install it on a shared disk.

The installation procedure is the same as the installation procedure in a non-cluster system. For details, see *3.1.3 Installation procedure*.

4.4.4 Setup procedure

This subsection describes how to set up Performance Management to run in a cluster system.

Setup must be performed first on the executing node and then on the standby node.

Executing indicates a task to be performed on the executing node. Standby indicates a task to be performed on the standby node. Option indicates a setup item that is required depending on the environment or an optional setup item used when the default is to be changed.

(1) Register PFM - Agent for Oracle Executing Standby Option

To perform integrated management of PFM - Agent for Oracle using PFM - Manager and PFM - Web Console, you must register PFM - Agent for Oracle with PFM - Manager and PFM - Web Console.

The conditions and procedure for registering PFM - Agent for Oracle are the same as when a cluster system is not used. For details about the conditions and procedure for registering PFM - Agent for Oracle, see *3.1.4(2) Register PFM - Agent for Oracle*.

(2) Mount the shared disk Executing

Make sure that the shared disk is mounted. If the shared disk is not mounted, use the cluster software or the volume manager to mount it.

(3) Set up the logical host environment for PFM - Agent for Oracle Executing

Execute the `jpcconf ha setup` command to set up the logical host environment. When you execute the command, the necessary files are copied to the shared disk, the logical host definition is set up, and the logical host environment is created.

Note:

Before you execute the command, stop all Performance Management programs and services throughout the Performance Management system. For details about how to stop services, see the chapter on operating Performance Management in the *JP1/Performance Management User's Guide*.

To set up the logical host environment:

1. Execute the `jpcconf ha setup` command to create the logical host environment for PFM - Agent for Oracle.

   Execute the command as follows:

   ```
   jpcconf ha setup -key Oracle -lhost jp1-halora -d /jp1
   ```

   Use the `-lhost` option to specify the logical host name. In the example above, the logical host name is `jp1-halora`. For DNS operation, specify a logical host name with the domain name portion removed.

   Specify the name of the shared disk directory as the environment directory name of the `-d` option. For example, if you specify `-d /jp1`, the directory `/jp1/jp1pc` is created, and the files for the logical host environment are created in that directory.

   Note:

   PFM - Agent for Oracle runs on file systems. When the database is built on RAW devices or ASM, the shared disk must be based on a file system.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Confirm that the logical host environment you created has been set up correctly.

(4) Set the connection-target PFM - Manager Executing

Execute the `jpcconf mgrhost define` command to specify which PFM - Manager is to manage PFM - Agent for Oracle.

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-halora
   ```

   Use the `-host` option to specify the host name of the connection-target PFM - Manager. If that PFM - Manager is running in a logical host environment, specify the logical host name of the PFM - Manager in the `-host` option. In the example above, the logical host name of PFM - Manager is `jp1-hal`.

   Use the `-lhost` option to specify the logical host name of PFM - Agent for Oracle. In the example above, the logical host name of PFM - Agent for Oracle is `jp1-halora`.

   Although an example of interactive command execution is shown here, the `jpcconf mgrhost define` command can be also executed non-interactively. For details about the `jpcconf mgrhost define` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(5) Create an Oracle account to be used Executing Option

Create an Oracle account that has specific system privileges to collect performance data for the Oracle Database monitored by PFM - Agent for Oracle.

For details about how to create an Oracle account, see *3.1.4(3) Create an Oracle account to be used in PFM - Agent for Oracle*.

This setup task is unnecessary when you use the `sys` account.

(6) Set up an instance environment Executing

You execute the `jpcconf inst setup` command to set up an instance environment for PFM - Agent for Oracle.

The setup procedure is the same as the setup procedure in a non-cluster system, except that, in a cluster system, you must specify the logical host name in the `-lhost` option when executing the `jpcconf inst setup` command.

In a cluster system, the `jpcconf inst setup` command is executed in the following format:

```
jpcconf inst setup -key Oracle -lhost logical-host-name -inst instance-name
```

Although an example of interactive command execution is shown here, the `jpcconf inst setup` command can be also executed non-interactively. For details about the `jpcconf inst setup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

As the agent log output folder (the value of `log_path`), specify the path of a folder that is on the shared disk.

For details about other settings and procedures, see *3.1.4(4) Set up an instance environment*.

(7) Set up the logical host environment for other Performance Management programs

Executing Option

At this point, set up any other Performance Management programs, such as PFM - Manager or PFM - Agent, on the same logical host.

For details about how to set up these products, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*, and the chapter on operation in a cluster system in the manuals for the applicable version of PFM - Agent.

(8) Specify network settings

Executing Option

This setup task is necessary only when you change the network settings to match the configuration in which Performance Management is used.

The following are the two network setting items:

- IP addresses

  To specify the IP address used by Performance Management in a network environment where multiple LANs are connected, directly edit the contents of the `jpchosts` file.

  After editing the file, copy it from the executing node to the standby node.

  For details about how to set IP addresses, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide.*

- Port numbers

  If Performance Management programs will communicate with each other through a firewall, use the `jpcconf port` command to set the port numbers.

  For details about how to set port numbers, see the chapter on installation and setup and the chapters on setup and operation in a cluster system in the *JP1/Performance Management Planning and Configuration Guide.*

(9) Change the size of log files

Executing Option

Performance Management outputs its operating status to a set of proprietary log files called a *common message log*. Perform this setting only if you want to change the default file size.

For details, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide.*

(10) Change the storage location of performance data

Executing Option

Perform this setting only if you want to change the folders where the database of performance data managed by PFM - Agent for Oracle is saved, backed up, exported, or imported.

For details, see *3.4.1 Changing the storage location of performance data*.

(11) Setting up the action log

Executing Option

This setting is required to output action log when alarm occur. 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 *K. Outputting Action Log Information*.

(12) Export the logical-host environment definition file Executing

Once you have created the logical host environment for PFM - Agent for Oracle, export the environment definition as a file. In the export process, a file containing the collective definition information for all Performance Management programs set up on that logical host is output. Export the environment definition only after you have set up all additional Performance Management programs on the logical host.

To export the environment definition of the logical host:

1. Execute the `jpcconf ha export` command to export the environment definition of the logical host.

   The definition information for the logical host environment you created is output to an export file. You can give this file any name you wish.

   For example, execute the command as follows to output the logical host environment definition to the file `lhostexp.txt`:

   ```
   jpcconf ha export -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha export` command can be also executed non-interactively. For details about the `jpcconf ha export` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(13) Copy the logical-host environment definition file to the standby node

Executing Standby

Copy the file you exported in *(12) Export the logical-host environment definition file* from the executing node to the standby node.

(14) Unmount the shared disk Executing Option

Complete setup by unmounting the file system. Note that if you intend to continue using the shared disk, you do not need to unmount the file system at this point.

Note:

Check whether the `jp1pc` directory and the files for the logical host environment exist on the local disk in the environment directory of the logical host that you set up. If they exist, this indicates that setup was performed without mounting the shared disk. In this case, take the following action:

1. Using the `tar` command, archive the `jp1pc` directory in the environment directory on the local disk.
2. Mount the shared disk.
3. If the environment directory you specified does not exist on the shared disk, create it now.
4. Extract the `tar` file into the environment directory on the shared disk.
5. Unmount the shared disk.
6. Delete the `jp1pc` directory and its contents from the environment directory on the local disk.

(15) Import the logical-host environment definition file Standby

On the standby node, import the exported file you copied from the executing node.

Use the `jpcconf ha import` command to set up the environment definition for the Performance Management programs of the logical host that you created on the executing node so that they to run on the standby node. If more than

one Performance Management program was installed on the logical host, the definition information for all of the programs is imported in batch form.

Note that the shared disk does not need to be online when you execute this command.

1. Execute the `jpcconf ha import` command to import the environment definition of the logical host.
   Execute the command as follows:

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha import` command can be also executed non-interactively. For details about the `jpcconf ha import` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.
   When you execute the command, the settings on the standby node are changed to reflect the environment described in the export file. This sets up the standby node to run PFM - Agent for Oracle as a logical host.
   If you used the `jpcconf port` command to assign fixed port numbers during setup, the same port numbers will take effect on the standby node.

2. Execute the `jpcconf ha list` command to check whether the logical host is set up correctly.
   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Check whether the same output is displayed as when you executed `jpcconf ha list` on the executing node.

(16) Register PFM - Agent for Oracle in the cluster software Executing Standby

If you intend to use Performance Management programs in a logical host environment, make sure that the programs are registered in the cluster software. Also, set up the environment so that the Performance Management programs are started and stopped based on instructions from the cluster software.

This subsection describes the settings to be specified to register PFM - Agent for Oracle in the cluster software.

In UNIX, you will typically need to register control methods for *Start*, *Stop*, *Process monitoring*, and *Forced stop* in the cluster software.

The following table shows how to set these control methods for PFM - Agent for Oracle.

Table 4–5: Control methods for PFM - Agent for Oracle registered in the cluster software

Item	Description

Start	Start PFM - Agent for Oracle by executing the following commands:
	`/opt/jp1pc/tools/jpcspm start -key AH -lhost logical-host-name`
	`/opt/jp1pc/tools/jpcspm start -key Oracle -lhost logical-host-name -inst instance-name`
	Start PFM - Agent for Oracle after the shared disk and the logical IP address have been made usable.

Stop	Stop PFM - Agent for Oracle by executing the following commands:
	`/opt/jp1pc/tools/jpcspm stop -key Oracle -lhost logical-host-name -inst instance-name`
	`/opt/jp1pc/tools/jpcspm stop -key AH -lhost logical-host-name`
	Stop PFM - Agent for Oracle before the shared disk and the logical IP address are made unusable.

Item	Description
Stop	If the service has stopped due to a failure, the `jpcspm stop` command returns `3`. In this case, assume that the command terminated normally, because the service has stopped. For cluster software that evaluates the execution result from a return value, convert the return value to `0`, for example.

Process monitoring	Use the `ps` command to check whether the monitored processes are running:
	`ps -ef | grep "`*process-name logical-host-name*`" | grep -v "grep `*monitored-process*`"`
	Monitored processes are as follows:
	• `jpcagto`
	• `agto/jpcsto`
	• `jpcah`
	For each of these processes, execute the following command:
	• For `jpcagto`
	`ps -ef | grep "jpcagto_`*instance-name logical-host-name*`" | grep -v "grep jpcagto"`
	• For `agto/jpcsto`
	`ps -ef | grep "agto/jpcsto_`*instance-name logical-host-name*`" | grep -v "grep agto/jpcsto"`
	• For `jpcah`
	• `ps -ef | grep "jpcah `*logical-host-name*`" | grep -v "grep jpcah"`
	Hitachi also recommends that you provide a control method that allows the cluster software to temporarily stop monitoring. This can be used when temporarily stopping Performance Management, for example, during system maintenance (thereby stopping the cluster software from monitoring files that are undergoing maintenance).

Forced stop	To perform a forced stop, execute the following command: `/opt/jp1pc/tools/jpcspm stop -key all -lhost `*logical-host-name*` -kill immediate` As the service key of the first argument, you can specify only `all`. Note: Execution of this command forcibly stops all Performance Management processes in the specified logical host environment by sending a `SIGKILL` signal to the processes. That is, the Performance Management system is stopped at the logical host level, not at the individual service level. Set up the cluster software to only perform a forced stop if a normal stop fails.

Notes:

- Because Performance Management programs that are to be registered in the cluster software must be started and stopped based on instructions from the cluster software, do not set them up to start automatically when the OS starts.
- If the cluster software evaluates execution results on the basis of the return value, set up the Performance Management programs to convert their command return values to values the cluster software expects. For the command return values of Performance Management programs, see the explanation of the command.
- If you use the `ps` command to monitor operations, check in advance the outputs of the `ps` command. The command outputs PFM - Agent for Oracle processes as the concatenations of the logical host name and the monitoring instance names. Make sure that the displayed PFM - Agent for Oracle processes are not truncated. If any one of the displayed PFM - Agent for Oracle processes is truncated, then shorten the instance names so that the displayed processes are no longer truncated.

  In addition, when using the `ps` command to identify the process name and logical host name, if you use `( )` (parentheses) or `[ ]` (square brackets), the process name and logical host name might not be acquired. If this happens, check the `ps` command reference of the OS, and then execute the command again.

- Start PFM - Agent for Oracle after starting Oracle, and stop it before stopping Oracle. If you are using the Oracle listener# for establishing connections and want to connect PFM - Agent for Oracle and the Oracle database, first start the listener and then start Agent for Oracle.

#

If `Y` was specified for `sqlnet` of the instance environment settings, the net service name specified in net_service_name uses the listener for connections. Before establishing connections, you must configure the environment on the Oracle database side to be able to establish connections by using the listener.

(17) Check whether services can be started and stopped from the cluster software Executing Standby

Check whether the cluster software is operating correctly by using it to issue start and stop requests to Performance Management programs on each node.

(18) Set up the environment in the cluster system Executing Standby

After setting up the Performance Management programs, use PFM - Web Console to set up the environment for the programs. You will then be able to display reports on the operating status of monitoring targets, and notify users whenever a problem occurs.

For details about setting up the environment for Performance Management programs, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.5 Setup cancellation and uninstallation (Windows)

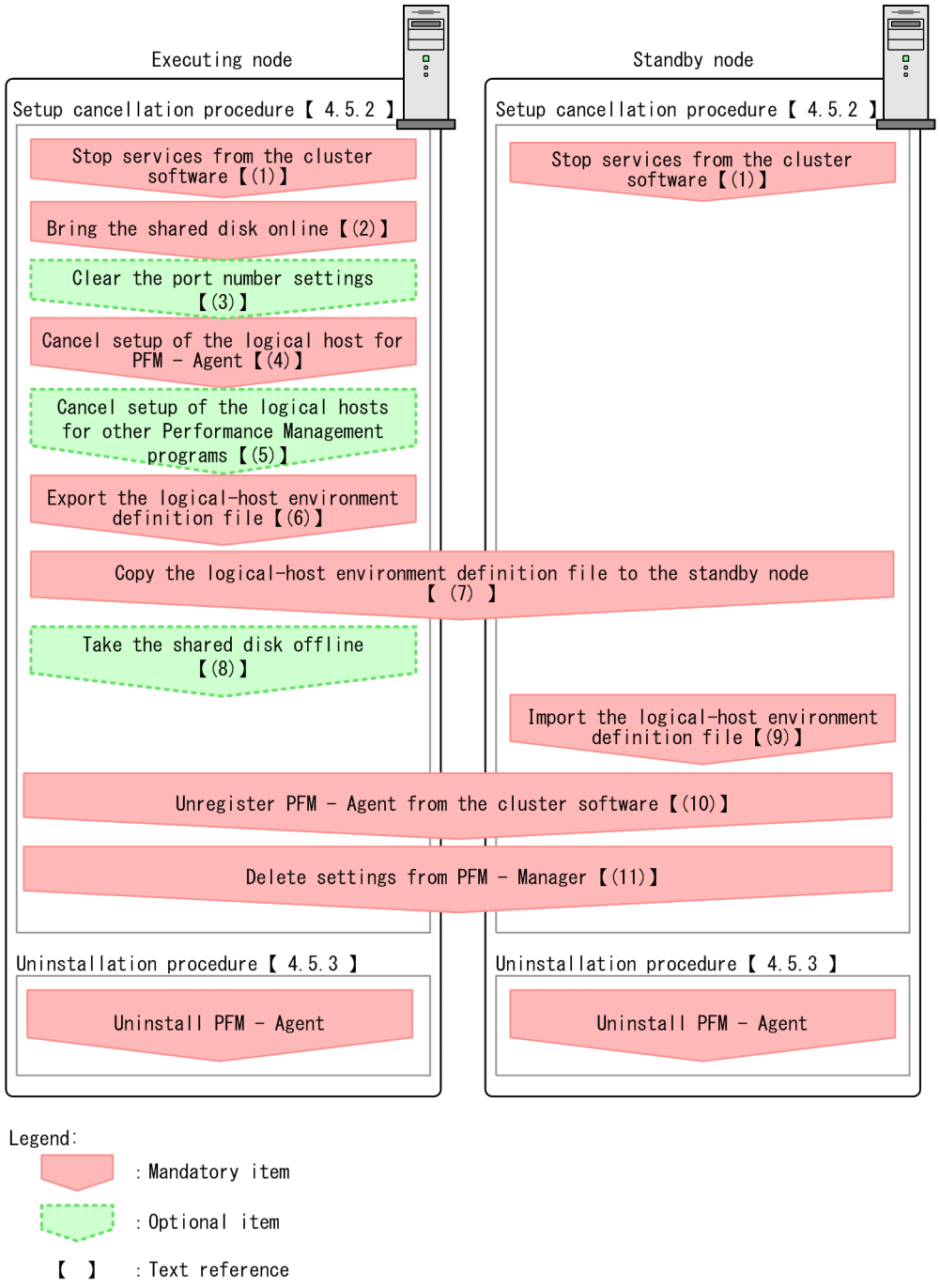
This section describes how to cancel setup of PFM - Agent for Oracle and how to uninstall PFM - Agent for Oracle in a cluster system.

For details about how to cancel setup of PFM - Manager and how to uninstall PFM - Manager, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.5.1 Setup cancellation and uninstallation workflow of PFM - Agent for Oracle

The following figure shows the setup cancellation and uninstallation workflow of PFM - Agent for Oracle in a cluster system.

Figure 4–6: Setup cancellation and uninstallation workflow of PFM - Agent for Oracle running on a logical host in a cluster system (Windows)



4.5.2 Setup cancellation procedure

This section describes the procedure for canceling setup of the logical host environment. There are separate procedures for the executing node and the standby node. Cancellation of setup must be performed first on the executing node and then on the standby node.

Executing indicates a task to be performed on the executing node. Standby indicates a task to be performed on the standby node. Option indicates a setup item that is required depending on the environment or an optional setup item used when the default is to be changed.

The following subsections describe how to cancel setup of PFM - Agent for Oracle.

(1) Stop services from the cluster software Executing Standby

Stop all the Performance Management programs and services running on the executing and standby nodes by using instructions from the cluster system. For details about how to stop the programs and services, see the documentation for your cluster software.

(2) Bring the shared disk online Executing

Make sure that the shared disk is online. If the shared disk is not online, use the cluster software or the volume manager to bring it online.

(3) Clear the port number settings Executing Option

Perform this step only if you are running a firewall environment, and you used the `jpcconf port` command to set port numbers during setup.

For details about how to clear the port number settings, see the chapter on installation and setup and the chapters on setup and operation in a cluster system in the *JP1/Performance Management Planning and Configuration Guide.*

(4) Canceling setup of the logical host environment for the executing node Executing

The following procedure shows how to cancel setup of the logical host environment for the executing node.

Note:

If the shared disk is not mounted when you delete the logical host environment, the logical host settings are deleted from the physical host only, and the directories and files on the shared disk will remain. In this case, bring the shared disk online, and then manually delete the `jp1pc` directory from the environment directory.

To cancel setup of the logical host environment for the executing node:

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-halora
   ```

   Before canceling setup of the logical host environment, check the current settings, including the logical host name and the path to the shared disk.

2. Clear the instance environment setting of PFM - Agent for Oracle.
   Execute the command as follows:

   ```
   jpcconf inst unsetup -key Oracle -lhost jp1-halora -inst SDC1
   ```

   Although an example of interactive command execution is shown here, the `jpcconf inst unsetup` command can be also executed non-interactively. For details about the `jpcconf inst unsetup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

When you execute the `jpcconf inst unsetup` command, the settings that allow the instance to start on the logical host are deleted. The files on the shared disk that relate to the logical host are also deleted.

3. Execute the `jpcconf ha unsetup` command to delete the logical host environment for PFM - Agent for Oracle.

   Execute the command as follows:

   ```
   jpcconf ha unsetup -key Oracle -lhost jp1-halora
   ```

   When you execute the `jpcconf ha unsetup` command, the settings that allow the instance to start on the logical host are deleted. Files on the shared disk that relate to the logical host are also deleted.

4. Use the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Confirm that PFM - Agent for Oracle has been deleted from the logical host environment.

5. Delete objects from the Oracle Database.

   Delete from Oracle the objects that PFM - Agent for Oracle requires to monitor Oracle (monitoring procedures, work tables, etc.). To delete these objects, use the SQL deletion script that PFM - Agent for Oracle provides. For details about how to delete objects from the Oracle Database, see *2.2.2(1)(b) Deleting the objects registered in the Oracle Database*.

6. Restore the Oracle initialization parameter to its original setting.

   If you have changed the value of the Oracle initialization parameter `TIMED_STATISTICS` for collecting records of PFM - Agent for Oracle, restore it to its original setting if necessary.

7. Delete the Oracle accounts and the tablespaces used by the accounts.

   Delete the Oracle accounts that have become unnecessary for PFM - Agent for Oracle. If the tablespaces that were used by deleted accounts are unnecessary, also delete the tablespaces.

   For details about how to delete Oracle accounts, see *2.2.2(2) Deleting an Oracle account used in PFM - Agent for Oracle*.

(5) Cancel setup of the logical host environments for other Performance Management programs Executing Option

If there are Performance Management programs for which you want to cancel setup in addition to PFM - Agent for Oracle on the same logical host, cancel setup of these at this point.

For details about how to cancel setup, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*, or the chapters on operation in a cluster system in the applicable PFM - Agent manual.

(6) Export the logical-host environment definition file Executing

After you have deleted PFM - Agent for Oracle from the logical host, export the environment definition as a file.

Performance Management allows you to synchronize the environments on the executing node and standby node by exporting the environment definition from one node and importing it into the other.

When you import the environment definition (without the Performance Management component) from the executing node into the standby node, the imported environment definition is compared with the existing environment definition (containing the Performance Management component) and the difference between the two is verified. The Performance

Management environment definition is then cleared from the standby node so that both nodes have the same environment.

To export the logical-host environment definition file:

1. Execute the `jpcconf ha export` command to export the logical host environment definition.

   The definition information for the logical host environment of Performance Management is output to an export file. You can give this file any name you wish.

   For example, execute the command as follows to output the logical host environment definition to the file `lhostexp.txt`:

   ```
   jpcconf ha export -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha export` command can be also executed non-interactively. For details about the `jpcconf ha export` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(7) Copy the logical-host environment definition file to the standby node

Executing Standby

Copy the file you exported in *(6) Export the logical-host environment definition file* from the executing node to the standby node.

(8) Take the shared disk offline Executing Option

Use the cluster software or the volume manager to take the shared disk offline. Note that if you intend to continue using the shared disk, you do not need to take it offline at this point.

(9) Import the logical-host environment definition file Standby

On the standby node, import the exported file you copied from the executing node in order to synchronize the environment definitions of both nodes. You do not need to take the shared disk offline on the standby node before importing the file. Note that the shared disk does not need to be offline when you execute this command.

To import the logical-host environment definition file:

1. Execute the `jpcconf ha import` command to import the environment definition of the logical host.

   Execute the command as follows:

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha import` command can be also executed non-interactively. For details about the `jpcconf ha import` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

   When you execute the command, the environment on the standby node is changed to reflect the environment described in the export file. This clears the settings that allow PFM - Agent for Oracle to start on the logical host. These settings are also cleared for any other Performance Management programs for which you cancel setup on the logical host.

   If you used the `jpcconf port` command during setup to assign fixed port numbers, the port numbers will no longer be assigned.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

```
jpcconf ha list -key all
```

Confirm whether the same output is displayed as when you executed the `jpcconf ha list` command on the executing node.

(10) Unregister PFM - Agent for Oracle in the cluster software Executing Standby

Delete the settings related to PFM - Agent for Oracle on the logical host from the cluster software.

For details about deleting these settings, see the documentation for your cluster software.

(11) Delete settings from PFM - Manager Executing Standby

Log in to PFM - Manager from PFM - Web Console, and delete the definition relating to the PFM - Agent for Oracle for which you want to cancel setup.

To delete the definition:

1. Start the PFM - Manager service.

   If you have stopped the PFM - Manager services from the cluster software as described in *(1) Stop services from the cluster software*, use the cluster software to start the PFM - Manager services. For details about how to start the services, see the cluster software documentation.

2. From PFM - Web Console, delete the agent.

3. Delete the agent information in PFM - Manager.

   For example, if PFM - Manager is running on the logical host `jp1-hal`, and PFM - Agent for Oracle is running on the logical host `jp1-halora`, execute the following command to delete the agent:

   ```
   jpctool service delete -id service-ID -host jp1-halora  -lhost jp1-hal
   ```

   In *service-ID*, specify the service ID of the agent you want to delete.

4. Restart the PFM - Manager service.

   For details about how to start services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

5. Apply the service information of the PFM - Manager host.

   In order to update the PFM - Web Console host to reflect the deletion of service information, synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host. Use the `jpctool service sync` command to synchronize the agent information.

4.5.3 Uninstallation procedure

Uninstall PFM - Agent for Oracle from the executing and standby nodes.

The uninstallation procedure is the same as the uninstallation procedure in a non-cluster system. For details, see *2.2.3 Procedure for uninstallation*.

Notes:

- Before you uninstall PFM - Agent for Oracle, stop all Performance Management programs and services on the node from which you are uninstalling PFM - Agent for Oracle.
- If you uninstalled PFM - Agent for Oracle without deleting the logical host environment, the environment directory may remain on the disk. In this case, delete the environment directory manually.

4.6 Setup cancellation and uninstallation (UNIX)

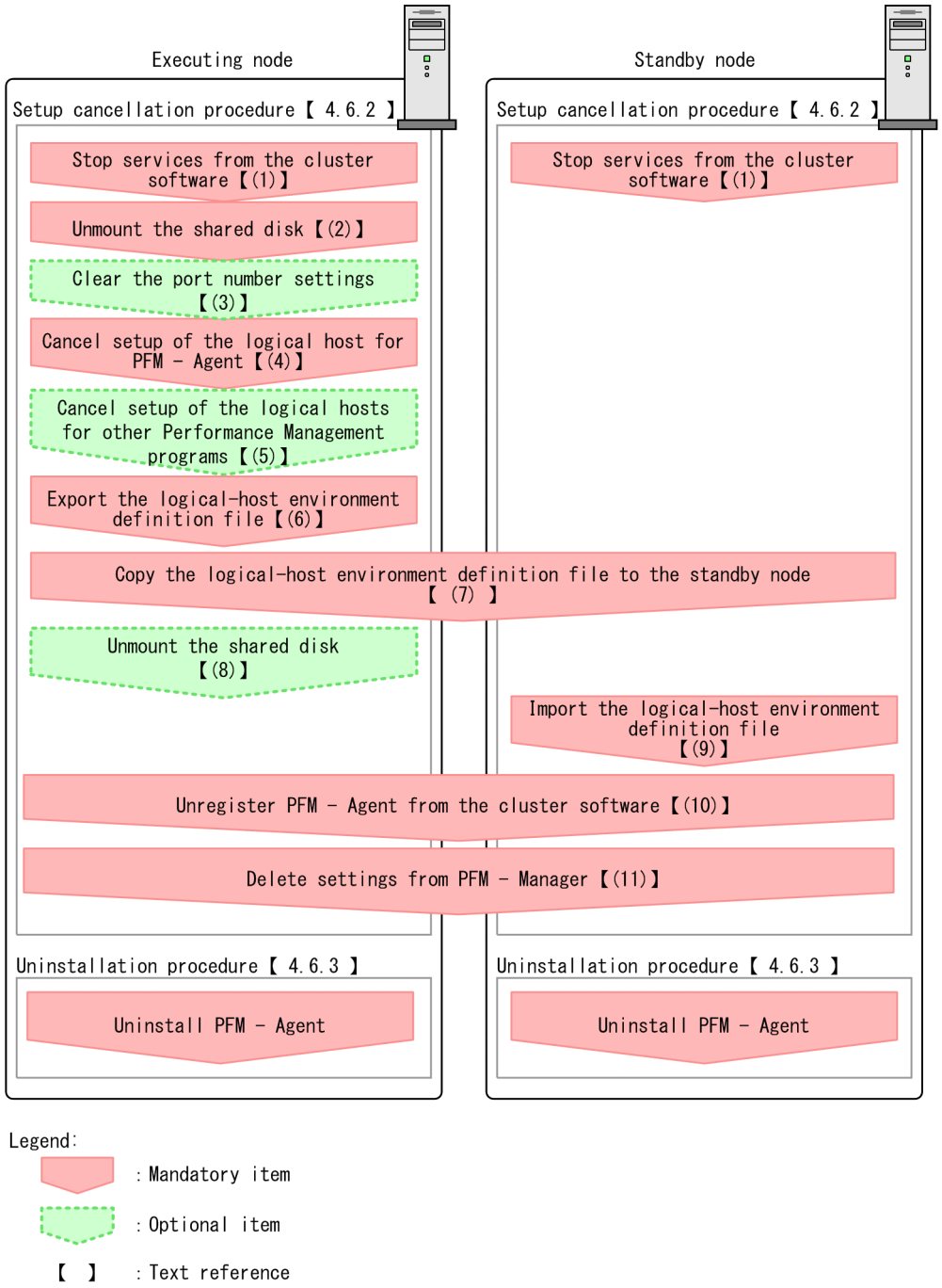
This section describes how to cancel setup of PFM - Agent for Oracle and how to uninstall PFM - Agent for Oracle in a cluster system.

For details about how to cancel setup of PFM - Manager and how to uninstall PFM - Manager, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*.

4.6.1 Setup cancellation and uninstallation workflow of PFM - Agent for Oracle

The following figure shows the setup cancellation and uninstallation workflow of PFM - Agent for Oracle in a cluster system.

Figure 4–7: Setup cancellation and uninstallation workflow of PFM - Agent for Oracle running on a logical host in a cluster system (UNIX)



4.6.2 Setup cancellation procedure

This section describes the procedure for canceling setup of the logical host environment. There are separate procedures for the executing node and the standby node. Cancellation of setup must be performed first on the executing node and then on the standby node.

Executing indicates a task to be performed on the executing node. Standby indicates a task to be performed on the standby node. Option indicates a setup item that is required depending on the environment or an optional setup item used when the default is to be changed.

The following subsections describe how to cancel setup of PFM - Agent for Oracle.

(1) Stop services from the cluster software Executing Standby

Stop all the Performance Management programs and services running on the executing and standby nodes by using instructions from the cluster system. For details about how to stop the programs and services, see the documentation for your cluster software.

(2) Mount the shared disk Executing

Make sure that the shared disk is mounted. If the shared disk is not mounted, mount it using the cluster software or the volume manager.

Note:

Check whether the `jp1pc` directory and the files for the logical host environment exist on the local disk in the environment directory of the logical host for which you are canceling setup. If they exist, this indicates that setup was performed without mounting the shared disk. In this case, take the following action:

1. On the local disk, use the `tar` command to archive the `jp1pc` directory in the environment directory of the logical host for which you are canceling setup.
2. Mount the shared disk.
3. If the environment directory for the logical host for which you are canceling setup does not exist on the shared disk, create it now.
4. On the shared disk, expand the `tar` file into the environment directory for which you are canceling setup.
5. Unmount the shared disk.
6. On the local disk, delete the `jp1pc` directory and its contents from the environment directory of the logical host.

(3) Clear the port number settings Executing Option

Perform this step only if you are running a firewall environment, and you used the `jpcconf port` command to set port numbers during setup.

For details about how to clear the port number settings, see the chapter on installation and setup and the chapters on setup and operation in a cluster system in the *JP1/Performance Management Planning and Configuration Guide.*

(4) Canceling setup of the logical host environment for PFM - Agent for Oracle Executing

The following procedure shows how to cancel setup of the logical host environment for PFM - Agent.

Note:

If the shared disk is not mounted when you delete the logical host environment, the logical host settings are deleted from the physical host only, and the directories and files on the shared disk will remain. In this case, bring the shared disk online, and then manually delete the `jp1pc` directory from 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-halora
   ```

   Before canceling setup of the logical host environment, check the current settings, including the logical host name and the path to the shared disk.

2. Clear the instance environment setting of PFM - Agent for Oracle.

   Execute the command as follows:

   ```
   jpcconf inst unsetup -key Oracle -lhost jp1-halora -inst SDC1
   ```

   When you execute the `jpcconf ha unsetup` command, `the settings that allow the instance to start on the logical host are deleted.` Files on the shared disk that relate to the instance on the logical host are also deleted.

3. Execute the `jpcconf ha unsetup` command to delete the logical host environment for PFM - Agent for Oracle.

   Execute the command as follows:

   ```
   jpcconf ha unsetup -key Oracle -lhost jp1-halora
   ```

   Although an example of interactive command execution is shown here, the `jpcconf inst unsetup` command can be also executed non-interactively. For details about the `jpcconf inst unsetup` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

   When you execute the `jpcconf ha unsetup` command, the settings that allow PFM - Agent for Oracle to start on the logical host are deleted. Files on the shared disk that relate to the logical host are also deleted.

4. Use the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Confirm that PFM - Agent for Oracle has been deleted from the logical host environment.

5. Delete objects from the Oracle Database.

   Delete from Oracle the objects that PFM - Agent for Oracle requires to monitor Oracle (monitoring procedures, work tables, etc.). To delete these objects, use the SQL deletion script that PFM - Agent for Oracle provides. For details about how to delete objects from the Oracle Database, see *3.2.2(1)(b) Deleting the objects registered in the Oracle Database*.

6. Restore the Oracle initialization parameter to its original setting.

   If you have changed the value of the Oracle initialization parameter `TIMED_STATISTICS` for collecting records of PFM - Agent for Oracle, restore it to its original setting if necessary.

7. Delete the Oracle accounts and the tablespaces used by the accounts.

   Delete the Oracle accounts that have become unnecessary for PFM - Agent for Oracle. If the tablespaces that were used by deleted accounts are unnecessary, also delete the tablespaces.

   For details about how to delete Oracle accounts, see *3.2.2(2) Deleting an Oracle account used in PFM - Agent for Oracle*.

(5) Cancel setup of the logical host environments for other Performance Management programs Executing Option

If there are Performance Management programs for which you want to cancel setup in addition to PFM - Agent for Oracle on the same logical host, cancel setup of these at this point.

For details about how to cancel setup, see the chapters on setup and operation in a cluster system in the *JP1/Performance Management User's Guide*, or the chapters on operation in a cluster system in the applicable PFM - Agent manual.

(6) Export the logical-host environment definition file Executing

After you have deleted PFM - Agent for Oracle from the logical host, export the environment definition as a file.

Performance Management allows you to synchronize the environments on the executing node and standby node by exporting the environment definition from one node and importing it into the other.

When you import the environment definition (without the Performance Management component) from the executing node into the standby node, the imported environment definition is compared with the existing environment definition (containing the Performance Management component) and the difference between the two is verified. The Performance Management environment definition is then cleared from the standby node so that both nodes have the same environment.

To export the logical-host environment definition file:

1. Execute the `jpcconf ha export` command to export the logical host environment definition.

   The definition information for the logical host environment of Performance Management is output to an export file. You can give this file any name you wish.

   For example, execute the command as follows to output the logical host environment definition to the file `lhostexp.txt`:

   ```
   jpcconf ha export -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha export` command can be also executed non-interactively. For details about the `jpcconf ha export` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

(7) Copy the logical-host environment definition file to the standby node

Executing Standby

Copy the file you exported in *(6) Export the logical-host environment definition file* from the executing node to the standby node.

(8) Unmount the shared disk Executing Option

To complete the task, unmount the file system. Note that if you intend to continue using the shared disk, you do not need to unmount it this point.

(9) Import the logical-host environment definition file Standby

On the standby node, import the exported file you copied from the executing node in order to synchronize the environment definitions of both nodes. You do not need to unmount the shared disk on the standby node before importing the file.

To import the logical-host environment definition file:

1. Execute the `jpcconf ha import` command to import the environment definition of the logical host.

   Execute the command as follows:

   ```
   jpcconf ha import -f lhostexp.txt
   ```

   Although an example of interactive command execution is shown here, the `jpcconf ha import` command can be also executed non-interactively. For details about the `jpcconf ha import` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*.

   When you execute the command, the environment on the standby node is changed to reflect the environment described in the export file. This clears the settings that allow PFM - Agent for Oracle to start on the logical host. These settings are also cleared for any other Performance Management programs for which you cancel setup on the logical host.

   If you used the `jpcconf port` command during setup to assign fixed port numbers, the port numbers will no longer be assigned.

2. Execute the `jpcconf ha list` command to check the logical host settings.

   Execute the command as follows:

   ```
   jpcconf ha list -key all
   ```

   Confirm whether the same output is displayed as when you executed the `jpcconf ha list` command on the executing node.

(10) Unregister PFM - Agent for Oracle in the cluster software Executing Standby

Delete the settings related to PFM - Agent for Oracle on the logical host from the cluster software.

For details about deleting these settings, see the documentation for your cluster software.

(11) Delete settings from PFM - Manager Executing Standby

Log in to PFM - Manager from PFM - Web Console, and delete the definition relating to the PFM - Agent for Oracle for which you want to cancel setup.

To delete the definition:

1. Start the PFM - Manager service.
2. If you have stopped the PFM - Manager services from the cluster software as described in *(1) Stop services from the cluster software*, use the cluster software to start the PFM - Manager services. For details about how to start the services, see the cluster software documentation.
3. From PFM - Web Console, delete the agent.
4. Delete the agent information in PFM - Manager.

   For example, if PFM - Manager is running on the logical host `jp1-hal`, and PFM - Agent for Oracle is running on the logical host `jp1-halora`, execute the following command to delete the agent:

   ```
   jpctool service delete -id service-ID -host jp1-halora -lhost jp1-hal
   ```

   In *service-ID*, specify the service ID of the agent you want to delete.

5. Restart the PFM - Manager service.

   For details about how to start services, see the chapter on starting and stopping Performance Management in the *JP1/Performance Management User's Guide*.

6. Apply the service information of the PFM - Manager host.

   In order to update the PFM - Web Console host to reflect the deletion of service information, synchronize the agent information of the PFM - Manager host and that of the PFM - Web Console host. Use the `jpctool service sync` command to synchronize the agent information.

4.6.3 Uninstallation procedure

Uninstall PFM - Agent for Oracle from the executing and standby nodes.

The uninstallation procedure is the same as the uninstallation procedure in a non-cluster system. For details, see *3.2.3 Procedure for uninstallation*.

Notes:

- Before you uninstall PFM - Agent for Oracle, stop all Performance Management programs and services on the node from which you are uninstalling PFM - Agent for Oracle.
- If you uninstalled PFM - Agent for Oracle without deleting the logical host environment, the environment directory may remain on the disk. In this case, delete the environment directory manually.

4.7 Notes on operating PFM - Agent for Oracle in a cluster system

This section provides notes on operating PFM - Agent for Oracle in a cluster system.

4.7.1 Host name in the collected performance data

The performance data PFM - Agent for Oracle collects includes a record that contains fields related to the host name. In the case of PFM - Agent for Oracle running on a logical host, the physical host name is stored in the indicated field of the record shown in the following table:

Record name	Field name	Stored host name	Description
Instance (PD_PDI)	Host	Physical host name	The name of the host on which the connected instance is running.
Collection Instance 2 (PD_PCI)	Host	Physical host name	The name of the host on which the connected instance is running.

4.8 Changing the system configuration of PFM - Agent for Oracle

Depending on the change in the network configuration of a monitored system or a change in the host name, you might need to change the system configuration of PFM - Agent for Oracle.

When you change the system configuration of PFM - Agent for Oracle, you also need to change the settings of PFM - Manager or PFM - Web Console. For details about how to change the system configuration of Performance Management, see the chapter that describes installation and setup in the *JP1/Performance Management Planning and Configuration Guide*. For some kinds of PFM - Agent, changing the logical host name requires additional tasks specific to the PFM - Agent. However, PFM - Agent for Oracle does not require such additional specific tasks.

4.9 Changing the operation of PFM - Agent for Oracle

This section describes how to change the operation of PFM - Agent for Oracle in a cluster system. For details about how to change the configuration of the whole Performance Management system, see the chapter on the installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

4.9.1 Updating an instance environment

When you update an instance environment in a cluster system, you first check the logical host name and the name of the instance that you want to update. You update the instance information on the executing node.

Before you change an information item, see the sections below in advance:

- Windows: *2.4.2 Updating an instance environment*
- UNIX: *3.4.2 Updating an instance environment*

For details about Oracle instance information, see your Oracle documentation.

Use the `jpcconf ha list` command to check the logical host name and the instance name. To update an instance environment, use the `jpcconf inst setup` command.

Updating an instance environment involves the steps described below. To update multiple instance environments, repeat the procedure for each instance environment.

1. Check the logical host name and the instance name.

   Execute the `jpcconf ha list` command specified with the service key that indicates the PFM - Agent for Oracle in the instance environment that is to be updated:

   For example, if you want to check the logical host name and instance name of PFM - Agent for Oracle, execute the following command:

   Windows:

   ```
   jpcconf ha list -key Oracle
   ```

   UNIX:

   ```
   jpcconf ha list -key Oracle
   ```

   For example, if you execute the command in the system with logical host name jp1_Ora and instance name Ora1, the command will list the following information:

   ```
   Logical Host Name     Key       Environment Directory
                                   Path to the logical host  Instance Name
                                   environment directory
   jp1_Ora               agto      Ora1
   ```

2. If the PFM - Agent for Oracle service is active in the instance environment that is to be updated, use the cluster software to stop the service.

3. If the shared disk is unmounted when you stop the service, use the cluster software or the volume manager to mount it.

4. Execute the `jpcconf inst setup` command specified with the service key and the instance name that indicate the PFM - Agent for Oracle in the instance environment that is to be updated.

   For example, if you are updating the instance environment with logical host name jp1_Ora and instance name Ora1, execute the following command:

Windows:

```
jpcconf inst setup -key Oracle -lhost jp1_Ora -inst Ora1
```

UNIX:

```
jpcconf inst setup -key Oracle -lhost jp1_Ora -inst Ora1
```

5. Update the instance information for Oracle.

   In Windows, enter the information shown in *2.4.2 Updating an instance environment* in accordance with the command's instructions. In Unix, enter the information shown in *3.4.2 Updating an instance environment* in accordance with the command's instructions. The current settings are displayed (except for the value of oracle_passwd). To use the displayed value, press the Enter key. When you have finished entering information, the instance environment is updated.

6. Use the cluster software to restart the service in the updated instance environment.

   For details about the starting the service, see the chapter on starting and stopping Performance Management in a cluster system in the *JP1/Performance Management User's Guide*.

Notes:

- If you want to change an item that cannot be updated, delete the instance environment and then re-create it.
- To change an account for monitoring Oracle, use the following procedure:

  1. Delete the objects created by the account you want to change.
  2. Register new objects after you change the account.

  Performance data is not deleted when an account is changed.

  For details about how to delete objects, see the following sections:

  Windows: see *2.2.2(1)(b) Deleting the objects registered in the Oracle Database*

  UNIX: see *3.2.2(1)(b) Deleting the objects registered in the Oracle Database*

  For details about how to register objects, see the following sections:

  Windows: see *2.1.4(3)(b) Registering objects in the Oracle Database*

  UNIX: see *3.1.4(4)(b) Registering objects in the Oracle Database*

For details about the commands used in this procedure, see the chapter on the commands in the manual *JP1/Performance Management Reference*.

4.9.2 Exporting and importing the logical-host environment definition file

You must export and import the logical-host environment definition file only if you perform any one of these operations listed below:

- When you set up the logical host environment or set up the instance environment, you change the node system on the logical host.

  For details about how to set up a logical host environment of PFM - Agent for Oracle, see the following sections:

  - Windows: see *4.3.4(3) Set up the logical host environment for PFM - Agent for Oracle*
  - UNIX: see *4.4.4(3) Set up the logical host environment for PFM - Agent for Oracle*

  For details about how to set up an instance environment, see the sections below:

  - Windows: see *4.3.4(6) Set up an instance environment*
  - UNIX: see *4.4.4(6) Set up an instance environment*

- When you set up the logical host environment of other Performance Management programs, you perform an operation that requires exporting and importing the logical-host environment definition file.
  - The following sections describe how to set up the logical host environment of other Performance Management programs:
  - Windows: see *4.3.4(7) Set up the logical host environment for other Performance Management programs*
  - UNIX: see *4.4.4(7) Set up the logical host environment for other Performance Management programs*
- When you specify network setting, you set the port numbers.
  For details about how to specify the network settings, see the sections below:
  - Windows: see *4.3.4(8) Specify network settings*
  - UNIX: see *4.4.4(8) Specify network settings*

The following sections describe how to export and import the logical-host environment definition file:

- Windows: From *4.3.4(12) Export the logical-host environment definition file* to *4.3.4(15) Import the logical-host environment definition file*
- UNIX: From *4.4.4(12) Export the logical-host environment definition file* to *4.4.4(15) Import the logical-host environment definition file*

Note that you do not have to export and import the logical-host environment definition file when you update only an instance environment.

For details about how to update an instance environment, see *4.9.1 Updating an instance environment*.

5 Monitoring template

This chapter describes the monitoring template for PFM - Agent for Oracle.

Overview of the monitoring template

The Performance Management products enable you to define alarms and reports by the following methods:

- Using the alarms and reports defined by PFM - Agent
- Copying and customizing the alarms and reports defined by PFM - Agent
- Using a wizard to define new information

A set of alarms and reports provided by PFM - Agent is called a *monitoring template*. Because the necessary information is predefined for the reports and alarms in the monitoring template, you can copy them in order to use the monitoring template as is or you can customize them as appropriate for your environment. This eliminates the need to use the wizard to create new definitions, thus simplifying the preparations for monitoring the operating status of desired programs.

This chapter describes the alarm and report settings in the monitoring template that have been defined by PFM - Agent for Oracle.

For details about using the monitoring template, see the chapter on creating reports used for operation analysis or the chapter on alarm-based operation monitoring in the *JP1/Performance Management User's Guide*.

Note:

The threshold specified for the alarms in the monitoring template is a reference example. When you use an alarm in the monitoring template, copy it and set an appropriate threshold according to the environment and the OS.

Format of alarm explanations

This section describes the format used to explain alarms. Alarms are presented in alphabetical order. The explanatory format for each alarm is as follows.

Alarm name

Indicates the name of the alarm name in the monitoring template.

Overview

Provides an overview of the programs that can be monitored by the alarm.

Main settings

Explains the main settings for this alarm in a tabular format. The alarm settings in the table correspond to the settings in the Properties window that appears when you click an alarm icon on the **Alarms** window of PFM - Web Console and then click the **Properties** method. For details about each alarm setting, see the Properties window for the particular alarm in PFM - Web Console.

Hyphens (--) in the *Setting* column of the table indicate that any value set for the item will be invalid.

If the abnormal condition is the same as the warning condition in a conditional expression, the system issues only the abnormal alarm event.

Related reports

Indicates the reports in the monitoring template that are associated with this alarm. You can view the reports by clicking an agent icon on the **Agents** window of PFM - Web Console, and then clicking the icon for the **Display Alarm Status** method.

List of alarms

An alarm table named `PFM Oracle Template Alarms 12.10` contains the alarms that are defined in the monitoring template for PFM - Agent for Oracle (`12.10` indicates the version of the alarm table). This alarm table is stored in the `Oracle` folder that is displayed on the **Alarms** window of PFM - Web Console. The following table lists the alarms defined in the monitoring template.

Note that if the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some records or fields collect the following performance data:

- Performance data for the PDBs and root container (`CDB$ROOT`) that are being monitored
- Performance data for all PDBs and common performance data for database instances

For details, see *6. List of records for PFM - Agent for Oracle*.

Table 5–1: List of alarms

Alarm name	What is monitored
Buffer Cache Usage	Buffer cache usage ratio
Buffer Cache Waits	Contention for data in the database and rollback blocks
Dict. Cache Usage	Shared pool
Disk Sorts	Percentage of all sort operations executed on disk using memory and disk I/O operations
Free List Waits	Contention on free lists
Full Table Scans	Percentage of full table scans
Library Cache Usage	Library cache
Redo Log Contention	Occurrence frequency of wait events
Server Status	Availability of an Oracle instance.
Tablespace Usage	Available tablespace

Notes:

- Due to the sort performed by PFM - Agent for Oracle, if you use the alarm table `PFM Oracle Template Alarms` *MM*`.`*NN*# provided by the monitoring template to perform monitoring, a Disk Sorts alarm may occur. If the error occurs, increase the value of the initial parameters `SORT AREA SIZE` and `SORT AREA RETAINED SIZE` of Oracle appropriately. When you set the new values, restart Oracle to enable the settings.
- Due to the table scan performed by PFM - Agent for Oracle, if you use the alarm table `PFM Oracle Template Alarms` *MM*`.`*NN*# provided by the monitoring template to perform monitoring while only PFM - Agent for Oracle is manipulating the database, a Full Table Scans alarm may occur.
- When PFM - Agent for Oracle does not connect to the monitoring target Oracle, PFM - Agent for Oracle does not detect any alarms except Server Status.

#

MM`.`*NN* is version of the data model of the alarm table.

Buffer Cache Usage

Overview

The Buffer Cache Usage alarm monitors the usage ratio of the buffer cache.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Buffer cache hit %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Cache Hit %`
	Abnormal condition	`Cache Hit % < 85`
	Warning condition	`Cache Hit % < 95`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Cache Usage (8.0)`

Buffer Cache Waits

Overview

The Buffer Cache Waits alarm monitors for contention for data in the database and rollback blocks.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Buffer cache wait %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Buffer Busy Wait %`
	Abnormal condition	`Buffer Busy Wait % > 5`
	Warning condition	`Buffer Busy Wait % > 3`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Cache Usage (8.0)`

Dict. Cache Usage

Overview

The Dict. Cache Usage alarm monitors the shared pool.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Dictionary cache miss %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Dict Cache Get Miss %`
	Abnormal condition	`Dict Cache Get Miss % > 15`
	Warning condition	`Dict Cache Get Miss % > 10`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Cache Usage (8.0)`

Disk Sorts

Overview

The Disk Sorts alarm monitors the percentage of all sort operations executed on disk using memory and disk I/O operations.

This alarm may occur when only PFM - Agent for Oracle is manipulating the Oracle Database and no other applications are running. If this is the case, you can suppress this alarm from occurring by increasing the value of `SORT_AREA_SIZE` (or the value of `SORT_AREA_RETAINED_SIZE`, if specified). The guideline for this value is 204,800. After specifying this value, restart the Oracle Database to apply the new setting.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`%CVS% sorts on disk`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Sort Overflow %`
	Abnormal condition	`Sort Overflow % > 15`
	Warning condition	`Sort Overflow % > 10`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Real-Time/Disk Sorts - Top 10 Sessions (9.1)`

Free List Waits

Overview

The Free List Waits alarm monitors for contention on free lists.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`%CVS free list waits`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Free List Wait Events`
	Abnormal condition	`Free List Wait Events > 2`
	Warning condition	`Free List Wait Events > 1`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Real-Time/Longest Transactions - Top 10 Sessions (9.1)`

Full Table Scans

Overview

The Full Table Scans alarm monitors the percentage of full table scans.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Non-index lookups %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Non-Index Lookups %`
	Abnormal condition	`Non-Index Lookups % > 10`
	Warning condition	`Non-Index Lookups % > 5`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Full Table Scans (8.0)`

Library Cache Usage

Overview

The Library Cache Usage alarm monitors the library cache.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Library cache miss %CVS%`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Selected
	occurrence(s) during	`2`
	interval(s)	`3`
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Lib Cache Miss %`
	Abnormal condition	`Lib Cache Miss % > 2`
	Warning condition	`Lib Cache Miss % > 1`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Cache Usage (8.0)`

Redo Log Contention

Overview

The Redo Log Contention alarm monitors the occurrence frequency of wait events concerning REDO log.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`%CVS redo log space requests waited`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	System Stat Summary Interval (`PI`)
	Field	`Redo Log Space Requests`
	Abnormal condition	`Redo Log Space Requests > 2`
	Warning condition	`Redo Log Space Requests > 1`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Troubleshooting/Recent Past/Redo Log Buffer Contention (8.0)`

Server Status

Overview

The Server Status alarm monitors the availability of Oracle instances.

The monitoring target of the Server Status alarm differs depending on the sqlnet value, which is instance information specified when an PFM - Agent for Oracle instance environment is configured.

Table 5–2: Relationship Between the sqlnet Value and the Value in the Availability Field

sqlnet value	Connections to Oracle Database	Monitoring Target of the Server Status alerm
Y	The listener is used to connect to Oracle Database.	Availability of the listener and Oracle Database.
N	The listener is not used to connect to Oracle Database.	Availability of Oracle Database only.

For details about how to configure the instance environment, see *2.1.4(3) Set up an instance environment* (for Windows), or *3.1.4(4) Set up an instance environment* (for UNIX).

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	Oracle(9.2)
	Alarm message	Database server availability = %CVS
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	Instance Availability (PD_PDIA)
	Field	Availability
	Abnormal condition	Availability = 0
	Warning condition	Availability = 0

Legend:

--: The setting is always invalid.

Related reports

Reports/Oracle/Status Reporting/Real-Time/System Overview (9.1)

Tablespace Usage

Overview

The Tablespace Usage alarm monitors available tablespace.

Main settings

Alarm properties in PFM - Web Console		Setting
Item	Detailed item	
Main Information	**Product**	`Oracle(9.2)`
	Alarm message	`Tablespace %CVS% free`
	Enable alarm	Selected
	Alarm notification	**Notify when the state changed**
	Notification target	**State changes for the alarm**
	Evaluate all data	Not selected
	Monitoring time range	Always
	Report alarm when the following damping condition is reached	Not selected
	occurrence(s) during	--
	interval(s)	--
Alarm Conditions	**Record**	Tablespace (`PD_PDTS`)
	Field	`Free %`
	Abnormal condition	`Free % < 10`
	Warning condition	`Free % < 20`
Actions	**E-mail**	--
	Command	--
	SNMP	**Abnormal**, **Warning**, **Normal**

Legend:

--: The setting is always invalid.

Related reports

`Reports/Oracle/Status Reporting/Real-Time/Tablespace Status (4.0)`

Format of report explanations

This section describes the format used to explain reports. The manual lists the reports in alphabetical order. Each report contains the following parts:

Report name

Indicates the report name for the monitoring template.

- A report whose name contains `(Multi-Agent)` presents information about multiple instances.
- A report whose name does not contain `(Multi-Agent)` presents information about a single instance.
- A report whose name contains `(9.2)` indicates that the data model of the record used for the report is 9.2.
- A report whose name contains `(9.1)` indicates that the data model of the record used for the report is 9.1.
- A report whose name contains `(8.0)` indicates that the data model of the record used for the report is 8.0.
- A report whose name contains `(5.0)` indicates that the data model of the record used for the report is 5.0.
- A report whose name contains `(4.0)` indicates that the data model of the record used for the report is 4.0.
- A report whose name does not contain `(9.2)`, `(9.1)`, `(8.0)`, `(5.0)`, `(4.0)` indicates that the data model of the record used for the report is 3.0.

For details about the data models, see *6. Records*.

Overview

Provides an overview of the information that can be displayed in the report.

Storage location

Indicates the storage location of the report.

Record

Indicates the record that contains the performance data used in the report. To display a historical report, you must specify information in advance in order to collect the record indicated in this column. Before displaying a report, display the agent properties in the Agents window of PFM - Web Console, and make sure that `Log` = `Yes` is set for this record. This setting is not needed to display a real-time report.

Fields

Provides a table that describes the fields used in the report.

Drilldown reports (report level)

Provides a table that lists other reports in the monitoring template that are related to this report. To display these drilldown reports, in the PFM - Web Console report window, select the name of a desired drilldown report from the **Drilldown report** drop-down list, and then click **Display Report**. Note that some reports do not have any drilldown reports.

Drilldown reports (field level)

Provides a table that describes reports in the monitoring template that are associated with fields used in this report. To display these drilldown reports, in the PFM - Web Console report window, choose the name of a desired field that is displayed under the graph or at the lower part of the report window. In the case of a historical report, choosing the time displayed in blue displays the report in smaller intervals. Note that some reports do not have any drilldown reports.

Organization of report folders

The following shows the organization of the report folders for PFM - Agent for Oracle. Angle brackets enclose folder names:

```
<Oracle>
 +-- <Monthly Trend>
 |    +-- Cache Usage Trend(Multi-Agent)(8.0)
 |    +-- Database Activity Trend(Multi-Agent)(8.0)
 |    +-- <Advanced>
 |         +-- Database Space Trend(Multi-Agent)(8.0)
 |         +-- Datafile I/O Trend Summary(8.0)
 |         +-- SGA Status Summary(9.1)
 |         +-- <Drilldown Only>
 |              +-- Datafile I/O Trend Detail(Reads)(8.0)
 |              +-- Datafile I/O Trend Detail(Writes)(8.0)
+-- <Status Reporting>
 |    +-- <Daily Trend>
 |    |    +-- Cache Usage Status(Multi-Agent)(8.0)
 |    |    +-- Database Activity Status(Multi-Agent)(8.0)
 |    |    +-- <Advanced>
 |    |         +-- Database Space Summary(Multi-Agent)(8.0)
 |    |         +-- Datafile I/O Status Summary(8.0)
 |    |         +-- <Drilldown Only>
 |    |             +-- Datafile I/O Status Detail(Reads)(8.0)
 |    |             +-- Datafile I/O Status Detail(Writes)(8.0)
 |    +-- <Real-Time>
 |         +-- CDB Audit Size Status(9.2)
 |         +-- CDB Data File Size Status (9.2)
 |         +-- CDB Diagnostics Size Status(9.2)
 |         +--CDB Status(9.2)
 |         +-- CDB Tablespace Status(9.2)
 |         +-- Database Activity Status(8.0)
 |         +-- Database Space Overview(8.0)
 |         +-- Fast Recovery Area Status(9.2)
 |         +-- Server Configuration Status(4.0)
 |         +-- SGA Status(9.1)
 |         +-- System Overview(9.1)
 |         +-- Tablespace Status(4.0)
 |         +-- <Drilldown Only>
 |              +-- Database Activity Status Detail(8.0)
 |              +-- Tablespace Status Detail(4.0)
+-- <Troubleshooting>
      +-- <Real-Time>
      |    +-- Blocking Locks(9.1)
      |    +-- Disk Sorts - Top 10 Sessions(9.1)
      |    +-- I/O Activity - Top 10 Datafiles(9.1)
      |    +-- Locked Objects(9.1)
      |    +-- Lock Usage - Top 10 Sessions(9.1)
      |    +-- Longest Transactions - Top 10 Sessions(9.1)
      |    +-- Memory Usage - Top 10 Sessions(9.1)
      |    +-- Physical I/O - Top 10 Sessions(9.1)
      |    +-- System Overview(9.1)
      |    +-- <Drilldown Only>
      |         +-- Datafile I/O Activity Detail(8.0)
      |         +-- Error Log(9.1)
      |         +-- Open Cursors(4.0)
```

```
|          +-- Session Detail(9.1)
|          +-- Session Statistics Detail(4.0)
|          +-- SQL Text(4.0)
+-- <Recent Past>
     +-- Cache Usage(8.0)
     +-- Full Table Scans(8.0)
     +-- Redo Log Buffer Contention(8.0)
```

The following describes each folder:

- `Monthly Trend` folder

  This folder contains reports that display daily information for the past month. Use it to check monthly trends in the system.

- `Status Reporting` folder

  This folder contains reports for displaying daily information. Use this folder to check the overall status of the system. You can display real-time reports as well as historical reports.

  - `Daily Trend` folder

    This folder contains reports for displaying hourly information for the past 24 hours. Use it to check the daily status of the system.

  - `Real-Time` folder

    This folder contains real-time reports for checking the system status.

- `Troubleshooting` folder

  This folder contains reports for displaying information that is useful for resolving problems. In the event of a system problem, use the reports in this folder to check the cause of the problem.

  - `Real-Time` folder

    This folder contains real-time reports for checking the current system status.

  - `Recent Past` folder

    This folder contains historical reports for displaying minute-by-minute information for the past hour.

These folders may also include the following folders. Which folder is included depends on the higher folder. The following describes each folder.

- `Advanced` folder

  This folder contains reports that use a record for which **Log** is set to **No** by default. To display any of these reports, you must use PFM - Web Console to change the record setting so that **Log** is set to **Yes**.

- `Drilldown Only` folder

  This folder contains reports that are displayed as drilldown reports (field level). Use it to display detailed information about fields contained in the parent report.

List of reports

Table 5-3 lists the reports defined in the monitoring template in alphabetical order.

Note that if the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some records or fields collect the following performance data:

- Performance data for the PDBs and root container (`CDB$ROOT`) that are being monitored
- Performance data for all PDBs and common performance data for database instances

For details, see *6. List of records for PFM - Agent for Oracle*.

Table 5–3: List of reports

Report name	Displayed information	Storage location
Blocking Locks (9.1)	Session that includes a lock that places another session in wait status	`Reports/Oracle/Troubleshooting/Real-Time/`
Cache Usage (8.0)	Buffer cache usage ratio per minute over the past hour	`Reports/Oracle/Troubleshooting/ Recent Past/`
Cache Usage Status (Multi-Agent) (8.0)	Overview of the buffer cache usage ratio per hour over the past 24 hours for multiple agents	`Reports/Oracle/Status Reporting/ Daily Trend/`
Cache Usage Trend (Multi-Agent) (8.0)	Overview of the buffer cache usage ratio per day over the past month for multiple agents	`Reports/Oracle/Monthly Trend/`
CDB Audit Size Status (9.2)	Information regarding the size of the unified audit file inside a PDB	`Reports/Oracle/Status Reporting/ Real-Time/`
CDB Data File Size Status(9.2)	Information regarding the size of the data file used by a PDB	`Reports/Oracle/Status Reporting/ Real-Time/`
CDB Diagnostics Size Status (9.2)	Information regarding the size of the diagnostic trace inside a PDB	`Reports/Oracle/Status Reporting/ Real-Time/`
CDB Status (9.2)	Information regarding all PDBs inside the CDB	`Reports/Oracle/Status Reporting/ Real-Time/`
CDB Tablespace Status (9.2)	Status of tablespaces that are present in all PDBs inside the CDB	`Reports/Oracle/Status Reporting/ Real-Time/`
Database Activity Status (8.0)	Database activity status	`Reports/Oracle/Status Reporting/ Real-Time/`
Database Activity Status Detail (8.0)	Details about database activity status	`Reports/Oracle/Status Reporting/ Real-Time/Drilldown Only/`
Database Activity Status (Multi-Agent) (8.0)	I/O statistical information on database instances per hour over the past 24 hours for multiple agents	`Reports/Oracle/Status Reporting/ Daily Trend/`
Database Activity Trend (Multi-Agent) (8.0)	Overview of I/O processing per day over the past month for multiple agents	`Reports/Oracle/Monthly Trend/`
Database Space Overview (8.0)	Performance data and general database information on tablespaces and data files	`Reports/Oracle/Status Reporting/ Real-Time/`
Database Space Summary (Multi-Agent) (8.0)	Overview of free space at the instance level per hour over the past 24 hours for multiple agents	`Reports/Oracle/Status Reporting/ Daily Trend/Advanced/`

Report name	Displayed information	Storage location
Database Space Trend (Multi-Agent) (8.0)	Overview of free space for instances per day over the past month for multiple agents	Reports/Oracle/Monthly Trend/ Advanced/
Datafile I/O Activity Detail (8.0)	Details about disk I/O operations on a data file	Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/
Datafile I/O Status Detail (Reads) (8.0)	Details about the average number of disk read operations on any data file per hour over the past 24 hours	Reports/Oracle/Status Reporting/ Daily Trend/Advanced/Drilldown Only/
Datafile I/O Status Detail (Writes) (8.0)	Details about the average number of disk write operations on any data file per hour over the past 24 hours	Reports/Oracle/Status Reporting/ Daily Trend/Advanced/Drilldown Only/
Datafile I/O Status Summary (8.0)	Number of disk I/O operations for each data file per hour over the past 24 hours	Reports/Oracle/Status Reporting/ Daily Trend/Advanced/
Datafile I/O Trend Detail (Reads) (8.0)	Details about disk read operations on any data file per day for one month	Reports/Oracle/Monthly Trend/ Advanced/Drilldown Only/
Datafile I/O Trend Detail (Writes) (8.0)	Details about disk write operations on any data file per day for one month	Reports/Oracle/Monthly Trend/ Advanced/Drilldown Only/
Datafile I/O Trend Summary (8.0)	Overview of disk I/O operations on data files per day over the past month	Reports/Oracle/Monthly Trend/ Advanced/
Disk Sorts - Top 10 Sessions (9.1)	Top 10 sessions in terms of the frequency of disk sort operations	Reports/Oracle/Troubleshooting/Real-Time/
Error Log (9.1)	All entries that have been recorded in the error log since the agent started	Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/
Fast Recovery Area Status (9.2)	Status of the fast recovery area	Reports/Oracle/Status Reporting/ Real-Time/
Full Table Scans (8.0)	Percentage of table lookups using no index per minute over the past hour	Reports/Oracle/Troubleshooting/ Recent Past/
I/O Activity - Top 10 Datafiles (9.1)	Top 10 data files in terms of the number of disk I/O operations	Reports/Oracle/Troubleshooting/Real-Time/
Lock Usage - Top 10 Sessions (9.1)	Top 10 sessions in terms of the number of locks being held	Reports/Oracle/Troubleshooting/Real-Time/
Locked Objects (9.1)	Objects that are locked by transactions	Reports/Oracle/Troubleshooting/Real-Time/
Longest Transactions - Top 10 Sessions (9.1)	Top 10 sessions in terms of the length of a transaction that placed another session in wait status	Reports/Oracle/Troubleshooting/Real-Time/
Memory Usage - Top 10 Sessions (9.1)	Top 10 sessions in terms of memory usage	Reports/Oracle/Troubleshooting/Real-Time/
Open Cursors (4.0)	Cursor opened by a session	Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/
Physical I/O - Top 10 Sessions (9.1)	Top 10 sessions in terms of concentration of I/O operations	Reports/Oracle/Troubleshooting/Real-Time/
Redo Log Buffer Contention (8.0)	Number of times a process waited for space in the REDO log buffer per minute over the past hour	Reports/Oracle/Troubleshooting/ Recent Past/
Server Configuration Status (4.0)	Information on setup parameters	Reports/Oracle/Status Reporting/ Real-Time/

Report name	Displayed information	Storage location
Session Detail (9.1)	Detailed session information	`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`
Session Statistics Detail (4.0)	Statistical information about a session	`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`
SGA Status (9.1)	Status of each component in SGA	`Reports/Oracle/Status Reporting/Real-Time/`
SGA Status Summary (9.1)	Overview of the status of each component in SGA per day over the past month	`Reports/Oracle/Monthly Trend/Advanced/`
SQL Text (4.0)	Performance data in the `SQL Text` and `Explain Plan` fields	`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`
System Overview (9.1) (real-time report on the overall status of instance)	Main performance data indicating the overall status of an instance in real-time	`Reports/Oracle/Status Reporting/Real-Time/`
System Overview (9.1) (real-time report on the general status of instance)	Performance data indicating the general status of an instance	`Reports/Oracle/Troubleshooting/Real-Time/`
Tablespace Status (4.0)	Status of all tablespaces in the database	`Reports/Oracle/Status Reporting/Real-Time/`
Tablespace Status Detail (4.0)	Details about a specified tablespace	`Reports/Oracle/Status Reporting/Real-Time/Drilldown Only/`

Blocking Locks(9.1)

Overview

The Blocking Locks(9.1) report displays real-time information about a session that includes a lock that places another session in wait status.

If you cannot display this report, create Oracle's static dictionary view DBA_WAITERS. To create the static dictionary view DBA_WAITERS, you must execute the CATBLOCK.SQL script provided by Oracle.

PFM - Agent for Oracle does not display this report if the Oracle Database does not have enough performance data to display the report.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Lock Waiters (PD_PDLW)

Fields

Field name	Description
Holding Session	Session ID that has the lock. To display the Locked Objects(9.1) report, choose this field.
Holding User	Name of the user that has the lock
Mode Held	Lock mode held during data collection
Mode Requested	Lock mode requested during data collection
Waiting Session	Session ID waiting for lock release. To display the Session Detail(9.1) report, choose this field.
Waiting User	Name of the user waiting for the lock to be released

Drilldown reports (report level)

Report name	Description
Longest Transactions - Top 10 Sessions(9.1)	Displays the top 10 transactions in terms of the length of a transaction that places another session in wait status.

Drilldown reports (field level)

Report name	Description
Locked Objects(9.1)	Displays the objects that are locked by the session. To display this report, choose the Holding Session field.
Session Detail(9.1)	Displays detailed information about the session. To display this report, choose the Waiting Session field.

Cache Usage(8.0)

Overview

The `Cache Usage(8.0)` report displays the usage ratio of the buffer cache per minute over the past hour.

Storage location

`Reports/Oracle/Troubleshooting/Recent Past/`

Record

System Stat Summary Interval (`PI`)

Fields

Field name	Description
Cache Hit %	Buffer cache usage
Dict Cache Get Miss %	Percentage of data resulting in cache miss
Lib Cache Miss %	Ratio of the objects loaded to the objects found in the library cache

Cache Usage Status(Multi-Agent)(8.0)

Overview

The Cache Usage Status(Multi-Agent)(8.0) report displays an overview of the buffer cache usage ratio per hour over the past 24 hours for multiple agents.

Storage location

Reports/Oracle/Status Reporting/Daily Trend/

Record

System Stat Summary Interval (PI)

Fields

Field name	Description
Cache Hit %	Buffer cache usage

Cache Usage Trend(Multi-Agent)(8.0)

Overview

The Cache Usage Trend(Multi-Agent)(8.0) report displays an overview of the buffer cache usage ratio per day over the past month for multiple agents.

Storage location

Reports/Oracle/Monthly Trend/

Record

System Stat Summary Interval (PI)

Fields

Field name	Description
Cache Hit %	Buffer cache usage

CDB Audit Size Status(9.2)

Overview

The CDB Audit Size Status(9.2) report displays the size of the unified audit file inside a PDB in real time.

This report is displayed when the root container or PDBs of a CDB in Oracle Database 12c R2 or later are monitored. This report is not displayed when a CDB configuration in Oracle Database 12c R1 or earlier or a conventional non-CDB configuration is monitored.

The following table describes the range of information acquired when the root container or PDBs in a CDB are monitored:

What is monitored	User performing the monitoring	Range of information acquired
Root container	Oracle administrative account (sys and system)	All PDBs inside a CDB
	Common user	No information can be acquired.
PDBs	All users	Performance data for the PDBs

If the size of the maximum space available for the unified audit file in a PDB (Max Audit Size) is either not set or set to *unlimited* in the Oracle Database to be monitored, values corresponding to the size of available maximum space and the size of available free space would be 0, and the percentage of available free space would be 100. Under this circumstance, field monitoring is not possible. For details, see *6. List of records for PFM - Agent for Oracle*.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

CDB Status (PD_CDCS)

Fields

Field name	Description
Container ID	Container ID
Name	PDB name
Audit File Size	Size of the unified audit file inside a PDB in megabytes
Max Audit Size	Size of the maximum space available for the unified audit file inside a PDB in megabytes
Free Audit Size	Size of free space available for the unified audit file inside a PDB in megabytes
Free Audit Size %	Percentage of free space available for the unified audit file inside a PDB

CDB Data File Size Status(9.2)

Overview

The CDB Data File Size Status(9.2) report displays the size of the data file used by a PDB in real time.

This report is displayed when the root container or PDBs of a CDB in Oracle Database 12c R2 or later are monitored. This report is not displayed when a CDB configuration in Oracle Database 12c R1 or earlier or a conventional non-CDB configuration is monitored.

The following table describes the range of information acquired when the root container or PDBs in a CDB are monitored:

What is monitored	User performing the monitoring	Range of information acquired
Root container	Oracle administrative account (sys and system)	All PDBs inside a CDB
	Common user	No information can be acquired.
PDBs	All users	Performance data for the PDBs

If the size of the maximum space available for both PDB data and temporary files (MAX_SIZE) is either not set or set to *unlimited* in the Oracle Database to be monitored, values corresponding to the size of available maximum space and the size of available free space would be 0, and the percentage of available free space would be 100. Under this circumstance, field monitoring is not possible. For details, see *6. List of records for PFM - Agent for Oracle*.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

CDB Status (PD_CDCS)

Fields

Field name	Description
Container ID	Container ID
Name	PDB name
Total Size	Size of the disk space for data and temporary files inside a PDB in megabytes
Max Size	Size of maximum space available for data and temporary files inside a PDB in megabytes
Free Size	Size of free space available for data and temporary files inside a PDB in megabytes
Free Size %	Percentage of free space available for data and temporary files inside a PDB

CDB Diagnostics Size Status(9.2)

Overview

The CDB Diagnostics Size Status(9.2) report displays the size of the diagnostic trace inside a PDB in real time.

This report is displayed when the root container or PDBs of a CDB in Oracle Database 12c R2 or later are monitored. This report is not displayed when a CDB configuration in Oracle Database 12c R1 or earlier or a conventional non-CDB configuration is monitored.

The following table describes the range of information acquired when the root container or PDBs in a CDB are monitored:

What is monitored	User performing the monitoring	Range of information acquired
Root container	Oracle administrative account (sys and system)	All PDBs inside a CDB
	Common user	No information can be acquired.
PDBs	All users	Performance data for the PDBs

If the size of the maximum space available for the diagnostic trace in a PDB (Max Diagnostics Size) is either not set or set to *unlimited* in the Oracle Database to be monitored, values corresponding to the size of available maximum space and the size of available free space would be 0, and the percentage of available free space would be 100. Under this circumstance, field monitoring is not possible. For details, see *6. List of records for PFM - Agent for Oracle*.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

CDB Status (PD_CDCS)

Fields

Field name	Description
Container ID	Container ID
Name	PDB name
Diagnostics Size	Size of the diagnostic trace in a PDB in megabytes
Max Diagnostics Size	Size of maximum space available for the diagnostic trace in a PDB in megabytes
Free Diagnostics Size	Size of free space available for the diagnostic trace in a PDB in megabytes
Free Diagnostics Size %	Percentage of free space available for the diagnostic trace in a PDB

CDB Status(9.2)

Overview

The CDB Status(9.2) report displays the state of a PDB in real time.

This report is displayed when the root container or PDBs of a CDB in Oracle Database 12c R2 or later are monitored. This report is not displayed when a CDB configuration in Oracle Database 12c R1 or earlier or a conventional non-CDB configuration is monitored.

The following table describes the range of information acquired when the root container or PDBs in a CDB are monitored:

What is monitored	User performing the monitoring	Range of information acquired
Root container	Oracle administrative account (sys and system)	All PDBs inside a CDB
	Common user	No information can be acquired.
PDBs	All users	Performance data for the PDBs

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

CDB Status (PD_CDCS)

Fields

Field name	Description
Container ID	Container ID
Name	PDB name
Open Mode	Open mode of the PDB
Restricted	Indicates whether only users with RESTRICTED SESSION privilege are allowed access.
Application Root	Indicates whether the PDB is an application root.
Application Seed	Indicates whether the PDB is an application seed.
Application PDB	Indicated whether the PDB is an application PDB.

CDB Tablespace Status(9.2)

Overview

The CDB Tablespace Status(9.2) report displays tablespace status in real time when Oracle Database 12c R2 or later is monitored.

The following table describes the range of information acquired when either a CDB or non-CDB configuration is monitored:

What is monitored	User performing the monitoring	Range of information acquired
Root container in a CDB configuration	Oracle administrative account (sys and system)	Root container and all PDBs inside a CDB
	Common user	Root container
PDBs in a CDB configuration#	All users	Performance data for the PDBs
Non-CDB configuration	All users	Database instance to be monitored

#
If the PDBs to be monitored are not open, information regarding the PDBs cannot be acquired.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

CDB Tablespace (PD_CDCS)

Fields

Field name	Description
Container ID	Container ID
Free %	Percentage of free space of the tablespace
Free Mbytes	Free space of the tablespace
Tablespace Name	Tablespace name
Used Mbytes	Size of used tablespace

Database Activity Status(8.0)

Overview

The Database Activity Status(8.0) report displays in real-time the database activity.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

Activity Summary (PD_PDAS)

Fields

Field name	Description
DML Locks %	Percentage of DML locks to the DML_LOCKS parameter in the init.ora initialization parameter file
Open Cursors %	Percentage of open cursors to the OPEN_CURSORS parameter in the init.ora initialization parameter file
Processes %	Percentage of processes to the PROCESSES parameter in the init.ora initialization parameter file
Sessions %	Percentage of sessions to the SESSIONS parameter in the init.ora initialization parameter file
Transactions %	Percentage of transactions to the TRANSACTIONS parameter in the init.ora initialization parameter file

Drilldown reports (report level)

Report name	Description
Database Activity Status Detail(8.0)	Displays details about the database activity status.

Database Activity Status Detail(8.0)

Overview

The Database Activity Status Detail(8.0) report displays in real-time details about the database activity status. This is a drilldown report.

Storage location

Reports/Oracle/Status Reporting/Real-Time/Drilldown Only/

Record

Activity Summary (PD_PDAS)

Fields

Field name	Description
Active Transactions	Number of active transactions in active sessions
DML Locks %	Percentage of DML locks to the DML_LOCKS parameter in the init.ora initialization parameter file
DML Locks Held	Number of current DML locks
Open Cursors	Number of current open cursors
Open Cursors %	Percentage of open cursors to the OPEN_CURSORS parameter in the init.ora initialization parameter file
Processes	Number of current Oracle processes
Processes %	Percentage of processes to the PROCESSES parameter in the init.ora initialization parameter file
Sessions	Number of current sessions
Sessions %	Percentage of sessions to the SESSIONS parameter in the init.ora initialization parameter file
Transactions %	Percentage of transactions to the TRANSACTIONS parameter in the init.ora initialization parameter file

Database Activity Status(Multi-Agent)(8.0)

Overview

The `Database Activity Status(Multi-Agent)(8.0)` report displays I/O statistical information on database instances per hour over the past 24 hours for multiple agents.

Storage location

`Reports/Oracle/Status Reporting/Daily Trend/`

Record

System Stat Summary Interval (`PI`)

Fields

Field name	Description
I/O Ops/sec	Number of I/O operations per second

Database Activity Trend(Multi-Agent)(8.0)

Overview

The Database Activity Trend(Multi-Agent)(8.0) report displays an overview of I/O processing per day over the past month for multiple agents.

Storage location

Reports/Oracle/Monthly Trend/

Record

System Stat Summary Interval (PI)

Fields

Field name	Description
I/O Ops/sec	Number of I/O operations per second

Database Space Overview(8.0)

Overview

The `Database Space Overview(8.0)` report displays in real-time performance data and general database information on tablespaces and data files.

Storage location

`Reports/Oracle/Status Reporting/Real-Time/`

Record

Database (`PD_PDDB`)

Fields

Field name	Description
Datafiles	Number of data files used by tablespaces
DB Name	Database name
Extents	Number of extents
Free %	Percentage of free space
Free Extents	Number of available extents
Free Mbytes	Free space in megabytes
Rollback Segments	Number of rollback segments
Segments	Number of segments
Tablespaces	Number of tablespaces
Used Mbytes	Used space in megabytes

Drilldown reports (report level)

Report name	Description
`Tablespace Status(4.0)`	Displays usage information on all tablespaces in the instance.

Database Space Summary(Multi-Agent)(8.0)

Overview

The `Database Space Summary(Multi-Agent)(8.0)` report displays an overview of free space at the instance level per hour over the past 24 hours for multiple agents.

Storage location

`Reports/Oracle/Status Reporting/Daily Trend/Advanced/`

Record

Database Interval (`PI_PIDB`)

Fields

Field name	Description
Datafiles	Number of data files used by the tablespace
Free %	Percentage of free space
Free Extents	Number of available extents
Mbytes	Size of the tablespace in megabytes
Tablespaces	Number of tablespaces

Database Space Trend(Multi-Agent)(8.0)

Overview

The `Database Space Trend(Multi-Agent)(8.0)` report displays an overview of free space for instances per day over the past month for multiple agents.

Storage location

`Reports/Oracle/Monthly Trend/Advanced/`

Record

Database Interval (`PI_PIDB`)

Fields

Field name	Description
Datafiles	Number of data files used by the tablespace
Free %	Percentage of free space
Free Extents	Number of available extents
Mbytes	Size of the tablespace in megabytes
Tablespaces	Number of tablespaces

Datafile I/O Activity Detail(8.0)

Overview

The `Datafile I/O Activity Detail(8.0)` report displays in real-time the details about disk I/O operations on a data file. This is a drilldown report.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`

Record

Data File (`PD_PDDF`)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Blocks Read	Number of physical block read operations
Physical Blocks Written	Number of physical block write operations
Physical Reads	Number of physical read operations
Physical Writes	Number of physical write operations
Tablespace Name	Tablespace name

Datafile I/O Status Detail(Reads)(8.0)

Overview

The `Datafile I/O Status Detail(Reads)(8.0)` report displays details about average disk read operations on any data file per hour over the past 24 hours. This is a drilldown report.

Storage location

`Reports/Oracle/Status Reporting/Daily Trend/Advanced/Drilldown Only/`

Record

Data File Interval (`PI_PIDF`)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Blocks Read	Number of physical block read operations
Physical Blocks Written	Number of physical block write operations
Physical Reads	Number of completed physical read operations
Physical Writes	Number of completed physical write operations
Tablespace Name	Tablespace name

Datafile I/O Status Detail(Writes)(8.0)

Overview

The `Datafile I/O Status Detail(Writes)(8.0)` report displays details about average disk write operations on any data file per hour over the past 24 hours. This is a drilldown report.

Storage location

`Reports/Oracle/Status Reporting/Daily Trend/Advanced/Drilldown Only/`

Record

Data File Interval (`PI_PIDF`)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Blocks Read	Number of physical block read operations
Physical Blocks Written	Number of physical block write operations
Physical Reads	Number of completed physical read operations
Physical Writes	Number of completed physical write operations
Tablespace Name	Tablespace name

Datafile I/O Status Summary(8.0)

Overview

The Datafile I/O Status Summary(8.0) report displays an overview of disk I/O operations for each data file per hour over the past 24 hours.

Storage location

Reports/Oracle/Status Reporting/Daily Trend/Advanced/

Record

Data File Interval (PI_PIDF)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Reads(Total)	Total number of physical read operations. Choose this field to display the Datafile I/O Status Detail(Reads)(8.0) report.
Physical Writes(Total)	Total number of physical write operations. Choose this field to display the Datafile I/O Status Detail(Writes)(8.0) report.

Drilldown reports (field level)

Report name	Description
Datafile I/O Status Detail(Reads)(8.0)	Displays details about average disk read operations on any data file per hour over the past 24 hours. To display this report, click the Physical Reads(Total) field.
Datafile I/O Status Detail(Writes)(8.0)	Displays details about average disk write operations on any data file per hour over the past 24 hours. To display this report, click the Physical Writes(Total) field.

Datafile I/O Trend Detail(Reads)(8.0)

Overview

The `Datafile I/O Trend Detail(Reads)(8.0)` report displays details about disk read operations on any data file per day for one month. This is a drilldown report.

Storage location

`Reports/Oracle/Monthly Trend/Advanced/Drilldown Only/`

Record

Data File Interval (`PI_PIDF`)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Blocks Read	Number of physical block read operations
Physical Blocks Written	Number of physical block write operations
Physical Reads	Number of completed physical read operations
Physical Writes	Number of completed physical write operations
Tablespace Name	Tablespace name

Datafile I/O Trend Detail(Writes)(8.0)

Overview

The `Datafile I/O Trend Detail(Writes)(8.0)` report displays details about disk write operations on any data file per day for one month. This is a drilldown report.

Storage location

`Reports/Oracle/Monthly Trend/Advanced/Drilldown Only/`

Record

Data File Interval (`PI_PIDF`)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Blocks Read	Number of physical block read operations
Physical Blocks Written	Number of physical block write operations
Physical Reads	Number of completed physical read operations
Physical Writes	Number of completed physical write operations
Tablespace Name	Tablespace name

Datafile I/O Trend Summary(8.0)

Overview

The Datafile I/O Trend Summary(8.0) report displays an overview of disk I/O operations on data files per day over the past month.

Storage location

Reports/Oracle/Monthly Trend/Advanced/

Record

Data File Interval (PI_PIDF)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Reads	Number of physical read operations that were completed during an interval. Choose this field to display the Datafile I/O Status Detail(Reads)(8.0) report.
Physical Writes	Number of physical write operations that were completed during an interval. Choose this field to display the Datafile I/O Status Detail(Writes)(8.0) report.

Drilldown reports (field level)

Report name	Description
Datafile I/O Trend Detail(Reads)(8.0)	Displays details about disk read operations on any data per day over the past one month. To display this report, choose the Physical Reads field.
Datafile I/O Trend Detail(Writes)(8.0)	Displays details about disk write operations on any data per day over the past one month. To display this report, choose the Physical Writes field.

Disk Sorts - Top 10 Sessions(9.1)

Overview

The `Disk Sorts - Top 10 Sessions(9.1)` report displays in real-time the top 10 sessions that frequently perform disk sort operations.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/`

Record

Session Statistics Summary (`PD_PDS2`)

Fields

Field name	Description
SID	Session ID
Sort Overflow %	Percentage of sort operations using temporary segments. To display the `Session Detail(9.1)` report, choose this field.
User	Oracle user name

Drilldown reports (field level)

Report name	Description
`Session Detail(9.1)`	Displays detailed information about a session. To display this report, choose the Sort Overflow % field.

Error Log(9.1)

Overview

The Error Log(9.1) report displays in real-time all entries that have been recorded in the error log since the agent started. This is a drilldown report.

PFM - Agent for Oracle does not display this report if the Oracle Database does not have enough performance data to display the report.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`

Record

Errorlog Detail (`PD_PDEL`)

Fields

Field name	Description
Container	PDB Container name and ID in which the error occured. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is blank.
Error #	Error number
Error File	Name of the file in which the error occurred
Error Time	Time the error occurred
Message	Error message

Fast Recovery Area Status(9.2)

Overview

The `Fast Recovery Area Status(9.2)` report displays in real-time the status of a fast recovery area.

Storage location

`Reports/Oracle/Status Reporting/Real-Time/`

Record

Collection Instance 2 (`PD_PCI`)

Fields

Field name	Description
Recovery File Dest	Location of the fast recovery area
Recovery Space Limit	Maximum size of the fast recovery area in megabytes
Recovery Space Used	Amount of space used in the fast recovery area in megabytes
Recovery Space Reclaimable	Size of obsolete, redundant, and other low priority files from the fast recovery area in megabytes
Recovery File Nums	Number of files in the fast recovery area
Recovery Space Free Mbytes	Amount of free space in the fast recovery area in megabytes
Recovery Space Free %	Percentage of free space in the fast recovery area

Full Table Scans(8.0)

Overview

The `Full Table Scans(8.0)` report displays the percentage of table lookups using no index per minute over the past hour.

Storage location

`Reports/Oracle/Troubleshooting/Recent Past/`

Record

System Stat Summary Interval (`PI`)

Fields

Field name	Description
Non-Index Lookups %	Percentage ratio of full table scans that do not involve caching

Drilldown reports (field level)

Report name	Description
`Cache Usage(8.0)`	Displays the buffer cache usage ratio. To display this report, choose the Non Index Lookups % field.

I/O Activity - Top 10 Datafiles(9.1)

Overview

The I/O Activity - Top 10 Datafiles(9.1) report displays in real-time the top 10 data files in terms of the number of disk I/O operations.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Data File (PD_PDDF)

Fields

Field name	Description
File #	File number
File Name	File name
Physical Reads	Number of physical read operations. To display the Datafile I/O Activity Detail(8.0) report, choose this field.
Physical Writes	Number of physical write operations. To display the Datafile I/O Activity Detail(8.0) report, choose this field.

Drilldown reports (report level)

Report name	Description
Datafile I/O Activity Detail(8.0)	Displays the details about I/O operations for all data files. To display this report, choose the Physical Read or Physical Writes field.
Physical I/O - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of the number of I/O operations.

Drilldown reports (field level)

Report name	Description
Datafile I/O Activity Detail(8.0)	Displays details about disk I/O operations for a specified data file. To display this report, choose the Physical Reads or Physical Writes field.

Lock Usage - Top 10 Sessions(9.1)

Overview

The Lock Usage - Top 10 Sessions(9.1) report displays in real-time the top 10 sessions in terms of the number of locks being held.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Session Detail (PD_PDS)

Fields

Field name	Description
Locks Held	Number of locks held during data collection. To display the Session Detail(9.1) report, choose this field.
SID	Session ID
User	Oracle user name

Drilldown reports (field level)

Report name	Description
Session Detail(9.1)	Display detailed information about a session. To display this report, choose the Locks Held field.

Locked Objects(9.1)

Overview

The Locked Objects(9.1) report displays real-time information about the objects that are locked by transactions.

PFM - Agent for Oracle does not display this report if the Oracle Database does not have enough performance data to display the report.

Note:

The drilldown report of the locked objects that are not performed transactions cannot be displayed correctly.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Transaction Lock (PD_PDTL)

Fields

Field name	Description
Locked Mode	Lock mode held. The valid values are as follows: • 1 (null) • 2 (row share) • 3 (row exclusive) • 4 (share) • 5 (share row exclusive) • 6 (exclusive)
Object Name	Object name
Object Type	Object type
Owner	Object owner
SID	Session ID. To display the Session Detail(9.1) report, choose this field.
User	Oracle user name

Drilldown reports (field level)

Report name	Description
Session Detail(9.1)	Displays detailed information about a session. To display this report, choose the SID field.

Longest Transactions - Top 10 Sessions(9.1)

Overview

The Longest Transactions - Top 10 Sessions(9.1) report displays in real-time the top 10 transactions in terms of the length of a transaction that placed another session in wait status.

PFM - Agent for Oracle does not display this report if the Oracle Database does not have enough performance data to display the report.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Transaction (PD_PDTR)

Fields

Field name	Description
Cache Hit %	Percentage of logical I/O operations to physical I/O operations.
Locks	Number of locks being held by the transaction. To display the Blocking Locks(9.1) report, choose this field.
Logical I/O	Logical I/O operations
Physical I/O	Physical I/O operations
SID	Session ID. To display the Open Cursors(4.0) report, choose this field.
Tran Secs	Number of seconds since the transaction started
User	Oracle user name

Drilldown reports (field level)

Report name	Description
Blocking Locks(9.1)	Displays a session that includes a lock that places another session in wait status. To display this report, choose the Locks field.
Open Cursors(4.0)	Displays the cursors that are opened by a session. To display this report, choose the SID field.

Memory Usage - Top 10 Sessions(9.1)

Overview

The Memory Usage - Top 10 Sessions(9.1) report displays in real-time the top 10 sessions in terms of memory usage.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

Session Statistics Summary (PD_PDS2)

Fields

Field name	Description
PGA Memory	Displays the PGA size for the sessions. To display a Session Detail(9.1) report, choose this field.
SID	Session ID
UGA Memory	Displays the UGA size for the sessions. To display a Session Detail(9.1) report, choose this field.
User	Oracle user name

Drilldown reports (field level)

Report name	Description
Session Detail(9.1)	Displays detailed information about a session. To display this report, choose the PGA Memory or UGA Memory field.

Open Cursors(4.0)

Overview

The `Open Cursors(4.0)` report displays in real-time a cursor opened by a session. This is a drilldown report.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`

Record

Open Cursor (`PD_PDOC`)

Fields

Field name	Description
Addrhash	Character string that identifies the SQL statement being executed
Program	Name of the program being executed
SID	Session ID
SQL Text	First 60 characters of the SQL statement that was analyzed by the open cursor. To display the `SQL Text(4.0)` report, choose this field.
User	Oracle user name

Drilldown reports (report level)

Report name	Description
`SQL Text(4.0)`	Displays the performance data in the SQL Text and Explain Plan fields. To display this field, choose the SQL Text field.

Physical I/O - Top 10 Sessions(9.1)

Overview

The `Physical I/O - Top 10 Sessions(9.1)` report displays in real-time the top 10 sessions in terms of concentration of I/O operations.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/`

Record

Session I/O Interval (`PI_PIIO`)

Fields

Field name	Description
Physical Reads	Number of physical read operations. To display the `Session Detail(9.1)` report, choose this field.
SID	Session ID
User	Oracle user name

Drilldown reports (report level)

Report name	Description
`I/O Activity - Top 10 Datafiles(9.1)`	Displays the top 10 data files in terms of the number of disk I/O operations.

Drilldown reports (field level)

Report name	Description
`Session Detail(9.1)`	Displays detailed information about a session. To display this report, choose the Physical Reads field.

Redo Log Buffer Contention(8.0)

Overview

The `Redo Log Buffer Contention(8.0)` report displays the number of times a process waited for space to be allocated in the REDO log entry per minute over the past hour.

Storage location

`Reports/Oracle/Troubleshooting/Recent Past/`

Record

System Stat Summary Interval (`PI`)

Fields

Field name	Description
Redo Log Space Requests	Number of times Oracle must wait for disk spaces to be allocated to REDO log entry because the active log file is full.

Server Configuration Status(4.0)

Overview

The `Server Configuration Status(4.0)` report displays real-time information on setup parameters.

Storage location

`Reports/Oracle/Status Reporting/Real-Time/`

Record

Parameter Values (`PD_PDP`)

Fields

Field name	Description
Is Default	Default value. The valid values are `TRUE` and `FALSE`.
Parameter Name	Parameter name. There are two parameters: • Parameter with size restriction that has no effect on performance • Parameter with no size restriction that has an effect on performance
Value	Parameter value

Session Detail(9.1)

Overview

The Session Detail(9.1) report displays in real-time detailed information about a session. This is a drilldown report.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/

Record

Session Detail (PD_PDS)

Fields

Field name	Description
Addrhash	Character string that identifies the SQL statement being executed
Blocking Locks	Number of locks that are blocking other locks
Command	Command being executed. To display the SQL Text(4.0) report, choose this field.
Container ID	The ID of the container to which this session belongs. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.
Locks Held	Number of locks held during data collection
Open Cursors	Number of open cursors. To display the Open Cursors(4.0) report, choose this field.
Program	Name of the program being executed
Session Events	Number of events the session is waiting for
Session Waits	Number of resources and events the session is waiting for
Sessions Blocked	Number of sessions that have been placed in wait status by this session
SID	Session ID. To display the Session Statistics Detail(4.0) report, choose this field.
Table Accesses	Number of table accesses
Transactions	Number of active transactions
User	Oracle user name

Drilldown reports (field level)

Report name	Description
Open Cursors(4.0)	Displays detailed information about the open cursors in the session. To display this report, choose the Open Cursors field.
Session Statistics Detail(4.0)	Displays detailed statistical information about a session. To display this report, choose the SID field.
SQL Text(4.0)	Displays the performance data in the SQL Text and Explain Plan fields. To display this report, choose the Command field.

Session Statistics Detail(4.0)

Overview

The `Session Statistics Detail(4.0)` report displays in real-time statistical information about a session. This is a drilldown report.

Storage location

`Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/`

Record

Session Statistics Summary (`PD_PDS2`)

Fields

Field name	Description
Blocking Locks	Number of locks in a session that are blocking other locks
Cache Hit %	Buffer cache usage
Disk Sorts	Number of disk sort operations
Lock Requests	Number of lock requests
Lock Waits	Number of times lock request was placed in wait status
Memory Sorts	Number of sort operations in memory
PGA Memory	Session's PGA size
Physical Reads	Number of real read operations on a database block from disk
Program	Program name
SID	Session ID
Sort Overflow %	Percentage of sort operations using temporary segments
UGA Memory	Session's UGA size
User	Oracle user name

SGA Status(9.1)

Overview

The SGA Status(9.1) report displays in real-time the status of a component in SGA.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

SGA Components (PD_PDSG)

Fields

Field name	Description
Bytes	Memory size (unit: bytes)
Component Name	SGA component name
Container ID	The ID of the container to which SGA component belongs. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.

SGA Status Summary(9.1)

Overview

The SGA Status Summary(9.1) report displays an overview of components in SGA per day over the past month.

Storage location

Reports/Oracle/Monthly Trend/Advanced/

Record

SGA Components (PD_PDSG)

Fields

Field name	Description
Bytes	Memory size (unit: bytes)
Component Name	SGA component name
Container ID	The ID of the container to which SGA component belongs. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.
PDB Bytes	Retrieves total size of memory for each Container ID field. The value can be either of the following: • If the Container ID is not 0, total size of memory of all SGA components that are allocated to PDBs or the root container (CDB$ROOT) • If the Container ID is 0, total size of memory of common SGA components of database instances This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.
Total Bytes	Total size of memory used by each SGA component (unit: bytes)

SQL Text(4.0)

Overview

The SQL Text(4.0) report displays in real-time the performance data in the SQL Text and Explain Plan fields. This is a drilldown report.

Note:

Do not use this report alone. This report is displayed in a drilldown from the Open Cursors(4.0) report or Session Detail(8.0) report.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/Drilldown Only/

Record

SQL Text (PD_PDSQ)

Fields

Field name	Description
Explain Plan	Execution plan on the SELECT, UPDATE, INSERT, and DELETE statements selected by the Oracle optimizer.
SQL Text	Part of the SQL text

System Overview(9.1)(real-time report on the overall status of instance)

Overview

The System Overview(9.1) report displays in real-time the main performance data indicating the overall status of an instance.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

System Stat Summary (PD)

Fields

Field name	Description
Cache Hit %	Buffer cache usage. To display the Physical I/O - Top 10 Sessions(9.1) report, choose this field.
Continued Row %	Percentage of the rows that are longer than one block or moved (continued rows or moved rows)
Current Logons	Number of logons to the Oracle Database during data collection
Deadlocks	Number of process deadlocks caused by enqueuing resulting from manipulation of DML
Disk Sorts	Number of disk sort operations. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Lock Requests	Number of lock requests. To display the Lock Usage - Top 10 Sessions(9.1) report, choose this field.
Memory Sorts	Number of memory sort operations. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Session CPU Usage	CPU time used in 1/100 seconds
Session PGA Memory	PGA size used by active sessions during data collection. To display the Memory Usage - Top 10 Sessions(9.1) report, choose this field.
Session UGA Memory	UGA size used by active sessions. To display the Memory Usage - Top 10 Sessions(9.1) report, choose this field.
Sort Overflow %	Percentage of sort operations using temporary segments. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Total SQL Executions	Number of SQL statement executions
User Calls	Number of requests from application to database that have been processed
User Commits	Number of transactions. To display the Longest Transactions - Top 10 Sessions(9.1) report, choose this field.
User Rollbacks	Number of rollbacks

Drilldown reports (report level)

Report name	Description
Database Activity Status(8.0)	Displays instance activity status.

Report name	Description
Database Space Overview(8.0)	Displays usage information about tablespaces and data files for the instance.
Error Log(9.1)	Displays the error messages that have been issued since the agent started.
Server Configuration Status(4.0)	Displays all setup parameter information for the server.
Tablespace Status(4.0)	Displays all tablespace information.

Drilldown reports (field level)

Report name	Description
Disk Sorts - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of the frequency of disk sort operations. To display this report, choose the following fields: • Disk Sorts • Memory Sorts • Sort Overflow %
Lock Usage - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of the number of locks held. To display this report, choose the Lock Requests field.
Longest Transactions - Top 10 Sessions(9.1)	Displays the top 10 transactions in terms of the length of a transaction that placed another session in wait status. To display this report, choose the User Commits field.
Memory Usage - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of memory usage. To display this report, choose the Session PGA Memory or Session UGA Memory field.
Physical I/O - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of concentration of I/O operations. To display this report, choose the Cache Hit % field.

System Overview(9.1)(real-time report on the general status of instance)

Overview

The System Overview(9.1) report displays in real-time the main performance data indicating the general status of an instance.

Storage location

Reports/Oracle/Troubleshooting/Real-Time/

Record

System Stat Summary (PD)

Fields

Field name	Description
Cache Hit %	Buffer cache usage. To display the Physical I/O - Top 10 Sessions(9.1) report, choose this field.
Continued Row %	Percentage of the rows that are longer than one block or moved (continued rows or moved rows)
Current Logons	Number of logons to the Oracle Database during data collection
Deadlocks	Number of process deadlocks caused by enqueuing resulting from manipulation of DML
Disk Sorts	Number of disk sort operations. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Lock Requests	Number of lock requests. To display the Lock Usage - Top 10 Sessions(9.1) report, choose this field.
Memory Sorts	Number of memory sort operations. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Session CPU Usage	CPU time used in 1/100 seconds
Session PGA Memory	PGA size used by active sessions during data collection. To display the Memory Usage - Top 10 Sessions(9.1) report, choose this field.
Session UGA Memory	UGA size used by active sessions. To display the Memory Usage - Top 10 Sessions(9.1) report, choose this field.
Sort Overflow %	Percentage of sort operations using temporary segments. To display the Disk Sorts - Top 10 Sessions(9.1) report, choose this field.
Total SQL Executions	Total number of SQL statement executions
User Calls	Number of requests from application to database that have been processed
User Commits	Number of transactions. To display the Longest Transactions - Top 10 Sessions(9.1) report, choose this field.
User Rollbacks	Number of rollbacks

Drilldown reports (report level)

Report name	Description
Database Activity Status(8.0)	Displays the instance activity status.

Report name	Description
Database Space Overview(8.0)	Displays usage information about tablespaces and data files for the instance.
Error Log(9.1)	Displays the error messages that have been issued since the agent started.
Server Configuration Status(4.0)	Displays all setup parameter information for the server.
Tablespace Status(4.0)	Displays all tablespace information.

Drilldown reports (field level)

Report name	Description
Disk Sorts - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of the frequency of disk sort operations. To display this report, choose the following fields: • Disk Sorts • Memory Sorts • Sort Overflow %
Lock Usage - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of the number of locks held. To display this report, choose the Lock Requests field.
Longest Transaction - Top 10 Sessions(9.1)	Displays the top 10 transactions in terms of the length of a transaction that placed another session in wait status. To display this report, choose the User Commits field.
Memory Usage - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of memory usage. To display this report, choose the Session PGA Memory or Session UGA Memory field.
Physical I/O - Top 10 Sessions(9.1)	Displays the top 10 sessions in terms of concentration of I/O operations. To display this report, choose the Cache Hit % field.

Tablespace Status(4.0)

Overview

The Tablespace Status (4.0) report displays in real-time the status of all tablespaces in the database.

Storage location

Reports/Oracle/Status Reporting/Real-Time/

Record

Tablespace (PD_PDTS)

Fields

Field name	Description
Free %	Percentage of free space
Free Mbytes	Free space in megabytes. To display the Tablespace Status Detail(4.0) report, choose this field.
Tablespace Name	Name of tablespace associated with the instance
Used Mbytes	Used space in megabytes. To display the Tablespace Status Detail(4.0) report, choose this field.

Drilldown reports (field level)

Report name	Description
Tablespace Status Detail(4.0)	Displays detailed information about a specified tablespace. To display this report, choose the Free Mbytes or Used Mbytes field.

Tablespace Status Detail(4.0)

Overview

The `Tablespace Status Detail(4.0)` report displays in real-time details about a specified tablespace. This is a drilldown report.

Storage location

`Reports/Oracle/Status Reporting/Real-Time/Drilldown Only/`

Record

Tablespace (`PD_PDTS`)

Fields

Field name	Description
Data Files	Number of data files used by the tablespace
Extents	Number of extents
Free %	Percentage of free space
Free Extents	Number of available extents
Free Mbytes	Free space in megabytes
Mbytes	Size of the tablespace in megabytes
Segments	Number of segments
Tablespace Name	Tablespace name

6 Records

This chapter describes the records for PFM - Agent for Oracle. For details about collecting performance data for each type of record, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide* or the chapter on management of operation monitoring data in the *JP1/Performance Management User's Guide*.

Data model

Each PFM - Agent records and fields are referred to collectively as a *data model*. There is a specific version number for each PFM - Agent and its data model.

For details about data model versions of PFM - Agent for Oracle, see *J. Version Compatibility*.

To check the data model version of each PFM - Agent for Oracle, use the Agents window in PFM - Web Console to display the agent properties.

For details about data models, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

Format of record explanations

This chapter describes the records for PFM - Agent for Oracle in alphabetical order. The explanation of each record consists of the following subsections:

Function

Provides an overview of the performance data that is stored in the record and includes important information that should be noted.

Default and changeable values

Consists of a table of the default values for the performance data under the collection conditions that are defined for the record, and indicates whether or not the values can be changed by the user. The table below lists and describes the items that appear in the Default and changeable values subsections. For details about each item in the table, see the chapter on management of operation monitoring data in the *JP1/Performance Management User's Guide*.

Table 6–1: Default and changeable value

Item	Default value	Changeable
Collection Interval	Performance data collection interval (in seconds)	Y: Changeable N: Not changeable
Collection Offset#1	Offset value for starting performance data collection (in seconds). For details about offset values, see the chapter on management of operation monitoring data in the *JP1/Performance Management User's Guide*. For collection start time for the performance data, see the chapter on the Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.	
Log	Whether or not collected performance data is stored in the Store database: `Yes`: Store (however, if `Collection Interval=0` is set, collected performance data is not stored). `No`: Do not store.	
LOGIF	Conditions for storing collected performance data in the Store database	
Over 10 Sec Collection Time#2,#3	Whether the collection of records might require 10 seconds or more. `Yes`: Might require 10 seconds or more. `No`: Does not require 10 seconds.	

#1

The range of values is from 0 to 32,767 (inclusive) seconds (within the value range specified for Collection Interval). This is used to distribute the collection processing workload because data collection is concentrated when multiple data items are collected. The data collection time that is recorded is the same as for the Collection Interval regardless of the value of Collection Offset.

If you change the value of Collection Offset, you should take into account the collection processing workload.

#2

This property is displayed if the collection of historical data is prioritized over the display processing of real-time reports. (if the functionality that prioritizes the collection of historical data is enabled) For details, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

#3

The value of this item is an approximation at best. Depending on the environment being monitored, the configuration, the load status, and even the collection of records whose value is `No` might require 10 seconds or more. In such cases, real-time reports might not be displayed.

In addition, depending on the collection interval and offset settings, real-time reports might not be displayed if the collection of historical data takes place successively for multiple records, even when the collection time for each record is short. In this case, check and, if necessary, revise the collection interval, or consider using historical reports.

ODBC key fields

These fields display the primary keys that are necessary to use the data retrieved from records stored in the Store database on either PFM - Manager or PFM - Base. Some ODBC key fields are common to all records, and some are specific to each record. This section presents the ODBC key fields that are specific to each record. Only the multi-instance records have specific ODBC key fields.

For details about the ODBC key fields common to all records, see *List of ODBC key fields* in this chapter.

Lifetime

Indicates the period during which consistency is guaranteed for the performance data that is collected in the record. For details about lifetimes, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.

Record size

Indicates the amount of performance data that can be collected and stored in each record at one time.

Fields

Provides a table that describes the fields of each record. The table contains the following items:

- PFM - View name (PFM - Manager name)
  - PFM - View name
    Indicates the field name that is displayed with PFM - Web Console (PFM - View name).
  - PFM - Manager name
    Field name (PFM - Manager name) to be specified in SQL statements when SQL statements are used from PFM - Manager to access the field data stored in the Store database.
    You specify the record ID at the beginning of an SQL statement. For example, to specify the Disk Sorts (`SORTS_DISK`) field of the System Stat Summary (`PD`) record, specify `PD_SORTS_DISK`.
- Description
  Explanation of the performance data that is stored in the field.
  Notes #1 and #2 in the table indicate the following:

  #1:
  A value in this field is the latest monitored value that the monitoring target returns at the time of data collection.

  #2:
  When this field is displayed in the historical report, the PFM - View name (Total) field is added.

  For each field, the following methods of calculating performance data are available:
  - Calculations (such as averages or percentages) based on data collected in the current and previous intervals.
  - Calculations based on data collected in the current interval only. This data includes the values accumulated internally by the OS (the data marked as #1 in the table).
  - Calculations based on data in other fields (See Data source in the table of each record's fields.)

  Unless otherwise noted, the value in each field is an unprocessed value that was collected at a specified data collection interval.

The following types of values are displayed in a historical report when records of the PI record type are summarized and displayed while the report interval setting is not "minute":

- The average value for the summarized interval
- The last collected value
- The sum total of values
- The minimum value
- The maximum value

Unless otherwise noted, the value in each field displays the average for the summarized interval.

- Summary

  The summarization method (Summary rules) used by Agent Store when summarizing data. For details, see *Summary rules* in this chapter.

- Format

  Data type of the field value, such as `double`. For details about the data types, see *List of data types* in this chapter.

- Delta

  In contrast to the data collected as the cumulative value, the so-called *delta* is the data that indicates the changed amount. For details about delta, see *Field values* in this chapter.

- Supported version

  Indicates the Oracle version that can use the field.

  If a version number is shown, the field is supported by that version and all subsequent versions. *All* means that all Oracle versions support the field. *Not supported* means that the field is not supported by Oracle.

- Data source

  Method used to obtain the field value or the source of the data. For details about field values, see *Field values* in this chapter.

List of ODBC key fields

Some ODBC key fields are common to all records, and some are specific to each record. This section presents the ODBC key fields common to all records. The ODBC key fields are necessary to use the data retrieved from records stored in the Store database on PFM - Manager.

The table below lists the ODBC key fields common to all records. For details about the ODBC key fields specific to each record, see the details of each record.

Table 6–2: List of ODBC key fields common to all records

ODBC key field	ODBC format	Data	Description
*record-ID*_DATE	SQL_INTEGER	Internal	Key in the record that indicates the record creation date
*record-ID*_DATETIME	SQL_INTEGER	Internal	Combination of the *record-ID*_DATE and *record-ID*_TIME fields
*record-ID*_DEVICEID	SQL_VARCHAR	Internal	*instance-name*[*host-name*]
*record-ID*_DRAWER_TYPE	SQL_VARCHAR	Internal	Type. Valid values are as follows: m: Minute H: Hour D: Day W: Week M: Month Y: Year
*record-ID*_PROD_INST	SQL_VARCHAR	Internal	Instance name of PFM - Agent for Oracle
*record-ID*_PRODID	SQL_VARCHAR	Internal	Product ID of PFM - Agent for Oracle
*record-ID*_RECORD_TYPE	SQL_VARCHAR	Internal	Identifier indicating the record type (4 bytes)
*record-ID*_TIME	SQL_INTEGER	Internal	Record creation time (Greenwich mean time (GMT))

Summary 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 in the summarization period must 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.

The fields referred to in the record descriptions in this chapter are known as *record-specific fields* to distinguish them from additional fields generated when data is summarized.

Additional fields have the following field names:

- Additional field contained in the Store database
  PFM - Manager name of the record-specific field, plus a suffix
- Additional field displayed in PFM - Web Console
  PFM - View name of the record-specific field, plus a suffix

The following table shows the suffix added to the PFM - Manager name, the suffix added to the corresponding PFM - View name, and the data stored in that field.

Table 6–3: List of suffixes in additional field names

Suffix added to the PFM - Manager name	Suffix added to the PFM - View name	Field data
_TOTAL	(Total)	Sum of the field values in all records in the summary period
_TOTAL_SEC	(Total)	Sum of the values of the field in the records collected within the summarization period (when the data type of the field is utime)
_COUNT	--	Number of records collected in the summary period
_HI	(Max)	Highest field value in the records in the summary period
_LO	(Min)	Lowest field value in the records in the summary period

Legend:
--: No additional field.

The table below lists the summary rules.

Table 6–4: List of summary rules

Summary rule name	Summary rules
COPY	Stores the actual field value of the most recent record in the summary period.
AVG	Stores the average field value of all field values in the summary period. The average value is calculated using the following expression: (*total-sum-of-the-field-values*)/(*number-of-collected-records*)

Summary rule name	Summary rules
AVG	Additional field (Store database) • _TOTAL • _TOTAL_SEC (when the data type of the field is utime) • _COUNT Additional field (PFM - Web Console) • (Total)
HILO	Stores the highest value, lowest value, and average value of all field values in the summary period. A record-specific field stores the average value. The highest value, lowest value, and average value is calculated using the following expression: (*total-sum-of-the-field-values*)/(*number-of-collected-records*) Additional field (Store database) • _HI • _LO • _TOTAL • _TOTAL_SEC (when the data type of the field is utime) • _COUNT Additional field (PFM - Web Console) #1,#2 • (Max) • (Min) • (Total)
--	No summarization

#1

For utime type fields whose Manager names contain the character string _AVG, the (Total) fields that are added in Performance Reporter cannot be used for historical reports.

#2

For fields whose Manager names contain any of the following character strings, the (Total) fields that are added in Performance Reporter cannot be used for historical reports:

PER, PCT, PERCENT, _AVG, _RATE_TOTAL

List of data types

Table 6-5 lists the data types for field values and the corresponding C and C++ data types. The values shown in the *Format* column of the record field tables are those shown below in the *Field* column under *Data type*.

Table 6–5: List of data types

Data type		Byte	Description
Field	C and C++		
`char`(*n*)	`char`( )	Number in parentheses	Character data of *n* bytes.
`Double`	`double`	8	Numeric value (1.7E±308 (15 digits))
`Long`	`long`	4	Numeric value (-2,147,483,648 to 2,147,483,647)
`Short`	`short`	2	Numeric value (-32,768 to 32,767)
`string`(*n*)	`char`[ ]	Number in parentheses	Character string of *n* bytes. (Only 7-bit ASCII characters can be set.) The last character is null.
`time_t`	`unsigned long`	4	Numeric value (0 to 4,294,967,295)
`Timeval`	Structure	8	Numeric value (first 4 bytes are seconds, next 4 bytes are microseconds)
`Ulong`	`unsigned long`	4	Numeric value (0 to 4,294,967,295)
`Ushort`	`unsigned short`	2	Numeric value (0 to 65,535)
`Utime`	Structure	8	Numeric value (first 4 bytes are seconds, next 4 bytes are microseconds)
`Word`	`unsigned short`	2	Numeric value (0 to 65,535)
(Not applicable)	`unsigned char`	1	Numeric value (0 to 255)

Field values

This section describes the values that are stored in the fields.

Data source

Each field contains a value obtained from a Performance Management product or program being monitored or the value derived there from by means of applying a formula. In the tables, the *Data source* column indicates the source of the value or the formula used to produce the value.

When a field's value is obtained by processing performance data acquired from Oracle, the character string in the *Data source* column indicates the method used to obtain the value that is set in the field. The following shows examples:

- When uppercase letters are shown:
  Uppercase letters indicate the table name of the Oracle Database when an Oracle Database is accessed. For example, the Cursor Open Hits (`CURSOR_OPEN_HITS`) field of the Activity Summary (`PD_PDAS`) record stores the value that is collected using OCI of Oracle. For details, see your Oracle documentation.
- When lowercase letters are shown:
  Lowercase letters indicate the key character string used to acquire the performance data that is stored in the Oracle Database table.
  For example, the Calls/Tran (`CALLS_PER_TRANSACTION`) field of the System Stat Summary (`PD`) record stores the value that is obtained by dividing the value obtained from `user calls` in the `Name` column of the `V$SYSSTAT` table by the value obtained from `user commits` in the Name column of the `V$SYSSTAT` table.
- When *Agent Collector* is shown:
  Agent Collector means that the value stored in the field was obtained from the *Agent Collector* service.
- When *init.ora parameter name* is shown:
  init.ora parameter name means that the value of the parameter name that is set in the `init.ora` Oracle initialization parameter file is used.
- When two dashes (--) are shown:
  Two dashes (--) means that the field's value is obtained without processing performance data.

Delta

In contrast to the data collected as the cumulative value, the so-called *delta* is the data that indicates the changed amount. For example, if the performance data value obtained during the first collection is 3 and the performance data value obtained during the second collection is 4, then the cumulative value is 7 and the changed amount is 1. In the tables, the Delta column indicates whether or not each field's value is a delta value. Note that since delta values are relative to previous data, they may be negative.

The following table explains the delta characteristics of performance data collected by PFM - Agent for Oracle:

Table 6–6: Performance data collected by PFM - Agent for Oracle

Record type	Delta	Data type	Indicate delta value#	Record value
PI record type	Yes	Real-time data	Selected	The displayed value is the changed amount.
			Not selected	The displayed value is the changed amount.
		- Historical data - Alarm monitoring data	N/A	The displayed value is the changed amount.

Record type	Delta	Data type	Indicate delta value#	Record value
PI record type	No	Real-time data	Selected	The displayed value was the actual value at the time of data collection.
			Not selected	The displayed value was the actual value at the time of data collection.
		- Historical data - Alarm monitoring data	N/A	The displayed value was the actual value at the time of data collection.
PD record type	Yes	Real-time data	Selected	The displayed value is the change.
			Not selected	The displayed value is the cumulative value.
		- Historical data - Alarm monitoring data	N/A	The displayed value is the cumulative value.
	No	Real-time data	Selected	The displayed value was the actual value at the time of data collection.
			Not selected	The displayed value was the actual value at the time of data collection.
		- Historical data - Alarm monitoring data	N/A	The displayed value was the actual value at the time of data collection.

Legend:

N/A: Not applicable

#

Indicates that the following check boxes are selected in the PFM - Web Console dialog box:

- **Indicate delta value** check box in the Report Wizard - Indication settings (Realtime) dialog box
- **Indicate delta value** check box in **Indication settings (Realtime)** on the **Properties** page of the Report window

The following points should be noted about collection of performance data:

- In order for a record of the PI record type to be saved, the performance data must be collected at least twice.

  For a record of the PI record type, performance data is collected at the interval set by PFM - Web Console. However, the performance data is not stored in the Store database when its collection is set by PFM - Web Console.

  Historical data for records of the PI record type requires two data collections in order to obtain the value (called delta) resulting from calculating the difference between two sets of data. Therefore, it takes up to twice as much time as the specified value before the historical data is stored in the Store database.

  For example, if PFM - Web Console sets the performance data collection interval at 300 seconds (5 minutes) at 18:32, the first data collection will begin at 18:35. The next data collection will begin at 18:40. Historical data is created from the data collected at 18:35 and 18:40, and is stored in the Store database as historical data at 18:40 (8 minutes after the time you entered the settings).
- In a real-time report, the value that is displayed is since the time the first data was collected. However, in reports requiring comparison with previous data, the initial values are shown as zero. Starting with the second cycle, the collection behavior depends on the type of report.
- The value of the collected data is displayed after the second data collection in the following cases:

    - The **Indicate delta value** check box is selected in the settings for real-time reports based on PI records.
    - The **Indicate delta value** check box is selected in the settings for real-time reports based on PD records.
- If the following setting is made, the difference between the first and second data collections is displayed at the second data collection; for the third and subsequent data collections, the value of the collected data is displayed when:
    - The **Indicate delta value** check box is selected in the settings for real-time reports based on PI records.
- While PFM - Agent for Oracle is running, if a monitored instance of Oracle is restarted, or a resource is reallocated while Oracle is running, the value of collected data may be negative. Note that for data for the second and subsequent collections, positive values are used as the data differential.
- When a record of the PI record type is displayed in a real-time report, the first time collection data of the field where the delta field is Yes is invalid.

Fields added only when data is stored in the Store database

The following table lists the fields that are added only when data is stored in the Store database:

Table 6–7: Fields added only when data is stored in the Store database

PFM - View name (PFM - Manager name)	Description	Format	Delta	Supported versions	Data source
Agent Host (DEVICEID)	Name of host where PFM - Agent for Oracle is running	string (256)	No	All	N/A
Agent Instance (PROD_INST)	Instance name of PFM - Agent for Oracle	string (256)	No	All	N/A
Agent Type (PROD_ID)	Product ID of PFM - Agent for Oracle (1-byte identifier)	char	No	All	N/A
Date (DATE)	Record creation date in (GMT)#1#2	char (3)	No	All	N/A
Date and Time (DATETIME)	Combination of the Date (DATE) and Time (TIME) fields#2	char (6)	No	All	N/A
Drawer Type (DRAWER_TYPE)	For a PI record, the data summarization type.	char	No	All	N/A
GMT Offset (GMT_ADJUST)	Difference (in seconds) between Greenwich Mean Time and local time	long	No	All	N/A
Time (TIME)	Record creation time (GMT)#1#2	char (3)	No	All	N/A

Legend:

N/A: The field value is specified without processing the performance data acquired from Oracle.

#1

A basic value is set because records of the PI record type are summarized. The following table shows the setting values for each record type.

Table 6–8: Setting value for each record type

Type	Setting value for each record type
Minute	0 second of the time when the record was created.
Hour	0 minute and 0 second of the time when the record was created.
Day	0:00 and 0 second of the day when the record was created.
Week	0:00 and 0 second on Monday of the week when the record was created.
Month	0:00 and 0 second on the 1st day of the month when the record was created.
Year	0:00 and 0 second on January 1st of the year when the record was created.

#2

When data is displayed in reports, the Date field is displayed in the format *YYYYMMDD*, the Date and Time field is displayed in the format *YYYYMMDD hh*:*mm*:*ss*, and the Time field is displayed in the format *hh*:*mm*:*ss*.

Fields 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. These fields are also added when data is stored in the Store database, but since they are used internally by PFM - Agent for Oracle, they are not used as fields displayed in reports, and should not be used during operation.

- *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*_COUNT
- *Record ID_PFM - Manager name*_SEC
- *Record ID_PFM - Manager name*_MSEC

Notes on records

Note the following when collecting records.

Results of record generation when no data can be acquired

The following explains the results of record generation when no data can be acquired.

- No records are generated

  In the following cases, no records are created.

  - PFM - Agent for Oracle cannot collect the performance data to be stored in a field that is defined as an ODBC key field.
  - PFM - Agent for Oracle cannot collect the performance data to be stored in a field that shows Oracle performance.

Notes on CPU usage

Collecting the following records can lead to increased CPU usage or a higher CPU I/O Wait value at the Oracle side:

- Applicable records
  - Cache Summary (`PD_PDCS`) record
  - Cache Summary Interval (`PI_PICS`) record
  - Shared Cursor Cache (`PD_PDC`) record
  - SQL Text - Performance Based (`PD_PDES`) record

  Caution is required for the above records when one of the following functions is to be used.
  - Collection of the above records

    To prevent this phenomenon, avoid collecting these records.
  - Display of the real-time report that uses the above records as the data sources

    To prevent this phenomenon, avoid displaying the real-time report that uses these records as the data sources.
  - Use of the alarm that uses the above records as the data sources

    To prevent this phenomenon, avoid using the alarm that uses these records as the data sources.

Notes on the Oracle version

- When Oracle 10g is being monitored, the Addr field value of both the `PD_PDLA` record and the `PI_PILA` record becomes invalid.

List of records for PFM - Agent for Oracle

This section lists the records that can be collected by PFM - Agent for Oracle.

Table 6-9 lists the records that can be collected by PFM - Agent for Oracle and the information that is stored in each record, in the order of record names and record IDs.

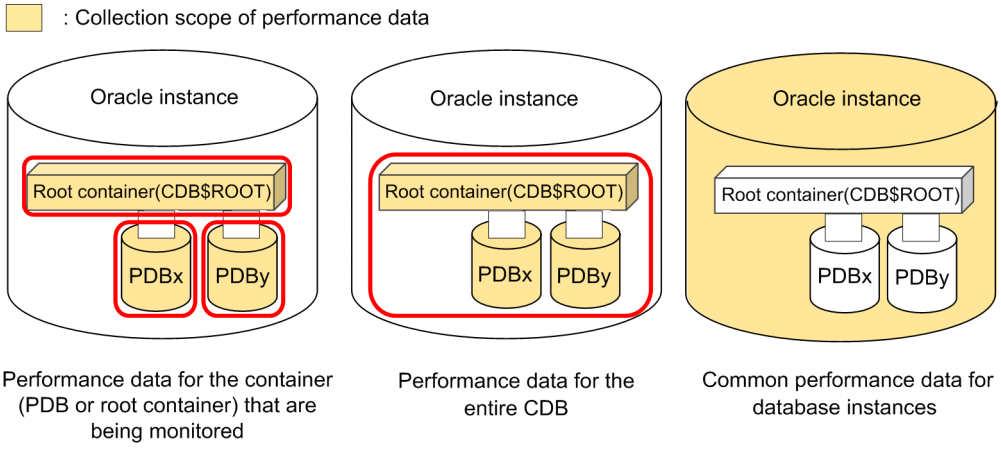
Note that if the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some records or fields collect the following performance data:

- Performance data for the PDBs and root container (`CDB$ROOT`) that are being monitored
- Performance data for all PDBs and common performance data for database instances

When monitoring a CDB configuration, the collection range of performance data might be different depending on the containers to be monitored and the fields to be acquired.

The following figures show differences in collection scope of performance data.

Figure 6–1: Differences in collection scope of performance data



Also, depending on containers and fields to be monitored, there are fields with precautions and fields that cannot collect effective performance data.

For details, see *Table 6-9* or the descriptions of each record field.

Table 6–9: List of records for PFM - Agent for Oracle (record names)

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Activity Summary	*PD_PDAS*	Performance data indicating the operating status (at a specific point in time) of the Oracle system.	Y	Y#3	N
ASM Disk	*PD_PDDK*	Performance data indicating the status (at a specific point in time) of the ASM disk managed by the ASM instance through which Oracle communicates.	Y	Y#4	Y#4

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
ASM Disk Group Interval	*PI_PIDG*	Performance data, taken at specific intervals, about the ASM disk group managed by the ASM instance through which Oracle communicates.	Y	Y#4	Y#4
Backup Async IO	*PD_PDBA*	Performance data indicating the status, at a specific point in time, of a file that was backed up or restored with asynchronous I/O operations and that is now being or has been executed by the Oracle recovery manager (RMAN).	Y	N	N
Backup Sync IO	*PD_PDBS*	Performance data indicating the status, at a specific point in time, of a file that was backed up or restored with synchronous I/O operations and that is now being or has been executed by the Oracle recovery manager (RMAN).	Y	N	N
Block Contention Interval	*PI_PIBC*	Performance data, taken at specific intervals, about each wait class.	Y	N	N
Block Contention Statistics	*PD_PDBC*	Performance data indicating the status of each wait class at a specific point in time.w	Y	N	N
Buffer Pool	*PD_PDBP*	Performance data indicating the status of the buffer pool available to instances at a specific point in time.	Y	N	N
Cache Summary	*PD_PDCS*	Performance data indicating the status of each cache in the system global area (SGA) at a specific point in time.	Y	Y#3	N
Cache Summary Interval	*PI_PICS*	Performance data, taken at specific intervals, about each cache in the system global area (SGA).	Y	Y#3	N
CDB Status	*PD_CDCS*	Performance data related to a PDB built inside a CDB, including open mode information and the status concerning the amount of disk space used.	N	Y#5	Y#4
CDB Tablespace	*PD_CDTS*	Performance data indicating the status of tablespaces present in both the root container and PDBs of a CDB at a certain point in time.	Y	Y#5	Y#4

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Circuit	*PD_PDCI*	Performance data indicating the status of circuits at a specific point in time.	Y	N	N
Collection Instance 2	*PD_PCI*	Performance data indicating the status (at a specific point in time) of an instance and the fast recovery area.	Y	Y#4	Y#4
Collection Tablespace 2	*PD_PCTS*	Performance data indicating the status (at a specific point in time) of tablespaces in a database.	Y	N	N
Control File	*PD_PDCF*	Performance data indicating the status of a control file at a specific point in time.	Y	N	N
Current Sessions Stat Summary	*PD_PDS3*	Performance data indicating the status of all active sessions for an instance at a specific point in time.	Y	Y#3	N
Data Dictionary Cache	*PD_PDDD*	Performance data indicating the usage of the data dictionary cache at a specific point in time.	Y	N	N
Data Dictionary Cache Interval	*PI_PIDD*	Performance data, taken at specific intervals, about the usage of the data dictionary cache.	Y	N	N
Data File	*PD_PDDF*	Performance data indicating the status (at a specific point in time) of data files.	Y	Y#5	N
Data File Interval	*PI_PIDF*	Performance data, taken at specific intervals, about data files.	Y	Y#5	N
Database	*PD_PDDB*	Performance data, taken at a specific point in time, indicating the following: • General information about a database • Statistical information on tablespaces • Statistical information on data files	Y	Y#3	N
Database Interval	*PI_PIDB*	Performance data, taken at specific intervals, about a database.	Y	Y#3	N
Database Object Cache	*PD_PDDO*	Performance data indicating the status of database objects in the library cache at a specific point in time.	Y	N	N
Dispatcher	*PD_PDDS*	Performance data indicating the status of dispatcher processes at a specific point in time.	Y	N	N

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Dispatcher Interval	*PI_PIDS*	Performance data, taken at specific intervals, about dispatcher processes.	Y	N	N
Errorlog Detail	*PD_PDEL*	Performance data indicating the status of error messages in the database alert file at a specific point in time.	Y	Y#5	N
GCS Stat Summary	*PD_PDGC*	Performance data indicating the status of the Global. Cache Service (GCS) at a specific point in time.	Y	Y#5	N
GCS Stat Summary Interval	*PI_PIGC*	Performance data, taken at specific intervals, about the Global Cache Service (GCS).	Y	Y#5	N
Instance	*PD_PDI*	Performance data indicating the status (at a specific point in time) of an instance.	Y	N	N
Instance Availability	*PD_PDIA*	Performance data indicating the status (at a specific point in time) of instance availability.	Y	Y#5	Y#6
Latch	*PD_PDLA*	Performance data indicating the status of latches at a specific point in time.	Y	Y#3	N
Latch Interval	*PI_PILA*	Performance data, taken at specific intervals, about latches.	Y	Y#3	N
Library Cache	*PD_PDLC*	Performance data indicating the status of library cache management at a specific point in time.	Y	N	N
Library Cache Interval	*PI_PILC*	Performance data, taken at specific intervals, about library cache management.	Y	N	N
Lock	*PD_PDLO*	Performance data indicating the status of locks at a specific point in time.	Y	N	N
Lock Activity Interval	*PI_PIPL*	Performance data, taken at specific intervals, about PCM lock conversions that have occurred.	N	N	N
Lock Interval	*PI_PILO*	Performance data, taken at specific intervals, about locks.	Y	N	N
Lock Waiters	*PD_PDLW*	Performance data indicating the status (at a specific point in time) of all sessions waiting for lock and all sessions holding lock.	Y	Y#5	N

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Minimum Database Interval 2	*PI_PMDB*	Performance data, taken at specific intervals, about a database.	Y	N	N
Minimum Data File Interval 2	*PI_PMDF*	Performance data, taken at specific intervals, about data files.	Y	N	N
Minimum Tablespace Interval 2	*PI_PMTS*	Performance data, taken at specific intervals, about tablespaces in a database.	Y	N	N
Multi - Threaded Server	*PD_PDMT*	Performance data indicating the status of the multithread server (MTS) options at a specific point in time.	Y	N	N
Multi - Threaded Server Interval	*PI_PIMT*	Performance data, taken at specific intervals, about the multi-thread server (MTS) options.	Y	N	N
Open Cursor	*PD_PDOC*	Performance data indicating the status (at a specific point in time) of cursors.	Y	Y#5	N
Options Installed	*PD_PDO*	Performance data indicating the status of the software options installed on the Oracle Server at a specific point in time.	Y	N	N
Parallel Query Server	*PD_PDPQ*	Performance data indicating the status of parallel query servers at a specific point in time.	Y	N	N
Parallel Query Server Interval	*PI_PIPQ*	Performance data, taken at specific intervals, about parallel query servers.	Y	N	N
Parallel Query Statistics	*PD_PDPS*	Performance data indicating the status of parallel query options at a specific point in time.	Y	N	N
Parameter Values	*PD_PDP*	Performance data indicating the status (at a specific point in time) of current parameter values.	Y	N	N
Process Detail	*PD_PDOP*	Performance data indicating the status of processes at a specific point in time.	Y	N	N
Queue Statistics	*PD_PDQU*	Performance data indicating the status of queues at a specific point in time.	Y	N	N
Resource Limit	*PD_PDRL*	Performance data indicating the usage of global system resources at a specific point in time.	Y	N	N
Rollback Segment	*PD_PDRS*	Performance data indicating the status of each rollback segment in a database at a specific point in time.	Y	N	N

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Rollback Segment Interval	*PI_PIRS*	Performance data, taken at specific intervals, about each rollback segment in a database.	Y	N	N
Segment Detail	*PD_PDSM*	Performance data indicating the status of database segments at a specific point in time.	Y	N	N
Server Status	*PD_STAT*	Performance data indicating the status of the Oracle server at a specific point in time.	Y	N	N
Session Detail	*PD_PDS*	Performance data indicating the status (at a specific point in time) of sessions.	Y	Y#3	N
Session Event	*PD_PDEV*	Performance data indicating the status of session events at a specific point in time.	Y	N	N
Session Event Interval	*PI_PIEV*	Performance data, taken at specific intervals, about session events.	Y	N	N
Session I/O Interval	*PI_PIIO*	Performance data, taken at specific intervals, about input/output of all active sessions.	Y	Y#3	N
Session Stat Summary Interval	*PI_PIS2*	Performance data, taken at specific intervals, about each session and performance indicator of an instance.	Y	Y#3	N
Session Statistics	*PD_PDSS*	Performance data indicating the status of sessions at a specific point in time.	Y	N	N
Session Statistics Summary	*PD_PDS2*	Performance data indicating the status (at a specific point in time) of each session and performance indicator of an instance.	Y	Y#3	N
Session Wait	*PD_PDWA*	Performance data indicating the status of the session waits at a specific point in time.	Y	N	N
SGA Components	*PD_PDSG*	Performance data indicating the status (at a specific point in time) of the system global area (SGA).	Y	Y#3	Y#7
Shared Cursor Cache	*PD_PDC*	Performance data indicating the status of the shared cursor cache at a specific point in time.	Y	N	N
Shared Server	*PD_PDSH*	Performance data indicating the status of the shared servers at a specific point in time.	Y	N	N
Shared Server Interval	*PI_PISH*	Performance data, taken at specific intervals, about the shared servers.	Y	N	N

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Sort Segment	*PD_PDSR*	Performance data indicating the status of sort segments in a database at a specific point in time.	Y	N	N
Sort Segment Interval	*PI_PISR*	Performance data, taken at specific intervals, about sort segments in a database.	Y	N	N
SQL Text	*PD_PDSQ*	Performance data indicating the status (at a specific point in time) of the SQL text for a cursor in the shared cursor cache.	Y	N	N
SQL Text - Performance Based	*PD_PDES*	Performance data indicating the status of the SQL text for an SQL statement that has resource requirements that exceed the specified maximum value at a specific point in time.	Y	N	N
*SQL*Net Listener*	*PD_PDNL*	Performance data indicating the status of the status and overview of an SQL*Net Listener at a specific point in time.	Y	N	N
*SQL*Net Listeners*	*PD_PDLS*	Performance data indicating the status and overview of each defined listener at a specific point in time.	Y	Y#5	N
System Event	*PD_PDSE*	Performance data indicating the status of each queued event in an instance at the system level at a specific point in time.	Y	N	N
System Event Interval	*PI_PISE*	Performance data, taken at specific intervals, about each queued event in an instance at the system level.	Y	N	N
System Stat Interval	*PI_PIST*	Metric information about sessions.	Y	N	N
System Stat Summary	*PD*	Performance data indicating the status (at a specific point in time) of key performance indicators after the start of an instance.	Y	Y#3	Y#4
System Stat Summary Interval	*PI*	Performance data, taken at specific intervals since the start of an instance, about key performance indicators.	Y	Y#3	Y#4
System Statistics	*PD_PDST*	Performance data indicating the status of all sessions in the system at a specific point in time.	Y	N	N
Table Access	*PD_PDTA*	Performance data indicating the status of a table accessed by	Y	N	N

Record name	Record ID	Information stored in record	Supported for monitoring		
			Non-CDB configuration #1	CDB configuration #2	
				PDB	Root container
Table Access	*PD_PDTA*	sessions during data collection at a specific point in time.	Y	N	N
Tablespace	*PD_PDTS*	Performance data indicating the status (at a specific point in time) of tablespaces in a database.	Y	Y#5	Y#6
Tablespace Fragmentation	*PD_PDTF*	Performance data indicating the status (at a specific point in time) of fragmentation of tablespaces in a database.	Y	Y#5	N
Tablespace Interval	*PI_PITS*	Performance data, taken at specific intervals, about tablespaces in a database.	Y	Y#5	N
Transaction	*PD_PDTR*	Performance data indicating the status (at a specific point in time) of transactions.	Y	Y#5	N
Transaction Interval	*PI_PITR*	Performance data, taken at specific intervals, about transactions.	Y	Y#5	N
Transaction Lock	*PD_PDTL*	Performance data indicating the status (at a specific point in time) of transaction locks.	Y	Y#5	N
Version	*PD_PDV*	Performance data, taken at a specific point in time, indicating the version number of a core component on the Oracle server.	Y	N	N
Database Link	PD_PDDL	This record is reserved and unavailable.	N	N	N
Ping Activity Interval	PI_PIPP		N	N	N
SQL*Net Handler	PD_PDNH		N	N	N

Legend:

Y: Supported

N: Not supported

#1

Applicable to Oracle Database 12c Release 1 or earlier, or Oracle Database 12c Release 2 in a non-CDB configuration.

#2

The Oracle CDB configuration is supported as of Oracle Database 12c Release 2 or later. It is not supported for Oracle Database 12c Release 1.

#3

Performance data for the PDBs that are being monitored and common performance data for database instances are collected.

#4

Performance data for all PDBs and the root container (CDB$ROOT), and common performance data for database instances are collected.

#5

Performance data for the PDBs that are being monitored is collected.

#6

Performance data for the root container (`CDB$ROOT`) that is being monitored is collected.

#7

The root container (`CDB$ROOT`) that is being monitored is collected following performance data:

- Use administrative accounts (`sys` or `system`)

  Performance data for all PDBs and the root container (`CDB$ROOT`), and common performance data for database instances are collected.

- Use common user

  Performance data for the root container (`CDB$ROOT`), and common performance data for database instances are collected.

This is difference of setting for collecting PDB's performance data. For details, see your Oracle documentation.

Activity Summary (PD_PDAS)

Function

The Activity Summary (PD_PDAS) record stores performance data indicating the operating status (at a specific point in time) of the Oracle system.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. For details, see the descriptions of each field.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 868 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Active Transactions (ACTIVE_TRANSACTIONS)	Number of active transactions in active sessions[#8]	--	long	No	All	SUM(V$ROLLSTAT.XACTS)
Avg Wait (AVERAGE_WAIT)	Average wait time for all events in all sessions. In centiseconds (1/100 of a second).[#7]	--	ulong	No	All	AVG(V$SESSION_EVENT.AVERAGE_WAIT)
Avg Wait String (AVERAGE_WAIT_STRING)	Average wait time for all events in all sessions (character string). In seconds.[#7]	--	string(30)	No	All	AVG(V$SESSION_EVENT.AVERAGE_WAIT) / 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cursor Open Hit % (CURSOR_OPEN_HIT_PERCENTAGE)	Percentage ratio of open cursors that were found during cursor search[#9]	--	double	No	All	V$SYSTEM_CURSOR_CACHE.HIT_RATIO * 100
Cursor Open Hits (CURSOR_OPEN_HITS)	Total cursor open hits[#9]	--	ulong	No	All	V$SYSTEM_CURSOR_CACHE.HITS
Cursor Opens (CURSOR_OPENS)	Total cursor opens[#9]	--	ulong	No	All	V$SYSTEM_CURSOR_CACHE.OPENS
DML Locks % (PERCENT_DML_LOCKS)	Percentage ratio of DML locks to the DML_LOCKS parameter value in the init.ora file[#7]	--	double	No	All	(COUNT(DBA_DML_LOCKS) / init.ora DML_LOCKS) * 100
DML Locks Held (DML_LOCKS_HELD)	Number of current DML locks[#7]	--	long	No	All	COUNT(DBA_DML_LOCKS)
Enqueue Resources % (PERCENT_ENQUEUE_RESOURCES)	Percentage ratio of locks to the value of the ENQUEUE_RESOURCES parameter in the init.ora file Always 0 if the monitoring target is Oracle 10g Release2 or later.	--	double	No	All	(COUNT(V$LOCK) where V$LOCK.LMODE is NOT NULL / init.ora ENQUEUE_RESOURCES) * 100
Locks Held (LOCKS_HELD)	Number of current locks[#7]	--	long	No	All	COUNT(V$LOCK) where V$LOCK.LMODE is NOT NULL
Open Cursors (OPEN_CURSORS)	Number of current open cursors[#8]	--	long	No	All	COUNT(V$OPEN_CURSOR)
Open Cursors % (PERCENT_OPEN_CURSORS)	Percentage ratio of open cursors to the value of the OPEN_CURSORS parameter in the init.ora file[#8]	--	double	No	All	(COUNT(V$OPEN_CURSOR) / init.ora OPEN_CURSORS) * 100
Processes (PROCESSES)	Number of current Oracle processes[#8]	--	ulong	No	All	COUNT(V$PROCESS)
Processes % (PERCENT_PROCESSES)	Percentage ratio of processes to the value of the PROCESSES parameter in the init.ora file[#8]	--	double	No	All	(COUNT(V$PROCESS) / init.ora PROCESSES) * 100
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDAS)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Session Events (SESSION_EVENTS)	Number of events queued by sessions[#7]	--	ulong	No	All	COUNT(V$SESSION_EVENT)
Session Waits (SESSION_WAITS)	Number of session waits[#7]	--	ulong	No	All	COUNT(V$SESSION_WAIT)
Sessions (SESSIONS)	Number of current sessions[#7]	--	ulong	No	All	COUNT(V$SESSION)
Sessions % (PERCENT_SESSIONS)	Percentage ratio of sessions to the value of the SESSIONS parameter in the init.ora file[#7]	--	double	No	All	(COUNT(V$SESSION) / init.ora SESSIONS) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
System Sessions (SESSIONS_SYSTEM)	Number of current system sessions[#7]	--	ulong	No	All	COUNT(V$SESSION) where V$SESSION.TYPE<>'USER'
Table Accesses (TABLE_ACCESSES)	Number of current table accesses[#7]	--	long	No	All	COUNT(V$ACCESS)
Time Waited (TIME_WAITED)	Total length of time that all sessions queued all events. In centiseconds (1/100 of a second).[#7]	--	double	No	All	SUM(V$SESSION_EVENT.TIME_WAITED)
Time Waited String (TIME_WAITED_STRING)	Total length of time that all sessions queued all events (character string). In seconds.[#7]	--	string(30)	No	All	SUM(V$SESSION_EVENT.TIME_WAITED) / 100
Total Timeouts (TOTAL_TIMEOUTS)	Total number of timeouts for all events in all sessions[#7]	--	ulong	No	All	SUM(V$SESSION_EVENT.TOTAL_TIMEOUTS)
Total Waits (TOTAL_WAITS)	Number of waits for all events in all sessions[#7]	--	ulong	No	All	SUM(V$SESSION_EVENT.TOTAL_WAITS)
Transactions % (PERCENT_TRANSACTIONS)	Percentage ratio of the number of transactions to the value of the TRANSACTIONS parameter in the init.ora file[#8]	--	double	No	All	(SUM(V$ROLLSTAT.XACTS) / init.ora TRANSACTION) * 100
User Sessions (SESSIONS_USER)	Number of current user sessions[#7]	--	ulong	No	All	COUNT(V$SESSION) where V$SESSION.TYPE = 'USER'

#7

Displays performance data for the PDBs that are being monitored and common performance data for database instances.

#8

Displays performance data for the PDBs that are being monitored.

#9

Displays performance data for the entire CDB.

ASM Disk (PD_PDDK)

Function

The ASM Disk (PD_PDDK) record stores performance data indicating the status (at a specific point in time) of the ASM disk, which is managed by the ASM instance through which Oracle communicates. This is a multi-instance record.

Notes

- The collection of ASM Disk (PD_PDDK) records is supported for ASM and Oracle Database if the versions of both are 11.2.0 or later. Collection is not supported in version 11.1.0 or earlier.
- Group Number becomes 0 and Disk Group Name is blank if no ASM disk group is mounted.
- If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record stores performance data for all ASM disks in the entire CDB, which are managed by the ASM instances through which Oracle communicates. The retrieved values will be same for both when monitoring PDBs and when monitoring the root container (CDB$ROOT).

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	260	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDDK_DISK_NUMBER
- PD_PDDK_GROUP_NUMBER

Lifetime

From creation to deletion of the ASM disk

Record size

- Fixed part: 678 bytes
- Variable part: 773 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cold Mbytes Read (COLD_MBYTES_READ)	Size of data (in megabytes) read from the cold region	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.COLD_BYTES_READ / (1024 * 1024)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cold Mbytes Written (COLD_MBYTES_WRITTEN)	Size of data (in megabytes) written to the cold region	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.COLD_BYTES_WRITTEN / (1024 * 1024)
Cold Reads (COLD_READS)	Number of reads from the cold region	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.COLD_READS
Cold Used Mbytes (COLD_USED_MBYTES)	Size of used area (in megabytes) in the cold region	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.COLD_USED_MB
Cold Writes (COLD_WRITES)	Number of writes to the cold region	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.COLD_WRITES
Create Date (CREATE_DATE)	Date the disk was added to the disk group	--	string(20)	No	Oracle 11g R2 or later	V$ASM_DISK.CREATE_DATE
Disk Group Name (DISK_GROUP_NAME)	Name of the group the disk belongs to. The value is blank if the disk does not belong to a group or the disk group is dismounted.	--	string(30)	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.NAME
Disk Name (NAME)	Disk name	--	string(30)	No	Oracle 11g R2 or later	V$ASM_DISK.NAME
Disk Number (DISK_NUMBER)	Number assigned to the disk in the disk group	--	ulong	No	Oracle 11g R2 or later	V$ASM_DISK.DISK_NUMBER
Fail Group (FAILGROUP)	Name of the fail group that includes the disk	--	string(30)	No	Oracle 11g R2 or later	V$ASM_DISK.FAILGROUP
Fail Group Type (FAILGROUP_TYPE)	Type of fail group	--	string(7)	No	Oracle 11g R2 or later	V$ASM_DISK.FAILGROUP_TYPE
Free % (PERCENT_FREE)	Percentage of unused disk space	--	double	No	Oracle 11g R2 or later	(V$ASM_DISK.FREE_MB / V$ASM_DISK.TOTAL_MB) * 100
Free Mbytes (FREE_MBYTES)	Size of unused disk space (in megabytes)	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.FREE_MB
Group Number (GROUP_NUMBER)	Number of the disk group that includes the disk. The value is 0 if the group is dismounted or the disk does not belong to a group.	--	ulong	No	Oracle 11g R2 or later	V$ASM_DISK.GROUP_NUMBER
Header Status (HEADER_STATUS)	Disk status	--	string(12)	No	Oracle 11g R2 or later	V$ASM_DISK.HEADER_STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Hot Mbytes Read (HOT_MBYTES_READ)	Size of data (in megabytes) read from the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_BYTES_READ / (1024 * 1024)
Hot Mbytes Written (HOT_MBYTES_WRITTEN)	Size of data (in megabytes) written to the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_BYTES_WRITTEN / (1024 * 1024)
Hot Reads (HOT_READS)	Number of reads from the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_READS
Hot Reads % (PERCENT_HOT_READS)	Percentage of reads from the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_READS / V $ASM_DISK.READS * 100
Hot Used % (PERCENT_HOT_USED)	Percentage of used area in the hot region	--	double	No	Oracle 11g R2 or later	(V $ASM_DISK.HOT_USED_MB / (V $ASM_DISK.HOT_USED_MB + V $ASM_DISK.COLD_USED_MB)) * 100
Hot Used Mbytes (HOT_USED_MBYTES)	Size of used area (in megabytes) in the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_USED_MB
Hot Writes % (PERCENT_HOT_WRITES)	Percentage of writes to the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_WRITES / V $ASM_DISK.WRITES * 100
Hot Writes (HOT_WRITES)	Number of writes to the hot region	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.HOT_WRITES
Label (LABEL)	Disk label	--	string(31)	No	Oracle 11g R2 or later	V$ASM_DISK.LABEL
Mode Status (MODE_STATUS)	Global status of the type of I/O request permitted for the disk	--	string(7)	No	Oracle 11g R2 or later	V $ASM_DISK.MODE_STATUS
Mount Date (MOUNT_DATE)	Date the disk was mounted	--	string(20)	No	Oracle 11g R2 or later	V $ASM_DISK.MOUNT_DATE
Mount Status (MOUNT_STATUS)	Status of the group the disk belongs to	--	string(7)	No	Oracle 11g R2 or later	V $ASM_DISK.MOUNT_STATUS
OS Mbytes (OS_MBYTES)	Disk size reported by the OS (in megabytes)	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.OS_MB
Path (PATH)	OS path	--	string(256)	No	Oracle 11g R2 or later	V$ASM_DISK.PATH

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Preferred Read (PREFERRED_READ)	Status of the preferred read fail group	--	string(1)	No	Oracle 11g R2 or later	V$ASM_DISK.PREFERRED_READ
Read Errs (READ_ERRS)	Number of I/O read errors for the disk	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.READ_ERRS
Read Mbytes (READ_MBYTES)	Size of data (in megabytes) read from the disk	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.BYTES_READ / (1024 * 1024)
Read Time (READ_TIME)	Total read request time for the disk in seconds. Set the TIMED_STATISTICS parameter in the init.ora file to TRUE to collect the value of this field.	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.READ_TIME
Reads (READS)	Number of I/O read requests for the disk	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.READS
Record Time (RECORD_TIME)	Time that collection of the performance data stored in the record finished	--	time_t	No	Oracle 11g R2 or later	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name. Always set to PDDK.	--	string(4)	No	Oracle 11g R2 or later	Agent Collector
Redundancy (REDUNDANCY)	Hardware redundancy of the disk	--	string(7)	No	Oracle 11g R2 or later	V$ASM_DISK.REDUNDANCY
Repair Timer (REPAIR_TIMER)	Remaining time until the disk is automatically deleted	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.REPAIR_TIMER
Sector Size (SECTOR_SIZE)	Physical block size in bytes	--	ushort	No	Oracle 11g R2 or later	V$ASM_DISK.SECTOR_SIZE
Start Time (START_TIME)	Start time for collection of the performance data stored in the record	--	time_t	No	Oracle 11g R2 or later	Agent Collector
State (STATE)	Global status of the disk in relation to the disk group	--	string(8)	No	Oracle 11g R2 or later	V$ASM_DISK.STATE
Total Mbytes (TOTAL_MBYTES)	Total disk capacity (in megabytes)	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.TOTAL_MB
UDID (UDID)	Unique device ID in the name returned by detection	--	string(64)	No	Oracle 11g R2 or later	V$ASM_DISK.UDID
Used Mbytes (USED_MBYTES)	Size of used disk space (in megabytes)	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.TOTAL_M

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Mbytes (USED_MBYTES)	Size of used disk space (in megabytes)	--	double	No	Oracle 11g R2 or later	B - V $ASM_DISK.FREE_MB
Voting File (VOTING_FILE)	Indicates whether a voting file is included in the disk. Y is set if the voting file is included. In other cases, N is set.	--	string(1)	No	Oracle 11g R2 or later	V $ASM_DISK.VOTING_FILE
Write Errs (WRITE_ERRS)	Number of disk write errors	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.WRITE_ERRS
Write Time (WRITE_TIME)	Total write request time to the disk in seconds. Set the TIMED_STATISTICS parameter in the init.ora file to TRUE to collect the value of this field.	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.WRITE_TIME
Writes (WRITES)	Number of I/O write requests to the disk	--	double	No	Oracle 11g R2 or later	V$ASM_DISK.WRITES
Written Mbytes (WRITTEN_MBYTES)	Size of data written to the disk (in megabytes)	--	double	No	Oracle 11g R2 or later	V $ASM_DISK.BYTES_WRITTEN / (1024 * 1024)

ASM Disk Group Interval (PI_PIDG)

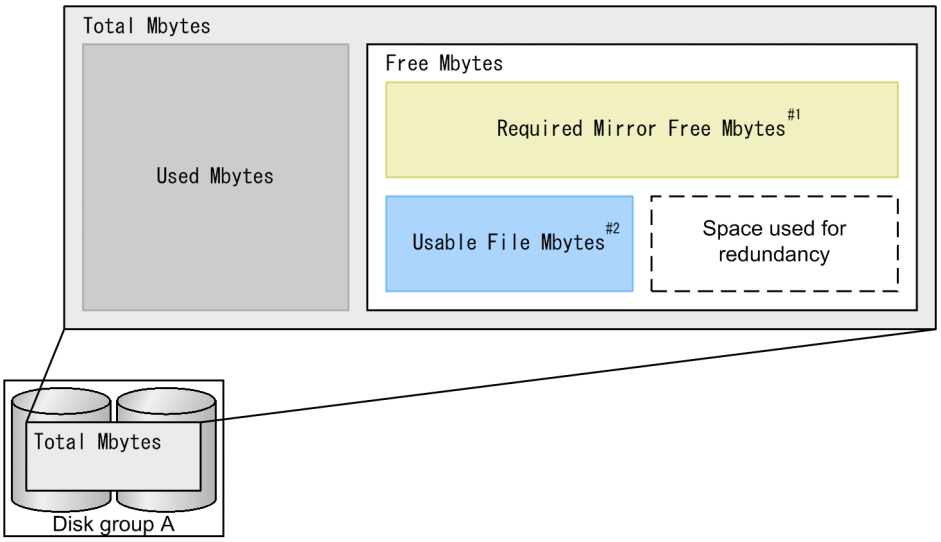
Function

The ASM Disk Group Interval (PI_PIDG) record stores, at specific intervals, the performance data of the ASM disk managed by the ASM instance through which Oracle communicates. This is a multi-instance record.

To use this record to monitor the free space in a non-mirrored configuration, we recommend monitoring the Free Mbyte field. In a mirrored configuration, we recommend monitoring the Usable File Mbytes field.

Using an ASM disk group with a redundant mirroring configuration as an example, the field configuration pertaining to the free space in the ASM Disk Group Interval (PI_PIDG) is shown in the following figure.

Figure 6–2: Field configuration pertaining to the free space in the ASM Disk Group Interval (PI_PIDG) record



#1

If a failure occurs in a mirrored ASM disk group, this capacity is needed for recovery. If this capacity is insufficient when a failure occurs, recovery might be impossible.

#2

This shows the space that can be used in a mirroring configuration. This is the value obtained by subtracting the Required Mirror Free Mbytes value from the Free Mbytes value. In a redundant (duplicated) mirroring configuration, if the difference is 4 gigabytes, the Usable File Mbytes value is halved (2 gigabytes).

If this capacity is insufficient, a new file might not be created, or file redundancy might not be maintained.

Depending on the values of the Free Mbyte and Required Mirror Free Mbytes fields, the capacity might be a negative value.

Notes

- The collection of the ASM Disk Group Interval (PI_PIDG) record is supported when the versions of both ASM and Oracle Database are 11.2.0 or later. Collection is not supported in version 11.1.0 or earlier.
- The capacity of the ASM disk group is calculated from the online ASM disks. As a result, the original capacity cannot be collected if any offline ASM disks exist.
- If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record stores performance data for all ASM disk group in the entire CDB, which are managed by the ASM instances through which Oracle communicates. The retrieved values will be same for both when monitoring PDBs and when monitoring the root container (CDB$ROOT).

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	260	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIDG_NAME

Lifetime

From creation to deletion of the ASM disk group

Record size

- Fixed part: 678 bytes
- Variable part: 280 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Allocation Unit Size (ALLOCATION_UNIT_SIZE)	Size of the allocation unit in bytes	COPY	ulong	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.ALLOCATION_UNIT_SIZE
Cold Used Mbytes (COLD_USED_MBYTES)	Size of used area (in megabytes) in the cold region	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.COLD_USED_MB
Free % (FREE_PERC)	Percentage of free space in the disk group	AVG	double	No	Oracle 11g R2 or later	(FREE_MBYTES / TOTAL_MBYTES) * 100
Free Mbytes (FREE_MBYTES)	Size of unused area (in megabytes) of the disk group	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.FREE_MB
Group Number (GROUP_NUMBER)	Cluster number assigned to the disk group	COPY	ulong	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.GROUP_NUMBER
Hot Used % (PERCENT_HOT_USED)	Percentage of use in the hot region	AVG	double	No	Oracle 11g R2 or later	(V$ASM_DISKGROUP.HOT_USED_MB / (V$ASM_DISKGROUP.HOT_USED_MB + V$ASM_DISKGROUP.COLD_USED_MB)) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Hot Used Mbytes (HOT_USED_MBYTES)	Size of used area (in megabytes) in the hot region	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.HOT_USED_MB
Name (NAME)	Disk group name	COPY	string(30)	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.NAME
Offline Disks (OFFLINE_DISKS)	Number of offline disks in the disk group	AVG	long	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.OFFLINE_DISKS
Record Time (RECORD_TIME)	Time that collection of the performance data stored in the record finished	COPY	time_t	No	Oracle 11g R2 or later	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIDG)	COPY	string(4)	No	Oracle 11g R2 or later	Agent Collector
Required Mirror Free Mbytes (REQUIRED_MIRROR_FREE_MBYTES)	Capacity (in megabytes) needed for recovery	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.REQUIRED_MIRROR_FREE_MB
Restore Area Lack (RESTORE_AREA_LACK)	Indicates whether the capacity is sufficient for recovery. 1 is set if the capacity is insufficient. 0 is set if the capacity is sufficient.	COPY	short	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.FREE_MB >= V$ASM_DISKGROUP.REQUIRED_MIRROR_FREE_MB
Sector Size (SECTOR_SIZE)	Physical block size in bytes	COPY	ushort	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.SECTOR_SIZE
Start Time (START_TIME)	Start time for collection of the performance data stored in the record	COPY	time_t	No	Oracle 11g R2 or later	Agent Collector
State (STATE)	Disk group status	COPY	string(11)	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.STATE
Total Mbytes (TOTAL_MBYTES)	Total capacity of the disk group (in megabytes)	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.TOTAL_MB
Type (TYPE)	Redundancy type of the disk group	COPY	string(6)	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.TYPE
Usable File Mbytes (USABLE_FILE_MBYTES)	Free space (in megabytes) that can be used in a mirroring configuration. This value can be negative.	AVG	double	No	Oracle 11g R2 or later	V$ASM_DISKGROUP.USABLE_FILE_MB
Usable File Free % (USABLE_PERC)	Percentage of space necessary for restoration. When this percentage turns	AVG	double	No	Oracle 11g R2 or later	(USABLE_FILE_MBYTES / TOTAL_MBYTES) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Usable File Free % (USABLE_PERC)	out to be a negative value, 0 is displayed instead.	AVG	double	No	Oracle 11g R2 or later	(USABLE_FILE_MBYTES / TOTAL_MBYTES) * 100
Used Mbytes (USED_MBYTES)	Size of used area (in megabytes) of the disk group	AVG	double	No	Oracle 11g R2 or later	V $ASM_DISKGROUP.TOTAL_MB - V $ASM_DISKGROUP.FREE_MB
Voting Files (VOTING_FILES)	Indicates whether a voting file is included in the disk group. Y is set if the voting file is included. In other cases, N is set.	COPY	string(1)	No	Oracle 11g R2 or later	V $ASM_DISKGROUP.VOTING_FILES

Backup Async IO (PD_PDBA)

Function

The Backup Async IO (PD_PDBA) record stores performance data indicating the status, at a specific point in time, of a file that was backed up or restored with asynchronous I/O operations and that is now being or has been executed by the Oracle recovery manager (RMAN). PFM - Agent for Oracle creates one record for each file that is backed up or restored with asynchronous I/O operations by the Oracle recovery manager (RMAN). This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	35	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

- PD_PDBA_FILE_NAME
- PD_PDBA_TYPE
- PD_PDBA_USE_COUNT

Lifetime

From the start to the end of RMAN

Record Size

- Fixed part: 678 bytes
- Variable part: 750 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Serial (SERIAL)	Number of session IDs being used for backup or restore processing	--	double	No	All	V $BACKUP_ASYNC_IO. SERIAL
Buffer Count (BUFFER_COUNT)	Number of buffers used for read or write operations	--	double	No	All	V $BACKUP_ASYNC_IO. BUFFER_COUNT
Buffer Size (BUFFER_SIZE)	Size of buffer (in kilobytes) used for file read or write operations	--	double	No	All	V $BACKUP_ASYNC_IO. BUFFER_SIZE / 1024

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Close Time (CLOSE_TIME)	Time file was closed (if the value of the Type field is AGGREGATE, this is the time all files were closed in batch mode)	--	string(20)	No	All	V$BACKUP_ASYNC_IO.CLOSE_TIME
Device Type (DEVICE_TYPE)	Type of device where the file is located	--	string(17)	No	All	V$BACKUP_ASYNC_IO.DEVICE_TYPE
Effective Rate (EFFECTIVE_KBYTES_PER_SEC)	Asynchronous I/O rate in kilobytes for the device under backup processing	--	ulong	No	All	V$BACKUP_ASYNC_IO.EFFECTIVE_BYTES_PER_SECOND / 1024
Elapsed Time (ELAPSED_TIME)	Length of time the file was open in seconds	--	double	Yes	All	V$BACKUP_ASYNC_IO.ELAPSED_TIME / 100
FileName (FILE_NAME)	Name of the backup file	--	string(513)	No	All	V$BACKUP_ASYNC_IO.FILENAME
IO Count (IO_COUNT)	Number of asynchronous file I/O operations	--	double	Yes	All	V$BACKUP_ASYNC_IO.IO_COUNT
Long Waits (LONG_WAITS)	Number of times the buffer became available after a block wait was issued	--	double	Yes	All	V$BACKUP_ASYNC_IO.LONG_WAITS
Max Long Wait Time (LONG_WAIT_TIME_MAX)	Maximum length of block wait time required for asynchronous I/O operations in seconds. However, the value indicated in this field is 1/100 of the actual value.	--	long	No	All	V$BACKUP_ASYNC_IO.LONG_WAIT_TIME_MAX / 100
Max Open Files (MAX_OPEN_FILES)	If the value of the Type field is AGGREGATE, the number of disk files that are open at any one time. If the value of the Type field is INPUT or OUTPUT, this field is empty.	--	ulong	No	All	V$BACKUP_ASYNC_IO.MAXOPENFILES
Max Short Wait Time (SHORT_WAIT_TIME_MAX)	Maximum unblocked polling time in seconds until asynchronous I/O operations were completed	--	double	No	All	V$BACKUP_ASYNC_IO.SHORT_WAIT_TIME_MAX / 100
Mbytes (MBYTES)	Number of megabytes read or written during the interval	--	double	Yes	All	V$BACKUP_ASYNC_IO.BYTES / (1024 * 1024)
Open Time (OPEN_TIME)	Time the file was opened (if the value of the Type field is AGGREGATE, the time the first file was opened)	--	string(20)	No	All	V$BACKUP_ASYNC_IO.OPEN_TIME

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Ready (READY)	Number of asynchronous requests for which buffer was made ready	--	double	No	All	V$BACKUP_ASYNC_IO.READY
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDBA)	--	string(4)	No	All	Agent Collector
SID (SID)	Oracle session ID of the session that is executing backup or restore processing	--	ulong	No	All	V$BACKUP_ASYNC_IO.SID
Set Count (SET_COUNT)	Number of backup sets under read or write operation	--	double	No	All	V$BACKUP_ASYNC_IO.SET_COUNT
Set Stamp (SET_STAMP)	Set stamp of the backup set under read or write operation	--	double	No	All	V$BACKUP_ASYNC_IO.SET_STAMP
Short Waits (SHORT_WAITS)	Number of times buffer became available after an asynchronous I/O operation and unblocked polling	--	double	Yes	All	V$BACKUP_ASYNC_IO.SHORT_WAITS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Backup or restoration status. Valid values are FINISHED, IN PROGRESS, and NOT STARTED.	--	string(12)	No	All	V$BACKUP_ASYNC_IO.STATUS
Total Long Wait Time (LONG_WAIT_TIME_TOTAL)	Total block wait time (in seconds) required to complete asynchronous I/O operations.	--	long	Yes	All	V$BACKUP_ASYNC_IO.LONG_WAIT_TIME_TOTAL / 100
Total Mbytes (TOTAL_MBYTES)	Total number of bytes in megabytes read or written (if the value is unknown, this field is empty)	--	double	Yes	All	V$BACKUP_ASYNC_IO.TOTAL_BYTES / (1024 * 1024)
Total Short Wait Time (SHORT_WAIT_TIME_TOTAL)	Total unblocked polling time (in seconds) before asynchronous I/O operation was completed	--	double	Yes	All	V$BACKUP_ASYNC_IO.SHORT_WAIT_TIME_TOTAL / 100
Type (TYPE)	Type of backup or restore processing. Valid values are AGGREGATE, INPUT, and OUTPUT.	--	string(10)	No	All	V$BACKUP_ASYNC_IO.TYPE
Use Count (USE_COUNT)	Counter used to identify a row from a different backup set	--	double	No	All	V$BACKUP_ASYNC_IO.USE_COUNT

Backup Sync IO (PD_PDBS)

Function

The Backup Sync IO (PD_PDBS) record stores performance data indicating the status, at a specific point in time, of a file that was backed up or restored with synchronous I/O operations and that is now being or has been executed by the Oracle recovery manager (RMAN). PFM - Agent for Oracle creates one record for each file that is backed up or restored with synchronous I/O operations by the Oracle recovery manager (RMAN). This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

- PD_PDBS_FILE_NAME
- PD_PDBS_TYPE
- PD_PDBS_USE_COUNT

Lifetime

From the start to the end of RMAN

Record Size

- Fixed part: 678 bytes
- Variable part: 718 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Serial (SERIAL)	Number of session IDs being used for backup or restore processing	--	double	No	All	V$BACKUP_SYNC_IO.SERIAL
Avg Transfer Rate (DISCRETE_KBYTES_PER_SEC)	Average file transfer rate in kilobytes	--	double	No	All	V$BACKUP_SYNC_IO.DISCRETE_BYTES_PER_SECOND / 1024
Buffer Count (BUFFER_COUNT)	Number of buffers used for file read or write operations	--	double	No	All	V$BACKUP_SYNC_IO.BUFFER_COUNT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Buffer Size (BUFFER_SIZE)	Size of buffer (in kilobytes) used for file read or write operations	--	double	No	All	V $BACKUP_SYNC_IO.BUFFER_SIZE / 1024
Close Time (CLOSE_TIME)	Time the file was closed (if the value of the Type field is AGGREGATE, the time the first file was closed)	--	string(20)	No	All	V $BACKUP_SYNC_IO.CLOSE_TIME
Device Type (DEVICE_TYPE)	Type of device where the file is located	--	string(17)	No	All	V $BACKUP_SYNC_IO.DEVICE_TYPE
Effective Rate (EFFECTIVE_KBYTES_PER_SEC)	I/O rate of the device under backup processing in kilobytes	--	double	No	All	V $BACKUP_SYNC_IO.EFFECTIVE_BYTES_PER_SECOND / 1024
Elapsed Time (ELAPSED_TIME)	Length of time the file was open in seconds	--	double	Yes	All	V $BACKUP_SYNC_IO.ELAPSED_TIME / 100
FileName (FILE_NAME)	Name of backup file	--	string(513)	No	All	V $BACKUP_SYNC_IO.FILENAME
IO Count (IO_COUNT)	Number of file I/O operations	--	double	Yes	All	V $BACKUP_SYNC_IO.IO_COUNT
Max IO Time (IO_TIME_MAX)	Maximum time required for a single I/O request in seconds	--	double	No	All	V $BACKUP_SYNC_IO.IO_TIME_MAX / 100
Max Open Files (MAX_OPEN_FILES)	If the value of the Type field is AGGREGATE, the number of disk files that are open at one time. If the value of the Type field is INPUT or OUTPUT, this field is empty.	--	long	No	All	V $BACKUP_SYNC_IO.MAXOPENFILES
Mbytes (MBYTES)	Number of megabytes read or written during the interval	--	double	Yes	All	V $BACKUP_SYNC_IO.BYTES / (1024 * 1024)
Open Time (OPEN_TIME)	Time the file was opened (if the value of the Type field is AGGREGATE, the time the first file was opened)	--	string(20)	No	All	V $BACKUP_SYNC_IO.OPEN_TIME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDBS)	--	string(4)	No	All	Agent Collector
SID (SID)	Oracle session ID of the session that is executing backup or restore processing	--	ulong	No	All	V $BACKUP_SYNC_IO.SID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Set Count (SET_COUNT)	Number of backup sets under read or write operation	--	double	No	All	V$BACKUP_SYNC_IO.SET_COUNT
Set Stamp (SET_STAMP)	Set stamp of the backup set under read or write operation	--	double	No	All	V$BACKUP_SYNC_IO.SET_STAMP
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Backup or restoration status. Valid values are FINISHED, IN PROGRESS, and NOT STARTED.	--	string(12)	No	All	V$BACKUP_SYNC_IO.STATUS
Total IO Time (IO_TIME_TOTAL)	Total I/O time on files in seconds.	--	double	Yes	All	V$BACKUP_SYNC_IO.IO_TIME_TOTAL / 100
Total Mbytes (TOTAL_MBYTES)	Total number of megabytes read or written (if this information is unknown, this field is empty)	--	double	Yes	All	V$BACKUP_SYNC_IO.TOTAL_BYTES / (1024 * 1024)
Type (TYPE)	Type of backup or restore processing. Valid values are AGGREGATE, INPUT, and OUTPUT.	--	string(10)	No	All	V$BACKUP_SYNC_IO.TYPE
Use Count (USE_COUNT)	Counter used to identify a row from a different backup set	--	double	No	All	V$BACKUP_SYNC_IO.USE_COUNT

Block Contention Interval (PI_PIBC)

Function

The Block Contention Interval (PI_PIBC) record stores performance data, taken at specific intervals, about each wait class. To collect this record, you must specify TRUE in the TIMED_STATISTICS parameter in the init.ora Oracle initialization parameter file.

PFM - Agent for Oracle creates one record for each wait class. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	5	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

PI_PIBC_CLASS

Lifetime

From the creation to the deletion of an Oracle instance

Record Size

- Fixed part: 678 bytes
- Variable part: 60 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Class (CLASS)	Block class[#1]	COPY	string(19)	No	All	V$WAITSTAT.CLASS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIBC)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Wait Count (WAIT_COUNT)	Number of OPERATION waits for the block's class.#2	AVG	double	Yes	All	V$WAITSTAT.COUNT
Wait Time (WAIT_TIME)	Total OPERATION wait time for the block's class#2	AVG	double	Yes	All	V$WAITSTAT.TIME

Block Contention Statistics (PD_PDBC)

Function

The Block Contention Statistics (PD_PDBC) record stores performance data indicating the status of each wait class at a specific point in time. To collect this record, you must specify TRUE in the TIMED_STATISTICS parameter in the init.ora Oracle initialization parameter file.

PFM - Agent for Oracle creates one record for each wait class. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	5	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

PD_PDBC_CLASS

Lifetime

From the creation to the deletion of an Oracle instance

Record Size

- Fixed part: 678 bytes
- Variable part: 36 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Class (CLASS)	Block class	--	string(19)	No	All	V$WAITSTAT.CLASS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDBC)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Wait Count (WAIT_COUNT)	Number of OPERATION waits for the block's class.	--	double	No	All	V$WAITSTAT.COUNT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Wait Time (WAIT_TIME)	Total OPERATION wait time for the block's class	--	double	No	All	V$WAITSTAT.TIME

Buffer Pool (PD_PDBP)

Function

The Buffer Pool (PD_PDBP) record stores performance data indicating the status of the buffer pool available to instances at a specific point in time. PFM - Agent for Oracle creates one record for each buffer pool. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	45	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

PD_PDBP_ID

Lifetime

From the start to the stop of an Oracle instance

Record Size

- Fixed part: 678 bytes
- Variable part: 177 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Buffer Busy Wait (BUFFER_BUSY_WAIT)	Statistics about buffer busy waits Number of buffers in buffer pool	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.BUFFER_BUSY_WAIT
Buffers (BUFFERS)	Statistics about consistent gets	--	double	No	All	V$BUFFER_POOL.BUFFERS
Consistent Gets (CONSISTENT_GETS)	Statistics about consistent gets	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.CONSISTENT_GETS
Db Block Change (DB_BLOCK_CHANGE)	Statistics about changes to database blocks	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.DB_BLOCK_CHANGE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Db Block Gets (DB_BLOCK_GETS)	Statistics about collected database blocks	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.DB_BLOCK_GETS
Dirty Buffers Inspected (DIRTY_BUFFERS_INSPECTED)	Information about dirty buffers	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.DIRTY_BUFFERS_INSPECTED
Free Buffer Inspected (FREE_BUFFER_INSPECTED)	Information about free buffers	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.FREE_BUFFER_INSPECTED
Free Buffer Wait (FREE_BUFFER_WAIT)	Statistics about free buffer waits	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.FREE_BUFFER_WAIT
Got Buffers (BUF_GOT)	Number of buffers collected by sets	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.BUF_GOT
ID (ID)	Buffer pool ID	--	ulong	No	All	V$BUFFER_POOL.ID
Max Set Size (SET_MSIZE)	Maximum size of buffer pool set	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.SET_MSIZE
Name (NAME)	Buffer pool name. Valid values are DEFAULT, KEEP, and RECYCLE.	--	string(20)	No	All	V$BUFFER_POOL.NAME
Physical Reads (PHYSICAL_READS)	Statistical value of physical read operations	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.PHYSICAL_READS
Physical Writes (PHYSICAL_WRITES)	Statistical value of physical write operations	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.PHYSICAL_WRITES
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDBP)	--	string(4)	No	All	Agent Collector
Repl Num (CNUM_REPL)	Number of buffers on exchange list	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.CNUM_REPL
Scan Sum (SUM_SCAN)	Number of scanned buffers in sets	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.SUM_SCAN

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Set Count (SET_COUNT)	Number of sets in buffer pool	--	double	No	All	V$BUFFER_POOL.SET_COUNT
Set Num (CNUM_SET)	Number of buffers in sets	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.CNUM_SET
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Write Complete Wait (WRITE_COMPLETE_WAIT)	Statistical value of write complete waits	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.WRITE_COMPLETE_WAIT
Write Num (CNUM_WRITE)	Number of buffers on write list	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.CNUM_WRITE
Write Sum (SUM_WRITE)	Number of buffers written in sets	--	double	Yes	All	V$BUFFER_POOL_STATISTICS.SUM_WRITE

Cache Summary (PD_PDCS)

Function

The Cache Summary (PD_PDCS) record stores performance data indicating the status of each cache in the system global area (SGA) at a specific point in time.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. For details, see the descriptions of each field.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	300	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC Key Fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record Size

- Fixed part: 886 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Cursor Open Hit % (CURSOR_OPEN_HIT_PERCENTAGE)	Percentage of cursors found in cursor searches that were open cursors[#10]	--	double	No	All	V $SYSTEM_CURSOR_CACHE.HIT_RATIO * 100
Cursor Open Hits (CURSOR_OPEN_HITS)	Total cursor open hits[#10]	--	double	No	All	V $SYSTEM_CURSOR_CACHE.HITS
Cursor Opens (CURSOR_OPENS)	Total cursor opens[#10]	--	double	No	All	V $SYSTEM_CURSOR_CACHE.OPENS
Database Object Cache Keeps	Number of objects kept[#9]	--	ulong	No	All	COUNT(V $DB_OBJECT_CACHE) where KEPT='YES'

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
(DATABASE_OBJECT_CACHE_KEEPS)	Number of objects kept[#9]	--	ulong	No	All	COUNT(V$DB_OBJECT_CACHE) where KEPT='YES'
Database Object Cache Locks (DATABASE_OBJECT_CACHE_LOCKS)	Number of users locking objects in cache[#9]	--	long	No	All	SUM(V$DB_OBJECT_CACHE.LOCKS)
Database Object Cache Pins (DATABASE_OBJECT_CACHE_PINS)	Number of users who have acquired objects in cache[#9]	--	long	No	All	SUM(V$DB_OBJECT_CACHE.PINS)
Dict Cache Fixed (DICTIONARY_CACHE_FIXED)	Number of entries fixed in cache[#10]	--	double	No	All	SUM(V$ROWCACHE.FIXED)
Dict Cache Flushes (DICTIONARY_CACHE_FLUSHES)	Number of disk flushes[#10]	--	double	No	All	SUM(V$ROWCACHE.FLUSHES)
Dict Cache Get Miss % (DICTIONARY_CACHE_GET_MISSES_PERCENTAGE)	Percentage of data requests for which a cache miss occurred[#10]	--	double	No	All	(SUM(V$ROWCACHE.GETMISSES) / SUM(V$ROWCACHE.GETS)) * 100
Dict Cache Get Misses (DICTIONARY_CACHE_GET_MISSES)	Number of data requests for which a cache miss occurred[#10]	--	double	No	All	SUM(V$ROWCACHE.GETMISSES)
Dict Cache Gets (DICTIONARY_CACHE_GETS)	Number of requests for information about data objects[#10]	--	double	No	All	SUM(V$ROWCACHE.GETS)
Dict Cache Modifications (DICTIONARY_CACHE_MODIFICATIONS)	Number of insert, update, and delete operations[#10]	--	double	No	All	SUM(V$ROWCACHE.MODIFICATIONS)
Dict Cache Scan Completes (DICTIONARY_CACHE_SCAN_COMPLETES)	Number of times scanning of a list of dependent entries was completed[#10]	--	double	No	All	SUM(V$ROWCACHE.SCANCOMPLETES)
Dict Cache Scan Misses (DICTIONARY_CACHE_SCAN_MISSES)	Number of times data was not found in cache by scanning[#10]	--	double	No	All	SUM(V$ROWCACHE.SCANMISSES)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Dict Cache Scan Misses % (DICTIONARY_CACHE_SCAN_MISSES_PERCENTAGE)	Percentage of times data was not found in cache by scanning#10	--	double	No	All	(SUM(V$ROWCACHE.SCANMISSES) / SUM(V$ROWCACHE.SCANS)) * 100
Dict Cache Scans (DICTIONARY_CACHE_SCANS)	Number of scan requests#10	--	double	No	All	SUM(V$ROWCACHE.SCANS)
Dict Cache Usage (DICTIONARY_CACHE_USAGE)	Number of cache entries with valid data#10	--	ulong	No	All	SUM(V$ROWCACHE.USAGE)
Lib Cache Get Hit % (LIBRARY_CACHE_GET_HIT_PERCENTAGE)	Percentage of times objects were found in library cache#10	--	double	No	All	(SUM(V$LIBRARYCACHE.GETHITS) / SUM(V$LIBRARYCACHE.GETS)) * 100
Lib Cache Get Hits (LIBRARY_CACHE_GET_HITS)	Number of times objects were found in library cache#10	--	double	No	All	SUM(V$LIBRARYCACHE.GETHITS)
Lib Cache Gets (LIBRARY_CACHE_GETS)	Number of locks requested for objects in library cache#10	--	double	No	All	SUM(V$LIBRARYCACHE.GETS)
Lib Cache Invalidations (LIBRARY_CACHE_INVALIDATIONS)	Number of times a nonpermane nt library object (such as a shared SQL area) was invalid#10	--	double	No	All	SUM(V$LIBRARYCACHE.INVALIDATIONS)
Lib Cache Miss % (LIBRARY_CACHE_MISS_PERCENTAGE)	Library cache miss percentage: The percentage of objects in the library cache that had to be reloaded into the library cache. A higher percentage indicates greater resource usage.#10	--	double	No	All	(SUM(V$LIBRARYCACHE.RELOADS) / SUM(V$LIBRARYCACHE.PINS)) * 100
Lib Cache Pin Hit % (LIBRARY_CACHE_PIN_HIT_PERCENTAGE)	Ratio (as a percent) of the LIBRARY_CACHE_PIN_HITS value to the LIBRARY_CACHE_PINS value#10	--	double	No	All	(SUM(V$LIBRARYCACHE.PINHITS) / SUM(V$LIBRARYCACHE.PINS)) * 100
Lib Cache Pin Hits (LIBRARY_CACHE_PIN_HITS)	Number of times an object acquired or accessed by the system had already been placed in cache and initialized#10	--	double	No	All	SUM(V$LIBRARYCACHE.PINHITS)
Lib Cache Pins (LIBRARY_CACHE_PINS)	Number of times the system acquired an object in cache to access it#10	--	double	No	All	SUM(V$LIBRARYCACHE.PINS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Lib Cache Reloads (LIBRARY_CACHE_RELOADS)	Number of times the system needed to reinitialize a library object and load it along with the data because it had not been used recently or it was invalid[#10]	--	double	No	All	SUM(V$LIBRARYCACHE.RELOADS)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDCS)	--	string(4)	No	All	Agent Collector
SQL Executing (SQL_EXECUTING)	Total number of users currently executing SQL statements[#9]	--	double	No	All	SUM(V$SQLAREA.USERS_EXECUTING)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Total SQL Executions (TOTAL_SQL_EXECUTIONS)	Total number of SQL executions[#9]	--	double	No	All	execute count

#9

Displays performance data for the PDBs that are being monitored.

#10

Displays common performance data for database instances.

Cache Summary Interval (PI_PICS)

Function

The Cache Summary Interval (PI_PICS) record stores performance data, taken at specific intervals, about each cache in the system global area (SGA).

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. For details, see the descriptions of each field.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	300	Y
Collection Offset	20	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC Key Fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record Size

- Fixed part: 1,202 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Cursor Open Hit % (CURSOR_OPEN_HIT_PERCENTAGE)	found in cursor searches that were open cursors[#2, #10]	AVG	double	No	All	V $SYSTEM_CURSOR_CACHE.HIT_RATIO * 100
Cursor Open Hits (CURSOR_OPEN_HITS)	Total cursor open hits[#2, #10]	AVG	double	Yes	All	V $SYSTEM_CURSOR_CACHE.HITS
Cursor Opens (CURSOR_OPENS)	Total cursor opens[#2, #10]	AVG	double	Yes	All	V $SYSTEM_CURSOR_CACHE.OPENS
Database Object Cache Keeps	Number of objects kept[#2]	AVG	ulong	No	All	COUNT(V $DB_OBJECT_CACHE) where KEPT='YES'

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
(DATABASE_OBJECT_CACHE_KEEPS)	Number of objects kept[#2]	AVG	ulong	No	All	COUNT(V $DB_OBJECT_CACHE) where KEPT='YES'
Database Object Cache Locks (DATABASE_OBJECT_CACHE_LOCKS)	Number of users locking objects in cache[#2, #9]	AVG	long	No	All	SUM(V $DB_OBJECT_CACHE.LOCKS)
Database Object Cache Pins (DATABASE_OBJECT_CACHE_PINS)	Number of users who have acquired objects in cache[#2, #9]	AVG	long	No	All	SUM(V $DB_OBJECT_CACHE.PINS)
Dict Cache Fixed (DICTIONARY_CACHE_FIXED)	Number of entries fixed in cache[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.FIXED)
Dict Cache Flushes (DICTIONARY_CACHE_FLUSHES)	Number of disk flushes[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.FLUSHES)
Dict Cache Get Misses (DICTIONARY_CACHE_GET_MISSES)	Number of data requests for which a cache miss occurred[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.GETMISSES)
Dict Cache Get Misses % (DICTIONARY_CACHE_GET_MISSES_PERCENTAGE)	Percentage of data requests for which a cache miss occurred[#2, #10]	AVG	double	No	All	(SUM(V $ROWCACHE.GETMISSES) / SUM(V $ROWCACHE.GETS)) * 100
Dict Cache Gets (DICTIONARY_CACHE_GETS)	Number of requests for information about data objects[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.GETS)
Dict Cache Modifications (DICTIONARY_CACHE_MODIFICATIONS)	Number of insert, update, and delete operations[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.MODIFICATIONS)
Dict Cache Scan Completes (DICTIONARY_CACHE_SCAN_COMPLETES)	Number of times scanning of a list of dependent entries was completed[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.SCANCOMPLETES)
Dict Cache Scan Miss % (DICTIONARY_CACHE_SCAN_MISSES_PERCENTAGE)	Percentage of times data was not found in cache by scanning[#2, #10]	AVG	double	No	All	(SUM(V $ROWCACHE.SCANMISSES) / SUM(V $ROWCACHE.SCANS)) * 100
Dict Cache Scan Misses (DICTIONARY_CACHE_SCAN_MISSES)	Number of times data was not found in cache by scanning[#2, #10]	AVG	double	Yes	All	SUM(V $ROWCACHE.SCANMISSES)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Dict Cache Scans (DICTIONARY_CACHE_SCANS)	Number of scan requests[#2, #10]	AVG	double	Yes	All	SUM(V$ROWCACHE.SCANS)
Dict Cache Usage (DICTIONARY_CACHE_USAGE)	Number of cache entries with valid data[#2, #10]	AVG	double	Yes	All	SUM(V$ROWCACHE.USAGE)
Lib Cache Get Hit % (LIBRARY_CACHE_GET_HIT_PERCENTAGE)	Percentage of times objects were found in library cache[#2, #10]	AVG	double	No	All	(SUM(V$LIBRARYCACHE.GETHITS) / SUM(V$LIBRARYCACHE.GETS)) * 100
Lib Cache Get Hits (LIBRARY_CACHE_GET_HITS)	Number of times objects were found in library cache[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.GETHITS)
Lib Cache Gets (LIBRARY_CACHE_GETS)	Number of locks requested for objects in library cache[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.GETS)
Lib Cache Invalidations (LIBRARY_CACHE_INVALIDATIONS)	Number of times a nonpermane nt library object (such as a shared SQL area) was invalid[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.INVALIDATIONS)
Lib Cache Miss % (LIBRARY_CACHE_MISS_PERCENTAGE)	Library cache miss percentage: The percentage of objects in the library cache that had to be reloaded into the library cache. A higher percentage indicates greater resource usage. [#2, #10]	AVG	double	No	All	(SUM(V$LIBRARYCACHE.RELOADS) / SUM(V$LIBRARYCACHE.PINS)) * 100
Lib Cache Pin Hit % (LIBRARY_CACHE_PIN_HIT_PERCENTAGE)	Ratio (as a percent) of the LIBRARY_CACHE_PIN_HITS value to the LIBRARY_CACHE_PINS value[#2, #10]	AVG	double	No	All	(SUM(V$LIBRARYCACHE.PINHITS) / SUM(V$LIBRARYCACHE.PINS)) * 100
Lib Cache Pin Hits (LIBRARY_CACHE_PIN_HITS)	Number of times an object acquired or accessed by the system had already been placed in cache and initialized[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.PINHITS)
Lib Cache Pins (LIBRARY_CACHE_PINS)	Number of times the system acquired an object in cache to access it[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.PINS)
Lib Cache Reloads (LIBRARY_CACHE_RELOADS)	Number of times the system needed to reinitialize a library object and load it along with the data because it had not been used recently or it was invalid[#2, #10]	AVG	double	Yes	All	SUM(V$LIBRARYCACHE.RELOADS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PICS)[#1]	COPY	string(4)	No	All	Agent Collector
SQL Executing (SQL_EXECUTING)	Total number of users currently executing SQL statements[#1, #9]	COPY	double	No	All	SUM(V$SQLAREA.USERS_EXECUTING)
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Total SQL Executions (TOTAL_SQL_EXECUTIONS)	Total number of users currently executing SQL statements[#1, #9]	COPY	double	Yes	All	execute count

#9

Displays performance data for the PDBs that are being monitored.

#10

Displays common performance data for database instances.

CDB Status (PD_CDCS)

Function

The CDB Status (PD_CDCS) record stores performance data related to a PDB built in a CDB, including open mode information and the amount of disk space used at a certain point in time. One record is created per PDB. This record is a multiple-instance record.

Notes

- The CDB Status (PD_CDCS) record is supported only when Oracle Database 12c R2 or later is monitored in a CDB configuration. This record cannot be collected when Oracle Database 12c R1 or earlier is monitored or when Oracle Database 12c R2 or later is monitored in a non-CDB configuration.
- When the CDB Status (PD_CDCS) record is to be monitored from the root container, you need to use an Oracle administrative account (sys or system). The CDB Status (PD_CDCS) record cannot be collected if you use a common user's account.
- The range of information collected varies depending on whether it is PDBs or the root container (CDB$ROOT) that is going to be monitored.
  - When monitoring PDBs

    Performance data regarding the monitored PDBs
  - When all PDBs are monitored from the root container (CDB$ROOT) to which they belong

    Performance data regarding all PDBs managed by the root container (CDB$ROOT)

If the PDB storage size (MAX_SIZE, MAX_DIAG_SIZE, or MAX_AUDIT_SIZE) is either not set or set to *unlimited* in the Oracle Database to be monitored, the following fields would always be 0 or 100 and therefore cannot be monitored:

Oracle item that is either not set or set to unlimited	Field name	Displayed value
MAX_SIZE	Max Size	Always 0
	Free Size	Always 0
	Free Size %	Always 100
MAX_DIAG_SIZE	Max Diagnostics Size	Always 0
	Free Diagnostics Size	Always 0
	Free Diagnostics Size %	Alway 100
MAX_AUDIT_SIZE	Max Audit Size	Always 0
	Free Audit Size	Always 0
	Free Audit Size %	Alway 100

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	0	Y
Log	No	Y

Item	Default Value	Changeable
LOGIF	(Blank)	Y

ODBC Key Fields

PD_CDCS_CON_ID

Lifetime

From the creation to the deletion of PDB

Record Size

- Fixed part: 678bytes
- Variable part: 262bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Container ID (CON_ID)	Container ID of a PDB	--	ushort	No	Oracle 12c R2 or later	V$PDBS.CON_ID
Container UID (CON_UID)	Unique identifier related to a PDB	--	ulong	No	Oracle 12c R2 or later	V$PDBS.CON_UID
DBID (DBID)	PDB identifier	--	ulong	No	Oracle 12c R2 or later	V$PDBS.DBID
Name (NAME)	PDB name	--	string(128)	No	Oracle 12c R2 or later	V$PDBS.NAME
Open Mode (OPEN_MODE)	Open mode • MOUNTED • READ WRITE • READ ONLY • MIGRATE	--	string(10)	No	Oracle 12c R2 or later	V$PDBS.OPEN_MODE
Restricted (RESTRICTED)	This field indicates YES or NO depending on whether only users with RESTRICTED SESSION privilege are allowed access.	--	string(3)	No	Oracle 12c R2 or later	V$PDBS.RESTRICTED
Total Size (TOTAL_SIZE)	Size of the disk space for data and temporary files inside a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V$PDBS.TOTAL_SIZE / (1024 * 1024)
Diagnostics Size (DIAG_SIZE)	Size of the diagnostic trace generated in a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V$PDBS.DIAGNOSTICS_SIZE / (1024 * 1024)
Audit File Size (AUDIT_SIZE)	Size of the unified audit file inside a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V$PDBS.AUDIT_FILE_SIZE / (1024 * 1024)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Max Size (MAX_SIZE)	Size of maximum space available for data and temporary files inside a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V$PDBS.MAX_SIZE / (1024 * 1024)
Max Diagnostics Size (MAX_DIAG_SIZE)	Size of maximum space available for the diagnostic trace in a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V $PDBS.MAX_DIAGNOSTICS_SIZE / (1024 * 1024)
Max Audit Size (MAX_AUDIT_SIZE)	Size of the maximum space available for the unified audit file inside a PDB in megabytes	--	double	No	Oracle 12c R2 or later	V $PDBS.MAX_AUDIT_SIZE / (1024 * 1024)
Free Size (TOTAL_FREE)	Size of free space available for data and temporary files inside a PDB in megabytes. When MAX_SIZE is 0, this field is always 0.	--	double	No	Oracle 12c R2 or later	MAX_SIZE - TOTAL_SIZE
Free Diagnostics Size (DIAG_FREE)	Size of free space available for the diagnostic trace in a PDB in megabytes. When MAX_DIAG_SIZE is 0, this field is always 0.	--	double	No	Oracle 12c R2 or later	MAX_DIAG_SIZE - DIAG_SIZE
Free Audit Size (AUDIT_FREE)	Size of free space available for the unified audit file inside a PDB in megabytes. When MAX_AUDIT_SIZE is 0, this field is always 0.	--	double	No	Oracle 12c R2 or later	MAX_AUDIT_SIZE - AUDIT_SIZE
Free Size % (PERC_FREE)	Percentage ratio of free space available for data and temporary files inside a PDB. When MAX_SIZE is 0, this field is always 100.	--	double	No	Oracle 12c R2 or later	((MAX_SIZE - TOTAL_SIZE) / MAX_SIZE) * 100
Free Diagnostics Size % (PERC_FREE_DIAG)	Percentage ratio of free space available for the diagnostic trace in a PDB. When MAX_DIAG_SIZE is 0, this field is always 100.	--	double	No	Oracle 12c R2 or later	((MAX_DIAG_SIZE - DIAG_SIZE) / MAX_DIAG_SIZE) * 100
Free Audit Size % (PERC_FREE_AUDIT)	Percentage ratio of free space available for the unified audit file inside a PDB. When MAX_AUDIT_SIZE is 0, this field is always 100.	--	double	No	Oracle 12c R2 or later	((MAX_AUDIT_SIZE - AUDIT_SIZE) / MAX_AUDIT_SIZE) * 100
Application Root (APPLICATION_ROOT)	This field indicates YES or NO depending on whether the PDB is an application root.	--	string(3)	No	Oracle 12c R2 or later	V $PDBS.APPLICATION_ROOT
Applicaton PDB (APPLICATION_PDB)	This field indicates YES or NO depending on whether	--	string(3)	No	Oracle 12c R2 or later	V $PDBS.APPLICATION_PDB

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Applicaton PDB (APPLICATION_PDB)	the PDB is an application PDB.	--	string(3)	No	Oracle 12c R2 or later	V$PDBS.APPLICATION_PDB
Application Seed (APPLICATION_SEED)	This field indicates YES or NO depending on whether the PDB is an application seed.	--	string(3)	No	Oracle 12c R2 or later	V$PDBS.APPLICATION_SEED
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	Oracle 12c R2 or later	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always CDCS)	--	string(4)	No	Oracle 12c R2 or later	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	Oracle 12c R2 or later	Agent Collector

CDB Tablespace(PD_CDTS)

Function

The CDB Tablespace (PD_CDTS) record stores performance data indicating the status concerning the size of the tablespaces of all the containers managed by the root container of the CDB at a certain point in time. One record is created per tablespace. This record is a multiple-instance record.

The CDB Tablespace (PD_CDTS) record differs from the Tablespace (PD_PDTS) record in the following respects.

Range of information collected

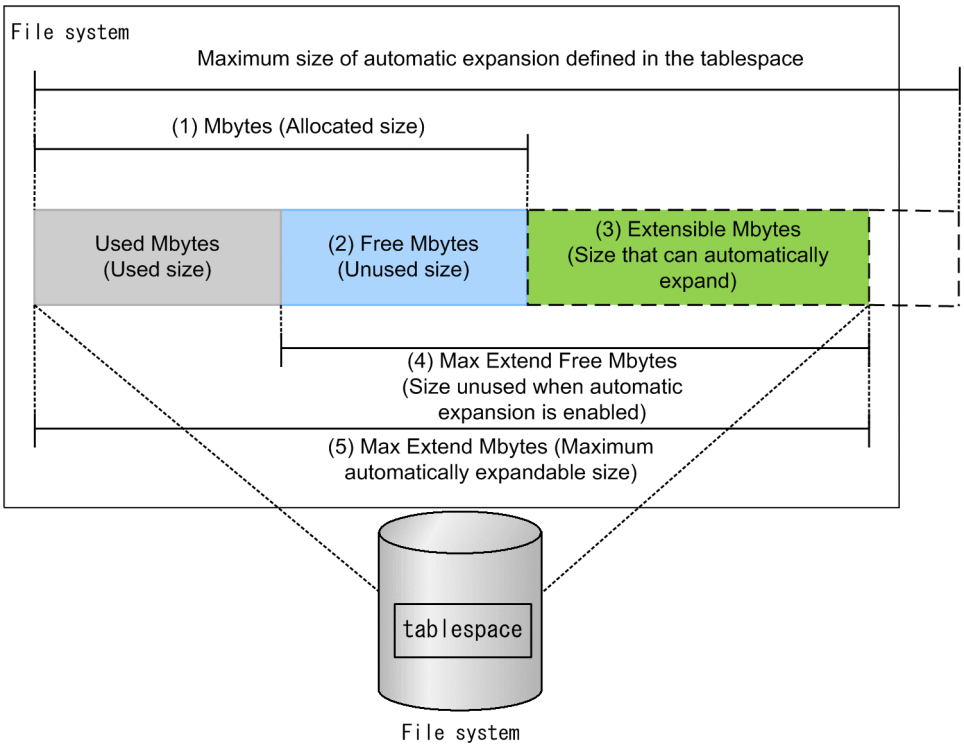
Record	Oracle Database to be monitored	Range of information collected
CDB Tablespace (CD_CDTS)	Root container (CDB$ROOT) in a CDB configuration	• When an Oracle administrative account (sys or system) is used: Performance data concerning the root container (CDB$ROOT) and all the PDBs managed by it • When a common user's account is used: Performance data concerning the root container (CDB$ROOT)
	PDBs in a CDB configuration	Performance data concerning monitored PDBs
	Non-CDB configuration	Performance data concerning the monitored Oracle Database
Tablespace (PD_PDTS)	Root container (CDB$ROOT) in a CDB configuration	Performance data concerning the root container (CDB$ROOT)
	PDBs in a CDB configuration	Performance data concerning the monitored PDBs
	Non-CDB configuration	Performance data concerning the monitored Oracle Database

Maximum size of automatic expansion

Record	Tablespace	Maximum size of automatic expansion
CDB Tablespace (CD_CDTS)	Permanent tablespace/ temporary tablespaces	The maximum size of automatic expansion defined in the tablespace or the free space of the file system, whichever is smaller
	UNDO tablespaces	The maximum size of automatic expansion defined in the tablespace
Tablespace (PD_PDTS)	All tablespaces	The maximum size of automatic expansion defined in the tablespace

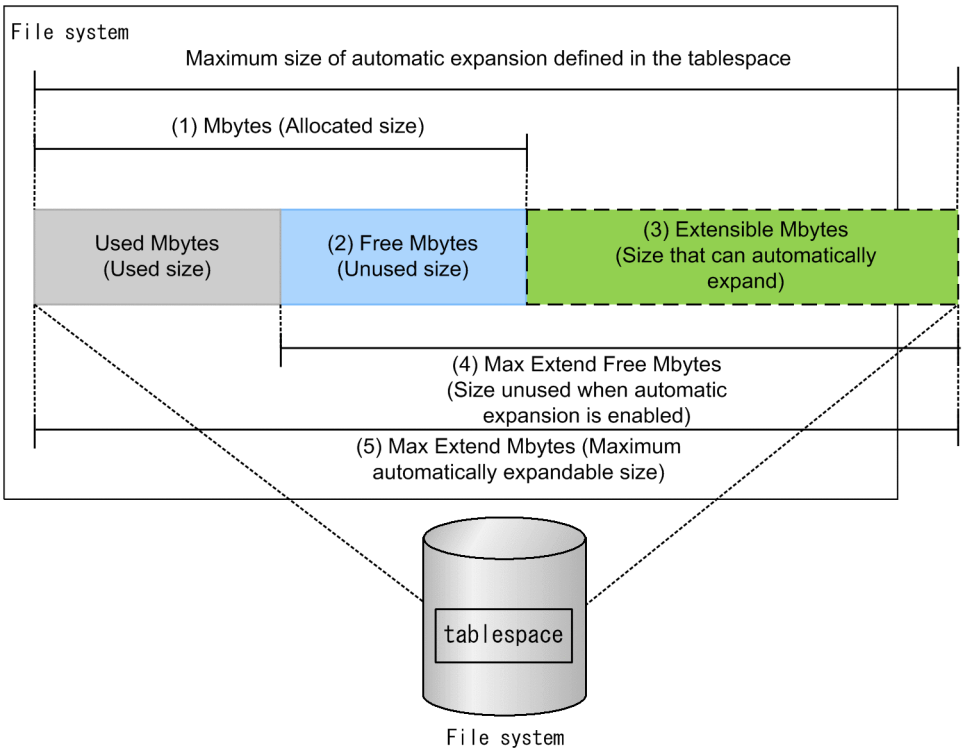
The following figure shows the relationship between the fields acquired with the CDB Tablespace (PD_CDTS) record.

Figure 6–3: Relationship between the fields acquired with tablespaces (except UNDO tablespaces)



When free space of the file system is smaller than the maximum size of automatic expansion defined in the tablespace, the value of the `Max Extend Mbytes` field is calculated based on the amount of free space. Other fields are calculated based on the value of the `Max Extend Mbytes` field.

Figure 6–4: Relationship between the fields acquired with tablespaces (UNDO tablespaces)



The maximum size of automatic expansion defined in the tablespace is displayed as the value of the Max Extend Mbytes field. If the maximum size is unlimited, the value is displayed based on the theoretical maximum size that can be set with the Oracle Database. Other fields are calculated based on the value of the Max Extend Mbytes field.

Expressions for calculating the values of fields related to the space size percentage:
The same expressions are used for all tablespaces.

- Free % (percentage of free space)
  ([Free Mbytes (2)] / [Mbytes (1)]) × 100
- Max Extend Free % (percentage of unused size in relation to the maximum size of automatic expansion)
  ([Max Extend Free Mbytes (4)] / [Max Extend Mbytes (5)]) × 100
- Extensible Mbytes % (percentage of remaining size that can automatically expand)
  ([Extensible Mbytes (3)] / [Max Extend Mbytes (5)]) × 100

The unused size (which is equivalent to setting localtemp_option to Y in instance information) is displayed as the size of a temporary tablespace. The displayed size of the UNDO tablespace varies depending on the specification of undospace_option in instance information. For more details, see the information about the effect of undospace_option provided in the *Tablespace (PD_PDTS)* record.

Notes

- The CDB Tablespace (PD_CDTS) record is supported only when Oracle Database 12c R2 or later is monitored. Information cannot be collected when Oracle Database 12c R1 or earlier is monitored.
- This record can be collected when either a CDB or non-CDB configuration is monitored.
- If a CDB configuration is monitored, the range of information collected varies depending on whether it is a PDB or the root container (CDB$ROOT) that is going to be monitored.

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y

ODBC key fields

- PD_CDTS_CON_ID
- PD_CDTS_TABLESPACE_NAME

Lifetime

From creation to deletion of a tablespace

Record size

- Fixed part: 678 bytes
- Variable part: 145 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Auto Extensible (AUTO_EXTENSIBLE)	This field indicates whether the automatic expansion of the tablespace is enabled. YES means enabled and NO means disabled. This field is set to YES when automatic expansion is enabled for any one of the data files of which the tablespace is comprised.	--	string(3)	No	Oracle 12c R2 or later	• For tablespaces other than temporary tablespaces: CDB_DATA_FILES.AUTOEXTENSIBLE • For temporary tablespaces: CDB_TEMP_FILES.AUTOEXTENSIBLE
Block Size (BLOCK_SIZE)	Block size of the tablespace	--	ulong	No	Oracle 12c R2 or later	CDB_TABLESPACES.BLOCK_SIZE
Container ID (CON_ID)	ID of the container to which the tablespace belongs	--	ushort	No	Oracle 12c R2 or later	CDB_TABLESPACES.CON_ID
Contents (CONTENTS)	Contents of the tablespace	--	string(21)	No	Oracle 12c R2 or later	CDB_TABLESPACES.CONTENTS
Extensible Mbytes (EXTENSIBLE_BYTES)	The remaining space available for automatic expansion in megabytes. This field is set to 0 when no automatic expansion is performed or when the maximum size is already reached.	--	double	No	Oracle 12c R2 or later	MAX_BYTES - BYTES
Extensible Mbytes % (PERCENT_EXTENSIBLE_BYTES)	Percentage ratio of the remaining space available for automatic expansion. This field is set to 0 when no automatic expansion is performed or when the maximum size is already reached.	--	double	No	Oracle 12c R2 or later	(EXTENSIBLE_BYTES / MAX_BYTES) * 100
Free % (PERCENT_FREE)	Percentage ratio of free space	--	double	No	Oracle 12c R2 or later	(FREE_BYTES / BYTES) * 100
Free Mbytes (FREE_BYTES)	Size of free space in megabytes	--	double	No	Oracle 12c R2 or later	• For UNDO tablespaces: CDB_DATA_FILES.BYTES - (CDB_TABLESPACE_USAGE_METRIC

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE_BYTES)	Size of free space in megabytes	--	double	No	Oracle 12c R2 or later	S.USED_SPACE * CDB_TABLESPACE.BLOCK_SIZE) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(CDB_FREE_SPACE.BYTES) + SUM(CDB_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is N: (SUM(CDB_FREE_SPACE.BYTES) / (1024 * 1024)
Max Extend Free % (MAX_PERCENT_FREE)	Percentage ratio of the unused size in relation to the maximum automatically expandable size. The value is the same as the Free % field when automatic expansion is disabled or the maximum size has already been reached.	--	double	No	Oracle 12c R2 or later	(MAX_FREE_BYTES / MAX_BYTES) * 100
Max Extend Free Mbytes (MAX_FREE_BYTES)	Size unused when automatic expansion is enabled. This size is the total of the unused size and the automatically expandable size in megabytes. The value is the same as the Free Mbytes field when automatic expansion is disabled or the maximum size has already been reached.	--	double	No	Oracle 12c R2 or later	MAX_BYTES - USED_BYTES
Max Extend Mbytes (MAX_BYTES)	Maximum automatic expandable size (in megabytes). The value is the same as	--	double	No	Oracle 12c R2 or later	• For tablespaces other than UNDO tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Max Extend Mbytes (MAX_BYTES)	the Mbytes field when automatic expansion is disabled or the maximum size has already been reached.	--	double	No	Oracle 12c R2 or later	(CDB_TABLESPACE_USAGE_METRICS.TABLESPACE_SIZE * CDB_TABLESPACES.BLOCK_SIZE) / (1024 * 1024) • For UNDO tablespaces: SUM(CDB_DATA_FILES.MAXBYTES) / (1024 * 1024)
Mbytes (BYTES)	Size of the tablespace in megabytes	--	double	No	Oracle 12c R2 or later	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(CDB_DATA_FILES.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces: SUM(CDB_TEMP_FILES.BYTES) / (1024 * 1024)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	Oracle 12c R2 or later	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always CDTS)	--	string(4)	No	Oracle 12c R2 or later	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	Oracle 12c R2 or later	Agent Collector
Status (STATUS)	Tablespace status. Valid values are INVALID (tablespace was deleted), OFFLINE, ONLINE, and READ ONLY.	--	string(9)	No	Oracle 12c R2 or later	CDB_TABLESPACES.STATUS
Tablespace Name (TABLESPACE_NAME)	Tablespace name	--	string(30)	No	Oracle 12c R2 or later	CDB_TABLESPACES.TABLESPACE_NAME

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Mbytes (USED_BYTES)	Size of used area in megabytes.#4	--	double	No	Oracle 12c R2 or later	• For tablespaces other than UNDO tablespaces: (CDB_TABLESPACE_USAGE_METRICS.USED_SPACE * CDB_TABLESPACES.BLOCK_SIZE) / (1024 * 1024) • For UNDO tablespaces: BYTES - FREE_BYTES

Circuit (PD_PDCI)

Function

The Circuit (PD_PDCI) record stores performance data indicating the status of circuits at a specific point in time. To collect this record, you must have a multi-thread server (MTS) configuration.

PFM - Agent for Oracle creates one record for each circuit of an instance. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	20	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

PD_PDCI_CIRCUIT

Lifetime

From the start to the stop of an Oracle instance

Record Size

- Fixed part: 678 bytes
- Variable part: 138 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Breaks (BREAKS)	Number of circuit breaks (pauses)	--	double	No	All	V$CIRCUIT.BREAKS
Bytes (BYTES)	Number of bytes that passed through the circuit	--	double	No	All	V$CIRCUIT.BYTES
Circuit (CIRCUIT)	Circuit address	--	string(16)	No	All	V$CIRCUIT.CIRCUIT
Circuit Server (SERVER)	Correct values cannot be collected for this field. Current server's process address	--	ulong	No	All	V$CIRCUIT.SERVER
Dispatcher (DISPATCHER)	Correct values cannot be collected for this field. Current dispatcher's process address	--	ulong	No	All	V$CIRCUIT.DISPATCHER

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Message0 (MESSAGE0)	Size of message in the first message buffer in bytes	--	double	No	All	V$CIRCUIT.MESSAGE0
Message1 (MESSAGE1)	Size of message in the second message buffer in bytes	--	double	No	All	V$CIRCUIT.MESSAGE1
Messages (MESSAGES)	Number of messages that passed through the circuit	--	double	No	All	V$CIRCUIT.MESSAGES
Queue (QUEUE)	Current queue with circuit. This field contains one of the following values: COMMON: Waiting for server process in common queue DISPATCHER: Waiting for dispatcher SERVER: Being processed OUTBOUND: Waiting for outbound connection NONE: Idle circuit	--	string(16)	No	All	V$CIRCUIT.QUEUE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDCI)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID bound to the circuit	--	ulong	No	All	V$SESSION.SID where V$CIRCUIT.SADDR = V$SESSION.SADDR
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Circuit status: BREAK: Pause EOF: Immediately before erasure OUTBOUND: External link to remote database NORMAL: Normal circuit to local database	--	string(16)	No	All	V$CIRCUIT.STATUS
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME
Waiter (WAITER)	Correct values cannot be collected for this field. Address of the server process waiting for a circuit that is currently busy.	--	ulong	No	All	V$CIRCUIT.WAITER

Collection Instance 2 (PD_PCI)

Function

The Collection Instance 2 (PD_PCI) record stores performance data indicating the status (at a specific point in time) of an instance.

Notes

This record collects the status of the fast recovery area managed by the Oracle Database.

If Oracle Database 12c R2 or later is monitored in a CDB configuration, the same values are collected regardless of whether it is the root container (CDB$ROOT) or the PDBs that are being monitored.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	600	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record Size

- Fixed part: 1,609 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Host (HOST)	Instance's host machine name	--	string(64)	No	All	V$INSTANCE.HOST_NAME
ORACLE_HOME (ORACLE_HOME)	ORACLE_HOME environment variable	--	string(255)	No	All	--
ORACLE_SID (ORACLE_SID)	ORACLE_SID environment variable	--	string(30)	No	All	--
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Record Type (INPUT_RECORD_TYPE)	Record name (always PCI)	--	string(4)	No	All	Agent Collector
Recovery File Dest (RECOVERY_FILE_DEST)	Location of the fast recovery area	--	string(513)	No	All	V$RECOVERY_FILE_DEST.NAME
Recovery Space Limit (RECOVERY_LIMIT)	Maximum size of the fast recovery area in megabytes	--	double	No	All	V$RECOVERY_FILE_DEST.SPACE_LIMIT / (1024 * 1024)
Recovery Space Used (RECOVERY_USED)	Used space of the fast recovery area in megabytes	--	double	No	All	V$RECOVERY_FILE_DEST.SPACE_USED / (1024 * 1024)
Recovery Space Reclaimable (RECOVERY_RECLAIMABLE)	Size of obsolete, redundant, and other low priority files from the fast recovery area in megabytes	--	double	No	All	V$RECOVERY_FILE_DEST.SPACE_RECLAIMABLE / (1024 * 1024)
Recovery File Nums (RECOVERY_FILES)	Number of files in the fast recovery area	--	ulong	No	All	V$RECOVERY_FILE_DEST.NUMBER_OF_FILES
Recovery Space Free Mbytes (RECOVERY_FREE_MB)	Free space in the fast recovery area in megabytes. When RECOVERY_LIMIT is 0, 0 is displayed.	--	double	No	All	RECOVERY_LIMIT - RECOVERY_USED
Recovery Space Free % (RECOVERY_FREE_RATIO)	Percentage of free space in the fast recovery area. When RECOVERY_LIMIT is 0, 100 is displayed.	--	double	No	All	((RECOVERY_LIMIT - RECOVERY_USED) / RECOVERY_LIMIT) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Version (VERSION)	Version of Oracle database 0 is set in lower than revision, if the monitoring target is Oracle Database 18c or later.	--	string(20)	No	All	PRODUCT_COMPONENT_VERSION

Collection Tablespace 2(PD_PCTS)

Function

The Collection Tablespace 2 (PD_PCTS) record stores performance data indicating the status (at a specific point in time) of tablespaces in a database. PFM - Agent for Oracle creates one record for each transaction. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	3600	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

PD_PCTS_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a tablespace

Record Size

- Fixed part: 678 bytes
- Variable part: 47 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Free Mbytes (FREE_BYTES)	Remaining free space in megabytes.	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For locally managed temporary tablespace when the value of

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Free Mbytes (FREE_BYTES)	Remaining free space in megabytes.	--	double	No	All	localtemp_option is Y: SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES) / (1024 * 1024) • For locally managed temporary tablespace when the value of localtemp_option is N: SUM(V$TEMP_SPACE_HEADER.BYTES_FREE)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Mbytes (BYTES)	Size of tablespace in megabytes	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_DATA_FILES.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces: SUM(DBA_TEMP_FILES.BYTES) / (1024 * 1024)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PCTS)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Tablespace name	--	string(30)	No	All	DBA_TABLESPACES.TABLESPACE_NAME

Control File (PD_PDCF)

Function

The Control File (PD_PDCF) record stores performance data indicating the status of a control file at a specific point in time.

PFM - Agent for Oracle creates one record for each control file. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	15	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDCF_NAME

Lifetime

From the creation to the deletion of a database

Record size

- Fixed part: 678 bytes
- Variable part: 522 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
File Name (NAME)	Name of the control file	--	string(513)	No	All	V$CONTROLFILE.NAME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDCF)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Status of control file: INVALID: Name cannot be identified	--	string(7)	No	All	V$CONTROLFILE.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	VALID: Name can be identified	--	string(7)	No	All	V$CONTROLFILE.STATUS

Current Sessions Stat Summary (PD_PDS3)

Function

The Current Sessions Stat Summary (PD_PDS3) record stores performance data indicating the status of all active sessions for an instance at a specific point in time.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record displays information about sessions of the PDBs that are being monitored, and about common sessions of database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	110	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 986 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes/Tran (BLOCK_CHANGES_PER_TRANSACTION)	Rate at which each transaction executed database manipulation language (DML) statements	--	double	No	All	db block changes / user commits
Block Visits/Tran (BLOCK_VISITS_PER_TRANSACTION)	Number of times a work database was loaded per transaction	--	double	No	All	(db block gets + consistent gets) / user commits
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage	--	double	No	All	(1 - (physical reads cache / (consistent gets from cache + db block gets from cache))) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Calls/Tran (CALLS_PER_TRANSACTION)	Rate at which client requests were executed per transaction	--	double	No	All	user calls / user commits
Changed Block % (CHANGED_BLOCK_PERCENTAGE)	Ratio (as a percent) of the number of queries that change data (insertion, update, and deletion) to the number of queries (search, insertion, update, deletion) executed on the database	--	double	No	All	(db block changes / (block gets + consistent gets)) * 100
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Ratio (as a percent) of the number of times rollback entries were applied to maintain read consistency to the number of read consistency requests by the application	--	double	No	All	(consistent changes / consistent gets) * 100
Continued Row % (CONTINUED_ROW_PERCENTAGE)	Percentage of fetched rows that were longer than one block or were moved	--	double	No	All	(table fetch continued row / (table fetch by rowid + table scan rows gotten)) * 100
Deadlocks (LOCK_DEADLOCKS)	Number of process deadlocks caused by DML enqueuing (locking) in active session	--	double	No	All	SUM(V$SESSTAT.VALUE)
Disk Sorts (SORTS_DISK)	Number of disk sort operations	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Conversions (LOCK_CONVERSIONS)	Number of enqueues (locks) whose mode changed (such as from share to lock)	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Releases (LOCK_RELEASES)	Number of times enqueuing (locking) was released by active session (this statistic is the same as the number of lock requests)	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Requests (LOCK_REQUESTS)	Number of times enqueuing (locking) was requested by active session	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Timeouts (LOCK_TIMEOUTS)	Number of times enqueuing (locking) request was not permitted by the active session within the specified wait time	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Waits (LOCK_WAITS)	Number of times lock request was placed in wait status by active session (the number of lock requests that were not placed in wait status equals the number of lock requests minus the number of enqueued waits)	--	double	No	All	SUM(V$SESSTAT.VALUE)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Logical Reads (LOGICAL_READS)	Sum of the number of logical read operations in read consistency mode and the number of requests to the current copy of the block	--	double	No	All	db block gets + consistent gets
Non-Index Lookups % (NON_INDEX_LOOKUPS)	Percentage of full table scans where no caching is performed	--	double	No	All	(table scans (long tables) / (table scans (short tables) + table scans (long tables))) * 100
Physical Reads (PHYSICAL_READS)	Number of times database block was actually read from disk by active session	--	double	No	All	physical reads - physical reads direct - physical reads direct (lob)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations on disk by DBWR	--	double	No	All	SUM(V$SESSTAT.VALUE)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDS3)	--	string(4)	No	All	Agent Collector
Recursive Calls (RECURSIVE_CALLS)	Number of user calls processed	--	double	No	All	SUM(V$SESSTAT.VALUE)
Recursive to User Call % (RECURSIVE_TO_USER_CALL_PERCENTAGE)	Correct values cannot be collected for this field. Ratio (as a percent) of recursive calls to user calls	--	double	No	All	(recursive calls / user calls) * 100
Redo Log Space Requests (REDO_LOG_SPACE_REQUESTS)	Number of times that, because the active log file was full, the Oracle server had to wait for disk space to be allocated for a REDO log entry.	--	double	No	All	SUM(V$SESSTAT.VALUE)
Redo Log Space Wait % (REDO_LOG_SPACE_WAIT_PERCENTAGE)	Wait rate for allocations of the disk area to the REDO log entry	--	double	No	All	(redo log space requests / redo entries) * 100
Row Source % (ROW_SOURCE_PERCENTAGE)	Percentage of all rows that were obtained by full table	--	double	No	All	(table scan rows gotten / (table fetch by rowid + table scan rows gotten)) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
SQL Net Bytes Rcvd (SQL_NET_BYTES_RECEIVED)	Number of bytes the active sessions received from clients via SQL*Net	--	double	No	All	SUM(V$SESSTAT.VALUE)
SQL Net Bytes Sent (SQL_NET_BYTES_SENT)	Number of bytes the active sessions sent to clients via SQL*Net	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session CPU Usage (SESSION_CPU_USAGE)	Total CPU time in hundredths of a second used to execute statements during data collection	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	Number of session cursors cached (the maximum number of cursors that can be cached is determined by the SESSION_CACHED_CURSORS parameter in the init.ora file)	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session Cursor Cache Hit % (SESSION_CURSOR_CACHE_HIT_PERCENTAGE)	Percentage of session cursors that were reused	--	double	No	All	(session cursor cache hits / session cursor cache count) * 100
Session Cursor Cache Hits (SESSION_CURSOR_CACHE_HITS)	Number of times cached session cursor was reused	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session PGA Memory (SESSION_PGA_MEMORY)	Size of PGA memory used by active session during data collection (in bytes)	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session UGA Memory (SESSION_UGA_MEMORY)	Size of UGA memory used by active session (in bytes)	--	double	No	All	SUM(V$SESSTAT.VALUE)
Sessions (SESSIONS)	Number of sessions during data collection	--	ulong	No	All	COUNT(V$SESSION)
Sort Overflow % (SORT_OVERFLOW_PERCENTAGE)	Percentage of sorts that used temporary segments	--	double	No	All	(sorts (disk) / (sorts (memory) + sorts (disk))) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
User Calls (USER_CALLS)	Number of user calls processed by active sessions	--	double	No	All	SUM(V$SESSTAT.VALUE)
User Calls / Parse (USER_CALLS_PER_PARSE)	Percentage indicating how well the application is managing the context area	--	double	No	All	user calls / parse count (total)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
User Commits (USER_COMMITS)	Number of transactions by active session	--	double	No	All	SUM(V$SESSTAT.VALUE)
User Rollback % (USER_ROLLBACK_PERCENTAGE)	Percentage of application transactions that failed (were rolledback)	--	double	No	All	(user rollbacks / (user commits + user rollbacks)) * 100
User Rollbacks (USER_ROLLBACKS)	Number of rollbacks by active session	--	double	No	All	SUM(V$SESSTAT.VALUE)
Write % (WRITE_PERCENTAGE)	Ratio (as a percent) of physical writes to all physical I/Os (reads and writes)	--	double	No	All	(physical writes / (physical reads + physical writes)) * 100

Data Dictionary Cache (PD_PDDD)

Function

The Data Dictionary Cache (PD_PDDD) record stores performance data indicating the usage of the data dictionary cache at a specific point in time. PFM - Agent for Oracle creates one record for each data dictionary cache. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDDD_CACHE_NUM
- PD_PDDD_SUBORDINATE_NUM

Lifetime

From the start to the stop of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 145 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cache # (CACHE_NUM)	ID number of row cache	--	double	No	All	V$ROWCACHE.CACHE#
Count (COUNT)	Total number of entries in cache	--	long	No	All	V$ROWCACHE.COUNT
Fixed (FIXED)	Number of entries fixed in cache	--	ulong	No	All	V$ROWCACHE.FIXED
Flushes (FLUSHES)	Number of times cache was flushed to disk	--	double	No	All	V $ROWCACHE.FLUSHES
Get Misses (GET_MISSES)	Number of data requests for which a cache miss occurred	--	double	No	All	V $ROWCACHE.GETMISSES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Get Misses % (GET_MISSES_PERCENTAGE)	Percentage of data requests for which a cache miss occurred during the interval	--	double	No	All	(V$ROWCACHE.GETMISSES / V$ROWCACHE.GETS) * 100
Gets (GETS)	Total number of requests to data object information	--	double	No	All	V$ROWCACHE.GETS
Modifications (MODIFICATIONS)	Number of insert, update, and delete operations	--	double	No	All	V$ROWCACHE.MODIFICATIONS
Parameter (PARAMETER)	Name of init.ora parameter that determines the number of entries in data dictionary cache	--	string(32)	No	All	V$ROWCACHE.PARAMETER
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDDD)	--	string(4)	No	All	Agent Collector
Scan Completes (SCAN_COMPLETES)	Number of times a list of subordinate entries was scanned completely	--	double	No	All	V$ROWCACHE.SCANCOMPLETES
Scan Misses (SCAN_MISSES)	Number of scans during which data was not found in cache	--	double	No	All	V$ROWCACHE.SCANMISSES
Scan Misses % (SCAN_MISSES_PERCENTAGE)	Percentage of scans in which data was not found in cache	--	double	No	All	(V$ROWCACHE.SCANMISSES / V$ROWCACHE.SCANS) * 100
Scans (SCANS)	Number of scan requests	--	double	No	All	V$ROWCACHE.SCANS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Subordinate # (SUBORDINATE_NUM)	Subordinate set number	--	double	No	All	V$ROWCACHE.SUBORDINATE#
Type (TYPE)	Type of master or subordinate row cache	--	string(11)	No	All	V$ROWCACHE.TYPE
Usage (USAGE)	Number of cache entries containing valid data	--	ulong	No	All	V$ROWCACHE.USAGE

Data Dictionary Cache Interval (PI_PIDD)

Function

data, taken at specific intervals, about usage of the data dictionary cache. PFM - Agent for Oracle creates one record for each data dictionary cache. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PIDD_CACHE_NUM
- PI_PIDD_SUBORDINATE_NUM

Lifetime

From the start to the stop of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 289 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cache # (CACHE_NUM)	ID number of row cache[#1]	COPY	double	No	All	V$ROWCACHE.CACHE#
Count (COUNT)	Total number of entries in cache[#2]	AVG	long	No	All	V$ROWCACHE.COUNT
Fixed (FIXED)	Number of entries fixed in cache[#2]	AVG	ulong	No	All	V$ROWCACHE.FIXED
Flushes (FLUSHES)	Number of times cache was flushed to disk[#2]	AVG	double	Yes	All	V $ROWCACHE.FLUSHES
Get Misses (GET_MISSES)	Number of data requests for which a cache miss occurred[#2]	AVG	double	Yes	All	V $ROWCACHE.GETMISSES
Get Misses %	Percentage of data requests for which a cache miss	AVG	double	No	All	(V $ROWCACHE.GETMISS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(GET_MISSES_PERCENTAGE)	occurred during the interval[#2]	AVG	double	No	All	ES / V $ROWCACHE.GETS) * 100
Gets (GETS)	Total number of requests to data object information[#2]	AVG	double	Yes	All	V$ROWCACHE.GETS
Modifications (MODIFICATIONS)	Number of insert, update, and delete operations[#2]	AVG	double	Yes	All	V $ROWCACHE.MODIFICATIONS
Parameter (PARAMETER)	Name of init.ora parameter that determines the number of entries in data dictionary cache[#1]	COPY	string(32)	No	All	V $ROWCACHE.PARAMETER
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIDD)[#1]	COPY	string(4)	No	All	Agent Collector
Scan Completes (SCAN_COMPLETES)	Number of times a list of subordinate entries was scanned completely[#2]	AVG	double	Yes	All	V $ROWCACHE.SCANCOMPLETES
Scan Misses (SCAN_MISSES)	Number of scans during which data was not found in cache[#2]	AVG	double	Yes	All	V $ROWCACHE.SCANMISSES
Scan Misses % (SCAN_MISSES_PERCENTAGE)	Percentage of scans in which data was not found in cache[#2]	AVG	double	No	All	(V $ROWCACHE.SCANMISSES / V $ROWCACHE.SCANS) * 100
Scans (SCANS)	Number of scan requests[#2]	AVG	double	Yes	All	V$ROWCACHE.SCANS
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Subordinate # (SUBORDINATE_NUM)	Subordinate set number[#1]	COPY	double	No	All	V $ROWCACHE.SUBORDINATE#
Type (TYPE)	Type of master or subordinate row cache[#1]	COPY	string(11)	No	All	V$ROWCACHE.TYPE
Usage (USAGE)	Number of cache entries containing valid data[#2]	AVG	ulong	No	All	V$ROWCACHE.USAGE

Data File (PD_PDDF)

Function

The Data File (PD_PDDF) record stores performance data indicating the status (at a specific point in time) of data files. PFM - Agent for Oracle creates one record for each data file in the database. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	10	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDDF_FILE_NUM
- PD_PDDF_NAME

Lifetime

From the creation to the deletion of a data file

Record size

- Fixed part: 678 bytes
- Variable part: 672 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Oracle block size	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_DATA_FILES.BLOCKS • For locally managed temporary tablespaces: DBA_TEMP_FILES.BLOCKS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Checkpoint Change # (CHECKPOINT_CHANGE_NUM)	System change number (SCN) at the last checkpoint	--	double	No	All	V $DATAFILE.CHECKPOINT_CHANGE#
Enabled (ENABLED)	This field contains one of the following values as the method for accessing a file using SQL: DISABLED READ ONLY READ WRITE UNKNOWN	--	string(10)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.ENABLED • For locally managed temporary tablespaces: V $TEMPFILE.ENABLED
File # (FILE_NUM)	File identification number	--	long	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.FILE# • For locally managed temporary tablespaces: V $TEMPFILE.FILE#
File Name (NAME)	File name	--	string(513)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.NAME • For locally managed temporary tablespaces: V $TEMPFILE.NAME
Free % (PERCENT_FREE)	Percentage ratio of free space	--	double	No	All	• For dictionary managed permanent tablespaces, locally

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free % (PERCENT_FREE)	Percentage ratio of free space	--	double	No	All	managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_FREE_SPACE.BYTES) / V$DATAFILE.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMP_SPACE_HEADER / DBA_TEMP_FILES.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is Y: ((V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED) / V$TEMPFILE.BYTES) * 100 • For the UNDO tablespaces when the value of undospace_option is Y: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / V$DATAFILE.BYTES) * 100
Free Mbytes (FREE)	Size of free space in megabytes	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE)	Size of free space in megabytes	--	double	No	All	temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
MBytes (BYTES)	Disk space in megabytes required on the file system	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$DATAFILE.BYTES / (1024 * 1024)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
MBytes (BYTES)	Disk space in megabytes required on the file system	--	double	No	All	• For locally managed temporary tablespaces: V$TEMPFILE.BYTES / (1024 * 1024)
Physical Blocks Read (PHYSICAL_BLOCKS_READ)	Number of physical block read operations	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYBLKRD • For locally managed temporary tablespaces: V$TEMPSTAT.PHYBLKRD
Physical Blocks Written (PHYSICAL_BLOCKS_WRITTEN)	Number of physical block write operations	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYBLKWRT • For locally managed temporary tablespaces: V$TEMPSTAT.PHYBLKWRT
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYRDS • For locally managed temporary tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed	--	double	No	All	V$TEMPSTAT.PHYRDS
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYWRTS • For locally managed temporary tablespaces: V$TEMPSTAT.PHYWRTS
Read Time (READ_TIME)	Read operation time. In centiseconds (1/100 of a second).	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.READTIM • For locally managed temporary tablespaces: V$TEMPSTAT.READTIM
Read Time (READ_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Time (RECORD_TIME)	Record name (always PDDF)	--	string(4)	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	File type (system file or user file) and file status (OFFLINE, SYSOFF, ONLINE, SYSTEM, or RECOVER)	--	string(7)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	File type (system file or user file) and file status (OFFLINE, SYSOFF, ONLINE, SYSTEM, or RECOVER)	--	string(7)	No	All	V$DATAFILE.STATUS • For locally managed temporary tablespaces: V$TEMPFILE.STATUS
Tablespace Name (TABLESPACE_NAME)	Tablespace name associated with the file	--	string(30)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_DATA_FILES.TABLESPACE_NAME • For locally managed temporary tablespaces: DBA_TEMP_FILES.TABLESPACE_NAME
Used Mbytes (USED)	Size of used area in megabytes	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (V$DATAFILE.BYTES - SUM(DBA_FREE_SPACE.BYTES)) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMPFILE.BYTES - V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Mbytes (USED)	Size of used area in megabytes	--	double	No	All	• For locally managed temporary tablespaces when the value of localtemp_option is Y: (V $TEMPFILE.BYTES - (V $TEMPFILE.BYTES - V $TEMP_EXTENT_POOL.BYTES_USED)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (V $DATAFILE.BYTES - SUM(DBA_FREE_SPACE.BYTES) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: (V $FILESTAT.PHYWRTS / (V $FILESTAT.PHYRDS + V $FILESTAT.PHYWRTS)) * 100 • For locally managed temporary tablespaces: ((V $TEMPSTAT.PHYWRTS / (V $TEMPSTAT.PHYRDS + V $TEMPSTAT.PHYWRTS)) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Write Time (WRITE_TIME)	Write operation time. In centiseconds (1/100 of a second).	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.WRITETIM • For locally managed temporary tablespaces: V$TEMPSTAT.WRITETIM

Data File Interval (PI_PIDF)

Function

The Data File Interval (PI_PIDF) record stores performance data, taken at specific intervals, about data files. PFM - Agent for Oracle creates one record for each data file in the database. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PIDF_FILE_NUM
- PI_PIDF_NAME

Lifetime

From the creation to the deletion of a data file

Record size

- Fixed part: 678 bytes
- Variable part: 892 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Oracle block size #1	COPY	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_DATA_FILES.BLOCKS • For locally managed temporary tablespaces: DBA_TEMP_FILES.BLOCKS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Checkpoint Change # (CHECKPOINT_CHANGE_NUM)	System change number (SCN) at the last checkpoint #1	COPY	double	No	All	V $DATAFILE.CHECKPOINT_CHANGE#
Enabled (ENABLED)	This field contains one of the following values as the method for accessing a file using SQL: #1 DISABLED READ ONLY READ WRITE UNKNOWN	COPY	string(10)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.ENABLED • For locally managed temporary tablespaces: V $TEMPFILE.ENABLED
File # (FILE_NUM)	File identification number #1	COPY	long	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.FILE# • For locally managed temporary tablespaces: V $TEMPFILE.FILE#
File Name (NAME)	File name #1	COPY	string(513)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.NAME • For locally managed temporary tablespaces: V $TEMPFILE.NAME
Free % (PERCENT_FREE)	Percentage ratio of free space #2	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free % (PERCENT_FREE)	Percentage ratio of free space [#2]	AVG	double	No	All	managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_FREE_SPACE.BYTES) / V$DATAFILE.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMP_SPACE_HEADER / V$TEMPFILE.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is Y: ((V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED) / V$TEMPFILE.BYTES) * 100 • For the UNDO tablespaces when the value of undospace_option is Y: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / V$DATAFILE.BYTES) * 100
Free Change (FREE_CHANGE)	Change to the free space (the difference between the value obtained this time and the value obtained the last time) in megabytes[#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Change (FREE_CHANGE)	Change to the free space (the difference between the value obtained this time and the value obtained the last time) in megabytes[#2]	AVG	double	No	All	temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Free Mbytes (FREE_BYTES)	Size of free space in megabytes [#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE_BYTES)	Size of free space in megabytes [#2]	AVG	double	No	All	SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
I/O Ops/sec (IO_RATE)	Number of I/O operations per second[#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: (V$FILESTAT.PHYRDS + V$FILESTAT.PHYWRTS) / seconds in interval • For locally managed temporary tablespaces: (V$TEMPSTAT.PHYRDS + V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
I/O Ops/sec (IO_RATE)	Number of I/O operations per second[#2]	AVG	double	No	All	$TEMPSTAT.PHYWRTS) / seconds in interval
Mbytes (BYTES)	Disk space in megabytes required on the file system[#1]	COPY	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $DATAFILE.BYTES / (1024 * 1024) • For locally managed temporary tablespaces: V $TEMPFILE.BYTES / (1024 * 1024)
Physical Blocks Read (PHYSICAL_BLOCKS_READ)	Number of physical block write operations during the interval[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $FILESTAT.PHYBLKRD • For locally managed temporary tablespaces: V $TEMPSTAT.PHYBLKRD
Physical Blocks Written (PHYSICAL_BLOCKS_WRITTEN)	Number of physical block read operations during the interval[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $FILESTAT.PHYBLKWRT • For locally managed temporary tablespaces: V $TEMPSTAT.PHYBLKWRT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Physical Reads (PHYSICAL_READS)	Number of physical block read operations that were completed during the interval[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYRDS • For locally managed temporary tablespaces: V$TEMPSTAT.PHYRDS
Physical Writes (PHYSICAL_WRITES)	Number of physical block write operations that were completed during the interval[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYWRTS • For locally managed temporary tablespaces: V$TEMPSTAT.PHYWRTS
Read Time (READ_TIME)	If the value of the TIMED_STATISTICS parameter in the init.ora file is TRUE, this field indicates the read operation time during the interval. If the parameter value is FALSE, this field contains 0. In centiseconds (1/100 of a second).[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.READTIM • For locally managed temporary tablespaces: V$TEMPSTAT.READTIM
Reads/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Reads/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYRDS / seconds in interval • For locally managed temporary tablespaces: V$TEMPSTAT.PHYRDS / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIDF)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	File type (system file or user file) and file status (OFFLINE, SYSOFF, ONLINE, SYSTEM, or RECOVER)[#1]	COPY	string(7)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces:V$DATAFILE.STATUS • For locally managed temporary tablespaces: V$TEMPFILE.STATUS
Tablespace Name (TABLESPACE_NAME)	Tablespace name associated with the file [#1]	COPY	string(30)	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_DATA_FILES.TABLESPACE_NAME • For locally managed temporary tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Tablespace Name (TABLESPACE_NAME)	Tablespace name associated with the file [#1]	COPY	string(30)	No	All	DBA_TEMP_FILES.TABLESPACE_NAME
Used Change (USED_CHANGE)	Size of used area (the difference between the value obtained this time and the value obtained the last time) in megabytes [#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (V$DATAFILE.BYTES - SUM(DBA_FREE_SPACE.BYTES)) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMPFILE.BYTES - V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$TEMPFILE.BYTES - (V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (V$DATAFILE.BYTES - SUM(DBA_FREE_SPACE.BYTES) - SUM(DBA_UNDO_E

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Change (USED_CHANGE)	Size of used area (the difference between the value obtained this time and the value obtained the last time) in megabytes #2	AVG	double	No	All	XTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Used Mbytes (USED_BYTES)	Change to the used space in megabytes. #2	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (V$DATAFILE.BYTES - SUM(DBA_FREE_SPACE.BYTES)) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (V$TEMPFILE.BYTES - V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$TEMPFILE.BYTES - (V$TEMPFILE.BYTES - V$TEMP_EXTENT_POOL.BYTES_USED)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (V$DATAFILE.BYTES - SUM(DBA_FREE_S

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Mbytes (USED_BYTES)	Change to the used space in megabytes. #2	AVG	double	No	All	PACE.BYTES) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations #2	AVG	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: (V$FILESTAT.PHYWRTS / (V$FILESTAT.PHYRDS + V$FILESTAT.PHYWRTS)) * 100 • For locally managed temporary tablespaces: ((V$TEMPSTAT.PHYWRTS / (V$TEMPSTAT.PHYRDS + V$TEMPSTAT.PHYWRTS)) * 100
Write Time (WRITE_TIME)	If the value of the TIMED_STATISTICS parameter in the init.ora file is TRUE, this field indicates the write operation time during the interval. If the parameter value is FALSE, this field contains 0. In centiseconds (1/100 of a second).#2	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.WRITETIM • For locally managed temporary tablespaces: V$TEMPSTAT.WRITETIM
Writes/sec (WRITES_RATE)	Number of write operations per second#2	AVG	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Writes/sec (WRITES_RATE)	Number of write operations per second[#2]	AVG	double	No	All	V$FILESTAT.PHYWRTS / seconds in interval • For locally managed temporary tablespaces: V$TEMPSTAT.PHYWRTS / seconds in interval

Database (PD_PDDB)

Function

The Database (PD_PDDB) record stores performance data, taken at a specific point in time, indicating the following:

- General information about a database
- Statistical information on tablespaces
- Statistical information on data files

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. For details, see the descriptions of each field.

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	20	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of a data file

Record size

- Fixed part: 914 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Archive Change # (ARCHIVE_CHANGE_NUM)	Last archived system change number (SCN)#10	--	double	No	All	V$DATABASE.ARCHIVE_CHANGE#
Blocks (BLOCKS)	Size of tablespace in Oracle blocks#8	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BLOCKS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Size of tablespace in Oracle blocks[#8]	--	double	No	All	• For Oracle that has locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BLOCKS) + SUM(DBA_TEMP_FILES.BLOCKS)
Checkpoint Change # (CHECKPOINT_CHANGE_NUM)	System change number (SCN) at the last checkpoint[#10]	--	double	No	All	V$DATABASE.CHECKPOINT_CHANGE#
Created (CREATED)	Creation date[#10]	--	string(20)	No	All	V$DATABASE.CREATED
DB Files % (PERCENT_DB_FILES)	Percentage ratio of the data files to the DB_FILES parameter value in the init.ora file[#8]	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: (COUNT(V$DATAFILE) / init.ora DB_FILES) * 100 • For Oracle that has locally managed temporary tablespaces: ((COUNT(V$DATAFILE) + COUNT(DBA_TEMP_FILES)) / init.ora DB_FILES) * 100
DB Name (NAME)	Database name[#10]	--	string(9)	No	All	V$DATABASE.NAME
Datafiles (DATAFILES)	Number of data files[#8]	--	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(V$DATAFILE) • For Oracle that has locally managed temporary tablespaces: COUNT(V$DATAFILE) + COUNT(DBA_TEMP_FILES)
Extents (EXTENTS)	Correct values cannot be collected in this field. Number of extents.	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Extents (EXTENTS)	Correct values cannot be collected in this field. Number of extents.	--	double	No	All	SUM(DBA_SEGMENTS.EXTENTS) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: SUM(DBA_SEGMENTS.EXTENTS) + SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: SUM(DBA_SEGMENTS.EXTENTS) + SUM(V$SORT_SEGMENT.TOTAL_EXTENTS)
Free % (PERCENT_FREE)	Percentage ratio of free space[#8]	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces or has UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_FREE_SPACE.BYTES) / DBA_DATA_FILES.BYTES) * 100 • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / (DBA_DATA_FILES.BYTES + DBA_TEMP_FILES.BYTES)) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free % (PERCENT_FREE)	Percentage ratio of free space[#8]	--	double	No	All	• For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(V$TEMP_SPACE_HEADER.BYTES_FREE)) / (DBA_DATA_FILES.BYTES + DBA_TEMP_FILES.BYTES)) * 100 • For the UNDO tablespaces when the value of undospace_option is Y: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / DBA_DATA_FILES.BYTES) * 100
Free Extents (FREE_EXTENTS)	Number of free extents[#8]	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_FREE_SPACE) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: COUNT(DBA_FREE_SPACE) + SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) - V$SORT_SEGMENT.USED_EXTENTS • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Extents (FREE_EXTENTS)	Number of free extents[#8]	--	double	No	All	COUNT(DBA_FREE_SPACE) + COUNT(V$TEMP_SPACE_HEADER)
Free Mbytes (FREE_BYTES)	Size of free space in megabytes[#8]	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: (SUM(DBA_FREE_SPACE.BYTES) + SUM(V$TEMP_SPACE_HEADER.BYTES_FREE)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE_BYTES)	Size of free space in megabytes[#8]	--	double	No	All	D') / (1024 * 1024)
High Max Extents (HIGH_MAX_EXTENTS)	Correct values cannot be collected in this field. Number of segments whose PCT_MAX_EXTENTS value exceeds 90%.	--	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where DBA_SEGMENTS.EXTENTS > 0.9 * DBA_SEGMENTS.MAX_EXTENTS • For Oracle that has locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where DBA_SEGMENTS.EXTENTS > 0.9 * DBA_SEGMENTS.MAX_EXTENTS + COUNT(V$SORT_SEGMENT) where V$SORT_SEGMENTS.TOTAL_EXTENTS > 0.9 * V$SORT_SEGMENT.MAX_SIZE
Links (LINKS)	This field is not supported. Number of database links	--	short	No	Not supported	COUNT(V$DBLINK)
Links In Tran (LINKS_IN_TRAN)	This field is not supported. Number of current database links in the transaction.	--	short	No	Not supported	SUM(V$DBLINK.IN_TRANSACTION)
Links Logged On (LINKS_LOGGED_ON)	This field is not supported. Number of database links currently logged in.	--	short	No	Not supported	SUM(V$DBLINK.LOGGED_ON)
Links Open Cursors (LINKS_OPEN_CURSORS)	This field is not supported. Number of database links with open cursor.	--	short	No	Not supported	SUM(V$DBLINK.OPEN_CURSORS)
Log Files % (PERCENT_LOG_FILES)	Percentage ratio of REDO log files to the LOG_FILES parameter value in the init.ora file. Correct values cannot be collected in this field (always 0).	--	double	No	All	--
Log Mode (LOG_MODE)	Archive log mode. Valid values are NOARCHIVE, LOG, and ARCHIVELOG.[#10]	--	string(12)	No	All	V$DATABASE.LOG_MODE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Mbytes (BYTES)	Size of a database file in megabytes[#8]	--	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BYTES) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces: (SUM(DBA_DATA_FILES.BYTES) + SUM(DBA_TEMP_FILES.BYTES)) / (1024 * 1024)
Next Alloc Fails (NEXT_ALLOC_FAILS)	Correct values cannot be collected in this field. Number of segments that exceed the maximum fragments permitted for NEXT_EXTENT. For locally managed temporary tablespaces, always 0.	--	ulong	No	All	COUNT(DBA_SEGMENTS) where NEXT_EXTENT > MAX(FETS$.LENGTH) * DB_BLOCK_SIZE
Overextended (OVEREXTENDED)	Correct values cannot be collected in this field. Number of segments with more than five extents.	--	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where EXTENTS > 5 • For Oracle that has locally managed temporary tables spaces: COUNT(DBA_SEGMENTS) where EXTENTS > 5 + COUNT(V$SORT_SEGMENT) where TOTAL_EXTENTS > 5
Physical Blocks Read (PHYSICAL_BLOCKS_READ)	Number of physical block read operations[#8]	--	double	No	All	SUM(V$FILESTAT.PHYBLKRD)
Physical Blocks Written (PHYSICAL_BLOCKS_WRITTEN)	Number of physical block write operations[#8]	--	double	No	All	SUM(V$FILESTAT.PHYBLKWRT)
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed[#8]	--	double	No	All	SUM(V$FILESTAT.PHYRDS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed[#8]	--	double	No	All	SUM(V$FILESTAT.PHYWRTS)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDDB)	--	string(4)	No	All	Agent Collector
Redo Files (REDO_FILES)	Number of REDO log files[#10]	--	ulong	No	All	COUNT(V$LOGFILE)
Rollback Segments (ROLLBACK_SEGMENTS)	Number of rollback segments[#8]	--	ulong	No	All	COUNT(V$ROLLNAME)
Rollback Segments Hit % (ROLLBACK_SEGMENTS_HIT_PERCENTAGE)	Rate at which the rollback segment header was obtained without waiting[#8]	--	double	No	All	((SUM(V$ROLLSTAT.GETS) - SUM(V$ROLLSTAT.WAITS)) / SUM(V$ROLLSTAT.GETS)) * 100
Rollback Segments Trans (ROLLBACK_SEGMENTS_TRANS)	Number of currently active transactions[#8]	--	ulong	No	All	SUM(V$ROLLSTAT.XACTS)
Segments (SEGMENTS)	Correct values cannot be collected in this field. Number of segments.	--	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: COUNT(DBA_SEGMENTS) + COUNT(DBA_TEMP_FILES GROUP BY TABLESPACE_NAME) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(DBA_SEGMENTS) +

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Segments (SEGMENTS)	Correct values cannot be collected in this field. Number of segments.	--	ulong	No	All	COUNT(V$SORT_SEGMENT)
Sort Segments (SORT_SEGMENTS)	Correct values cannot be collected in this field. Number of sort segments.	--	ulong	No	All	COUNT(V$SORT_SEGMENT)
Sorting Users (SORTING_USERS)	Number of active users for the current sort segment#8	--	long	No	All	SUM(V$SORT_SEGMENT.CURRENT_USERS)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Tablespaces (TABLESPACES)	Number of tablespaces#8	--	double	No	All	COUNT(DBA_TABLESPACES)
Used Mbytes (USED_BYTES)	Size of used area in megabytes. If the monitoring target is locally managed temporary tablespaces, performance data are not collected.#8	--	double	No	All	• For Oracle that does not have any UNDO tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(sm$ts_used.bytes)/(1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(sm$ts_used.bytes) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations#8	--	double	No	All	(SUM(V$FILESTAT.PHYWRTS) / (SUM(V$FILESTAT.PHYRDS) + SUM(V$FILESTAT.PHYWRTS))) * 100

#8

Displays performance data for the PDBs that are being monitored.

#10

Displays common performance data for database instances.

Database Interval (PI_PIDB)

Function

The Database Interval (PI_PIDB) record stores performance data, taken at specific intervals, about a database.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. For details, see the descriptions of each field.

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	10	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of a database

Record size

- Fixed part: 1, 406 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Archive Change # (ARCHIVE_CHANGE_NUM)	Last archived system change number (SCN)#1, #10	COPY	double	No	All	V$DATABASE.ARCHIVE_CHANGE#
Blocks (BLOCKS)	Size of the database in Oracle blocks#2, #8	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BLOCKS) • For Oracle that has locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BLOCKS) +

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Size of the database in Oracle blocks[#2, #8]	AVG	double	No	All	SUM(DBA_TEMP_FILES.BLOCKS)
Checkpoint Change # (CHECKPOINT_CHANGE_NUM)	System change number (SCN) at the last checkpoint[#1, #10]	COPY	double	No	All	V$DATABASE.CHECKPOINT_CHANGE#
Created (CREATED)	Creation date [#1, #10]	COPY	string(20)	No	All	V$DATABASE.CREATED
DB Files % (PERCENT_DB_FILES)	Percentage ratio of the data files to the DB_FILES parameter value in the init.ora file [#2, #8]	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: (COUNT(V$DATAFILE) / init.ora DB_FILES) * 100 • For Oracle that has locally managed temporary tablespaces: ((COUNT(V$DATAFILE) + COUNT(DBA_TEMP_FILES)) / init.ora DB_FILES) * 100
DB Name (NAME)	Database name [#1, #10]	COPY	string(9)	No	All	V$DATABASE.NAME
Datafiles (DATAFILES)	Number of data files [#2, #8]	AVG	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(V$DATAFILE) • For Oracle that has locally managed temporary tablespaces: COUNT(V$DATAFILE) + COUNT(DBA_TEMP_FILES)
Extents (EXTENTS)	Correct values cannot be collected in this field. Number of extents. [#2]	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: SUM(DBA_SEGMENTS.EXTENTS) • For Oracle that has locally managed temporary tablespaces when the value of

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Extents (EXTENTS)	Correct values cannot be collected in this field. Number of extents. #2	AVG	double	No	All	localtemp_option is Y: SUM(DBA_SEGMENTS.EXTENTS) + SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: SUM(DBA_SEGMENTS.EXTENTS) + SUM(V$SORT_SEGMENT.TOTAL_EXTENTS)
Free % (PERCENT_FREE)	Percentage ratio of free space #2, #8	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces or has UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_FREE_SPACE.BYTES) / DBA_DATA_FILES.BYTES) * 100 • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: (((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / (DBA_DATA_FILES.BYTES + DBA_TEMP_FILES.BYTES)) * 100 • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option isN:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free % (PERCENT_FREE)	Percentage ratio of free space #2, #8	AVG	double	No	All	((SUM(DBA_FREE_SPACE.BYTES) + SUM(V$TEMP_SPACE_HEADER.BYTES_FREE)) / (DBA_DATA_FILES.BYTES +DBA_TEMP_FILES.BYTES)) * 100 • For the UNDO tablespaces when the value of undospace_option is Y: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / DBA_DATA_FILES.BYTES) * 100
Free Change (FREE_CHANGE)	Change to the free space in bytes#2, #8	AVG	double	No	All	• For Oracle that does not have any UNDO tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED')
Free Extents (FREE_EXTENTS)	Number of free extents #2, #8	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_FREE_SPACE) • For Oracle that has locally managed temporary tablespaces when the value of

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Extents (FREE_EXTENTS)	Number of free extents #2, #8	AVG	double	No	All	localtemp_option is Y: COUNT(DBA_FREE_SPACE) + SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) - V$SORT_SEGMENT.USED_EXTENTS • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(DBA_FREE_SPACE) + COUNT(V$TEMP_SPACE_HEADER)
Free Mbytes (FREE_BYTES)	Size of free space in megabytes #2, #8	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES)) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: (SUM(DBA_FREE_SPACE.BYTES) + SUM(V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE_BYTES)	Size of free space in megabytes #2, #8	AVG	double	No	All	$TEMP_SPACE_HEADER.BYTES_FREE)) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
High Max Extent (HIGH_MAX_EXTENTS)	Correct values cannot be collected in this field. Number of segments whose PCT_MAX_EXTENTS value exceeds 90%. #2	HILO	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where DBA_SEGMENTS.EXTENTS >0.9 * DBA_SEGMENTS.MAX_EXTENTS • For Oracle that has locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where DBA_SEGMENTS.EXTENTS > 0.9 * DBA_SEGMENTS.MAX_EXTENTS + COUNT(V$SORT_SEGMENT) where V$SORT_SEGMENTS.TOTAL_EXTENTS > 0.9 * V$SORT_SEGMENT.MAX_SIZE
I/O Ops/sec (IO_RATE)	Number of I/O operations per second #2, #8	AVG	double	No	All	(SUM(V$FILESTAT.PHYRDS) + SUM(V$FILESTAT.PHYWRTS)) / seconds in interval
Links (LINKS)	This field is not supported. Number of database links. #2	AVG	long	No	Not supported	COUNT(V$DBLINK)
Links In Tran (LINKS_IN_TRAN)	This field is not supported.	AVG	short	No	Not supported	SUM(V$DBLINK.IN_TRANSACTION)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Links In Tran (LINKS_IN_TRAN)	Number of current database links in the transaction. #2	AVG	short	No	Not supported	SUM(V$DBLINK.IN_TRANSACTION)
Links Logged On (LINKS_LOGGED_ON)	This field is not supported. Number of database links currently logged on. #2	AVG	short	No	Not supported	SUM(V$DBLINK.LOGGED_ON)
Links Open Cursors (LINKS_OPEN_CURSORS)	This field is not supported. Number of database links with open cursor. #2	AVG	short	No	Not supported	SUM(V$DBLINK.OPEN_CURSORS)
Log Files % (PERCENT_LOG_FILES)	Percentage ratio of REDO log files to the LOG_FILES parameter value in the init.ora file #2 Correct values cannot be collected in this field (always 0).	AVG	double	No	All	--
Log Mode (LOG_MODE)	Archive log mode. Valid values are NOARCHIVE, LOG, and ARCHIVELOG. #1, #10	COPY	string(12)	No	All	V$DATABASE.LOG_MODE
Mbytes (BYTES)	Size of a database file in megabytes #2, #8	AVG	double	No	All	• For Oracle that does not have any locally managed temporary tablespaces: SUM(DBA_DATA_FILES.BYTES) / (1024 * 1024) • For Oracle that has locally managed temporary tablespaces: (SUM(DBA_DATA_FILES.BYTES) + SUM(DBA_TEMP_FILES.BYTES)) / (1024 * 1024)
Next Alloc Fails (NEXT_ALLOC_FAILS)	Correct values cannot be collected in this field. Number of segments that exceed the maximum fragments permitted for NEXT_EXTENT. For locally managed temporary tablespaces, always 0.#2	HILO	ulong	No	All	COUNT(DBA_SEGMENTS) where NEXT_EXTENT > MAX(FET$.LENGTH) * DB_BLOCK_SIZE
Overextended (OVEREXTENDED)	Correct values cannot be collected in this field. Number of segments with more than five extents. #2	HILO	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where EXTENTS > 5

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Overextended (OVEREXTENDED)	Correct values cannot be collected in this field. Number of segments with more than five extents. #2	HILO	ulong	No	All	• For Oracle that has locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) where EXTENTS > 5 + COUNT(V$SORT_SEGMENT) where TOTAL_EXTENTS > 5
Physical Blocks Read (PHYSICAL_BLOCKS_READ)	Number of physical block read operations #2, #8	AVG	double	Yes	All	SUM(V$FILESTAT.PHYBLKRD)
Physical Blocks Written (PHYSICAL_BLOCKS_WRITTEN)	Number of physical block write operations #2, #8	AVG	double	Yes	All	SUM(V$FILESTAT.PHYBLKWRT)
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed #2, #8	AVG	double	Yes	All	SUM(V$FILESTAT.PHYRDS)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed #2, #8	AVG	double	Yes	All	SUM(V$FILESTAT.PHYWRTS)
Reads/sec (READ_RATE)	Number of read operations per second #2, #8	AVG	double	No	All	SUM(V$FILESTAT.PHYRDS) / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record#1	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIDB)#1	COPY	string(4)	No	All	Agent Collector
Redo Files (REDO_FILES)	Number of REDO log files #2, #10	AVG	ulong	No	All	COUNT(V$LOGFILE)
Rollback Segments (ROLLBACK_SEGMENTS)	Number of rollback segments #2, #8	AVG	ulong	No	All	COUNT(V$ROLLNAME)
Rollback Segments Hit % (ROLLBACK_SEGMENTS_HIT_PERCENTAGE)	Rate at which the rollback segment header was obtained without waiting #2, #8	AVG	double	No	All	((SUM(V$ROLLSTAT.GETS) - SUM(V$ROLLSTAT.WAITS)) / SUM(V$ROLLSTAT.GETS)) * 100
Rollback Segments Trans	Number of currently active transactions #2, #8	AVG	long	No	All	SUM(V$ROLLSTAT.XACTS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(ROLLBACK_SEGMENTS_TRANS)	Number of currently active transactions #2, #8	AVG	long	No	All	SUM(V$ROLLSTAT.XACTS)
Segments (SEGMENTS)	Correct values cannot be collected in this field. Number of segments. #2	AVG	ulong	No	All	• For Oracle that does not have any locally managed temporary tablespaces: COUNT(DBA_SEGMENTS) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is Y: COUNT(DBA_SEGMENTS) + COUNT(DBA_TEMP_FILES GROUP BY TABLESPACE_NAME) • For Oracle that has locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(DBA_SEGMENTS) + COUNT(V$SORT_SEGMENT)
Sort Segments (SORT_SEGMENTS)	Correct values cannot be collected in this field. Number of sort segments. #2	AVG	ulong	No	All	COUNT(V$SORT_SEGMENT)
Sorting Users (SORTING_USERS)	Number of active users for the current sort segment #2, #8	AVG	long	No	All	SUM(V$SORT_SEGMENT.CURRENT_USERS)
Start Time (START_TIME)	Collection start time for the performance data stored in the record#1	COPY	time_t	No	All	Agent Collector
Tablespaces (TABLESPACES)	Number of tablespaces #2, #8	AVG	double	No	All	COUNT(DBA_TABLESPACES)
Used Change (USED_CHANGE)	Change to the used space in bytes#2, #8	AVG	double	No	All	• For Oracle that does not have any UNDO tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(sm$ts_uses.bytes)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Change (USED_CHANGE)	Change to the used space in bytes[#2, #8]	AVG	double	No	All	• For the UNDO tablespaces when the value of undospace_option is Y: (SUM(sm $ts_uses.bytes) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED')
Used Mbytes (USED_BYTES)	Size of used area in megabytes. If the monitoring target is locally managed temporary tablespaces, performance data are not collected. [#2, #8]	AVG	double	No	All	• For Oracle that does not have any UNDO tablespaces or has UNDO tablespaces when the value of undospace_option is N: SUM(sm $ts_used.bytes) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(sm $ts_used.bytes) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations [#2, #8]	AVG	double	No	All	(SUM(V $FILESTAT.PHYWRTS) / (SUM(V $FILESTAT.PHYRDS) + SUM(V $FILESTAT.PHYWRTS))) * 100
Writes/sec (WRITES_RATE)	Number of write operations per second[#2, #8]	AVG	double	No	All	SUM(V $FILESTAT.PHYWRTS) / seconds in interval

#8

Displays performance data for the PDBs that are being monitored.

#10

Displays common performance data for database instances.

Database Object Cache (PD_PDDO)

Function

The Database Object Cache (PD_PDDO) record stores performance data indicating the status of database objects in the library cache at a specific point in time. PFM - Agent for Oracle creates one record for each database object. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC key fields

PD_PDDO_NAME

Lifetime

From the loading to erasing of data in the library cache of the database object

Record size

- Fixed part: 678 bytes
- Variable part: 298 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
DB Link (DB_LINK)	Name of the database link, if it exists	--	string(64)	No	All	V$DB_OBJECT_CACHE.DB_LINK
Executions (EXECUTIONS)	Number of times object was executed	--	double	No	All	V$DB_OBJECT_CACHE.EXECUTIONS
Kept (KEPT)	If the object was retained by the DBMS_SHARED_POOL.KEEP PL/SQL procedure, the value of this field is YES. Otherwise, the value is NO.	--	string(3)	No	All	V$DB_OBJECT_CACHE.KEPT
Loads (LOADS)	Number of times the object was loaded (this value	--	double	No	All	V$DB_OBJECT_CACHE.LOADS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Loads (LOADS)	increases even when the object is invalid)	--	double	No	All	V$DB_OBJECT_CACHE.LOADS
Locks (LOCKS)	Number of users currently locking this object	--	long	No	All	V$DB_OBJECT_CACHE.LOCKS
Namespace (NAMESPACE)	Object's namespace in librarycache. Valid values are TABLE/PROCEDURE, BODY, TRIGGER, INDEX, CLUSTER, and OBJECT.	--	string(15)	No	All	V$DB_OBJECT_CACHE.NAMESPACE
Object Name (NAME)	Object name	--	string(100)	No	All	V$DB_OBJECT_CACHE.NAME
Owner (OWNER)	Object's owner	--	string(64)	No	All	V$DB_OBJECT_CACHE.OWNER
Pins (PINS)	Number of users who currently have the object	--	long	No	All	V$DB_OBJECT_CACHE.PINS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDDO)	--	string(4)	No	All	Agent Collector
Sharable Mem (SHARABLE_MEM)	Size of shared memory (in bytes) used by the object in shared pool	--	double	No	All	V$DB_OBJECT_CACHE.SHARABLE_MEM
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Type (TYPE)	Object type. Valid values are INDEX, TABLE, CLUSTER, VIEW, SET, SYNONYM, SEQUENCE, PROCEDURE, FUNCTION, PACKAGE, PACKAGEBODY, TRIGGER, CLASS, OBJECT, USER, and DBLINK.	--	string(14)	No	All	V$DB_OBJECT_CACHE.TYPE

Dispatcher (PD_PDDS)

Function

The Dispatcher (PD_PDDS) record stores performance data indicating the status of dispatcher processes at a specific point in time. To collect this record, you must have a multi-thread server (MTS) configuration.

PFM - Agent for Oracle creates one record for each dispatcher in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	45	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDDS_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 228 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Accept (ACCEPT)	If the dispatcher accepts a new connection, the value of this field is YES, otherwise the value is NO.	--	string(3)	No	All	V$DISPATCHER.ACCEPT
Breaks (BREAKS)	Number of breaks (pauses) in this connection	--	double	No	All	V$DISPATCHER.BREAKS
Busy (BUSY)	Dispatcher's total busy time in hundredths of a second	--	double	No	All	V$DISPATCHER.BUSY
Busy % (PERCENT_BUSY)	Percentage of time the dispatcher was busy	--	double	No	All	(V$DISPATCHER.BUSY / (V$DISPATCHER.BUSY + V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Busy % (PERCENT_BUSY)	Percentage of time the dispatcher was busy	--	double	No	All	$DISPATCHER.IDLE)) * 100
Bytes (BYTES)	Size of message processed by the dispatcher in bytes	--	double	No	All	V $DISPATCHER.BYTES
Created (CREATED)	Number of circuits created by the dispatcher	--	ulong	No	All	V $DISPATCHER.CREATED
Idle (IDLE)	Dispatcher's idle time in hundredths of a second	--	double	No	All	V$DISPATCHER.IDLE
Idle % (PERCENT_IDLE)	Percentage of time the dispatcher was in idle status	--	double	No	All	(V $DISPATCHER.IDLE / (V $DISPATCHER.BUSY + V $DISPATCHER.IDLE)) * 100
Listener (LISTENER)	Most recent Oracle error number the dispatcher received from listener	--	long	No	All	V $DISPATCHER.LISTENER
Messages (MESSAGES)	Number of messages processed by the dispatcher	--	double	No	All	V $DISPATCHER.MESSAGES
Network (NETWORK)	Network protocols supported by the dispatcher (such as TCP or DECNET)	--	string(128)	No	All	V $DISPATCHER.NETWORK
Oracle PID (PID)	Dispatcher process's Oracle process ID	--	ulong	No	All	V$PROCESS.PID where V $DISPATCHER.PADDR = V$PROCESS.ADDR
Owned (OWNED)	Number of circuits owned by the dispatcher	--	ulong	No	All	V $DISPATCHER.OWNED
Process Name (NAME)	Name of dispatcher process	--	string(5)	No	All	V$DISPATCHER.NAME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDDS)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Dispatcher status: WAIT: Idle SEND: Sending a message connection	--	string(16)	No	All	V $DISPATCHER.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	RECEIVE: Receiving a message CONNECT: Establishing connection DISCONNECT: Processing a disconnection request BREAK: Engaged in break processing OUTBOUND: Establishing outbound connection TERMINATE: Under termination processing ACCEPT: Accepting connection (no available information) REFUSE: Refusing connection (no available information)	--	string(16)	No	All	V$DISPATCHER.STATUS

Dispatcher Interval (PI_PIDS)

Function

The Dispatcher Interval (PI_PIDS) record stores performance data indicating the status of dispatcher processes at a specific point in time. To collect this record, you must have a multi-thread server (MTS) configuration.

PFM - Agent for Oracle creates one record for each dispatcher in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	35	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIDS_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 334 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Accept (ACCEPT)	If the dispatcher accepts a new connection, the value of this field is YES, otherwise the value is NO.[#1]	COPY	string(3)	No	All	V$DISPATCHER.ACCEPT
Breaks (BREAKS)	Number of breaks (pauses) in this connection[#2]	AVG	double	Yes	All	V$DISPATCHER.BREAKS
Busy (BUSY)	Dispatcher's total busy time in hundredths of a second[#2]	AVG	double	Yes	All	V$DISPATCHER.BUSY
Busy % (PERCENT_BUSY)	Percentage of time the dispatcher was busy[#2]	AVG	double	No	All	(V$DISPATCHER.BUSY / (V$DISPATCHER.BUSY + V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Busy % (PERCENT_BUSY)	Percentage of time the dispatcher was busy[#2]	AVG	double	No	All	$DISPATCHER.IDLE)) * 100
Bytes (BYTES)	Size of message processed by the dispatcher in bytes[#2]	AVG	double	Yes	All	V $DISPATCHER.BYTES
Created (CREATED)	Number of circuits created by the dispatcher[#2]	AVG	ulong	Yes	All	V $DISPATCHER.CREATED
Idle (IDLE)	Dispatcher's idle time in hundredths of a second[#2]	AVG	double	Yes	All	V$DISPATCHER.IDLE
Idle % (PERCENT_IDLE)	Percentage of time the dispatcher was in idle status[#2]	AVG	double	No	All	(V $DISPATCHER.IDLE / (V $DISPATCHER.BUSY + V $DISPATCHER.IDLE)) * 100
Listener (LISTENER)	Most recent Oracle error number the dispatcher received from listener[#1]	COPY	short	No	All	V $DISPATCHER.LISTENER
Messages (MESSAGES)	Number of messages processed by the dispatcher[#2]	AVG	double	Yes	All	V $DISPATCHER.MESSAGES
Network (NETWORK)	Network protocols supported by the dispatcher (such as TCP or DECNET) [#1]	COPY	string(128)	No	All	V $DISPATCHER.NETWORK
Oracle PID (PID)	Dispatcher process's Oracle process ID[#1]	COPY	ulong	No	All	V$PROCESS.PID WHERE V $PROCESS.ADDR = V $DISPATCHER.PADDR
Owned (OWNED)	Number of circuits owned by the dispatcher[#2]	AVG	ulong	No	All	V $DISPATCHER.OWNED
Process Name (NAME)	Name of dispatcher process[#1]	COPY	string(5)	No	All	V$DISPATCHER.NAME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIDS)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	Dispatcher status:[#1] WAIT: Idle	COPY	string(16)	No	All	V $DISPATCHER.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	SEND: Sending a message connection RECEIVE: Receiving a message CONNECT: Establishing connection DISCONNECT: Processing a disconnection request BREAK: Engaged in break processing OUTBOUND: Establishing outbound connection TERMINATE: Under termination processing ACCEPT: Accepting connection (no available information) REFUSE: Refusing connection (no available information)	COPY	string(16)	No	All	V$DISPATCHER.STATUS

Errorlog Detail (PD_PDEL)

Function

The Errorlog Detail (PD_PDEL) record stores performance data indicating the status of error messages in the database alert file at a specific point in time. PFM - Agent for Oracle creates one record for each error message. This is a multi-instance record.

The applicable messages begin with one of the following codes:

- DBA-
- EXP-
- IMP-
- LCC-
- OER-
- ORA-
- PCC-
- PLS-
- RTL-
- TNS-

Note:

- If the value of the following initialization parameter for the Oracle database has not been set, you cannot collect this record:

  - For a version earlier than Oracle 11g: background_dump_dest

  - For Oracle 11g or a later version: diagnostic_dest

- If you are collecting this record, do not delete an alert file while PFM - Agent for Oracle is running.
- This record is created for any error message whose first line in the alert file begins with one of the codes indicated above. This record is not created for an error message whose format ends with the code shown below:

  message (ORA-XXXX)

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	35	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDEL_ERROR_TIME
- PD_PDEL_ERROR_NUM

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 843 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Container (CONTAINER)	The container name and ID of the PDB which an error has been occurred. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is blank.	--	string(36)	No	Oracle 12c R2 or later	Obtained from the alert files of the database and background process
Error # (ERROR_NUM)	Error number	--	string(10)	No	All	Obtained from the alert files of the database and background process
Error File (ERROR_FILE)	Name of the file resulting in the error. The value of the field is output as an absolute path.	--	string(256)	No	All	Obtained from the alert files of the database and background process
Error Time (ERROR_TIME)	Time the error occurred	--	string(24)	No	All	Obtained from the alert files of the database and background process
Message (MESSAGE)	Error message	--	string(512)	No	All	Obtained from the alert files of the database and background process
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDEL)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

GCS Stat Summary (PD_PDGC)

Function

The GCS Stat Summary (PD_PDGC) record stores performance data indicating the status of the Global Cache Service (GCS) at a specific point in time. This record is for Oracle Real Application Clusters.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle Real Application Clusters instance

Record size

- Fixed part: 754 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
GC Blocks Corrupt (GLOBAL_CACHE_BLOCKS_CORRUPT)	Number of blocks that resulted in some damage or checksum error during interconnection	--	ulong	No	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc blocks corrupt' GROUP BY INST_ID
GC Blocks Lost (GLOBAL_CACHE_BLOCKS_LOST)	Number of times a global cache request resulted in a timeout due to damage or checksum errors during interconnection	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc blocks lost' GROUP BY INST_ID
GC CRBlock Rec Per MilliSec (GLOBAL_CACHE_CRBLOCK_RECEIVE_PER_MILLISEC)	Length of time that the foreground process waited for each CR block that was sent via an interconnection (in milliseconds)	--	double	No	All	(GLOBAL_CACHE_CRBLOCK_RECEIVE_TIME * 10) / GLOBAL_CACHE_CRBLOCKS_RECEIVED GROUP BY INST_ID
GC CRBlock Receive Time	Total length of time the foreground process waited	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc cr

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(GLOBAL_CACHE_CR BLOCK_RECEIVE_T IME)	for a CR block that was sent via an interconnection	--	double	No	All	block receive time' GROUP BY INST_ID
GC CRBlocks Received (GLOBAL_CACHE_CR BLOCKS_RECEIVED)	Total number of blocks received	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc cr blocks received' GROUP BY INST_ID
GC Convert Time (GLOBAL_CACHE_CO NVERT_TIME)	Total time that elapsed during lock conversion	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache convert time' GROUP BY INST_ID
GC Converts (GLOBAL_CACHE_CO NVERTS)	Number of lock conversions in the global cache	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache convert' GROUP BY INST_ID
GC Get Per MilliSec (GLOBAL_CACHE_GE T_PER_MILLISEC)	Wait time per request (in milliseconds)	--	double	No	All	(GLOBAL_CACHE_GE T_TIME * 10) / GLOBAL_CACHE_GE TS GROUP BY INST_ID
GC Get Time (GLOBAL_CACHE_GE T_TIME)	Total wait time	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache get time' GROUP BY INST_ID
GC Gets (GLOBAL_CACHE_GE TS)	Number of locks obtained	--	double	No	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache gets' GROUP BY INST_ID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TY PE)	Record name (always PDGC)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

GCS Stat Summary Interval (PI_PIGC)

Function

GCS Stat Summary Interval (PI_PIGC) record stores performance data, taken at specific intervals, about the Global Cache Service (GCS). This record is for Oracle Real Application Clusters.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle Real Application Clusters instance

Record size

- Fixed part: 874 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
GC Blocks Corrupt (GLOBAL_CACHE_BLOCKS_CORRUPT)	Number of blocks that resulted in some damage or checksum error during interconnection[#2]	AVG	ulong	Yes	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc blocks corrupt' GROUP BY INST_ID
GC Blocks Lost (GLOBAL_CACHE_BLOCKS_LOST)	Number of times a global cache request resulted in a timeout due to damage or checksum errors during interconnection[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc blocks lost' GROUP BY INST_ID
GC CRBlock Rec Per MilliSec (GLOBAL_CACHE_CRBLOCK_RECEIVE_PER_MILLISEC)	Length of time that the foreground process waited for each CR block that was sent via an interconnection (in milliseconds) [#2]	AVG	double	No	All	(GLOBAL_CACHE_CRBLOCK_RECEIVE_TIME * 10) / GLOBAL_CACHE_CRBLOCKS_RECEIVED GROUP BY INST_ID
GC CRBlock Receive Time	Total length of time the foreground process waited	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME= 'gc

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(GLOBAL_CACHE_CR BLOCK_RECEIVE_T IME)	for a CR block that was sent via an interconnection[#2]	AVG	double	Yes	All	cr block receive time' GROUP BY INST_ID
GC CRBlocks Received (GLOBAL_CACHE_CR BLOCKS_RECEIVED)	Total number of blocks received[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME= gc cr blocks received' 'GROUP BY INST_ID
GC Convert Time (GLOBAL_CACHE_CO NVERT_TIME)	Total time that elapsed during lock conversion[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache convert time' GROUP BY INST_ID
GC Converts (GLOBAL_CACHE_CO NVERTS)	Number of lock conversions in the global cache[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache convert' GROUP BY INST_ID
GC Get Per MilliSec (GLOBAL_CACHE_GE T_PER_MILLISEC)	Wait time per request (in milliseconds) [#2]	AVG	double	No	All	(GLOBAL_CACHE_GET _TIME * 10) / GLOBAL_CACHE_GETS GROUP BY INST_ID
GC Get Time (GLOBAL_CACHE_GE T_TIME)	Total wait time[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache get time' GROUP BY INST_ID
GC Gets (GLOBAL_CACHE_GE TS)	Number of locks obtained[#2]	AVG	double	Yes	All	GV$SYSSTAT.VALUE WHERE NAME = 'global cache gets' GROUP BY INST_ID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TY PE)	Record name (always PIGC)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

Instance (PD_PDI)

Function

The Instance (PD_PDI) record stores performance data indicating the status (at a specific point in time) of an instance.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 1,128 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Host (HOST)	Name of the physical host connected to the running instance	--	string(64)	No	All	V$INSTANCE.HOST_NAME
ORACLE_HOME (ORACLE_HOME)	ORACLE_HOME environment variable	--	string(255)	No	All	--
ORACLE_SID (ORACLE_SID)	ORACLE_SID environment variable	--	string(30)	No	All	--
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDI)	--	string(4)	No	All	Agent Collector
Restricted Mode (RESTRICTED_MODE)	This field is not supported. The value of this field is 1 if the instance is in restricted mode	--	short	No	Not supported	V$INSTANCE.LOGINS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Restricted Mode (RESTRICTED_MODE)	and 0 if the instance is not in restricted mode.	--	short	No	Not supported	V$INSTANCE.LOGINS
SGA Database Buffers (SGA_DATABASE_BUFFERS)	Size of SGA database buffer in bytes	--	double	No	All	V$SGA.VALUE where V$SGA.NAME = 'Database Buffers'
SGA Fixed Size (SGA_FIXED_SIZE)	Size of SGA fixed memory in bytes	--	double	No	All	V$SGA.VALUE where V$SGA.NAME = 'Fixed Size'
SGA Redo Buffers (SGA_REDO_BUFFERS)	Size of SGA REDO buffer in bytes	--	double	No	All	V$SGA.VALUE where V$SGA.NAME = 'Redo Buffers'
SGA Variable Size (SGA_VARIABLE_SIZE)	Size of SGA variable memory in bytes	--	double	No	All	V$SGA.VALUE where V$SGA.NAME = 'Variable Size'
Session Current (SESSION_CURRENT)	Current number of concurrent user sessions	--	ulong	No	All	V$LICENSE.SESSIONS_CURRENT
Session Highwater (SESSION_HIGHWATER)	Maximum number of concurrent user sessions since the instance started	--	ulong	No	All	V$LICENSE.SESSIONS_HIGHWATER
Sessions Max (SESSIONS_MAX)	Maximum number of concurrent user sessions permitted for the instance	--	ulong	No	All	V$LICENSE.SESSIONS_MAX
Sessions Warning (SESSIONS_WARNING)	Warning limit for the concurrent user sessions for the instance	--	ulong	No	All	V$LICENSE.SESSIONS_WARNING
Shutdown Pending (SHUTDOWN_PENDING)	The value of this field is 1 if shutdown is on hold, and 0 if shutdown is not on hold.	--	short	No	All	V$INSTANCE.SHUTDOWN_PENDING
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Startup Time (STARTUP_TIME)	Start date and time	--	string(20)	No	All	V$INSTANCE.STARTUP_TIME
Users Max (USERS_MAX)	Maximum number of users permitted for the database	--	ulong	No	All	V$LICENSE.USERS_MAX
Version (VERSION)	Oracle Database version 0 is set in lower than revision, if the monitoring target is Oracle Database 18c or later.	--	string(20)	No	All	PRODUCT_COMPONENT_VERSION

Instance Availability (PD_PDIA)

Function

An Instance Availability (PD_PDIA) record stores performance data indicating the availability of an instance at a specific point in time. This record is immediately disconnected when a connection to an Oracle Database is attempted and successful. As such, due to data collection for other records, if an Oracle Database is already connected, there may temporarily be two connections.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 695 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Availability (AVAILABILITY)	Availability status[#3, #4] Valid values are 0 (stopped) or 1 (running). If connection to an Oracle Database is impossible, 0 (stopped). If connection to an Oracle Database is possible, 1 (running).	--	short	No	All	Agent Collector
Collect Time (COLLECT_TIME)	Time required for processing to connect to and disconnect from an Oracle Database (in milliseconds)	--	long	No	All	Agent Collector
Error # (ERROR_NUM)	Error code when a connection to an Oracle	--	string(10)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Error # (ERROR_NUM)	Database is attempted and results in an error[#5] This is blank when the connection is successful.	--	string(10)	No	All	Agent Collector
Record Time (RECORD_TIME)	Collection end time of performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name. This is always PDIA.	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time of performance data stored in the record	--	time_t	No	All	Agent Collector

#3

The value in the Availability field differs depending on the sqlnet value, which is instance information specified when an PFM - Agent for Oracle instance environment is configured. Table 6-11 Relationship Between the sqlnet Value and the Value in the Availability Field.

Table 6–10: Relationship Between the sqlnet Value and the Value in the Availability Field

Sqlnet value	Connection to Oracle Database	Value in the Availability field
Y	The listener is used to connect to Oracle Database.	The status of the availability of the listener and Oracle Database
N	The listener is not used to connect to Oracle Database.	The status of the availability of Oracle Database only

For details about other settings and procedures, see *2.1.4(3) Set up an instance environment* or *3.1.4(4) Set up an instance environment*.

#4

If the maximum number of concurrent user sessions that can be recognized by an Oracle instance has been reached before PD_PDIA record collection (due to failed attempts to connect to the Oracle Database), the Availability field of a PD_PDIA record may be displayed as 0 (stopped), but other records may be obtained normally.

Likewise, when an Oracle instance stops running during record collection, the Availability field of a PD_PDIA record may be displayed as 0 (stopped), but other records may be obtained normally.

Also, when an Oracle instance stopped during record collection starts again, the Availability field of a PD_PDIA record may be displayed as 1 (running), but other records may not be able to be obtained.

#5

The following table lists example output for the Error # field and the corresponding message code. The error codes in this table are output when Oracle is physically disconnected. For details about the messages that correspond to the Oracle message codes, see the Oracle documentation.

Table 6–11: Examples of Error # field output and corresponding message codes

Example of Error # field output (Oracle error code)	Corresponding Oracle message code
28	ORA-00028
1012	ORA-01012

Example of Error # field output (Oracle error code)	Corresponding Oracle message code
3113	ORA-03113
3114	ORA-03114
12571	ORA-12571

Latch (PD_PDLA)

Function

The Latch (PD_PDLA) record stores performance data indicating the status of latches at a specific point in time. PFM - Agent for Oracle creates one record for each latch in an instance. This is a multi-instance record.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record includes latch information of monitoring PDB and common latch information for database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	10	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDLA_LATCH_NUM
- PD_PDLA_LEVEL_NUM

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 284 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addr (ADDR)	Latch address	--	string(16)	No	All	V$LATCH.ADDR
Gets (GETS)	Number of times latches were obtained for requests in the willingto-wait mode	--	double	No	All	V$LATCH.GETS
Immediate Gets (IMMEDIATE_GETS)	Number of times latches were obtained for requests in the no wait mode	--	double	No	All	V$LATCH.IMMEDIATE_GETS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Immediate Hit % (IMMEDIATE_HIT_PERCENTAGE)	Percentage of times that latches were obtained in the no wait mode (Rate at which latches were obtained in the first attempt)	--	double	No	All	(V $LATCH.IMMEDIATE_GETS/(V $LATCH.IMMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES))* 100
Immediate Misses (IMMEDIATE_MISSES)	Number of times latch acquisition failed for requests in the no wait mode	--	double	No	All	V $LATCH.IMMEDIATE_MISSES
Latch # (LATCH_NUM)	Latch number	--	short	No	All	V$LATCH.LATCH#
Latch Name (NAME)	Latch name	--	string(50)	No	All	V $LATCHNAME.NAME
Level # (LEVEL_NUM)	Latch level	--	double	No	All	V$LATCH.LEVEL#
Misses (MISSES)	Number of times the first attempt at latch acquisition failed for requests in the willingto-wait mode	--	double	No	All	V$LATCH.MISSES
OS PID (OS_PID)	OS's client process ID	--	string(12)	No	All	V $SESSION.PROCESS where V $LATCHHOLDER.SID = V $SESSION.SID
OS User (OS_USER)	OS's client user name	--	string(30)	No	All	V $SESSION.OSUSER where V $LATCHHOLDER.SID = V $SESSION.SID
Oracle PID (PID)	Process ID to which the latch belongs	--	ulong	No	All	V $LATCHHOLDER.PID
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V $SESSION.PROGRAM where V $LATCHHOLDER.SID = V $SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDLA)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
SID (SID)	Session ID to which the latch belongs	--	ulong	No	All	V$LATCHHOLDER.SID
Sleeps (SLEEPS)	Number of times the system slept when it needed to wait	--	double	No	All	V$LATCH.SLEEPS
Spin Gets (SPIN_GETS)	Number of latch requests that can wait that failed the first time but were successful during spinning	--	double	No	All	V$LATCH.SPIN_GETS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME
Waiters Woken (WAITERS_WOKEN)	Number of times standby sleep was released (always 0 for Oracle 10g Release 2 or later)	--	double	No	All	V$LATCH.WAITERS_WOKEN
Waits Holding Latch (WAITS_HOLDING_LATCH)	Number of waits when other latches were held (always 0 for Oracle 10g Release 2 or later)	--	double	No	All	V$LATCH.WAITS_HOLDING_LATCH
Willing To Wait Hit % (WILLING_TO_WAIT_HIT_PERCENTAGE)	Percentage of times that latches were obtained in the willingto-wait mode (Rate at which latches were obtained in the first attempt)	--	double	No	All	((V$LATCH.GETS - V$LATCH.MISSES) / V$LATCH.GETS) * 100

Latch Interval (PI_PILA)

Function

The Latch Interval (PI_PILA) record stores performance data, taken at specific intervals, about latches. PFM - Agent for Oracle creates one record for each latch in an instance. This is a multi-instance record.

For the monitored Oracle version, if the data in the ADDR column in the Oracle static dictionary view V$LATCH is the same in every row, this record does not operate normally. To monitor latches in this case, use the Latch (PD_PDLA) record.

You can use the Oracle command sqlplus to check whether the data in the ADDR column is the same in every row. To do so, execute the following SQL statement:

```
SELECT ADDR FROM V$LATCH
```

For details about the sqlplus command, see the Oracle documentation.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record includes latch information of monitoring PDB and common latch information for database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	5	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PILA_ADDR

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 416 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addr	Latch address[#1]	COPY	string(16)	No	All	V$LATCH.ADDR

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(ADDR)	Latch address[#1]	COPY	string(16)	No	All	V$LATCH.ADDR
Gets (GETS)	Number of times latches were obtained for requests in the willingto-wait mode[#2]	AVG	double	Yes	All	V$LATCH.GETS
Immediate Gets (IMMEDIATE_GETS)	Number of times latches were obtained for requests in the no wait mode[#2]	AVG	double	Yes	All	V $LATCH.IMMEDIATE_GETS
Immediate Hit % (IMMEDIATE_HIT_PERCENTAGE)	Percentage of times that latches were obtained in the no wait mode (Rate at which latches were obtained in the first attempt) [#2]	AVG	double	No	All	(V $LATCH.IMMEDIATE_GETS/(V $LATCH.IMMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES)) * 100
Immediate Misses (IMMEDIATE_MISSES)	Number of times latch acquisition failed for requests in the no wait mode[#2]	AVG	double	Yes	All	V $LATCH.IMMEDIATE_MISSES
Latch # (LATCH_NUM)	Latch number[#1]	COPY	double	No	All	V$LATCH.LATCH#
Latch Name (NAME)	Latch name[#1]	COPY	string(50)	No	All	V $LATCHNAME.NAME
Level # (LEVEL_NUM)	Latch level[#1]	COPY	double	No	All	V$LATCH.LEVEL#
Misses (MISSES)	Number of times the first attempt at latch acquisition failed for requests in the willingto-wait mode[#2]	AVG	double	Yes	All	V$LATCH.MISSES
OS PID (OS_PID)	OS's client process ID[#1]	COPY	string(12)	No	All	V $SESSION.PROCESS where V $LATCHHOLDER.SID = V $SESSION.SID
OS User (OS_USER)	OS's client user name[#1]	COPY	string(30)	No	All	V $SESSION.OSUSER where V $LATCHHOLDER.SID = V $SESSION.SID
Oracle PID (PID)	Process ID to which the latch belongs[#1]	COPY	ulong	No	All	V $LATCHHOLDER.PID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Program (PROGRAM)	Name of the program being executed[#1]	COPY	string(48)	No	All	V$SESSION.PROGRAM where V$LATCHHOLDER.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PILA)[#1]	COPY	string(4)	No	All	Agent Collector
SID (SID)	Session ID to which the latch belongs[#1]	COPY	ulong	No	All	V$LATCHHOLDER.SID
Sleeps (SLEEPS)	Number of times the system slept when it needed to wait[#2]	AVG	double	Yes	All	V$LATCH.SLEEPS
Spin Gets (SPIN_GETS)	Number of latch requests that can wait that failed the first time but were successful during spinning[#2]	AVG	double	Yes	All	V$LATCH.SPIN_GETS
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
User (USERNAME)	Oracle user name[#1]	COPY	string(30)	No	All	V$SESSION.USERNAME
Waiters Woken (WAITERS_WOKEN)	Number of times standby sleep was released[#2] (always 0 for Oracle 10g Release 2 or later)	AVG	double	Yes	All	V$LATCH.WAITERS_WOKEN
Waits Holding Latch (WAITS_HOLDING_LATCH)	Number of waits when other latches were held[#2] (always 0 for Oracle 10g Release 2 or later)	AVG	double	Yes	All	V$LATCH.WAITS_HOLDING_LATCH
Willing To Wait Hit % (WILLING_TO_WAIT_HIT_PERCENTAGE)	Percentage of times that latches were obtained in the willingto-wait mode (Rate at which latches were obtained in the first attempt) [#2]	AVG	double	No	All	((V$LATCH.GETS - V$LATCH.MISSES) / V$LATCH.GETS) * 100

Library Cache (PD_PDLC)

Function

The Library Cache (PD_PDLC) record stores performance data indicating the status of library cache management at a specific point in time. PFM - Agent for Oracle creates one record for each library cache. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	35	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDLC_NAMESPACE

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 88 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Get Hit % (GET_HIT_PERCENTAGE)	Ratio (as a percent) of the value of the Get Hits field to the value of the Gets field	--	double	No	All	(V $LIBRARYCACHE.GETHITS / V $LIBRARYCACHE.GETS) * 100
Get Hits (GET_HITS)	Number of times the handle was already in cache (if the handle was not in cache, a miss results and the system allocates the handle and places it in cache)	--	double	No	All	V $LIBRARYCACHE.GETHITS
Gets (GETS)	Number of times the system requested handle to a library object in this namespace	--	double	No	All	V $LIBRARYCACHE.GETS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Invalidations (INVALIDATIONS)	Number of times a nonpermane nt library object (such as a shared SQL area) was invalid	--	double	No	All	V $LIBRARYCACHE.INVALIDATIONS
Miss % (MISS_PERCENTAGE)	Ratio (as a percent) of the number of reloads to the number of acquisition requests to objects in the library cache issued by the system	--	double	No	All	(V $LIBRARYCASHE.RELOADS / V $LIBRARYCASHE.PINS) * 100
Namespace (NAMESPACE)	Namespace in the library cache	--	string(15)	No	All	V $LIBRARYCACHE.NAMESPACE
Pin Hit % (PIN_HIT_PERCENTAGE)	Ratio (as a percent) of the value of the Pin Hits field to the value of the Pins field	--	double	No	All	(V $LIBRARYCACHE.PINHITS / V $LIBRARYCACHE.PINS) * 100
Pin Hits (PIN_HITS)	Number of times the object acquired by the system was already in cache and initialized	--	double	No	All	V $LIBRARYCACHE.PINHITS
Pins (PINS)	Number of times the system issued an acquisition request to an object in cache to access it	--	double	No	All	V $LIBRARYCACHE.PINS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDLC)	--	string(4)	No	All	Agent Collector
Reloads (RELOADS)	Number of times the system needed to reinitialize a library object and load data because it had not been used recently or was invalid	--	double	No	All	V $LIBRARYCACHE.RELOADS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

Library Cache Interval (PI_PILC)

Function

The Library Cache Interval (PI_PILC) record stores performance data, taken at specific intervals, about the library cache. PFM - Agent for Oracle creates one record for each library cache. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	25	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PILC_NAMESPACE

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 196 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Get Hit % (GET_HIT_PERCENTAGE)	Ratio (as a percent) of the value of the Get Hits field to the value of the Gets field during the interval[#2]	AVG	double	No	All	(V $LIBRARYCACHE.GETHITS / V $LIBRARYCACHE.GETS) * 100
Get Hits (GET_HITS)	Number of times the handle was already in cache during the interval[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.GETHITS
Gets (GETS)	Number of times the system requested handle to a library object in this namespace during the interval[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.GETS
Invalidations (INVALIDATIONS)	Number of times a nonpermane nt library object (such as a	AVG	double	Yes	All	V $LIBRARYCACHE.INVALIDATIONS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Invalidations (INVALIDATIONS)	shared SQL area) was invalid[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.INVALIDATIONS
Miss % (MISS_PERCENTAGE)	Ratio (as a percent) of the number of reloads to the number of acquisition requests to objects in the library cache issued by the system[#2]	AVG	double	No	All	(V $LIBRARYCACHE.RELOADS / V $LIBRARYCASHE.PINS) * 100
Namespace (NAMESPACE)	Namespace in the library cache[#1]	COPY	string(15)	No	All	V $LIBRARYCACHE.NAMESPACE
Pin Hit % (PIN_HIT_PERCENTAGE)	Ratio (as a percent) of the value of the Pin Hits field to the value of the Pins field during the interval[#2]	AVG	double	No	All	(V $LIBRARYCACHE.PINHITS / V $LIBRARYCACHE.PINS) * 100
Pin Hits (PIN_HITS)	Number of times the object accessed by the system was already in cache and initialized during the interval[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.PINHITS
Pins (PINS)	Number of times the system issued an acquisition request to an object in cache to access it during the interval[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.PINS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PILC)[#1]	COPY	string(4)	No	All	Agent Collector
Reloads (RELOADS)	Number of times the system needed to reinitialize a library object and load data because it had not been used recently or was invalid during the interval[#2]	AVG	double	Yes	All	V $LIBRARYCACHE.RELOADS
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

Lock (PD_PDLO)

Function

The Lock (PD_PDLO) record stores performance data indicating the status of locks at a specific point in time. PFM - Agent for Oracle creates one record for each lock in an instance. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create this view, you must execute the CATBLOCK.SQL script that is provided by Oracle.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	55	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDLO_ADDR
- PD_PDLO_SID

Lifetime

From the start to the end of a lock

Record size

- Fixed part: 678 bytes
- Variable part: 197 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addr (ADDR)	Lock address	--	string(16)	No	All	V$LOCK.KADDR
Blocking (BLOCKING)	Whether this lock blocks another lock (If it does, the value in this field is 1. Otherwise, the value is 0.)	--	short	No	All	V$LOCK.BLOCK
Blocking Sessions (BLOCKING_SESSIONS)	Number of sessions that are blocked because they are waiting for a lock to be released	--	ulong	No	All	count(WAITING_SESSION) from DBA_WAITERS where DBA_WAITERS.LOCK

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocking Sessions (BLOCKING_SESSIONS)	Number of sessions that are blocked because they are waiting for a lock to be released	--	ulong	No	All	_ID1 = ID1 and DBA_WAITERS.LOCK _ID2 = ID2 and DBA_WAITERS.HOLD ING_SESSION = SID and DBA_WAITERS.MODE _HELD = LOCK_MODE
Current Mode Time (CURRENT_MODE_TIME)	Time since the current mode was authorized	--	ulong	No	All	V$LOCK.CTIME
ID1 (ID1)	Lock ID 1	--	ulong	No	All	V$LOCK.ID1
ID2 (ID2)	Lock ID 2	--	ulong	No	All	V$LOCK.ID2
Lock Mode (LOCK_MODE)	Lock mode. The following are the valid values for this field: Null, Row-S (SS), Row-X (SX), Share, S/Row-X (SSX), and Exclusive. The following are the values of the fields that are displayed as abbreviations : Row-S (SS) : row share Row-X (SX) : row exclusive S/Row-X (SSX) : share row exclusive	--	string(20)	No	All	V$LOCK.LMODE
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V $SESSION.PROGRAM where V$LOCK.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDLO)	--	string(4)	No	All	Agent Collector
Request Mode (REQUEST_MODE)	Requested lock mode. The following are the valid values for this field: Null, Row-S (SS), Row-X (SX), Share, S/Row-X (SSX), and Exclusive. The following are the values of the fields that are displayed as abbreviations : Row-S (SS) : row share	--	string(20)	No	All	V$LOCK.REQUEST

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Request Mode (REQUEST_MODE)	Row-X (SX) : row exclusive S/Row-X (SSX) : share row exclusive	--	string(20)	No	All	V$LOCK.REQUEST
SID (SID)	Session ID holding the lock	--	ulong	No	All	V$LOCK.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Type (TYPE)	Lock type	--	string(2)	No	All	V$LOCK.TYPE
Type Text (TYPE_TEXT)	Details of lock type	--	string(32)	No	All	V$LOCK.TYPE
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$LOCK.SID = V$SESSION.SID

Lock Activity Interval (PI_PIPL)

Function

The Lock Activity Interval (PI_PIPL) record stores performance data, taken at specific intervals, about PCM lock conversions that have occurred. PFM - Agent for Oracle creates one record each time a PCM lock conversion occurs. This is a multi-instance record for Oracle Real Application Clusters.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	55	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PIPL_INITIAL_STATE
- PI_PIPL_FINAL_STATE

Lifetime

From the start to the stop of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 84 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Action (ACTION)	Description of lock conversion.#1 The correct values cannot be collected because the values are collected from a performance view that is not recommended by Oracle.	COPY	string(55)	No	All	V$LOCK_ACTIVITY.ACTION_VAL
Conversions (CONVERSIONS)	Number of times a lock operation was executed.#2 The correct values cannot be collected because the values are collected from a	AVG	long	Yes	All	V$LOCK_ACTIVITY.COUNTER

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Conversions (CONVERSIONS)	performance view that is not recommended by Oracle.	AVG	long	Yes	All	V$LOCK_ACTIVITY.COUNTER
Final State (FINAL_STATE)	Final PCM lock status.#1 The correct values cannot be collected because the values are collected from a performance view that is not recommended by Oracle.	COPY	string(5)	No	All	V$LOCK_ACTIVITY.TO_VAL
Initial State (INITIAL_STATE)	Initial PCM lock status.#1 The correct values cannot be collected because the values are collected from a performance view that is not recommended by Oracle.	COPY	string(5)	No	All	V$LOCK_ACTIVITY.FROM_VAL
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record#1	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIPL)#1	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record#1	COPY	time_t	No	All	Agent Collector

Lock Interval (PI_PILO)

Function

The Lock Interval (PI_PILO) record stores performance data, taken at specific intervals, about locks. PFM - Agent for Oracle creates one record for each lock in an instance. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create this view, you must execute the CATBLOCK.SQL script that is provided by Oracle.

When Log is set to Yes, if you collect history over a long period of time, because of the short lifetime, it is not summarized in units of years or months. All instances are retained, resulting in a bloa ted store database. In addition, when the collected history is summarized, more memory is used than necessary. The memory shortage might cause monitoring to stop. If you want to collect history over a long period of time, use the Lock (PD_PDLD) record for monitoring.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	55	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PILO_ADDR
- PI_PILO_SID

Lifetime

From the activation to release of a lock

Record size

- Fixed part: 678 bytes
- Variable part: 217 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addr (ADDR)	Lock address[#1]	COPY	string(16)	No	All	V$LOCK.KADDR
Blocking (BLOCKING)	Whether this lock blocks another lock (If it does, the value in this field is 1. Otherwise, the value is 0.)[#1]	COPY	short	No	All	V$LOCK.BLOCK

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocking Sessions (BLOCKING_SESSIONS)	Number of sessions that are blocked because they are waiting for a lock to be released[#1]	COPY	ulong	No	All	count(WAITING_SESSION) from DBA_WAITERS where DBA_WAITERS.LOCK_ID1 = ID1 and DBA_WAITERS.LOCK_ID2 = ID2 and DBA_WAITERS.HOLDING_SESSION = SID and DBA_WAITERS.MODE_HELD = LOCK_MODE
Current Mode Time (CURRENT_MODE_TIME)	Time since the current mode was authorized[#2]	AVG	ulong	Yes	All	V$LOCK.CTIME
ID1 (ID1)	Lock ID 1[#1]	COPY	double	No	All	V$LOCK.ID1
ID2 (ID2)	Lock ID 2[#1]	COPY	double	No	All	V$LOCK.ID2
Lock Mode (LOCK_MODE)	Lock mode. The following are the valid values for this field: Null, Row-S (SS), Row-X (SX), Share, S/Row-X (SSX), and Exclusive. The following are the values of the fields that are displayed as abbreviations : Row-S (SS) : row share Row-X (SX) : row exclusive S/Row-X (SSX) : share row exclusive[#1]	COPY	string(20)	No	All	V$LOCK.LMODE
Program (PROGRAM)	Name of the program being executed[#1]	COPY	string(48)	No	All	V$SESSION.PROGRAM where V$LOCK.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PILO)[#1]	COPY	string(4)	No	All	Agent Collector
Request Mode (REQUEST_MODE)	Requested lock mode. The following are the valid values for this field: Null, Row-S (SS), Row-X (SX), Share, S/Row-X (SSX), and Exclusive. The following are the values of the fields that are displayed as abbreviations : Row-S (SS) : row share	COPY	string(20)	No	All	V$LOCK.REQUEST

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Request Mode (REQUEST_MODE)	Row-X (SX) : row exclusive S/Row-X (SSX) : share row exclusive[#1]	COPY	string(20)	No	All	V$LOCK.REQUEST
SID (SID)	Session ID holding or having the lock[#1]	COPY	ulong	No	All	V$LOCK.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Type (TYPE)	Lock type[#1]	COPY	string(2)	No	All	V$LOCK.TYPE
Type Text (TYPE_TEXT)	Details of lock type[#1]	COPY	string(32)	No	All	V$LOCK.TYPE
User (USERNAME)	Oracle user name[#1]	COPY	string(30)	No	All	V $SESSION.USERNAME where V$LOCK.SID = V$SESSION.SID

Lock Waiters (PD_PDLW)

Function

The Lock Waiters (PD_PDLW) record stores performance data indicating the status (at a specific point in time) of all sessions waiting for lock and all sessions holding lock. PFM - Agent for Oracle creates one record for each lock that a session is waiting for. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create the static dictionary view DBA_WAITERS, you need to execute the CATBLOCK.SQL script that is provided by Oracle.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	60	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the start to the end of a deadlock

Record size

- Fixed part: 678 bytes
- Variable part: 428 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Holding OS PID (HOLDING_PID)	OS's client process ID corresponding to the session holding the lock	--	string(30)	No	All	V$SESSION.PROCESS
Holding Session (HOLDING_SESSION)	Session ID holding the lock	--	ulong	No	All	DBA_WAITERS.HOLDING_SESSION
Holding User (HOLDING_USER)	User name for the session holding the lock	--	string(30)	No	All	V$SESSION.USERNAME
Lock ID1 (LOCK_ID1)	Lock ID 1	--	string(40)	No	All	DBA_WAITERS.LOCK_ID1

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Lock ID2 (LOCK_ID2)	Lock ID 2	--	string(40)	No	All	DBA_WAITERS.LOCK_ID2
Lock Type (TYPE)	Lock type	--	string(30)	No	All	DBA_WAITERS.TYPE
Mode Held (MODE_HELD)	Lock mode that was held during data collection	--	string(40)	No	All	DBA_WAITERS.MODE_HELD
Mode Requested (MODE_REQUESTED)	Lock mode that was requested during data collection	--	string(40)	No	All	DBA_WAITERS.MODE_REQUESTED
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDLW)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Waiting OS PID (WAITING_PID)	OS's client process ID corresponding to the session waiting for the lock	--	string(30)	No	All	V$SESSION.PROCESS
Waiting Session (WAITING_SESSION)	Session ID waiting for the lock	--	ulong	No	All	DBA_WAITERS.WAITING_SESSION
Waiting User (WAITING_USER)	User name for the session waiting for the lock	--	string(30)	No	All	V$SESSION.USERNAME
XID (XID)	ID used internally to identify the record	--	string(100)	No	All	DBA_WAITERS.LOCK_ID1 + DBA_WAITERS.LOCK_ID2 + DBA_WAITERS.HOLDING_SESSION + DBA_WAITERS.WAITING_SESSION

Minimum Database Interval 2 (PI_PMDB)

Function

The Minimum Database Interval 2 (PI_PMDB) record stores performance data, taken at specific intervals, about a database.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	3600	Y
Collection Offset	1810	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

None

Lifetime

From the creation to the deletion of a database

Record Size

- Fixed part: 688 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
DB Name (NAME)	Database name	COPY	string(9)	No	All	V$DATABASE.NAME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PMDB)	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	COPY	time_t	No	All	Agent Collector

Minimum Data File Interval 2 (PI_PMDF)

Function

The Minimum Data File Interval 2 (PI_PMDF) record stores performance data, taken at specific intervals, about data files. PFM - Agent for Oracle creates one record for each data file in a database. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	300	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC Key Fields

- PI_PMDF_FILE_NUM
- PI_PMDF_NAME

Lifetime

From the creation to the deletion of a data file

Record Size

- Fixed part: 678 bytes
- Variable part: 655 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
File # (FILE_NUM)	File identification number	COPY	ushort	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$DATAFILE.FILE# For locally managed temporary tablespaces: V$TEMPFILE.FILE#
File Name (NAME)	File name	COPY	string(513)	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$DATAFILE.NAME

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
File Name (NAME)	File name	COPY	string(513)	No	All	For locally managed temporary tablespaces: V$TEMPFILE.NAME
I/O Ops/sec (IO_RATE)	Number of I/O operations per second	AVG	double	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: (V$FILESTAT.PHYRDS + V$FILESTAT.PHYWRTS) / seconds in interval For locally managed temporary tablespaces: (V$TEMPSTAT.PHYRDS + V$TEMPSTAT.PHYWRTS) / seconds in interval
Mbytes (BYTES)	Percentage of disk space used by a file system, in megabytes	COPY	double	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$DATAFILE.BYTES / (1024 * 1024) For locally managed temporary tablespaces: V$TEMPFILE.BYTES / (1024 * 1024)
Physical Reads (PHYSICAL_READS)	Number of physical block read operations that were completed during the interval[#2]	AVG	double	Yes	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYRDS For locally managed temporary tablespaces: V$TEMPSTAT.PHYRDS
Physical Writes (PHYSICAL_WRITES)	Number of physical block write operations that were completed during the interval[#2]	AVG	double	Yes	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V$FILESTAT.PHYWRTS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Physical Writes (PHYSICAL_WRITES)	Number of physical block write operations that were completed during the interval[#2]	AVG	double	Yes	All	For locally managed temporary tablespaces: V $TEMPSTAT.PHYWRTS
Reads/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $FILESTAT.PHYRDS / seconds in interval For locally managed temporary tablespaces: V $TEMPSTAT.PHYRDS / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PMDF) [#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Tablespace name associated with the file[#1]	COPY	string(30)	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_DATA_FILES.TABLESPACE_NAME For locally managed temporary tablespaces: DBA_TEMP_FILES.TABLESPACE_NAME
Writes/sec (WRITES_RATE)	Number of write operations per second[#2]	AVG	double	No	All	For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: V $FILESTAT.PHYWRTS / seconds in interval For locally managed temporary tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Writes/sec (WRITES_RATE)	Number of write operations per second#2	AVG	double	No	All	V $TEMPSTAT.PHYWRTS / seconds in interval

Minimum Tablespace Interval 2 (PI_PMTS)

Function

The Minimum Tablespace Interval 2 (PI_PMTS) record stores performance data, taken at specific intervals, about tablespaces in a database. PFM - Agent for Oracle creates one record for each tablespace in a database. This is a multi-instance record.

Default and changeable values

Item	Default Value	Changeable
Collection Interval	3600	Y
Collection Offset	1510	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PMTS_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a tablespace

Record Size

- Fixed part: 678 bytes
- Variable part: 163 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
I/O Ops/sec (IO_RATE)	Number of I/O operations per second#2	AVG	double	No	All	(SUM(V $FILESTAT.PHYRDS) + SUM(V $FILESTAT.PHYWRTS)) / seconds in interval
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed#2	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V $FILESTAT.PHYRDS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed[#2]	AVG	double	Yes	All	• For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYRDS)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYWRTS) • For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYWRTS)
Reads/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	SUM(V$FILESTAT.PHYRDS) / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PMTS)[#1]	COPY	string(4)	No	All	Agent Collector
Rollback Segments (ROLLBACK_SEGMENTS)	Number of rollback segments. Performance data about a locally managed tablespace is not collected.[#2]	AVG	ulong	No	All	COUNT(DBA_ROLLBACK_SEGS)
Sort Segments (SORT_SEGMENTS)	Number of sort segments. Performance data about a locally managed permanent tablespace is not collected.[#2]	AVG	ulong	No	All	COUNT(V$SORT_SEGMENT)
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Tablespace name[#1]	COPY	string(30)	No	All	DBA_TABLESPACES.TABLESPACE_NAME
Writes/sec (WRITES_RATE)	Number of write operations per second[#2]	AVG	double	No	All	• For dictionary managed permanent

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported Version	Data Source
Writes/sec (WRITES_RATE)	Number of write operations per second#2	AVG	double	No	All	tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYWRTS) / seconds in interval • For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYWRTS) / seconds in interval

Multi - Threaded Server (PD_PDMT)

Function

The Multi - Threaded Server (PD_PDMT) record stores performance data indicating the status of the multi-thread server (MTS) options at a specific point in time.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance in an MTS environment

Record size

- Fixed part: 894 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Queue Wait (QUEUES_AVERAGE_WAIT)	Average wait time per item in hundredths of a second	--	double	No	All	V$QUEUE.WAIT / V$QUEUE.TOTALQ
Circuits (CIRCUITS)	Number of circuits	--	ulong	No	All	COUNT(V$CIRCUIT)
Dispatchers (DISPATCHERS)	Number of dispatchers	--	ulong	No	All	COUNT(V$DISPATCHER)
Dispatchers Busy (DISPATCHERS_BUSY)	Total busy time of all dispatchers in hundredths of a second	--	double	No	All	SUM(V$DISPATCHER.BUSY)
Dispatchers Busy % (DISPATCHERS_PERCENT_BUSY)	Percentage of the time all dispatchers were busy	--	double	No	All	(SUM(V$DISPATCHER.BUSY) / (SUM(V$DISPATCHER.IDLE) + SUM(V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Dispatchers Busy % (DISPATCHERS_PERCENT_BUSY)	Percentage of the time all dispatchers were busy	--	double	No	All	$DISPATCHER.BUSY)) * 100
Dispatchers Idle (DISPATCHERS_IDLE)	Total idle time of all dispatchers in hundredths of a second	--	double	No	All	SUM(V $DISPATCHER.IDLE)
Dispatchers Idle % (DISPATCHERS_PERCENT_IDLE)	Percentage of the time all dispatchers were idle	--	double	No	All	(SUM(V $DISPATCHER. IDLE) / (SUM(V $DISPATCHER.IDLE) + SUM(V $DISPATCHER.BUSY)) * 100
Dispatchers Messages (DISPATCHERS_MESSAGES)	Total number of messages processed by all dispatchers	--	double	No	All	SUM(V $DISPATCHER.MESSAGES)
Items Queued (QUEUES_QUEUED)	Total number of items currently in all queues	--	double	No	All	SUM(V $QUEUE.QUEUED)
MTS Max Servers % (PERCENT_MTS_MAX_SERVERS)	Ratio (as a percent) of the number of MTS servers to the MTS_MAX_SERVERS parameter value in the init.ora file. The value is always 0.	--	double	No	All	(COUNT(V $SHARED_SERVER) / init.ora MTS_MAX_SERVERS) * 100
MTS Servers Highwater (SERVERS_HIGHWATER)	Maximum number of multithread servers (MTS) that were running at any one time since the instance started.	--	long	No	All	• Which does not have any locally managed temporary tablespaces: V $MTS.SERVERS_HIGHWATER • Which has locally managed temporary tablespaces: V $SHARED_SERVER_MONITOR.SERVERS_HIGHWATER
MTS Servers Started (SERVERS_STARTED)	Started Total number of multithread servers (MTS) since the instance started (this value does not include multithread servers that started during startup processing)	--	long	No	All	• Which does not have any locally managed temporary tablespaces: V $MTS.SERVERS_STARTED • Which has locally managed temporary tablespaces: V $SHARED_SERVER_MONITOR.SERVERS_STARTED

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
MTS Servers Terminated (SERVERS_TERMINATED)	Total number of multithread servers (MTS) that were terminated by Oracle since the instance started	--	double	No	All	• Which does not have any locally managed temporary tablespaces: V$MTS.SERVERS_TERMINATED • Which has locally managed temporary tablespaces: V$SHARED_SERVER_MONITOR.SERVERS_TERMINATED
PQS Busy (PQS_BUSY)	Length of time all parallel query servers were busy (in minutes)	--	double	No	All	SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL)
PQS Busy % (PQS_PERCENT_BUSY)	Percentage of the time all parallel query servers were busy	--	double	No	All	(SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL) / (SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) + SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL))) * 100
PQS CPU (PQS_CPU)	CPU time used by all parallel query servers to process SQL statements	--	double	No	All	SUM(V$PQ_SLAVE.CPU_SECS_TOTAL)
PQS Idle (PQS_IDLE)	Length of time all parallel query servers were idle (in minutes)	--	double	No	All	SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL)
PQS Idle % (PQS_PERCENT_IDLE)	Percentage of the time all parallel query servers were idle	--	double	No	All	(SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) / (SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) + SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL))) * 100
PQS Msgs Rcvd (PQS_MSGS_RCVD)	Total number of messages received by all parallel query servers	--	double	No	All	SUM(V$PQ_SLAVE.MSGS_RCVD_TOTAL)
PQS Msgs Sent (PQS_MSGS_SENT)	Total number of messages sent by all parallel query servers	--	double	No	All	SUM(V$PQ_SLAVE.MSGS_SENT_TOTAL)
PQS Sessions (PQS_SESSIONS)	Number of sessions being used by parallel query servers	--	ulong	No	All	SUM(V$PQ_SLAVE.SESSIONS)
Parallel Query Servers (PARALLEL_QUERY_SERVERS)	Number of parallel query servers	--	long	No	All	COUNT(V$PQ_SLAVE)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Queues (QUEUES)	Number of queues	--	ulong	No	All	COUNT(V$QUEUE)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDMT)	--	string(4)	No	All	Agent Collector
Shared Servers (SHARED_SERVERS)	Number of shared servers	--	long	No	All	COUNT(V$SHARED_SERVER)
Shared Servers Busy (SHARED_SERVERS_BUSY)	Total length of time during which all shared servers were busy, in hundredths of a second	--	double	No	All	SUM(V$SHARED_SERVER.BUSY)
Shared Servers Busy % (SHARED_SERVERS_PERCENT_BUSY)	Percentage of the time during which all shared servers were busy	--	double	No	All	(V$SHARED_SERVER.BUSY / (V$SHARED_SERVER.IDLE + V$SHARED_SERVER.BUSY)) * 100
Shared Servers Idle (SHARED_SERVERS_IDLE)	Total length of time during which all shared servers were idle, in hundredths of a second	--	double	No	All	SUM(V$SHARED_SERVER.IDLE)
Shared Servers Idle % (SHARED_SERVERS_PERCENT_IDLE)	Percentage of the time during which all shared servers were idle	--	double	No	All	(V$SHARED_SERVER.IDLE / (V$SHARED_SERVER.IDLE + $SHARED_SERVER.BUSY)) * 100
Shared Servers Messages (SHARED_SERVERS_MESSAGES)	Total number of messages processed by all shared servers	--	double	No	All	SUM(V$SHARED_SERVER.MESSAGES)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Total Items Queued (QUEUES_TOTAL_QUEUED)	Total number of items in all queues	--	double	No	All	SUM(V$QUEUE.TOTALQ)
Total Queues Wait (QUEUES_WAIT)	Total length of time all items were queued in hundredths of a second	--	double	No	All	SUM(V$QUEUE.WAIT)

Multi - Threaded Server Interval (PI_PIMT)

Function

The Multi - Threaded Server Interval (PI_PIMT) record stores performance data, taken at specific intervals, about the multi-thread server (MTS) options.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance in an MTS environment

Record size

- Fixed part: 1,274 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Queue Wait (QUEUES_AVERAGE_WAIT)	Average wait time per item in hundredths of a second[#2]	AVG	double	No	All	V$QUEUE.WAIT / V$QUEUE.TOTALQ
Circuits (CIRCUITS)	Number of circuits[#2]	AVG	ulong	No	All	COUNT(V$CIRCUIT)
Dispatchers (DISPATCHERS)	Number of dispatchers[#2]	AVG	ulong	No	All	COUNT(V$DISPATCHER)
Dispatchers Busy (DISPATCHERS_BUSY)	Total busy time of all dispatchers in hundredths of a second[#2]	AVG	double	Yes	All	SUM(V$DISPATCHER.BUSY)
Dispatchers Busy % (DISPATCHERS_PERCENT_BUSY)	Percentage of the time all dispatchers were busy[#2]	AVG	double	No	All	(SUM(V$DISPATCHER.BUSY) / (SUM(V$DISPATCHER.IDLE) + SUM(V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Dispatchers Busy % (DISPATCHERS_PERCENT_BUSY)	Percentage of the time all dispatchers were busy[#2]	AVG	double	No	All	$DISPATCHER.BUSY)) * 100
Dispatchers Idle (DISPATCHERS_IDLE)	Total idle time of all dispatchers in hundredths of a second[#2]	AVG	double	Yes	All	SUM(V $DISPATCHER.IDLE)
Dispatchers Idle % (DISPATCHERS_PERCENT_IDLE)	Percentage of the time all dispatchers were idle[#2]	AVG	double	No	All	(SUM(V $DISPATCHER.IDLE) / (SUM(V $DISPATCHER.IDLE) + SUM(V $DISPATCHER.BUSY)) * 100
Dispatchers Messages (DISPATCHERS_MESSAGES)	Total number of messages processed by all dispatchers[#2]	AVG	double	Yes	All	SUM(V $DISPATCHER.MESSAGES)
Items Queued (QUEUES_QUEUED)	Total number of items currently in all queues[#2]	AVG	double	No	All	SUM(V $QUEUE.QUEUED)
MTS Max Servers % (PERCENT_MTS_MAX_SERVERS)	Ratio (as a percent) of the number of MTS servers to the MTS_MAX_SERVERS parameter value in the init.ora file. The value is always 0.[#2]	AVG	double	No	All	(COUNT(V $SHARED_SERVER) / init.ora MTS_MAX_SERVERS) * 100
MTS Servers Highwater (SERVERS_HIGHWATER)	Maximum number of multi-thread servers (MTS) that were running at any one time since the instance started[#2]	HILO	long	No	All	• Which does not have any locally managed temporary tablespaces: V $MTS.SERVERS_HIGHWATER • Which has locally managed temporary tablespaces: V $SHARED_SERVER_MONITOR.SERVERS_HIGHWATER
MTS Servers Started (SERVERS_STARTED)	Total number of multithread servers (MTS) since the instance started (this value does not include multi-thread servers that started during startup processing)[#2]	AVG	long	Yes	All	• Which does not have any locally managed temporary tablespaces: V $MTS.SERVERS_STARTED • Which has locally managed temporary tablespaces: V $SHARED_SERVER_MONITOR.SERVERS_STARTED

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
MTS Servers Terminated (SERVERS_TERMINATED)	Total number of multithread servers (MTS) that were terminated by Oracle since the instance started[#2]	AVG	double	Yes	All	• Which does not have any locally managed temporary tablespaces: V$MTS.SERVERS_TERMINATED • Which has locally managed temporary tablespaces: V$SHARED_SERVER_MONITOR.SERVERS_TERMINATED
PQS Busy (PQS_BUSY)	Length of time all parallel query servers were busy (in minutes)[#2]	AVG	double	Yes	All	SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL)
PQS Busy % (PQS_PERCENT_BUSY)	Percentage of the time all parallel query servers were busy[#2]	AVG	double	No	All	(SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL) / (SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) + SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL))) * 100
PQS CPU (PQS_CPU)	CPU time used by all parallel query servers to process SQL statements[#2]	AVG	double	Yes	All	SUM(V$PQ_SLAVE.CPU_SECS_TOTAL)
PQS Idle (PQS_IDLE)	Length of time all parallel query servers were idle (in minutes)[#2]	AVG	double	Yes	All	SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL)
PQS Idle % (PQS_PERCENT_IDLE)	Percentage of the time all parallel query servers were idle[#2]	AVG	double	No	All	(SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) / (SUM(V$PQ_SLAVE.IDLE_TIME_TOTAL) + SUM(V$PQ_SLAVE.BUSY_TIME_TOTAL))) * 100
PQS Msgs Rcvd (PQS_MSGS_RCVD)	Total number of messages received by all parallel query servers[#2]	AVG	double	Yes	All	SUM(V$PQ_SLAVE.MSGS_RCVD_TOTAL)
PQS Msgs Sent (PQS_MSGS_SENT)	Total number of messages sent by all parallel query servers[#2]	AVG	double	Yes	All	SUM(V$PQ_SLAVE.MSGS_SENT_TOTAL)
PQS Sessions (PQS_SESSIONS)	Number of sessions being used by parallel query servers[#2]	AVG	ulong	Yes	All	SUM(V$PQ_SLAVE.SESSIONS)
Parallel Query Servers (PARALLEL_QUERY_SERVERS)	Number of parallel query servers[#2]	AVG	long	No	All	COUNT(V$PQ_SLAVE)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Queues (QUEUES)	Number of queues[#2]	AVG	ulong	No	All	COUNT(V$QUEUE)
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIMT)[#1]	COPY	string(4)	No	All	Agent Collector
Shared Server Idle % (SHARED_SERVERS_PERCENT_IDLE)	Percentage of the time during which all shared servers were idle[#2]	AVG	double	No	All	(V$SHARED_SERVER.IDLE / (V$SHARED_SERVER.IDLE + $SHARED_SERVER.BUSY)) * 100
Shared Servers (SHARED_SERVERS)	Number of shared servers[#2]	AVG	long	No	All	COUNT(V$SHARED_SERVER)
Shared Servers Busy (SHARED_SERVERS_BUSY)	Total length of time during which all shared servers were busy, in hundredths of a second[#2]	AVG	double	Yes	All	SUM(V$SHARED_SERVER.BUSY)
Shared Servers Busy % (SHARED_SERVERS_PERCENT_BUSY)	Percentage of the time during which all shared servers were busy[#2]	AVG	double	No	All	(V$SHARED_SERVER.BUSY / (V$SHARED_SERVER.IDLE + V$SHARED_SERVER.BUSY)) * 100
Shared Servers Idle (SHARED_SERVERS_IDLE)	Total length of time during which all shared servers were idle, in hundredths of a second[#2]	AVG	double	Yes	All	SUM(V$SHARED_SERVER.IDLE)
Shared Servers Messages (SHARED_SERVERS_MESSAGES)	Total number of messages processed by all shared servers[#2]	AVG	double	Yes	All	SUM(V$SHARED_SERVER.MESSAGES)
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Total ItemsQueued (QUEUES_TOTAL_QUEUED)	Total number of items in all queues[#2]	AVG	double	Yes	All	SUM(V$QUEUE.TOTALQ)
Total Queue Wait (QUEUES_WAIT)	Total length of time all items were queued in hundredths of a second[#2]	AVG	double	Yes	All	SUM(V$QUEUE.WAIT)

Open Cursor (PD_PDOC)

Function

The Open Cursor (PD_PDOC) record stores performance data indicating the status (at a specific point in time) of cursors. PFM - Agent for Oracle creates one record for each open cursor in the system. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	75	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDOC_SID
- PD_PDOC_ADDRHASH

Lifetime

From the opening to the closing of the cursor

Record size

- Fixed part: 678 bytes
- Variable part: 184 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addrhash (ADDRHASH)	Character string that identifies the SQL statement being executed	--	string(38)	No	All	V$OPEN_CURSOR.ADDRESS + V$OPEN_CURSOR.HASH_VALUE
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V$SESSION.PROGRAM where V$OPEN_CURSOR.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDOC)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
SID (SID)	Session ID	--	ulong	No	All	V$SESSION.SID where V$OPEN_CURSOR.SID = V$SESSION.SID
SQL Text (SQL_TEXT)	First 60 characters of the SQL statement that is analyzed by the open cursor	--	string(60)	No	All	V$OPEN_CURSOR.SQL_TEXT
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$OPEN_CURSOR.SID = V$SESSION.SID

Options Installed (PD_PDO)

Function

The Options Installed (PD_PDO) record stores performance data indicating the status of the software options installed on the Oracle Server at a specific point in time. PFM - Agent for Oracle creates one record for each installation option. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	70	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDO_PARAMETER

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 130 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Parameter (PARAMETER)	Option name	--	string(64)	No	All	V$OPTION.PARAMETER
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDO)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Value (VALUE)	Whether the option has been specified. Valid values are TRUE (specified) and FALSE (not specified).	--	string(64)	No	All	V$OPTION.VALUE

Parallel Query Server (PD_PDPQ)

Function

The Parallel Query Server (PD_PDPQ) record stores performance data indicating the status of parallel query servers at a specific point in time. PFM - Agent for Oracle creates one record for each parallel query server of an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	55	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDPQ_SLAVE_NAME

Lifetime

From the start to the end of a Parallel Execution Server

Record size

- Fixed part: 678 bytes
- Variable part: 90 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Busy % (PERCENT_BUSY)	Percentage of time the query server was busy	--	double	No	All	(V $PQ_SLAVE.BUSY_TIME_TOTAL / (V $PQ_SLAVE.IDLE_TIME_TOTAL + V $PQ_SLAVE.BUSY_TIME_TOTAL)) * 100
Current Busy Time (BUSY_TIME_CUR)	Length of time the query server was busy during the SQL statement processing in the session that was active during data collection	--	ulong	No	All	V $PQ_SLAVE.BUSY_TIME_CUR
Current CPU Secs (CPU_SECS_CUR)	CPU time used in the session that was active during data collection	--	ulong	No	All	V $PQ_SLAVE.CPU_SECS_CUR

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Current Idle Time (IDLE_TIME_CUR)	Length of time the query server was idle during SQL statement processing in the session that was active during data collection	--	ulong	No	All	V $PQ_SLAVE.IDLE_TIME_CUR
Current Msgs Rcvd (MSGS_RCVD_CUR)	Number of messages received during SQL statement processing by the session that was active during data collection	--	ulong	No	All	V $PQ_SLAVE.MSGS_RCVD_CUR
Current Msgs Sent (MSGS_SENT_CUR)	Number of messages sent during SQL statement processing by the session that was active during data collection	--	ulong	No	All	V $PQ_SLAVE.MSGS_SENT_CUR
Idle % (PERCENT_IDLE)	Number of messages sent during SQL statement processing by the session that was active during data collection	--	double	No	All	(V $PQ_SLAVE.IDLE_TIME_TOTAL / (V $PQ_SLAVE.IDLE_TIME_TOTAL + V $PQ_SLAVE.BUSY_TIME_TOTAL)) * 100
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDPQ)	--	string(4)	No	All	Agent Collector
Sessions (SESSIONS)	Number of sessions in use by the parallel query server	--	ulong	No	All	V $PQ_SLAVE.SESSIONS
Slave Name (SLAVE_NAME)	Name of the parallel query server	--	string(4)	No	All	V $PQ_SLAVE.SLAVE_NAME
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Status of the parallel query server during data collection. Valid values are BUSY and IDLE.	--	string(4)	No	All	V$PQ_SLAVE.STATUS
Total Busy Time (BUSY_TIME_TOTAL)	Total length of time the query server was active during the interval	--	double	No	All	V $PQ_SLAVE.BUSY_TIME_TOTAL
Total CPU Secs (CPU_SECS_TOTAL)	Total CPU time used by the query server to process SQL statements during the interval	--	double	No	All	V $PQ_SLAVE.CPU_SECS_TOTAL

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Total Idle Time (IDLE_TIME_TOTAL)	Total length of time the query server was idle during the interval	--	double	No	All	V$PQ_SLAVE.IDLE_TIME_TOTAL
Total Msgs Rcvd (MSGS_RCVD_TOTAL)	Total number of messages received by the query server during the interval	--	double	No	All	V$PQ_SLAVE.MSGS_RCVD_TOTAL
Total Msgs Sent (MSGS_SENT_TOTAL)	Total number of messages sent by the query server during the interval	--	double	No	All	V$PQ_SLAVE.MSGS_SENT_TOTAL

Parallel Query Server Interval (PI_PIPQ)

Function

The Parallel Query Server Interval (PI_PIPQ) record stores performance data, taken at specific intervals, about parallel query servers. PFM - Agent for Oracle creates one record for each parallel query server of an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	45	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIPQ_SLAVE_NAME

Lifetime

From the start to the end of a Parallel Execution Server

Record size

- Fixed part: 678 bytes
- Variable part: 246 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Busy % (PERCENT_BUSY)	Percentage of time the query server was busy[#2]	AVG	double	No	All	(V $PQ_SLAVE.BUSY_TIME_TOTAL / (V $PQ_SLAVE.IDLE_TIME_TOTAL + V $PQ_SLAVE.BUSY_TIME_TOTAL)) * 100
Current Busy Time (BUSY_TIME_CUR)	Length of time the query server was busy during the SQL statement processing in the session that was active during data collection[#2]	AVG	ulong	No	All	V $PQ_SLAVE.BUSY_TIME_CUR
Current CPU Secs (CPU_SECS_CUR)	CPU time used in the session that was active during data collection[#2]	AVG	ulong	No	All	V $PQ_SLAVE.CPU_SECS_CUR

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Current Idle Time (IDLE_TIME_CUR)	Length of time the query server was idle during SQL statement processing in the session that was active during data collection[#2]	AVG	ulong	No	All	V $PQ_SLAVE.IDLE_TIME_CUR
Current Msgs Rcvd (MSGS_RCVD_CUR)	Number of messages received during SQL statement processing by the session that was active during data collection[#2]	AVG	ulong	No	All	V $PQ_SLAVE.MSGS_RCVD_CUR
Current Msgs Sent (MSGS_SENT_CUR)	Number of messages sent during SQL statement processing by the session that was active during data collection[#2]	AVG	ulong	No	All	V $PQ_SLAVE.MSGS_SENT_CUR
Idle % (PERCENT_IDLE)	Percentage of time the query server was idle[#2]	AVG	double	No	All	(V $PQ_SLAVE.IDLE_TIME_TOTAL / (V $PQ_SLAVE.IDLE_TIME_TOTAL + V $PQ_SLAVE.BUSY_TIME_TOTAL)) * 100
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIPQ)[#1]	COPY	string(4)	No	All	Agent Collector
Sessions (SESSIONS)	Number of sessions in use by the parallel query server[#2]	AVG	ulong	Yes	All	V $PQ_SLAVE.SESSIONS
Slave Name (SLAVE_NAME)	Name of the parallel query server[#1]	COPY	string(4)	No	All	V $PQ_SLAVE.SLAVE_NAME
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	Status of the parallel query server during data collection. Valid values are BUSY and IDLE.[#1]	COPY	string(4)	No	All	V$PQ_SLAVE.STATUS
Total Busy Time (BUSY_TIME_TOTAL)	Total length of time the query server was active during the interval[#2]	AVG	double	Yes	All	V $PQ_SLAVE.BUSY_TIME_TOTAL
Total CPU Secs (CPU_SECS_TOTAL)	Total CPU time used by the query server to process SQL statements during the interval[#2]	AVG	double	Yes	All	V $PQ_SLAVE.CPU_SECS_TOTAL

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Total Idle Time (IDLE_TIME_TOTAL)	Total length of time the query server was idle during the interval[#2]	AVG	double	Yes	All	V$PQ_SLAVE.IDLE_TIME_TOTAL
Total Msgs Rcvd (MSGS_RCVD_TOTAL)	Total number of messages received by the query server during the interval[#2]	AVG	double	Yes	All	V$PQ_SLAVE.MSGS_RCVD_TOTAL
Total Msgs Sent (MSGS_SENT_TOTAL)	Total number of messages sent by the query server during the interval[#2]	AVG	double	Yes	All	V$PQ_SLAVE.MSGS_SENT_TOTAL

Parallel Query Statistics (PD_PDPS)

Function

PThe Parallel Query Statistics (PD_PDPS) record stores performance data indicating the status of parallel query options at a specific point in time. PFM - Agent for Oracle creates one record for each statistica l value. To collect this record, parallel query servers must be running.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	90	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDPS_STATISTIC

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 39 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDPS)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Statistic (STATISTIC)	Name of the statistics	--	string(30)	No	All	V$PQ_SYSSTAT.NAME
Value (VALUE)	Statistical value	--	double	No	All	V $PQ_SYSSTAT.VALUE

Parameter Values (PD_PDP)

Function

The Parameter Values (PD_PDP) record stores performance data indicating the status (at a specific point in time) of current parameter values. PFM - Agent for Oracle creates one record for each parameter. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	85	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDP_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 588 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Is Default (IS_DEFAULT)	Whether or not the value is the default. Valid values are TRUE and FALSE.	--	string(9)	No	All	V$PARAMETER.ISDEFAULT
Parameter Name (NAME)	Parameter name	--	string(64)	No	All	V$PARAMETER.NAME
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDP)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Value (VALUE)	Parameter value	--	string(512)	No	All	V$PARAMETER.VALUE

Process Detail (PD_PDOP)

Function

The Process Detail (PD_PDOP) record stores performance data indicating the status of processes at a specific point in time. PFM - Agent for Oracle creates one record for each process in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	80	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDOP_PID

Lifetime

From the start to the end of a process

Record size

- Fixed part: 678 bytes
- Variable part: 133 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Background (BACKGROUND)	Whether this is a background process. Valid values are Y (background process) and N (normal process).	--	string(1)	No	All	V$PROCESS.BACKGROUND
Latchspin (LATCHSPIN)	Address of the latch in spin status (if there is no such latch, the value is null)	--	string(16)	No	All	V$PROCESS.LATCHSPIN
Latchwait (LATCHWAIT)	Address of the latch in wait status (if there is no such latch, the value is null)	--	string(16)	No	All	V$PROCESS.LATCHWAIT
Oracle PID (PID)	Oracle process ID	--	ulong	No	All	V$PROCESS.PID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V$PROCESS.PROGRAM
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDOP)	--	string(4)	No	All	Agent Collector
SPID (SPID)	OS's process ID	--	string(12)	No	All	V$PROCESS.SPID
Serial # (SERIAL_NUM)	Process serial number	--	ulong	No	All	V$PROCESS.SERIAL#
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Terminal (TERMINAL)	OS's terminal ID	--	string(10)	No	All	V$PROCESS.TERMINAL
User (USERNAME)	OS's process user name (for a 2-task user that accesses via a network, -T is added to the user name)	--	string(15)	No	All	V$PROCESS.USERNAME

Queue Statistics (PD_PDQU)

Function

The Queue Statistics (PD_PDQU) record stores performance data indicating the status of queues at a specific point in time. PFM - Agent for Oracle creates one record for each queue in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	95	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDQU_PID
- PD_PDQU_TYPE

Lifetime

From the start to the end of a process that has a queue

Record size

- Fixed part: 678 bytes
- Variable part: 47 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait (AVERAGE_WAIT)	Average wait time per item in hundredths of a second	--	double	No	All	V$QUEUE.WAIT / V$QUEUE.TOTALQ
Oracle PID (PID)	Queue's Oracle process ID	--	ulong	No	All	V$PROCESS.PID where V$QUEUE.PADDR = V$PROCESS.ADDR
Queued (QUEUED)	Number of items in the queue	--	double	No	All	V$QUEUE.QUEUED
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDQU)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Total Queued (TOTAL_QUEUED)	Total number of items in the queue	--	double	No	All	V$QUEUE.TOTALQ
Type (TYPE)	Type of queue. Valid values are COMMON (process for each server), DISPATCHER, and OUTBOUND (in use by remote server).	--	string(10)	No	All	V$QUEUE.TYPE
Wait (WAIT)	Total length of time all items were queued in hundredths of a second	--	double	No	All	V$QUEUE.WAIT

Resource Limit (PD_PDRL)

Function

The Resource Limit (PD_PDRL) record stores performance data indicating the usage of global system resources at a specific point in time. PFM - Agent for Oracle creates one record for each system resource. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	25	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDRL_RESOURCE_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 77 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Current Utilization (CURRENT_UTILIZATION)	Number of locks, resources, or processes currently in use	--	double	No	All	V$RESOURCE_LIMIT.CURRENT_UTILIZATION
Initial Allocation (INITIAL_ALLOCATION)	Initial allocation (value specified in initialization parameter file)	--	string(10)	No	All	V$RESOURCE_LIMIT.INITIAL_ALLOCATION
Limit Value (LIMIT_VALUE)	Limit value for locks and resources (the value of this field can be greater than the initial allocation value; in the case of unlimited allocation, the value of this field is UNLIMITED)	--	string(10)	No	All	V$RESOURCE_LIMIT.LIMIT_VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Max Utilization (MAX_UTILIZATION)	Maximum amount of resources used since the last instance started	--	double	No	All	V$RESOURCE_LIMIT.MAX_UTILIZATION
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDRL)	--	string(4)	No	All	Agent Collector
Resource Name (RESOURCE_NAME)	Resource name	--	string(30)	No	All	V$RESOURCE_LIMIT.RESOURCE_NAME
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Utilization % (UTILIZATION_PERCENT)	Ratio (as a percent) of the current utilization value to the maximum utilization value	--	double	No	All	CURRENT_UTILIZATION / MAX_UTILIZATION * 100

Rollback Segment (PD_PDRS)

Function

The Rollback Segment (PD_PDRS) record stores performance data indicating the status of each rollback segment in a database at a specific point in time. PFM - Agent for Oracle creates one record for each rollback segment. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDRS_USN

Lifetime

From the creation to the deletion of a rollback segment

Record size

- Fixed part: 678 bytes
- Variable part: 190 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Active (AVG_ACTIVE)	Average size of an active extent that has uncommitted transaction data during data collection	--	double	No	All	V$ROLLSTAT.AVEACTIVE
Avg Shrink (AVG_SHRINK)	Total size of released extents divided by the number of shrinks Extends	--	double	No	All	V$ROLLSTAT.AVESHRINK
Extends (EXTENDS)	Number of times the rollback segment was extended to obtain new extents	--	double	No	All	V$ROLLSTAT.EXTENDS
Extents (EXTENTS)	Number of extents in the rollback segment	--	ulong	No	All	V$ROLLSTAT.EXTENTS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Gets (GETS)	Number of times header was obtained	--	double	No	All	V$ROLLSTAT.GETS
HWM Size (HWM_SIZE)	Maximum size of rollback segment	--	double	No	All	V$ROLLSTAT.HWMSIZE
Hit % (HIT_PERCENTAGE)	Percentage of times that a segment header was obtained without waiting	--	double	No	All	((V$ROLLSTAT.GETS - V$ROLLSTAT.WAITS) / V$ROLLSTAT.GETS) * 100
Mbytes (BYTES)	Size of rollback segment in megabytes	--	double	No	All	V$ROLLSTAT.RSSIZE / (1024 * 1024)
Optimal Size (OPT_SIZE)	Optimum size of rollback segment	--	double	No	All	V$ROLLSTAT.OPTSIZE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDRS)	--	string(4)	No	All	Agent Collector
Segment Name (NAME)	Name of the rollback segment	--	string(30)	No	All	V$ROLLNAME.NAME
Shrinks (SHRINKS)	Number of times the rollback segment shrank by deleting at least one additional extent	--	double	No	All	V$ROLLSTAT.SHRINKS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Status. Valid values are ONLINE (segment is online) and PENDINGOFFLINE (segment is offline, but several active transactions are used in the rollback segment; segment is placed offline when transactions are completed).	--	string(15)	No	All	V$ROLLSTAT.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Tablespace Name (TABLESPACE_NAME)	Name of tablespace that contains the segment	--	string(30)	No	All	DBA_ROLLBACK_SEGS.TABLESPACE_NAME
Transactions (TRANSACTIONS)	Number of active transactions	--	long	No	All	V$ROLLSTAT.XACTS
USN (USN)	Rollback segment number	--	double	No	All	V$ROLLSTAT.USN
Waits (WAITS)	Number of header waits	--	double	No	All	V$ROLLSTAT.WAITS
Wraps (WRAPS)	Number of times rollback segment was wrapped from one extent to another	--	double	No	All	V$ROLLSTAT.WRAPS
Writes (WRITES)	Number of bytes written in the rollback segment	--	double	No	All	V$ROLLSTAT.WRITES

Rollback Segment Interval (PI_PIRS)

Function

The Rollback Segment Interval (PI_PIRS) record stores performance data, taken at specific intervals, about each rollback segment in a database. PFM - Agent for Oracle creates one record for each rollback segment. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	20	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIRS_USN

Lifetime

From the creation to the deletion of a rollback segment

Record size

- Fixed part: 678 bytes
- Variable part: 426 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Active (AVG_ACTIVE)	Average size of an active extent that has uncommitted transaction data during data collection[#2]	AVG	double	No	All	V$ROLLSTAT.AVEACTIVE
Avg Shrink (AVG_SHRINK)	Total size of released extents divided by the number of shrinks[#2]	AVG	double	No	All	V$ROLLSTAT.AVESHRINK
Extends (EXTENDS)	Number of times the rollback segment was extended to obtain new extents[#2]	HILO	double	Yes	All	V$ROLLSTAT.EXTENDS
Extents (EXTENTS)	Number of extents in the rollback segment[#2]	AVG	ulong	No	All	V$ROLLSTAT.EXTENTS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Gets (GETS)	Number of times header was obtained[#2]	AVG	double	Yes	All	V$ROLLSTAT.GETS
HWM Size (HWM_SIZE)	Maximum size of rollback segment[#2]	AVG	double	No	All	V$ROLLSTAT.HWMSIZE
Hit % (HIT_PERCENTAGE)	Percentage of times that a segment header was obtained without waiting[#2]	AVG	double	No	All	((V$ROLLSTAT.GETS - V$ROLLSTAT.WAITS) / V$ROLLSTAT.GETS) * 100
Mbytes (BYTES)	Size of rollback segment in megabytes[#2]	AVG	double	No	All	V$ROLLSTAT.RSSIZE / (1024 * 1024)
Optimal Size (OPT_SIZE)	Optimum size of rollback segment[#2]	AVG	double	No	All	V$ROLLSTAT.OPTSIZE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIRS)[#1]	COPY	string(4)	No	All	Agent Collector
Segment Name (NAME)	Name of the rollback[#1]	COPY	string(30)	No	All	V$ROLLNAME.NAME
Shrinks (SHRINKS)	Number of times the rollback segment shrank by deleting at least one additional extent[#2]	HILO	double	Yes	All	V$ROLLSTAT.SHRINKS
Size Change (SIZE_CHANGE)	Change to the size of rollback segment[#2]	AVG	double	No	All	V$ROLLSTAT.RSSIZE
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	Status. Valid values are ONLINE (segment is online) and PENDINGOFFLINE (segment is offline, but several active transactions are used in the rollback segment; segment is placed offline when	COPY	string(15)	No	All	V$ROLLSTAT.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	transactions are completed).[#1]	COPY	string(15)	No	All	V$ROLLSTAT.STATUS
Tablespace Name (TABLESPACE_NAME)	Name of tablespace that contains the segment[#1]	COPY	string(30)	No	All	DBA_ROLLBACK_SEGS.TABLESPACE_NAME
Transactions (TRANSACTIONS)	Number of active transactions[#2]	AVG	long	No	All	V$ROLLSTAT.XACTS
USN (USN)	Rollback segment number[#1]	COPY	double	No	All	V$ROLLSTAT.USN
Waits (WAITS)	Number of header waits[#2]	AVG	double	Yes	All	V$ROLLSTAT.WAITS
Wraps (WRAPS)	Number of times rollback segment was wrapped from one extent to another[#2]	HILO	double	Yes	All	V$ROLLSTAT.WRAPS
Writes (WRITES)	Number of bytes written in the rollback segment[#2]	AVG	double	Yes	All	V$ROLLSTAT.WRITES

Segment Detail (PD_PDSM)

Function

The Segment Detail (PD_PDSM) record stores performance data indicating the status of database segments at a specific point in time. PFM - Agent for Oracle creates one record for each segment. This is a multi-instance record.

In an environment where many (more than hundreds of thousands of) segments exist, the following might occur:

- The size of the Store database increases.
- Data collection requires a long time.

If you do not need performance data for each segment, consider using alternative records. For example, you can use the Tablespace (PD_PDTS) record to monitor the size of tablespaces and check the number of segments.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	120	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC key fields

- PD_PDSM_SEGMENT_NAME
- PD_PDSM_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a segment

Record size

- Fixed part: 678 bytes
- Variable part: 231 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Segment size in Oracle blocks	--	ulong	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocks (BLOCKS)	Segment size in Oracle blocks	--	ulong	No	All	DBA_SEGMENTS.BLOCKS • For locally managed temporary tablespaces: V$SORT_SEGMENT.TOTAL_BLOCKS
Bytes (BYTES)	Segment size in bytes	--	double	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_SEGMENTS.BYTES • For locally managed temporary tablespaces: DBA_TEMP_FILES.BYTES
Extents (EXTENTS)	Number of extents allocated to this segment	--	ulong	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_SEGMENTS.EXTENTS • For locally managed temporary tablespaces: V$SORT_SEGMENT.TOTAL_EXTENTS
Free list Groups (FREELIST_GROUPS)	Number of free list groups allocated to the segment. Performance data about the locally managed temporary tablespace is not collected.	--	ulong	No	All	DBA_SEGMENTS.FREELIST_GROUPS
Free lists (FREELISTS)	Number of process free lists allocated to the segment. Performance data about the locally managed temporary tablespace is not collected.	--	ulong	No	All	DBA_SEGMENTS.FREELISTS
Header Block (HEADER_BLOCK)	Block ID including the segment header. Performance data about the locally managed temporary tablespace is not collected.	--	ulong	No	All	DBA_SEGMENTS.HEADER_BLOCK

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Header File (HEADER_FILE)	File ID including the segment header. Performance data about the locally managed temporary tablespace is not collected.	--	ushort	No	All	DBA_SEGMENTS.HEADER_FILE
Increase % (PCT_INCREASE)	The percentage increase in size of the extent due to the next extent allocation. Performance data about the locally managed temporary tablespace is not collected.	--	short	No	All	DBA_SEGMENTS.PCT_INCREASE
Initial Extent (INITIAL_EXTENT)	Size of the initial extent for the segment in bytes. Performance data about the locally managed temporary tablespace is not collected.	--	double	No	All	DBA_SEGMENTS.INITIAL_EXTENT
Max Extents (MAX_EXTENTS)	Maximum number of extents permitted for the segment. Performance data about the locally managed temporary tablespace is not collected.	--	ulong	No	All	DBA_SEGMENTS.MAX_EXTENTS
Max Extents % (PERCENT_MAX_EXTENTS)	Ratio (as a percent) of the currently allocated extents to the maximum number of extents permitted for the segment. Performance data about the locally managed temporary tablespace is not collected.	--	double	No	All	(DBA_SEGMENTS.EXTENTS / DBA_SEGMENTS.MAX_EXTENTS) * 100
Min Extents (MIN_EXTENTS)	Minimum number of extents permitted for the segment	--	ulong	No	All	DBA_SEGMENTS.MIN_EXTENTS
Next Alloc Fails (NEXT_ALLOC_FAILS)	Correct values cannot be collected for this field. Failure of the next extent allocation. If it fails, the value of this field is 1.	--	short	No	All	DBA_SEGMENTS.NEXT_EXTENT > MAX(FET$.LENGTH) * DB_BLOCK_SIZE
Next Extent (NEXT_EXTENT)	Size of the next extent for the segment in bytes	--	double	No	All	DBA_SEGMENTS.NEXT_EXTENT
Overextended (OVEREXTENDED)	Correct values cannot be collected for this field. If the number of extents is greater than the specified value (initial value is 5), the value of this field is 1; Otherwise, the value of this field is 0.	--	short	No	All	• For dictionary managed permanent tablespaces, dictionary managed temporary tablespaces, or locally managed permanent tablespaces: DBA_SEGMENTS.EXTENTS > 5 • For locally managed temporary tablespaces:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Overextended (OVEREXTENDED)	Correct values cannot be collected for this field. If the number of extents is greater than the specified value (initial value is 5), the value of this field is 1; Otherwise, the value of this field is 0.	--	short	No	All	V$SORT_SEGMENT.TOTAL_EXTENTS > 5
Owner (OWNER)	Segment owner's user name. Performance data about the locally managed temporary tablespace is not collected.	--	string(30)	No	All	DBA_SEGMENTS.OWNER
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSM)	--	string(4)	No	All	Agent Collector
Segment Name (SEGMENT_NAME)	Segment name. Performance data about the locally managed temporary tablespace is not collected.	--	string(81)	No	All	DBA_SEGMENTS.SEGMENT_NAME
Segment Type (SEGMENT_TYPE)	Segment type. Valid values are CACHE, CLUSTER, DEFERRED ROLLBACK, INDEX, ROLLBACK, TABLE, and TEMPORARY. Performance data about the locally managed temporary tablespace is not collected.	--	string(18)	No	All	DBA_SEGMENTS.SEGMENT_TYPE
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Name of tablespace that contains the segment	--	string(30)	No	All	DBA_SEGMENTS.TABLESPACE_NAME

Server Status (PD_STAT)

Function

The Server Status (PD_STAT) record stores performance data indicating the status of the Oracle database at a specific point in time.

Notes

If you start the JP1/PFM - Agent for Oracle service with startup_always set to Y, the Server Status (PD_STAT) record might not be collected until the Oracle Database is up and running.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 686 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Availability (AVAILABILITY)	Availability status; this value can be either 0 (inactive) or 1 (active)	--	ulong	No	All	Agent Collector
Change Time (CHANGE_TIME)	Last time Availability was changed	--	time_t	No	All	Agent Collector
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always STAT)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

Session Detail (PD_PDS)

Function

The Session Detail (PD_PDS) record stores the performance data indicating the status (at a specific point in time) of sessions. PFM - Agent for Oracle creates one record for each session in an instance. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create the static dictionary view DBA_WAITERS, you need to execute the CATBLOCK.SQL script that is provided by Oracle.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record displays the session of monitoring PDB and common sessions for database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	100	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDS_SID
- PD_PDS_SERIAL_NUM

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 727 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Action (ACTION)	Name of the action that is specified by calling the DBMS_APPLICATION_INFO.SET_ACTION procedure during data collection	--	string(32)	No	All	V$SESSION.ACTION

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addrhash (ADDRHASH)	Character string that identifies the SQL statement being executed	--	string(38)	No	All	V$SESSION.SQL_ADDRESS + V$SESSION.SQL_HASH_VALUE
Auditing SID (AUDSID)	Auditing session ID. The value of this field is -1 when Oracle Database 10g is being monitored or when the value of the User field is SYS.	--	ulong	No	All	V$SESSION.AUDSID
Avg Wait (AVERAGE_WAIT)	Average time of all events that the session is waiting for. In centiseconds (1/100 of a second). To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	ulong	No	All	AVG(V$SESSION_EVENT.AVERAGE_WAIT)
Avg Wait String (AVERAGE_WAIT_STRING)	Average time (character string) of all events that the session is waiting for. In seconds. To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	string(30)	No	All	AVG(V$SESSION_EVENT.AVERAGE_WAIT) / 100
Blocking Locks (BLOCKING_LOCKS)	Number of locks blocking another lock	--	double	No	All	COUNT(V$LOCK) where V$LOCK.BLOCK > 0
Client Info (CLIENT_INFO)	Information specified by calling the DBMS_APPLICATION_INFO.SET_CLIE NT_INFO procedure	--	string(64)	No	All	V$SESSION.CLIENT_INFO
Client PID (PROCESS)	Client ID of the OS	--	string(12)	No	All	V$SESSION.PROCESS
Command (COMMAND)	Command being executed or command number	--	string(32)	No	All	V$SESSION.COMMAND
Container ID (CONTAINER_ID)	The ID of the container to which this session belongs.	--	word	No	Oracle 12c R2 or later	V$SESSION.CON_ID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Container ID (CONTAINER_ID)	This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.	--	word	No	Oracle 12c R2 or later	V$SESSION.CON_ID
Fixed Table Sequence (FIXED_TABLE_SEQUENCE)	Value to be increased each time the session completes a call to the database	--	double	No	All	V$SESSION.FIXED_TABLE_SEQUENCE
Locks Held (LOCKS_HELD)	Number of locks held by the session during data collection	--	double	No	All	COUNT(V$LOCKS) where V$LOCK.LMODE is NOT NULL
Locks Requested (LOCKS_REQUESTED)	Number of requested locks that the session was not holding	--	double	No	All	COUNT(V$LOCKS) where V$LOCK.LMODE is NULL
Lockwait (LOCKWAIT)	Address of the lock the session is waiting for. If there is no such lock, the value is null.	--	string(16)	No	All	V$SESSION.LOCKWAIT
Logon Seconds (LOGON_SECONDS)	Number of seconds since login	--	ulong	No	All	V$SESSION.LOGON_TIME
Logon Time (LOGON_TIME)	Session connection time	--	string(20)	No	All	V$SESSION.LOGON_TIME
Machine (MACHINE)	OS machine name	--	string(64)	No	All	V$SESSION.MACHINE
Module (MODULE)	Name of the module being executed that is specified by calling the DBMS_APPLICATION_INFO.SET_MODULE during data collection	--	string(48)	No	All	V$SESSION.MODULE
OS User (OSUSER)	Client user name of the OS	--	string(30)	No	All	V$SESSION.OSUSER
Open Cursors (OPEN_CURSORS)	Number of open cursors[#11]	--	ulong	No	All	COUNT(V$OPEN_CURSOR)
Oracle PID (PID)	Oracle process ID	--	ulong	No	All	V$PROCESS.PID where V$SESSION.PADDR = V$PROCESS.ADDR

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Oracle Server (SERVER)	Oracle server type. Valid values are DEDICATED, NONE, PSEUDO, and SHARED.	--	string(9)	No	All	V$SESSION.SERVER
Program (PROGRAM)	Program name of OS	--	string(64)	No	All	V$SESSION.PROGRAM
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDS)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID	--	ulong	No	All	V$SESSION.SID
Schema # (SCHEMA_NUM)	Schema user ID	--	long	No	All	V$SESSION.SCHEMA#
Schema Name (SCHEMANAME)	Schema user name	--	string(30)	No	All	V$SESSION.SCHEMANAME
Serial # (SERIAL_NUM)	Session serial number that identifies the session object. The session serial number guarantees that a session-level command is applied to the correct session object even when one session ends and another session with the same session ID starts.	--	ulong	No	All	V$SESSION.SERIAL#
Session Events (SESSION_EVENTS)	Number of events placed in wait status by the session	--	short	No	All	COUNT(V$SESSION_EVENT)
Session Waits (SESSION_WAITS)	Number of waits caused by the session	--	ulong	No	All	COUNT(V$SESSION_WAIT)
Sessions Blocked (SESSIONS_BLOCKED)	Number of sessions blocked by the session[#11]	--	ulong	No	All	COUNT(DBA_WAITERS)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Session status. Valid values are ACTIVE, INACTIVE,	--	string(8)	No	All	V$SESSION.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	KILLED, CACHED, and SNIPED.	--	string(8)	No	All	V$SESSION.STATUS
Table Accesses (TABLE_ACCESSES)	Number of table accesses[#11]	--	double	No	All	COUNT(V$ACCESS)
Terminal (TERMINAL)	Terminal name of the OS	--	string(16)	No	All	V$SESSION.TERMINAL
Time Waited (TIME_WAITED)	Total length of time the session waited for all events. In centiseconds (1/100 of a second). To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	ulong	No	All	SUM(V$SESSION_EVENT.TIME_WAITED)
Time Waited String (TIME_WAITED_STRING)	Total length of time (character string) the session waited for all events. In seconds. To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	string(30)	No	All	SUM(V$SESSION_EVENT.TIME_WAITED) / 100
Total Timeouts (TOTAL_TIMEOUTS)	Total number of timeouts for the session events	--	ulong	No	All	SUM(V$SESSION_EVENT.TOTAL_TIMEOUTS)
Total Waits (TOTAL_WAITS)	Number of waits for all events of the session	--	double	No	All	SUM(V$SESSION_EVENT.TOTAL_WAITS)
Transaction Address (TRANSACTION_ADDRESS)	Address of the transaction state object	--	string(16)	No	All	V$SESSION.TADDR
Transactions (TRANSACTIONS)	Number of active transactions[#11]	--	ulong	No	All	COUNT(V$TRANSACTION)
Type (TYPE)	Session type	--	string(10)	No	All	V$SESSION.TYPE
User (USERNAME)	Oracle user name This field always includes a record for which NULL is set as information for the SYS user. Since null cannot be specified conditionally for connections from	--	string(30)	No	All	V$SESSION.USERNAME

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
User (USERNAME)	users other than special user A, use the User # field to specify the following conditional expressions: User<> "A" AND User #<>"0"	--	string(30)	No	All	V$SESSION.USERNAME
User # (USER_NUM)	Oracle user ID	--	long	No	All	V$SESSION.USER#

#11

The value is 0 for a common session of database instances.

Session Event (PD_PDEV)

Function

The Session Event (PD_PDEV) record stores performance data indicating the status of session events at a specific point in time. PFM - Agent for Oracle creates one record for each event the session waits for. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	45	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDEV_SID
- PD_PDEV_EVENT

Lifetime

From the start to the end of a session event

Record size

- Fixed part: 678 bytes
- Variable part: 241 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait (AVERAGE_WAIT)	Average time of all events the session is waiting for in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	double	No	All	V$SESSION_EVENT.AVERAGE_WAIT
Avg Wait String (AVERAGE_WAIT_STRING)	Average time (character string) of all events the session is waiting for in seconds (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	string(21)	No	All	V$SESSION_EVENT.AVERAGE_WAIT / 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Event (EVENT)	Name of the event the session is waiting for	--	string(64)	No	All	V$SESSION_EVENT.EVENT
Program (PROGRAM)	Name of the program being executed	--	string(64)	No	All	V$SESSION.PROGRAM where V$SESSION_EVENT.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDEV)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID	--	ulong	No	All	V$SESSION_EVENT.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Time Waited (TIME_WAITED)	Total time the session was waiting for events in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	double	No	All	V$SESSION_EVENT.TIME_WAITED
Time Waited String (TIME_WAITED_STRING)	Total time the session was waiting for events in seconds (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	string(21)	No	All	V$SESSION_EVENT.TIME_WAITED / 100
Total Timeouts (TOTAL_TIMEOUTS)	Total number of timeouts for the session events	--	double	No	All	V$SESSION_EVENT.TOTAL_TIMEOUTS
Total Waits (TOTAL_WAITS)	Session's total number of waits for events	--	double	No	All	V$SESSION_EVENT.TOTAL_WAITS
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$SESSION_EVENT.SID = V$SESSION.SID

Session Event Interval (PI_PIEV)

Function

The Session Event Interval (PI_PIEV) record stores performance data, taken at specific intervals, about session events. PFM - Agent for Oracle creates one record for each event the session waits for. This is a multi-instance record.

When Log is set to Yes, if you collect history over a long period of time, because of the short lifetime, it is not summarized in units of years or months. All instances are retained, resulting in a bloated store database. In addition, when the collected history is summarized, more memory is used than necessary. The memory shortage might cause monitoring to stop. If you want to collect history over a long period of time, use the Session Event (PD_PDEV) record for monitoring.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PIEV_SID
- PI_PIEV_EVENT

Lifetime

From the start to the end of a session event

Record size

- Fixed part: 678 bytes
- Variable part: 289 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait (AVERAGE_WAIT)	Average time of all events the session is waiting for in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file) #2	AVG	double	No	All	V$SESSION_EVENT.AVERAGE_WAIT
Avg Wait String (AVERAGE_WAIT_STRING)	Average time (character string) of all events the session is waiting for in seconds (to collect the value of this field, the	COPY	string(21)	No	All	V$SESSION_EVENT.AVERAGE_WAIT / 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait String (AVERAGE_WAIT_STRING)	TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)[#1]	COPY	string(21)	No	All	V$SESSION_EVENT.AVERAGE_WAIT / 100
Event (EVENT)	Name of the event the session is waiting for[#1]	COPY	string(64)	No	All	V$SESSION_EVENT.EVENT
Program (PROGRAM)	Name of the program being executed[#1]	COPY	string(64)	No	All	V$SESSION.PROGRAM where V$SESSION_EVENT.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIEV)[#1]	COPY	string(4)	No	All	Agent Collector
SID (SID)	Session ID[#1]	COPY	ulong	No	All	V$SESSION_EVENT.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Time Waited (TIME_WAITED)	Total time the session was waiting for event in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)[#2]	AVG	double	Yes	All	V$SESSION_EVENT.TIME_WAITED
Time Waited String (TIME_WAITED_STRING)	Total time the session was waiting for event in seconds (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)[#1]	COPY	string(21)	No	All	V$SESSION_EVENT.TIME_WAITED / 100
Total Timeouts (TOTAL_TIMEOUTS)	Total number of timeouts for the session event[#2]	AVG	double	Yes	All	V$SESSION_EVENT.TOTAL_TIMEOUTS
Total Waits (TOTAL_WAITS)	Session's total number of waits for the events[#2]	AVG	double	Yes	All	V$SESSION_EVENT.TOTAL_WAITS
User (USERNAME)	Oracle user name[#1]	COPY	string(30)	No	All	V$SESSION.USERNAME where V$SESSION_EVENT.SID = V$SESSION.SID

Session I/O Interval (PI_PIIO)

Function

The Session I/O Interval (PI_PIIO) record stores performance data, taken at specific intervals, about input/output of all active sessions. PFM - Agent for Oracle creates one record for each active session. This is a multi-instance record.

Note that this record is valid if PFM - Agent for Oracle monitors information about a session whose connection continues for a long time in the Oracle database being monitored. A session of a connection pool is an example of such a session. However, valid data cannot be retrieved when connection and disconnection take place repeatedly because the ODBC key field is PI_PIIO_SID. In this case, when the difference from the previous value is retrieved for a delta, a possibility of retrieving the difference from a different session exists.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record includes I/O information about active sessions of the PDBs that are being monitored, and about common active sessions of database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	25	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIIO_SID

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 186 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes (BLOCK_CHANGES)	Number of times changes were made to session blocks[#2]	AVG	double	Yes	All	V$SESS_IO.BLOCK_CHANGES
Block Gets (BLOCK_GETS)	Number of times session blocks were acquired[#2]	AVG	double	Yes	All	V$SESS_IO.BLOCK_GETS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage[#2]	AVG	double	No	All	100 * (BLOCK_GETS + CONSISTENT_GETS - PHYSICAL_READS) / (BLOCK_GETS + CONSISTENT_GETS)
Consistent Changes (CONSISTENT_CHANGES)	Number of times a consistent change was made in the session[#2]	AVG	double	Yes	All	V$SESS_IO.CONSISTENT_CHANGES
Consistent Gets (CONSISTENT_GETS)	Number of times a consistent acquisition was made in the session[#2]	AVG	double	Yes	All	V$SESS_IO.CONSISTENT_GETS
OS PID (PID)	Client process ID of the OS[#1]	COPY	string(30)	No	All	V$SESSION.PROCESS
Physical Reads (PHYSICAL_READS)	Number of physical read operations in the session[#2]	AVG	double	Yes	All	V$SESS_IO.PHYSICAL_READS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIIO)[#1]	COPY	string(4)	No	All	Agent Collector
SID (SID)	Session ID[#1]	COPY	ulong	No	All	V$SESS_IO.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
User (USERNAME)	User name of the session[#1]	COPY	string(30)	No	All	V$SESSION.USERNAME

Session Stat Summary Interval (PI_PIS2)

Function

The Session Stat Summary Interval (PI_PIS2) record stores performance data, taken at specific intervals, about each session and performance indicator of an instance. PFM - Agent for Oracle creates one record for each session in an instance. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create this view, you must execute the CATBLOCK.SQL script that is provided by Oracle.

This record is useful when you want to monitor session information in the monitored Oracle Database that has a long connection period, such as a connection pool. However, if you repeatedly connect and disconnect during system operation, you can monitor session information using the Session Statistics Summary (PD_PDS2) record. If sessions are frequently connected and disconnected, valid data cannot be obtained under the following circumstance: when attempting to obtain the difference for values of delta items between a current connection and a previous connection, information derived from another session might be substituted for the previous connection values because the key field is PI_PIS2_SID.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record includes information about sessions and performance indicators of the PDBs that are being monitored, and about common sessions and performance indicators of database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	105	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIS2_SID

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 904 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes/Tran (BLOCK_CHANGES_PER_TRANSACTION)	Rate at which each transaction executed database manipulation language (DML) statements[#2]	AVG	double	No	All	db block changes / user commits
Block Visits/Tran (BLOCK_VISITS_PER_TRANSACTION)	Number of work database read operations executed per transaction[#2]	AVG	double	No	All	(db block gets + consistent gets) / user commits
Blocking Locks (BLOCKING_LOCKS)	Number of locks owned by the session that are blocking another lock.[#2] Correct values cannot be collected for this field, and 0 is always displayed.	AVG	double	No	All	Agent Collector
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage[#2]	AVG	double	No	All	(1 - (physical reads cache / (consistent gets from cache + db block gets from cache))) * 100
Calls/Tran (CALLS_PER_TRANSACTION)	Rate at which client requests were executed per transaction[#2]	AVG	double	No	All	user calls / user commits
Changed Block % (CHANGED_BLOCK_PERCENTAGE)	Ratio (as a percent) of the number of queries that change data (insertion, update, and deletion) to the number of queries (search, insertion, update, deletion) executed on the database[#2]	AVG	double	No	All	(db block changes / (block gets + consistent gets)) * 100
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Ratio (as a percent) of the number of times rollback entries were applied to maintain read consistency to the number of read consistency requests by the application[#2]	AVG	double	No	All	(consistent changes / consistent gets) * 100
Continued Row % (CONTINUED_ROW_PERCENTAGE)	Percentage of rows obtained that were longer than one block or had been moved[#2]	AVG	double	No	All	(table fetch continued row / (table fetch by rowid + table scan rows gotten)) * 100
Deadlocks (LOCK_DEADLOCKS)	Number of process deadlocks caused by enqueuing (locking) of DML processing[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Disk Sorts (SORTS_DISK)	Number of disk sort operations[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Lock Conversions (LOCK_CONVERSIONS)	Number of enqueues (locks) whose mode changed (such	AVG	double	Yes	All	V$SESSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Lock Conversions (LOCK_CONVERSIONS)	as from SHARE to EXCLUSIVE)[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Lock Releases (LOCK_RELEASES)	Number of times enqueuing (locking) was released (this statistic is the same as the lock request count)[#2]	AVG	double	Yes	All	SUM(V$SESSTAT.VALUE)
Lock Requests (LOCK_REQUESTS)	Number of times enqueuing (locking) was requested[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Lock Timeouts (LOCK_TIMEOUTS)	Number of times enqueuing (locking) request was not permitted within the specified wait time[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Lock Waits (LOCK_WAITS)	Number of times lock request was placed in wait status (the number of lock requests not placed in wait status equals the number of lock requests minus the number of enqueuing waits)[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Logical Reads (LOGICAL_READS)	Sum of the number of logical read operations in read consistency mode and the number of requests to the current copy of the block[#2]	AVG	double	Yes	All	db block gets + consistent gets
Memory Sorts (SORTS_MEMORY)	Number of sort operations in memory[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Non-Index Lookups % (NON_INDEX_LOOKUPS)	Percentage of full table scans for which no caching is performed[#2]	AVG	double	No	All	(table scans (long tables) / (table scans (short tables) + table scans (long tables))) * 100
PGA Memory (PGA_MEMORY)	Amount of PGA memory in use during data collection (in bytes)[#2]	AVG	double	No	All	V$SESSTAT.VALUE
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations onto disk by DBWR[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Physical Reads (PHYSICAL_READS)	Number of times a database block was actually read from disk[#2]	AVG	double	Yes	All	physical reads - physical reads direct - physical reads direct (lob)
Program (PROGRAM)	OS's program name[#1]	COPY	string(48)	No	All	V$SESSION.PROGRAM
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Type (INPUT_RECORD_TYPE)	Record name (always PIS2)[#1]	COPY	string(4)	No	All	Agent Collector
Recursive Calls (RECURSIVE_CALLS)	Number of user calls processed[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Recursive to User Call % (RECURSIVE_TO_USER_CALL_PERCENTAGE)	Correct values cannot be collected for this field. Percentage indicating overhead[#2]	AVG	double	No	All	(recursive calls / user calls) * 100
Redo Log Space Requests (REDO_LOG_SPACE_REQUESTS)	Number of times that, because the active log file was full, the Oracle server had to wait for disk space to be allocated for a REDO log entry.[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Redo Log Space Wait % (REDO_LOG_SPACE_WAIT_PERCENTAGE)	Wait rate for allocations of the disk area to the REDO log entry[#2]	AVG	double	No	All	(redo log space requests / redo entries) * 100
Row Source % (ROW_SOURCE_PERCENTAGE)	Percentage of obtained rows that were obtained by full-table scans[#2]	AVG	double	No	All	(table scan rows gotten / (table fetch by rowid + table scan rows gotten)) * 100
SID (SID)	Session ID[#1]	COPY	ulong	No	All	V$SESSION.SID
SQL Net Bytes Rcvd (SQL_NET_BYTES_RECEIVED)	Number of bytes received from clients via SQL*Net[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
SQL Net Bytes Sent (SQL_NET_BYTES_SENT)	Number of bytes sent to clients via SQL*Net[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	Number of session cursors cached (the maximum number of cursors that can be cached is determined by the SESSION_CACHED_CURSORS parameter in the init.ora file)[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Session Cursor Cache Hit % (SESSION_CURSOR_CACHE_HIT_PERCENTAGE)	Percentage of the number of session cursors stored in the cache that were found in the cache after an access was made[#2]	AVG	double	No	All	(session cursor cache hits / session cursor cache count) * 100
Session Cursor Cache Hits	Recorded session cursor cache hit count[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(SESSION_CURSOR_CACHE_HITS)	Recorded session cursor cache hit count[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Sort Overflow % (SORT_OVERFLOW_PERCENTAGE)	Percentage of sorts that used temporary segments[#2]	AVG	double	No	All	(sorts (disk) / (sorts (memory) + sorts (disk))) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Statement CPU (STATEMENT_CPU)	Total CPU time used by active statements during data collection in hundredths of a second[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
UGA Memory (UGA_MEMORY)	Amount of session memory used (in bytes)[#2]	AVG	double	No	All	V$SESSTAT.VALUE
User (USERNAME)	Oracle user name[#1]	COPY	string(30)	No	All	V$SESSION.USERNAME
User Calls (USER_CALLS)	Number of user calls processed[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
User Calls / Parse (USER_CALLS_PER_PARSE)	Percentage indicating how well the application is managing the context area[#2]	AVG	double	No	All	user calls / parse count (total)
User Commits (USER_COMMITS)	Number of transactions[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
User Rollback % (USER_ROLLBACK_PERCENTAGE)	Percentage of application transactions that failed[#2]	AVG	double	No	All	(user rollbacks / (user commits + user rollbacks)) * 100
User Rollbacks (USER_ROLLBACKS)	Number of rollbacks[#2]	AVG	double	Yes	All	V$SESSTAT.VALUE
Waiting Locks (WAITING_LOCKS)	Number of locks owned by another session that this session is waiting for. Correct values cannot be collected for this field, and 0 is always displayed.[#2]	AVG	double	No	All	--
Write % (WRITE_PERCENTAGE)	Ratio (as a percent) of physical writes to all physical I/Os (reads and writes)[#2]	AVG	double	No	All	(physical writes / (physical reads + physical writes)) * 100

Session Statistics (PD_PDSS)

Function

The Session Statistics (PD_PDSS) record stores performance data indicating the status of sessions at a specific point in time. PFM - Agent for Oracle creates one record for each session in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	125	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC key fields

- PD_PDSS_SID
- PD_PDSS_STATISTIC_NUM

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 186 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Class (CLASS)	Statistic class	--	string(20)	No	All	V$STATNAME.CLASS where V$SESSTAT.STATISTIC# = V$STATNAME.STATISTIC#
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V$SESSION.PROGRAM where V$SESSTAT.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSS)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID	--	ulong	No	All	V$SESSTAT.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Statistic # (STATISTIC_NUM)	Statistic number	--	double	No	All	V$SESSTAT.STATISTIC#
Statistic Name (NAME)	Statistic name	--	string(64)	No	All	V$STATNAME.NAME where V$SESSTAT.STATISTIC# = V$STATNAME.STATISTIC#
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$SESSTAT.SID = V$SESSION.SID
Value (VALUE)	Statistical value	--	double	No	All	V$SESSTAT.VALUE

Session Statistics Summary (PD_PDS2)

Function

The Session Statistics Summary (PD_PDS2) record stores performance data indicating the status (at a specific point in time) of each session and performance indicator of an instance. PFM - Agent for Oracle create one record for each session in an instance. This is a multi-instance record.

If you cannot view the performance data in this record, create Oracle's static data dictionary view DBA_WAITERS. To create the static dictionary view DBA_WAITERS, you need to execute the CATBLOCK.SQL script that is provided by Oracle.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record includes information about sessions and performance indicators of the PDBs that are being monitored, and about common sessions and performance indicators of database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	105	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDS2_SID

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 412 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes/Tran (BLOCK_CHANGES_PER_TRANSACTION)	Rate at which each transaction executed DML operation	--	double	No	All	db block changes / user commits
Block Visits/Tran (BLOCK_VISITS_PER_TRANSACTION)	Number of work database read operations executed per transaction	--	double	No	All	(db block gets + consistent gets) / user commits

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Blocking Locks (BLOCKING_LOCKS)	Number of locks owned by the session that are blocking another lock Correct values cannot be collected in this field, and 0 is always displayed.	--	double	No	All	Agent Collector
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage	--	double	No	All	(1 - (physical reads cache / (consistent gets from cache + db block gets from cache))) * 100
Calls/Tran (CALLS_PER_TRANSACTION)	Rate at which client requests were executed per transaction	--	double	No	All	user calls / user commits
Changed Block % (CHANGED_BLOCK_PERCENTAGE)	Percentage indicating the balance between queries and DML in the database application. This value changes according to indexes and application utilization status.	--	double	No	All	(db block changes / (block gets + consistent gets)) * 100
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Percentage indicating the extents for which the application needed to check the consistency of read operations	--	double	No	All	(consistent changes / consistent gets) * 100
Continued Row % (CONTINUED_ROW_PERCENTAGE)	Percentage ratio of rows obtained that were longer than one block or had been moved	--	double	No	All	(table fetch continued row / (table fetch by rowid + table scan rows gotten)) * 100
Deadlocks (LOCK_DEADLOCKS)	Number of process deadlocks caused by enqueuing (locking) of DML processing	--	double	No	All	V$SESSTAT.VALUE
Disk Sorts (SORTS_DISK)	Number of disk sort operations	--	double	No	All	V$SESSTAT.VALUE
Lock Conversions (LOCK_CONVERSIONS)	Number of enqueues (locks) whose mode was changed (such as from share to lock)	--	double	No	All	V$SESSTAT.VALUE
Lock Releases (LOCK_RELEASES)	Number of times enqueuing (locking) was released. This	--	double	No	All	SUM(V$SESSTAT.VALUE)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Lock Releases (LOCK_RELEASES)	statistical information is the same as the number of lock requests.	--	double	No	All	SUM(V$SESSTAT.VALUE)
Lock Requests (LOCK_REQUESTS)	Number of times enqueuing (locking) was requested	--	double	No	All	V$SESSTAT.VALUE
Lock Timeouts (LOCK_TIMEOUTS)	Number of times enqueuing (locking) request was not permitted within the specified wait time	--	double	No	All	V$SESSTAT.VALUE
Lock Waits (LOCK_WAITS)	Number of times the lock request was placed in wait status. The number of lock requests that were not placed in wait status is obtained by subtracting the number of enqueued waits from the number of lock requests.	--	double	No	All	V$SESSTAT.VALUE
Logical Reads (LOGICAL_READS)	Sum of the number of logical read operations in read consistency mode and the number of requests to the current copy of block	--	double	No	All	db block gets + consistent gets
Memory Sorts (SORTS_MEMORY)	Number of sort operations in memory	--	double	No	All	V$SESSTAT.VALUE
Non-Index Lookups % (NON_INDEX_LOOKUPS)	Percentage ratio of full table scans that do not involve caching	--	double	No	All	(table scans (long tables) / (table scans (short tables) + table scans (long tables))) * 100
PGA Memory (PGA_MEMORY)	Size of PGA memory in use during data collection	--	double	No	All	V$SESSTAT.VALUE
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations on the disk by DBWR	--	double	No	All	V$SESSTAT.VALUE
Physical Reads (PHYSICAL_READS)	Number of times a database block was actually read from disk	--	double	No	All	physical reads - physical reads direct - physical reads direct (lob)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Program (PROGRAM)	Program name of OS	--	string(48)	No	All	V$SESSION.PROGRAM
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDS2)	--	string(4)	No	All	Agent Collector
Recursive Calls (RECURSIVE_CALLS)	Number of user calls processed	--	double	No	All	V$SESSTAT.VALUE
Recursive to User Call % (RECURSIVE_TO_USER_CALL_PERCENTAGE)	Correct values cannot be collected in this field. Percentage indicating overhead.	--	double	No	All	(recursive calls / user calls) * 100
Redo Log Space Requests (REDO_LOG_SPACE_REQUESTS)	Number of times Oracle must wait until disk space is allocated for the REDO log entry, because the active log file is full	--	double	No	All	V$SESSTAT.VALUE
Redo Log Space Wait % (REDO_LOG_SPACE_WAIT_PERCENTAGE)	Percentage of waiting time until disk space is allocated for the REDO log entry	--	double	No	All	(redo log space requests / redo entries) * 100
Row Source % (ROW_SOURCE_PERCENTAGE)	Percentage ratio of all rows obtained by full table scan	--	double	No	All	(table scan rows gotten / (table fetch by rowid + table scan rows gotten)) * 100
SID (SID)	Session ID	--	ulong	No	All	V$SESSION.SID
SQL Net Bytes Rcvd (SQL_NET_BYTES_RECEIVED)	Number of bytes received from clients via SQL*Net	--	double	No	All	V$SESSTAT.VALUE
SQL Net Bytes Sent (SQL_NET_BYTES_SENT)	Number of bytes sent to clients via SQL*Net	--	double	No	All	V$SESSTAT.VALUE
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	Total number of references to the cursor	--	double	No	All	V$SESSTAT.VALUE
Session Cursor Cache Hit % (SESSION_CURSOR_CACHE_HIT_PERCENTAGE)	Number of times an access to the session cursor was executed but the session cursor was not found in the cache	--	double	No	All	(session cursor cache hits / session cursor cache count) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Session Cursor Cache Hits (SESSION_CURSOR_CACHE_HITS)	Recorded count of session cursor cache hits	--	double	No	All	V$SESSTAT.VALUE
Sort Overflow % (SORT_OVERFLOW_PERCENTAGE)	Percentage ratio of sort operations using temporary segments	--	double	No	All	(sorts (disk) / (sorts (memory) + sorts (disk))) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Statement CPU (STATEMENT_CPU)	Total CPU time used by active statements during data collection. In centiseconds (1/100 of a second). To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	double	No	All	V$SESSTAT.VALUE
UGA Memory (UGA_MEMORY)	Number of session memory segments used	--	double	No	All	V$SESSTAT.VALUE
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME
User Calls (USER_CALLS)	Number of user calls processed by active sessions	--	double	No	All	V$SESSTAT.VALUE
User Calls / Parse (USER_CALLS_PER_PARSE)	Percentage indicating how well the application is managing the context area	--	double	No	All	user calls / parse count (total)
User Commits (USER_COMMITS)	Number of transactions	--	double	No	All	V$SESSTAT.VALUE
User Rollback % (USER_ROLLBACK_PERCENTAGE)	Percentage ratio of application transactions that failed (rolled back)	--	double	No	All	(user rollbacks / (user commits + user rollbacks)) * 100
User Rollbacks (USER_ROLLBACKS)	Number of rollbacks	--	double	No	All	V$SESSTAT.VALUE
Waiting Locks (WAITING_LOCKS)	Number of locks owned by another session that this session is waiting for	--	double	No	All	--

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Waiting Locks (WAITING_LOCKS)	Correct values cannot be collected in this field, and 0 is always displayed.	--	double	No	All	--
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations to all physical I/O operations	--	double	No	All	(physical writes / (physical reads + physical writes)) * 100

Session Wait (PD_PDWA)

Function

The Session Wait (PD_PDWA) record stores performance data indicating the status of session waits at a specific point in time. PFM - Agent for Oracle creates one record for each wait in a single session. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	155	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDWA_SID
- PD_PDWA_SEQ_NUM

Lifetime

From the start to the end of a session

Record size

- Fixed part: 678 bytes
- Variable part: 409 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Event (EVENT)	Resource or event the session is waiting for	--	string(64)	No	All	V$SESSION_WAIT.EVENT
P1 (P1)	Additional parameter 1	--	ulong	No	All	V$SESSION_WAIT.P1
P1 Text (P1_TEXT)	Description of additional parameter 1	--	string(64)	No	All	V$SESSION_WAIT.P1TEXT
P2 (P2)	Additional parameter 2	--	ulong	No	All	V$SESSION_WAIT.P2
P2 Text (P2_TEXT)	Description of additional parameter 2	--	string(64)	No	All	V$SESSION_WAIT.P2TEXT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
P3 (P3)	Additional parameter 3	--	ulong	No	All	V$SESSION_WAIT.P3
P3 Text (P3_TEXT)	Description of additional parameter 3	--	string(64)	No	All	V$SESSION_WAIT.P3TEXT
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V$SESSION.PROGRAM where V$SESSION_WAIT.SID = V$SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDWA)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID	--	ulong	No	All	V$SESSION_WAIT.SID
Seq # (SEQ_NUM)	Number identifying the wait (the value of this field is incremented for each wait)	--	double	No	All	V$SESSION_WAIT.SEQ#
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
State (STATE)	Status of the shared server. Valid values are WAITING (waiting for the data collection time), WAITED KNOWN TIME (the value of the WAIT_TIME field is the previous wait time), WAITED SHORT TIME (the last wait is within 100 seconds), and WAITED UNKNOWN TIME (the last wait is unknown).	--	string(19)	No	All	V$SESSION_WAIT.STATE
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$SESSION_WAIT.SID = V$SESSION.SID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Wait Time (WAIT_TIME)	Session wait time. 0 indicates that the session is currently in wait status. Otherwise, the value indicates the session's previous wait time in hundredths of a second. To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	ulong	No	All	V$SESSION_WAIT.WAIT_TIME
Wait Time String (WAIT_TIME_STRING)	Session wait time. 0 indicates that the session is currently in wait status. Otherwise, the value indicates the session's previous wait time in hundredths of a second. To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	string(20)	No	All	V$SESSION_WAIT.WAIT_TIME

SGA Components (PD_PDSG)

Function

The SGA Components (PD_PDSG) record stores performance data indicating the status (at a specific point in time) of the system global area (SGA). PFM - Agent for Oracle creates one record for each system global area (SGA). This is a multi-instance record.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, this record collects different performance data depending on whether PDBs are being monitored or the root container (CDB$ROOT) is being monitored.

When monitoring PDBs

- Performance data for SGA allocated to the PDBs that are being monitored
- Performance data for common SGA of database instances

When monitoring the root container (CDB$ROOT)

- Use administrative accounts (sys or system)
  - Performance data for SGA allocated to all PDBs
  - Performance data for SGA allocated to the root container (CDB$ROOT)
  - Performance data for common SGA of database instances
- Use common user
  - Performance data for SGA allocated to the root container (CDB$ROOT)
  - Performance data for common SGA of database instances

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	115	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDSG_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 66 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Bytes (BYTES)	Size of allocated memory (in bytes)	--	double	No	All	• V $SGASTAT.BYTES • V $SGA_DYNAMIC_COMPONENTS.CURRENT_SIZE
Component Name (NAME)	Name of the SGA component	--	string(26)	No	All	• V $SGASTAT.NAME • V $SGA_DYNAMIC_COMPONENTS.COMPONENT
Container ID (CONTAINER_ID)	The ID of the container to which SGA component belongs. This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.	--	word	No	Oracle 12c R2 or later	V $SGASTAT.CON_ID
PDB Bytes (PDB_BYTES)	Retrieves total size of memory for each Container ID field. The value can be either of the following: • If the Container ID is not 0, total size of memory of all SGA components that are allocated to PDBs or the root container (CDB $ROOT) • If the Container ID is 0, total size of memory of common SGA components of database instances This field is valid if the monitoring target is Oracle Database 12c R2 or later in a CDB configuration. Otherwise, the value is 0.	--	double	No	Oracle 12c R2 or later	V$SGASTAT.BYTES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Pool (POOL)	Pools in which there is component name memory. There are four pools: • shared pool • large pool • java pool • streams pool This field is blank when there is no applicable pool.	--	string(12)	No	10-00 or later	• V$SGASTAT.POOL
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSG)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Total Bytes (TOTAL_BYTES)	Total size (in bytes) of memory allocated to all SGA components If the monitoring target is Oracle 12c R2 or later in a CDB configuration, total size of memory allocated to the entire CDB.	--	double	No	All	V$SGASTAT.BYTES

Shared Cursor Cache (PD_PDC)

Function

The Shared Cursor Cache (PD_PDC) record stores performance data indicating the status of the shared cursor cache at a specific point in time. PFM - Agent for Oracle creates one record for each shared cursor cache. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	10	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	Yes	N

ODBC key fields

PD_PDC_ADDRHASH

Lifetime

From the loading to the unloading from the shared SQL area

Record size

- Fixed part: 678 bytes
- Variable part: 1,311 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Action (ACTION)	Name of the action that was executing when the first SQL statement was analyzed	--	string(32)	No	All	V$SQLAREA.ACTION
Addrhash (ADDRHASH)	Value identifying the SQL statement being executed	--	string(38)	No	All	V$SQLAREA.ADDRESS V$SQLAREA.HASH_VALUE
Buffer Gets (BUFFER_GETS)	Total buffer acquisitions over all child cursors	--	double	No	All	V$SQLAREA.BUFFER_GETS
Command Type (COMMAND_TYPE)	Command type	--	string(32)	No	All	V$SQLAREA.COMMAND_TYPE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Disk Reads (DISK_READS)	Number of disk blocks read by this cursor and by all cursors executed by this cursor	--	double	No	All	V$SQLAREA.DISK_READS
Executions (EXECUTIONS)	Number of times SQL statement was executed	--	double	No	All	V$SQLAREA.EXECUTIONS
First Load Time (FIRST_LOAD_TIME)	First time the cursor was loaded into system global area (SGA)	--	string(19)	No	All	V$SQLAREA.FIRST_LOAD_TIME
Invalidations (INVALIDATIONS)	Invalidations (INVALIDATIONS) Number of times the cursor's context was invalid for any of the following reasons: • Table referenced by cursor was deleted • Validity was checked • Index was specified	--	double	No	All	V$SQLAREA.INVALIDATIONS
Kept Versions (KEPT_VERSIONS)	Number of cursors of the same SQL statement, package, procedure, function, and trigger anonymous PL/SQL block in the shared pool	--	ulong	No	All	V$SQLAREA.KEPT_VERSIONS
Loaded Versions (LOADED_VERSIONS)	Number of cursors that have been entirely loaded because no outdated information exists	--	ulong	No	All	V$SQLAREA.LOADED_VERSIONS
Loads (LOADS)	Number of times a cursor was loaded because the cursor body was not used while the text of the SQL statement was in cache or because the cursor was no longer valid	--	double	No	All	V$SQLAREA.LOADS
Module (MODULE)	Module name when the first SQL statement was analyzed	--	string(64)	No	All	V$SQLAREA.MODULE
Open Versions (OPEN_VERSIONS)	Number of open cursors that belong to the user	--	ulong	No	All	V$SQLAREA.OPEN_VERSIONS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Parse Calls (PARSE_CALLS)	Number of times a user issued an analysis call to a cursor	--	double	No	All	V$SQLAREA.PARSE_CALLS
Parsing Schema ID (PARSING_SCHEMA_ID)	Schema ID used to analyze SQL statements	--	long	No	All	V$SQLAREA.PARSING_SCHEMA_ID
Parsing User ID (PARSING_USER_ID)	User ID that analyzed the SQL statements	--	long	No	All	V$SQLAREA.PARSING_USER_ID
Persistent Mem (PERSISTENT_MEM)	Amount of memory (in bytes) per user while the cursor was valid	--	double	No	All	V$SQLAREA.PERSISTENT_MEM
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDC)	--	string(4)	No	All	Agent Collector
Rows Processed (ROWS_PROCESSED)	Number of rows returned by processing or SQL statements	--	double	No	All	V$SQLAREA.ROWS_PROCESSED
Runtime Mem (RUNTIME_MEM)	Amount of memory (in bytes) required per user only during the execution	--	double	No	All	V$SQLAREA.RUNTIME_MEM
SQL Text (SQL_TEXT)	SQL text or PL/SQL statement that requires the cursor	--	string(1000)	No	All	V$SQLAREA.SQL_TEXT
Sharable Mem (SHARABLE_MEM)	Amount of memory that can be shared by users	--	double	No	All	V$SQLAREA.SHARABLE_MEM
Sorts (SORTS)	Number of sort operations executed by SQL statements	--	double	No	All	V$SQLAREA.SORTS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Users Executing (USERS_EXECUTING)	Number of users currently executing the cursor	--	long	No	All	V$SQLAREA.USERS_EXECUTING
Users Opening SQL (USERS_OPENING_SQL)	Number of users currently using the open cursor to analyze statements	--	long	No	All	V$SQLAREA.USERS_OPENING
Version Count (VERSION_COUNT)	Number of cursors (Multiple users might execute the same SQL	--	ulong	No	All	V$SQLAREA.VERSION_COUNT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Version Count (VERSION_COUNT)	statement on different versions of a single table.)	--	ulong	No	All	V $SQLAREA.VERSION_COUNT

Shared Server (PD_PDSH)

Function

The Shared Server (PD_PDSH) record stores performance data indicating the status of the shared servers at a specific point in time. PFM - Agent for Oracle creates one record for each shared server in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDSH_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 108 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Breaks (BREAKS)	Number of breaks (pauses)	--	double	No	All	V$SHARED_SERVER.BREAKS
Busy (BUSY)	Total busy time in hundredths of a second	--	double	No	All	V$SHARED_SERVER.BUSY
Busy % (PERCENT_BUSY)	Percentage of time the shared server was busy	--	double	No	All	(V$SHARED_SERVER.BUSY / (V$SHARED_SERVER.BUSY + V$SHARED_SERVER.IDLE)) * 100
Bytes (BYTES)	Total size of messages in bytes	--	double	No	All	V$SHARED_SERVER.BYTES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Circuit (CIRCUIT)	Address of the circuit that is currently providing the service	--	string(16)	No	All	V$SHARED_SERVER.CIRCUIT
Idle (IDLE)	Total idle time in hundredths of a second	--	double	No	All	V$SHARED_SERVER.IDLE
Idle % (PERCENT_IDLE)	Percentage of time the shared server was idle	--	double	No	All	(V$SHARED_SERVER.IDLE / (V$SHARED_SERVER.BUSY + V$SHARED_SERVER.IDLE)) * 100
Messages (MESSAGES)	Number of messages processed	--	double	No	All	V$SHARED_SERVER.MESSAGES
Oracle PID (PID)	process ID of the shared server process	--	ulong	No	All	V$PROCESS.PID where V$SHARED_SERVER.PADDR = V$PROCESS.ADDR
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSH)	--	string(4)	No	All	Agent Collector
Requests (REQUESTS)	Number of requests retrieved from the common queue while the server was running	--	double	No	All	V$SHARED_SERVER.REQUESTS
Shared Server Name (NAME)	Name of the shared server	--	string(5)	No	All	V$SHARED_SERVER.NAME
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Status (STATUS) Status; one of the following: EXEC: Under SQL execution WAIT (ENQ): Waiting for lock WAIT (SEND): Waiting to transfer data to user WAIT (COMMON): Idle, waiting for a user request	--	string(16)	No	All	V$SHARED_SERVER.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	WAIT (RESET): Waiting for a circuit to reset after break QUIT: Terminating	--	string(16)	No	All	V$SHARED_SERVER.STATUS

Shared Server Interval (PI_PISH)

Function

The Shared Server Interval (PI_PISH) record stores performance data, taken at specific intervals, about shared servers. PFM - Agent for Oracle creates one record for each shared server in an instance. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PISH_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 204 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Breaks (BREAKS)	Number of breaks (pauses)[#2]	AVG	double	Yes	All	V $SHARED_SERVER.BREAKS
Busy (BUSY)	Total busy time in hundredths of a second[#2]	AVG	double	Yes	All	V $SHARED_SERVER.BUSY
Busy % (PERCENT_BUSY)	Percentage of time the shared server was busy[#2]	AVG	double	No	All	(V $SHARED_SERVER.BUSY / (V $SHARED_SERVER.BUSY + V $SHARED_SERVER.IDLE)) * 100
Bytes (BYTES)	Total size of messages in bytes[#2]	AVG	double	Yes	All	V $SHARED_SERVER.BYTES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Circuit (CIRCUIT)	Address of the circuit that is currently providing the service[#1]	COPY	string(16)	No	All	V$SHARED_SERVER.CIRCUIT
Idle (IDLE)	Total idle time in hundredths of a second[#2]	AVG	double	Yes	All	V$SHARED_SERVER.IDLE
Idle % (PERCENT_IDLE)	Percentage of time the shared server was idle[#2]	AVG	double	No	All	(V$SHARED_SERVER.IDLE / (V$SHARED_SERVER.BUSY + V$SHARED_SERVER.IDLE)) * 100
Messages (MESSAGES)	Number of messages processed[#2]	AVG	double	Yes	All	V$SHARED_SERVER.MESSAGES
Oracle PID (PID)	Oracle process ID of the shared server process[#1]	COPY	ulong	No	All	V$PROCESS.PID where V$SHARED_SERVER.PADDR = V$PROCESS.ADDR
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PISH)[#1]	COPY	string(4)	No	All	Agent Collector
Requests (REQUESTS)	Number of requests retrieved from the common queue while the server was running[#2]	AVG	double	Yes	All	V$SHARED_SERVER.REQUESTS
Shared Server Name (NAME)	Name of the shared server[#1]	COPY	string(5)	No	All	V$SHARED_SERVER.NAME
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	Status; one of the following:[#1] EXEC: Under SQL execution WAIT (ENQ): Waiting for a lock to be released WAIT (SEND): Waiting to transfer data to user WAIT (COMMON): Idle, waiting for a user request WAIT (RESET): Waiting for a circuit to reset after break	COPY	string(16)	No	All	V$SHARED_SERVER.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Status (STATUS)	QUIT: Terminating	COPY	string(16)	No	All	V$SHARED_SERVER.STATUS

Sort Segment (PD_PDSR)

Function

The Sort Segment (PD_PDSR) record stores performance data indicating the status of sort segments in a database at a specific point in time. PFM - Agent for Oracle creates one record for each sort segment. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	25	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDSR_TABLESPACE_NAME
- PD_PDSR_SEGMENT_FILE
- PD_PDSR_SEGMENT_BLOCK

Lifetime

From the start to the stop of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 188 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Added Extents (ADDED_EXTENTS)	Number of extents allocated	--	double	No	All	V$SORT_SEGMENT.ADDED_EXTENTS
Current Users (CURRENT_USERS)	Number of active users for the segment	--	double	No	All	V$SORT_SEGMENT.CURRENT_USERS
Extent Hits (EXTENT_HITS)	Number of times an unused extent was found in the pool	--	double	No	All	V$SORT_SEGMENT.EXTENT_HITS
Extent Size (EXTENT_SIZE)	Size of extent	--	double	No	All	V$SORT_SEGMENT.EXTENT_SIZE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Blocks (FREE_BLOCKS)	Blocks that are not allocated to any sort operation	--	double	No	All	V$SORT_SEGMENT.FREE_BLOCKS
Free Extents (FREE_EXTENTS)	Extents that are not allocated to any sort operation	--	double	No	All	V$SORT_SEGMENT.FREE_EXTENTS
Free Requests (FREE_REQUESTS)	Number of allocation release requests	--	double	No	All	V$SORT_SEGMENT.FREE_REQUESTS
Freed Extents (FREED_EXTENTS)	Number of extents whose allocation was released	--	double	No	All	V$SORT_SEGMENT.FREED_EXTENTS
Max Blocks (MAX_BLOCKS)	Maximum number of blocks used	--	double	No	All	V$SORT_SEGMENT.MAX_BLOCKS
Max Size (MAX_SIZE)	Maximum number of extents used	--	double	No	All	V$SORT_SEGMENT.MAX_SIZE
Max Sort Blocks (MAX_SORT_BLOCKS)	Maximum number of blocks used in individual sort operations	--	double	No	All	V$SORT_SEGMENT.MAX_SORT_BLOCKS
Max Sort Size (MAX_SORT_SIZE)	Maximum number of extents used in individual sort operations	--	double	No	All	V$SORT_SEGMENT.MAX_SORT_SIZE
Max Used Blocks (MAX_USED_BLOCKS)	Maximum number of blocks used in all sort operations	--	double	No	All	V$SORT_SEGMENT.MAX_USED_BLOCKS
Max Used Size (MAX_USED_SIZE)	Maximum number of extents used in all sort operations	--	double	No	All	V$SORT_SEGMENT.MAX_USED_SIZE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSR)	--	string(4)	No	All	Agent Collector
Segment Block (SEGMENT_BLOCK)	Block number of first extent	--	double	No	All	V$SORT_SEGMENT.SEGMENT_BLOCK
Segment File (SEGMENT_FILE)	File number of first extent	--	ulong	No	All	V$SORT_SEGMENT.SEGMENT_FILE
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Tablespace Name	Name of tablespace	--	string(31)	No	All	V$SORT_SEGMENT.TABLESPACE_NAME

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(TABLESPACE_NAME)	Name of tablespace	--	string(31)	No	All	V$SORT_SEGMENT.TABLESPACE_NAME
Total Blocks (TOTAL_BLOCKS)	Number of blocks in the segment	--	double	No	All	V$SORT_SEGMENT.TOTAL_BLOCKS
Total Extents (TOTAL_EXTENTS)	Number of extents for the segment	--	double	No	All	V$SORT_SEGMENT.TOTAL_EXTENTS
Used Blocks (USED_BLOCKS)	Number of blocks allocated for active sort operation	--	double	No	All	V$SORT_SEGMENT.USED_BLOCKS
Used Extents (USED_EXTENTS)	Number of extents allocated for active sort operation	--	double	No	All	V$SORT_SEGMENT.USED_EXTENTS

Sort Segment Interval (PI_PISR)

Function

The Sort Segment Interval (PI_PISR) record stores performance data, taken at specific intervals, about sort segments in a database. PFM - Agent for Oracle creates one record for each sort segment. This is a multi-instance record.

Notes

When multiple instances are collected for the Sort Segment Interval (PI_PISR) record, the collected values are incorrect.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	15	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PISR_SEGMENT_FILE
- PI_PISR_SEGMENT_BLOCK

Lifetime

From the start to the stop of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 392 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Added Extents (ADDED_EXTENTS)	Number of extents allocated#2	AVG	double	Yes	All	V$SORT_SEGMENT.ADDED_EXTENTS
Current Users (CURRENT_USERS)	Number of active users for the segment#2	AVG	double	No	All	V$SORT_SEGMENT.CURRENT_USERS
Extent Hits (EXTENT_HITS)	Number of times an unused extent was found in the pool#2	AVG	double	Yes	All	V$SORT_SEGMENT.EXTENT_HITS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Extent Size (EXTENT_SIZE)	Size of extent[#1]	COPY	double	No	All	V$SORT_SEGMENT.EXTENT_SIZE
Free Blocks (FREE_BLOCKS)	Blocks that are not allocated to any sort operation[#2]	AVG	double	No	All	V$SORT_SEGMENT.FREE_BLOCKS
Free Extents (FREE_EXTENTS)	Extents that are not allocated to any sort operation[#2]	AVG	double	No	All	V$SORT_SEGMENT.FREE_EXTENTS
Free Requests (FREE_REQUESTS)	Number of allocation release requests[#2]	AVG	double	Yes	All	V$SORT_SEGMENT.FREE_REQUESTS
Freed Extents (FREED_EXTENTS)	Number of extents whose allocation was released[#2]	AVG	double	Yes	All	V$SORT_SEGMENT.FREED_EXTENTS
Max Blocks (MAX_BLOCKS)	Maximum number of blocks used[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_BLOCKS
Max Size (MAX_SIZE)	Maximum number of extents used[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_SIZE
Max Sort Blocks (MAX_SORT_BLOCKS)	Maximum number of blocks used in individual sort operations[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_SORT_BLOCKS
Max Sort Size (MAX_SORT_SIZE)	Maximum number of extents used in individual sort operations[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_SORT_SIZE
Max Used Blocks (MAX_USED_BLOCKS)	Maximum number of blocks used in all sort operations[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_USED_BLOCKS
Max Used Size (MAX_USED_SIZE)	Maximum number of extents used in all sort operations[#2]	AVG	double	No	All	V$SORT_SEGMENT.MAX_USED_SIZE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PISR)[#1]	COPY	string(4)	No	All	Agent Collector
Segment Block (SEGMENT_BLOCK)	Block number of first extent[#1]	COPY	double	No	All	V$SORT_SEGMENT.SEGMENT_BLOCK
Segment File (SEGMENT_FILE)	File number of first extent[#1]	COPY	ulong	No	All	V$SORT_SEGMENT.SEGMENT_FILE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Name of tablespace[#1]	COPY	string(31)	No	All	V$SORT_SEGMENT.TABLESPACE_NAME
Total Blocks (TOTAL_BLOCKS)	Number of blocks in the segment[#2]	AVG	double	No	All	V$SORT_SEGMENT.TOTAL_BLOCKS
Total Extents (TOTAL_EXTENTS)	Number of extents for the segment[#2]	AVG	double	No	All	V$SORT_SEGMENT.TOTAL_EXTENTS
Used Blocks (USED_BLOCKS)	Number of blocks allocated for active sort operation[#2]	AVG	double	No	All	V$SORT_SEGMENT.USED_BLOCKS
Used Extents (USED_EXTENTS)	Number of extents allocated for active sort operation[#2]	AVG	double	No	All	V$SORT_SEGMENT.USED_EXTENTS

SQL Text (PD_PDSQ)

Function

The SQL Text (PD_PDSQ) record stores performance data indicating the status (at a specific point in time) of the SQL text for a cursor in the shared cursor cache. PFM - Agent for Oracle creates one record for each SQL text for a cursor in the shared cursor cache. This is a multi-instance record.

Since this record in available only in real-time, it is not displayed in the Properties window that is displayed by clicking the agent icon on the **Agents** page of PFM - Web Console, and then clicking the **Properties** method.

You use this record only when calling the SQL Text report provided by the monitoring template as a drilldown report. You cannot display a report using this record individually.

If you are displaying a report using this record with SQL and the user executing the SQL statement is not the user who created LSC_13_PLAN_TABLE in the sp_inst.sql script, the system issues a FAILED message to the Explain Plan (EXPLAIN_PLAN) field. To display the correct report, the user specified in the oracle_user property must execute the sp_inst.sql script.

This record displays a drilldown report of the SELECT, INSERT, UPDATE, and DELETE statements. It does not display a drilldown report of any other SQL statement or any PL/SQL package.

Notes

- Before creating an account, check whether you want to acquire the value of the Explain Plan (EXPLAIN_PLAN) field in the SQL Text (PD_PDSQ) record for operations on the objects that belong to the SYS schema. If you want to do so, use sys as the account to be used by PFM - Agent for Oracle. If you use an account other than sys, you will no longer be able to acquire the value of that field. If the value of the EXPLAIN_PLAN field cannot be acquired, message Explain Plan Failed is stored in the field.
- If the account used by PFM - Agent for Oracle has no privileges to access, or fails to reference, an object that belongs to a schema of the user who executed SQL, the following value cannot be acquired:

  The value of the Explain Plan (EXPLAIN_PLAN) field in the SQL Text (PD_PDSQ) record

  If the value of the EXPLAIN_PLAN field cannot be acquired, message Explain Plan Failed is stored in the field. If you want to acquire the value of the Explain Plan (EXPLAIN_PLAN) field, execute the SQL for manipulating the field in the *owner.table-name* format.

Default and changeable values

None

ODBC key fields

PD_PDSQ_ADDRHASH

Lifetime

None

Record size

- Fixed part: 678 bytes
- Variable part: 30,051 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addrhash (ADDRHASH)	Address and hash value used to identify the cached cursor	--	string(38)	No	All	V$SQLTEXT.ADDRESS V$SQLTEXT.HASH_VALUE
Command Type (COMMAND_TYPE)	Type of the SQL statement	--	string(10)	No	All	V$SQLTEXT.COMMAND_TYPE
Explain Plan (EXPLAIN_PLAN)	Execution plan for SELECT, UPDATE, INSERT, and DELETE statements chosen by the Oracle optimizer	--	string(30000)	No	All	Agent Collector
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDSQ)	--	string(4)	No	All	Agent Collector
SQL Text (SQL_TEXT)	Portion of the SQL text	--	string(30000)	No	All	V$SQLTEXT.SQL_TEXT
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

SQL Text - Performance Based (PD_PDES)

Function

The SQL Text - Performance Based (PD_PDES) record stores performance data indicating the status of SQL text that satisfies the following conditions at a specific point in time (see Note). PFM - Agent for Oracle creates one record for each SQL statement that satisfies the conditions. This is a multi-instance record. A maximum of five instances can be created for this record.

Note:

The following describes the conditions:

1. Data is sorted from the object with the most read operations per execution to the object with the least read operations per execution.
2. The result of step 1 indicates that the number of disk read operations per execution of an object exceeds 1,000. Alternatively, the number of I/O operations per execution of an object exceeds 10,000.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	40	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDES_ADDRHASH

Lifetime

From the loading to the unloading from the shared SQL area

Record size

- Fixed part: 678 bytes
- Variable part: 10,131 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Addrhash (ADDRHASH)	Address and hash value used to identify the cached cursor	--	string(38)	No	All	V$SQLAREA.ADDRESS + V$SQLAREA.HASH_VALUE
Buffer Gets (BUFFER_GETS)	Total buffer acquisitions over all	--	double	No	All	V$SQLAREA.BUFFER_GETS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Buffer Gets (BUFFER_GETS)	child cursors Disk Reads	--	double	No	All	V $SQLAREA.BUFFER_GETS
Disk Reads (DISK_READS)	Total disk read operations over all child cursors	--	double	No	All	V $SQLAREA.DISK_READS
Disk Reads/Exec (DISK_READS_PER_EXECUTION)	Number of physical read operations required per execution	--	double	No	All	V $SQLAREA.DISK_READS / V $SQLAREA.EXECUTIONS
Executions (EXECUTIONS)	Number of times the object was executed after this object was placed in the library cache	--	double	No	All	V $SQLAREA.EXECUTIONS
Hit % (HIT_PERCENTAGE)	Ratio (as a percent) of buffer read operations to all read operations	--	double	No	All	100 * (V $SQLAREA.BUFFER_GETS - V $SQLAREA.DISK_READS) / V $SQLAREA.BUFFER_GETS
Logical Reads/Exec (LOGICAL_IO_PER_EXECUTION)	Number of logical read operations required per execution	--	double	No	All	V $SQLAREA.BUFFER_GETS / V $SQLAREA.EXECUTIONS
Parsing User (PARSING_USER)	User who analyzed the SQL statement (if connecting to Oracle 12c R1 or later, the user ID)	--	string(30)	No	All	USER$.NAME where USER$.USER# = V $SQLAREA.PARSING_USER_ID • In Oracle 12c R1 and later: V $SQLAREA.PARSING_USER_ID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDES)	--	string(4)	No	All	Agent Collector
SEQNO (SEQUENCE)	Number of times the cursor was found during data collection	--	ulong	No	All	Agent Collector
SQL Text (SQL_TEXT)	SQL text for the cursor	--	string(10000)	No	All	V $SQLTEXT.SQL_TEXT

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Sorts (SORTS)	Total number of sort operations executed on all child cursors	--	double	No	All	V$SQLAREA.SORTS
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector

SQL*Net Listener (PD_PDNL)

Function

The SQL*Net Listener (PD_PDNL) record stores performance data indicating the status and overview of an SQL*Net Listener at a specific point in time. This status information includes the status of, and overview information about, the default SQL*Net Listener. If the listener is not active, the Alias field is blank.

If SQL*Net has not been installed, an attempt to display a report that uses this record will display the No records returned message.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of a listener instance

Record size

- Fixed part: 1,366 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Alias (ALIAS)	Name of the listener	--	string(20)	No	All	lsnrctl stat
Availability (AVAILABILITY)	Listener task's availability status; this value can be either 0 (inactive) or 1 (active)	--	short	No	All	Agent Collector
Change Time (CHANGE_TIME)	Time the availability task was changed	--	time_t	No	All	Agent Collector
Handlers (HANDLERS)	Number of services processed by the listener task	--	ushort	No	All	lsnrctl stat

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Log File (LOG_FILE)	Location of log file	--	string(256)	No	All	lsnrctl stat
Parameter File (PARAMETER_FILE)	Location of parameter file. This is Blank when the listener.or a file does not exist in the Oracle environment. For details about the listener.ora file, see the Oracle documentation.	--	string(256)	No	All	lsnrctl stat
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDNL)	--	string(4)	No	All	Agent Collector
Security (SECURITY)	Security status	--	string(5)	No	All	lsnrctl stat
Start Date (START_DATE)	Listener start date and time	--	string(20)	No	All	lsnrctl stat
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Trace Level (TRACE_LEVEL)	Listener's trace level	--	string(5)	No	All	lsnrctl stat
Up Time (UP_TIME)	Total run time	--	string(30)	No	All	lsnrctl stat
Version (VERSION)	Version Listener's version	--	string(80)	No	All	lsnrctl stat

SQL*Net Listeners (PD_PDLS)

Function

The SQL*Net Listeners (PD_PDLS) record stores performance data indicating the status and overview of each defined listener at a specific point in time. PFM - Agent for Oracle creates one record for each name specified in listener_name. Only one listener can be monitored.

If you omit the listener name when setting an instance, PFM - Agent for Oracle monitors the default listener. If the listener is not active, the Alias field is blank.

If SQL*Net has not been installed, an attempt to display a report that uses this record will display the No records returned message.

Notes

When setting an alarm monitoring SQL * Net Listeners (PD_PDLS) records, if you set the name of a listener that does not exist in the listener_name of the instance information, the Listener Status alarm status will always be normal. Upon the occurrence of this phenomenon, the following message is output to the common message log:

KAVF12412-W The listener does not exist. (ListenerName=*listener-name*)

If this message is output, correct the LISTENER_NAME property value.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDLS_ALIAS

Lifetime

From the creation to the deletion of a listener instance

Record size

- Fixed part: 678 bytes
- Variable part: 688 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Alias (ALIAS)	Name of the listener	--	string(20)	No	All	lsnrctl stat

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Availability (AVAILABILITY)	Listener task's availability status; this value can be either 0 (inactive) or 1 (active)	--	short	No	All	Agent Collector
Change Time (CHANGE_TIME)	Time the availability task was changed	--	time_t	No	All	Agent Collector
Handlers (HANDLERS)	Number of services processed by the listener task	--	short	No	All	lsnrctl stat
Log File (LOG_FILE)	Location of log file	--	string(256)	No	All	lsnrctl stat
Parameter File (PARAMETER_FILE)	Location of parameter file. This is Blank when the listener.or a file does not exist in the Oracle environment. For details about the listener.ora file, see the Oracle documentation.	--	string(256)	No	All	lsnrctl stat
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDLS)	--	string(4)	No	All	Agent Collector
Security (SECURITY)	Security status	--	string(5)	No	All	lsnrctl stat
Start Date (START_DATE)	Listener start date and time	--	string(20)	No	All	lsnrctl stat
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Trace Level (TRACE_LEVEL)	Listener's trace level	--	string(5)	No	All	lsnrctl stat
Up Time (UP_TIME)	Total run time	--	string(30)	No	All	lsnrctl stat
Version (VERSION)	Listener's version	--	string(80)	No	All	lsnrctl stat

System Event (PD_PDSE)

Function

The System Event (PD_PDSE) record stores the performance data indicating the status of each queued event in an instance at the system level at a specific point in time. PFM - Agent for Oracle creates one record for each queued event. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	15	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDSE_EVENT

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 97 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait (AVERAGE_WAIT)	Average wait time for the event in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	double	No	All	V$SYSTEM_EVENT.AVERAGE_WAIT
Event (EVENT)	Name of the event the system is waiting for	--	string(64)	No	All	V$SYSTEM_EVENT.EVENT
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type	Record name (always PDSE)	--	string(4)	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
(INPUT_RECORD_TYPE)	Record name (always PDSE)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Time Waited (TIME_WAITED)	Total wait time for the event in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)	--	double	No	All	V$SYSTEM_EVENT.TIME_WAITED
Total Timeouts (TOTAL_TIMEOUTS)	Number of timeouts for the event	--	double	No	All	V$SYSTEM_EVENT.TOTAL_TIMEOUTS
Total Waits (TOTAL_WAITS)	Number of waits for the event	--	double	No	All	V$SYSTEM_EVENT.TOTAL_WAITS

System Event Interval (PI_PISE)

Function

The System Event Interval (PI_PISE) record stores performance data, taken at specific intervals, about each queued event in an instance at the system level. PFM - Agent for Oracle creates one record for each queued event. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	10	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PISE_EVENT

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 145 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Wait (AVERAGE_WAIT)	Average wait time for the event in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)[#2]	AVG	double	No	All	V$SYSTEM_EVENT.AVERAGE_WAIT
Event (EVENT)	Name of the event the system is waiting for[#1]	COPY	string(64)	No	All	V$SYSTEM_EVENT.EVENT
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Type (INPUT_RECORD_TYPE)	Record name (always PISE)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Time Waited (TIME_WAITED)	Total wait time for the event in hundredths of a second (to collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file)[#2]	AVG	double	Yes	All	V$SYSTEM_EVENT.TIME_WAITED
Total Timeouts (TOTAL_TIMEOUTS)	Number of timeouts for the event[#2]	AVG	double	Yes	All	V$SYSTEM_EVENT.TOTAL_TIMEOUTS
Total Waits (TOTAL_WAITS)	Number of waits for the event[#2]	AVG	double	Yes	All	V$SYSTEM_EVENT.TOTAL_WAITS

System Stat Interval (PI_PIST)

Function

The System Stat Interval (PI_PIST) record stores metric information, taken at specific intervals, about sessions. PFM - Agent for Oracle creates one record for each metric information item. It provides this metric information about all sessions as information about the entire system. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PIST_STATISTIC_NUM

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 114 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Class (CLASS)	Statistic class[#1]	COPY	string(20)	No	All	V$SYSSTAT.CLASS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PIST)[#1]	COPY	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Statistic # (STATISTIC_NUM)	Statistic number[#1]	COPY	double	No	All	V$SYSSTAT.STATISTIC#

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Statistic Name (NAME)	Statistic name#1	COPY	string(64)	No	All	V$SYSSTAT.NAME
Value (VALUE)	Statistical value#2	AVG	double	Yes	All	V$SYSSTAT.VALUE

System Stat Summary (PD)

Function

The System Stat Summary (PD) record stores performance data, taken at a specific point in time, indicating cumulative values from the start of the instance.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. The following table shows the relationship between the fields and performance data.

Fields	Performance data	
	When monitoring PDBs	When monitoring the root container (CDB$ROOT)
Dict Cache Get Misses % Lib Cache Miss % Redo Alloc Immediate % Redo Alloc Willing to Wait % Redo Copy Immediate % Redo Copy Willing to Wait %	Displays common performance data for database instances. Note The fields collect the same value when monitoring either the PDB or the root container (CDB$ROOT). To prevent alarm notification from multiple instances of PFM - Agent for Oracle, select the instance to be monitored.	
Free List Wait Events Sys Undo Blk Wait Events Sys Undo Hdr Wait Events Undo Blk Wait Events Undo Hdr Wait Events	Displays performance data for the PDBs that are being monitored.	• Use administrative accounts (sys or system) Performance data for all PDBs and the root container (CDB$ROOT) are collected. • Use common user Performance data for the root container (CDB$ROOT) is collected.
Recursive To User Call % SQL Executing	As with non CDB, unable to collect valid performance data.	
Fields except for above	Displays performance data for the PDBs that are being monitored.	Displays common performance data for database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	0	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 1,130 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes/Tran (BLOCK_CHANGES_PER_TRANSACTION)	Rate at which each transaction executed DML operation	--	double	No	All	db block changes / user commits
Block Visits/Tran (BLOCK_VISITS_PER_TRANSACTION)	Number of times a work database was loaded per transaction	--	double	No	All	(db block gets + consistent gets) / user commits
Buffer Busy Wait % (BUFFER_BUSY_WAIT_PERCENTAGE)	Percentage ratio of buffer busy waits	--	double	No	All	(V$SYSTEM_EVENT.TOTAL_WAITS where EVENT = 'buffer busy waits' / (consistent gets + db block gets)) * 100
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage	--	double	No	All	(1 - (physical reads cache / (consistent gets from cache + db block gets from cache))) * 100
Calls/Tran (CALLS_PER_TRANSACTION)	Rate at which client requests were executed per transaction	--	double	No	All	user calls / user commits
Changed Block % (CHANGED_BLOCK_PERCENTAGE)	Percentage ratio of difference between query and database manipulation language (DML) in the database application. This value changes according to indexes and application utilization status.	--	double	No	All	(db block changes / (block gets + consistent gets)) * 100
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Percentage ratio of extents used for consistency of application read operations	--	double	No	All	(consistent changes / consistent gets) * 100
Continued Row % (CONTINUED_ROW_PERCENTAGE)	Percentage ratio of continued rows. The value is close to 0 unless the application handles LONG columns.	--	double	No	All	(table fetch continued row / (table fetch by rowid + table scan rows gotten)) * 100
Current Logons (CURRENT_LOGONS)	Number of current login to Oracle Database	--	long	No	All	V$SYSSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Deadlocks (LOCK_DEADLOCKS)	Number of deadlocks caused by locked DML processing	--	double	No	All	V$SYSSTAT.VALUE
Dict Cache Get Misses % (DICTIONARY_CACHE_GET_MISSES_PERCENTAGE)	Percentage ratio of data requests issued due to cache miss	--	double	No	All	(SUM(V $ROWCACHE.GETMISSES) / SUM(V $ROWCACHE.GETS)) * 100
Disk Sorts (SORTS_DISK)	Number of disk sort operations	--	double	No	All	V$SYSSTAT.VALUE
Free List Wait Events (FREE_LIST_WAIT_EVENTS)	Number of wait events in the free list	--	double	No	All	V$WAITSTAT.COUNT where class = 'free list'
Lib Cache Miss % (LIBRARY_CACHE_MISS_PERCENTAGE)	Library cache miss rate. This field means the ratio of times the allocated objects in library cache are reloaded. As the value of this field increases, the amount of resources in use also increases.	--	double	No	All	(SUM(V $LIBRARYCACHE.RELOADS) /SUM(V $LIBRARYCACHE.PINS)) * 100
Lock Conversions (LOCK_CONVERSIONS)	Number of enqueues (locks) whose mode was changed (such as from share to lock)	--	double	No	All	V$SYSSTAT.VALUE
Lock Releases (LOCK_RELEASES)	Number of times enqueuing (locking) was released	--	double	No	All	V$SYSSTAT.VALUE
Lock Requests (LOCK_REQUESTS)	Number of times enqueuing (locking) was requested	--	double	No	All	V$SYSSTAT.VALUE
Lock Timeouts (LOCK_TIMEOUTS)	Number of times an enqueuing (locking) request was not permitted within the allocated time	--	double	No	All	V$SYSSTAT.VALUE
Lock Waits (LOCK_WAITS)	Number of times an enqueuing (locking) request was placed in wait status. The difference between the enqueuing requests count and the enqueuing waits count is the number of times the request was not treated as an enqueuing request.	--	double	No	All	V$SYSSTAT.VALUE
Logical Reads (LOGICAL_READS)	Number of logical read operations in read consistency mode and number of requests to the current copy of blocks	--	double	No	All	db block gets + consistent gets
Memory Sorts (SORTS_MEMORY)	Number of sort operations in memory	--	double	No	All	V$SYSSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Non-Index Lookups % (NON_INDEX_LOOKUPS)	Percentage ratio of full table scans that do not involve caching	--	double	No	All	(table scans (long tables) / (table scans (short tables) + table scans (long tables))) * 100
Physical Reads (PHYSICAL_READS)	Number of physical read operations on database block from disk	--	double	No	All	physical reads - physical reads direct - physical reads direct (lob)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations on the disk by DBWR	--	double	No	All	V$SYSSTAT.VALUE
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PD)	--	string(4)	No	All	Agent Collector
Recursive Calls (RECURSIVE_CALLS)	Number of user calls processed	--	double	No	All	V$SYSSTAT.VALUE
Recursive To User Call % (RECURSIVE_TO_USER_CALL_PERCENTAGE)	Correct values cannot be collected in this field. Percentage indicating overhead	--	double	No	All	(recursive calls / user calls) * 100
Redo Alloc Immediate % (REDO_ALLOC_IMMEDIATE_HIT_PERCENTAGE)	Success rate of immediately acquiring REDO allocation latch	--	double	No	All	(1 - (V $LATCH.IMMEDIATE_MISSES / (V $LATCH.IMMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES))) * 100 where V $LATCH.NAME = 'redo allocation'
Redo Alloc Willing to Wait % (REDO_ALLOC_WILLING_TO_WAIT_HIT_PERCENTAGE)	Success rate of acquiring REDO allocation latch from cache	--	double	No	All	(1 - (V $LATCH.MISSES / V $LATCH.GETS)) * 100 where name = 'redo allocation'
Redo Copy Immediate % (REDO_COPY_IMMEDIATE_HIT_PERCENTAGE)	Success rate of immediately acquiring REDO copy latch	--	double	No	All	(1 - (V $LATCH.IMMEDIATE_MISSES / (V $LATCH.MMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES))) * 100 where name = 'redo copy'

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Redo Copy Willing to Wait % (REDO_COPY_WILLING_TO_WAIT_HIT_PERCENTAGE)	Success rate of acquiring REDO copy latch from cache	--	double	No	All	(1 - (V$LATCH.MISSES / V$LATCH.GETS)) * 100 where name = 'redo copy'
Redo Log Buffer Alloc Retries (REDO_LOG_BUFFER_ALLOC_RETRIES)	Number of waits that occurred when an attempt was made to allocate the REDO log buffer	--	double	No	All	V$SYSSTAT.VALUE
Redo Log Buffer Wait % (REDO_LOG_BUFFER_WAIT_PERCENTAGE)	Percentage of waits that occurred when an attempt was made to allocate the REDO log buffer	--	double	No	All	(redo buffer allocation retries / redo entries) * 100
Redo Log Space Requests (REDO_LOG_SPACE_REQUESTS)	Number of times Oracle must wait for disk spaces to be allocated to REDO log entry because the active log file is full.	--	double	No	All	V$SYSSTAT.VALUE
Redo Log Space Wait % (REDO_LOG_SPACE_WAIT_PERCENTAGE)	Percentage of waiting time for disk space to be allocated for the REDO log entry.	--	double	No	All	(redo log space requests / redo entries) * 100
Row Source % (ROW_SOURCE_PERCENTAGE)	Percentage ratio of rows obtained by full-table scan	--	double	No	All	(table scan rows gotten / (table fetch by rowid + table scan rows gotten)) * 100
SQL Executing (SQL_EXECUTING)	This field is not supported. Number of current SQL executions	--	double	No	Not supported	Agent Collector
SQL Net Bytes Rcvd (SQL_NET_BYTES_RECEIVED)	Amount of data received from clients via SQL*Net	--	double	No	All	V$SYSSTAT.VALUE
SQL Net Bytes Sent (SQL_NET_BYTES_SENT)	Amount of data sent to clients via SQL*Net	--	double	No	All	V$SYSSTAT.VALUE
Session CPU Usage (SESSION_CPU_USAGE)	CPU time used in 1/100 seconds. To collect the value of this field, the TIMED_STATISTICS parameter must be set to TRUE in the init.ora file.	--	double	No	All	V$SYSSTAT.VALUE
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	Number of session cursors cached. The SESSION_CACHED_CURSORS parameter in the init.ora file specifies	--	double	No	All	V$SYSSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	the maximum value of this field.	--	double	No	All	V$SYSSTAT.VALUE
Session Cursor Cache Hit % (SESSION_CURSOR_CACHE_HIT_PERCENTAGE)	Hit rate of analysis calls in session's cursor cache	--	double	No	All	(session cursor cache hits / session cursor cache count) * 100
Session Cursor Cache Hits (SESSION_CURSOR_CACHE_HITS)	Number of times a cursor was found in the session's cursor cache by analysis call	--	double	No	All	V$SYSSTAT.VALUE
Session PGA Memory (SESSION_PGA_MEMORY)	Size of PGA memory currently being used in bytes	--	double	No	All	SUM(V$SESSTAT.VALUE)
Session UGA Memory (SESSION_UGA_MEMORY)	Size of used session memory in bytes	--	double	No	All	SUM(V$SESSTAT.VALUE)
Sort Overflow % (SORT_OVERFLOW_PERCENTAGE)	Percentage ratio of sort operations using a temporary segment	--	double	No	All	(sorts (disk) / (sorts (memory) + sorts (disk))) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Sys Undo Blk Wait Events (SYSTEM_UNDO_BLOCK_WAIT_EVENTS)	Number of block wait events for the system's rollback segment	--	double	No	All	V$WAITSTAT.COUNT where class = 'system undo block'
Sys Undo Hdr Wait Events (SYSTEM_UNDO_HEADER_WAIT_EVENTS)	Number of header block wait events for the system's rollback segment	--	double	No	All	V$WAITSTAT.COUNT where class = 'system undo header'
Total Logons (TOTAL_LOGONS)	Number of logins to the server	--	double	No	All	V$SYSSTAT.VALUE
Total SQL Executions (TOTAL_SQL_EXECUTIONS)	Total number of SQL executions	--	double	No	All	execute count
Undo Blk Wait Events (UNDO_BLOCK_WAIT_EVENTS)	Number of block wait events for other rollback segments	--	double	No	All	V$WAITSTAT.COUNT where class = 'undo block'
Undo Hdr Wait Events (UNDO_HEADER_WAIT_EVENTS)	Number of header block wait events for other rollback segments	--	double	No	All	V$WAITSTAT.COUNT where class = 'undo header'

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
User Calls (USER_CALLS)	Number of user calls processed	--	double	No	All	V$SYSSTAT.VALUE
User Calls / Parse (USER_CALLS_PER_PARSE)	Application management status in the context area	--	double	No	All	user calls / parse count (total)
User Commits (USER_COMMITS)	Number of transactions	--	double	No	All	V$SYSSTAT.VALUE
User Rollback % (USER_ROLLBACK_PERCENTAGE)	Failure rate of application transactions	--	double	No	All	(user rollbacks / (user commits + user rollbacks)) * 100
User Rollbacks (USER_ROLLBACKS)	Number of rollbacks	--	double	No	All	V$SYSSTAT.VALUE
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations	--	double	No	All	(physical writes / (physical reads + physical writes)) * 100

System Stat Summary Interval (PI)

Function

The System Stat Summary Interval (PI) record stores performance data, taken at specific intervals since the start of an instance, about key performance indicators.

Notes

If the monitoring target is Oracle Database 12c Release 2 or later in a CDB configuration, some fields in this record collect common performance data for database instances. The following table shows the relationship between the fields and performance data.

Fields	Performance data	
	When monitoring PDBs	When monitoring the root container (CDB$ROOT)
Dict Cache Get Misses % Lib Cache Miss % Redo Alloc Immediate % Redo Alloc Willing to Wait % Redo Copy Immediate % Redo Copy Willing to Wait %	Displays common performance data for database instances. Note The fields collect the same value when monitoring either the PDB or the root container (CDB$ROOT). To prevent alarm notification from multiple instances of PFM - Agent for Oracle, select the instance to be monitored.	
Free List Wait Events Sys Undo Blk Wait Events Sys Undo Hdr Wait Events Undo Blk Wait Events Undo Hdr Wait Events	Displays performance data for the PDBs that are being monitored.	• Use administrative accounts (sys or system) Performance data for all PDBs and the root container (CDB$ROOT) are collected. • Use common user Performance data for the root container (CDB$ROOT) is collected.
Recursive To User Call % SQL Executing	As with non CDB, unable to collect valid performance data.	
Fields except for above	Displays performance data for the PDBs that are being monitored.	Displays common performance data for database instances.

Default and changeable values

Item	Default value	Changeable
Collection Interval	60	Y
Collection Offset	0	Y
Log	Yes	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

None

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 1,974 bytes
- Variable part: 0 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Block Changes/Tran (BLOCK_CHANGES_PER_TRANSACTION)	Rate at which each transaction executed DML operation [#2]	AVG	double	Yes[#6]	All	db block changes / user commits
Block Get/sec (BLOCK_GET_RATE)	Rate at which the application referenced the database[#2]	AVG	double	Yes[#6]	All	(db block gets + consistent gets) / second in interval
Block Visits/Tran (BLOCK_VISITS_PER_TRANSACTION)	Number of times a work database was loaded per transaction[#2]	AVG	double	Yes[#6]	All	(db block gets + consistent gets) / user commits
Buffer Busy Wait % (BUFFER_BUSY_WAIT_PERCENTAGE)	Percentage ratio of buffer busy waits[#2]	AVG	double	Yes[#6]	All	(V $SYSTEM_EVENT.TOTAL_WAITS where EVENT = 'buffer busy waits' / (consistent gets + db block gets)) * 100
Cache Hit % (CACHE_HIT_PERCENTAGE)	Buffer cache usage[#2]	AVG	double	Yes[#6]	All	(1 - (physical reads cache / (consistent gets from cache + db block gets from cache))) * 100
Call/sec (CALL_RATE)	Work demand rate applied to the instance by all work resources[#2]	AVG	double	Yes[#6]	All	recursive calls + user calls / seconds in interval
Calls/Tran (CALLS_PER_TRANSACTION)	Rate at which client requests were executed per transaction. You can use the value of this field to detect a change in the application or the utilization status. This value may increase significantly if unique queries increase.[#2]	AVG	double	No	All	user calls / user commits
Changed Block % (CHANGED_BLOCK_PERCENTAGE)	Percentage ratio of difference between query and database manipulation language (DML) in the database application. This value changes according to indexes and application utilization status.[#2]	AVG	double	Yes[#6]	All	(db block changes / (block gets + consistent gets)) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Percentage indicating the extents for which the application needed to check the consistency of read operations[#2]	AVG	double	Yes[#6]	All	(consistent changes / consistent gets) * 100
Continued Row % (CONTINUED_ROW_PERCENTAGE)	Percentage ratio of continued rows. The value is close to 0 unless the application handles LONG columns. [#2]	AVG	short	Yes[#6]	All	(table fetch continued row / (table fetch by rowid + table scan rows gotten)) * 100
Current Logons (CURRENT_LOGONS)	Number of current logons to Oracle Database[#2]	AVG	ulong	No	All	V$SYSSTAT.VALUE
Deadlocks (LOCK_DEADLOCKS)	Number of deadlocks caused by locked DML processing[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Dict Cache Get Miss % (DICTIONARY_CACHE_GET_MISSES_PERCENTAGE)	Percentage ratio of data requests issued due to cache miss[#2]	AVG	double	Yes[#6]	All	(SUM(V$ROWCACHE.GETMISSES) / SUM(V$ROWCACHE.GETS)) * 100
Disk Sorts (SORTS_DISK)	Number of disk sort operations[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Free List Wait Events (FREE_LIST_WAIT_EVENTS)	Number of wait events in the free list[#2]	AVG	double	Yes	All	V$WAITSTAT.COUNT where class = 'free list'
I/O Ops/sec (IO_RATE)	Number of I/O operations per second[#2]	AVG	double	No	All	(physical reads + physical writes) / seconds in interval
Lib Cache Miss % (LIBRARY_CACHE_MISS_PERCENTAGE)	Library cache miss rate. This field means the ratio of times the allocated objects in library cache are reloaded. As the value of this field increases, the amount of resources in use also increases. [#2]	AVG	double	Yes[#6]	All	(SUM(V$LIBRARYCACHE.RELOADS) SUM(V$LIBRARYCACHE.PINS)) * 100
Lock Conversions (LOCK_CONVERSIONS)	Number of enqueues (locks) whose mode was changed (such as from share to lock)[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Lock Hit % (LOCK_HIT_PERCENTAGE)	Percentage ratio of the number of accesses to data blocks to the total number of data block accesses. However, this is limited to cases that do not request a lock conversion. Only valid with an Oracle RAC configuration. [#2]	AVG	double	Yes[#6]	All	((consistent gets - global enqueue gets async) / consistent gets) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Lock Releases (LOCK_RELEASES)	Number of times enqueuing (locking) was released[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Lock Requests (LOCK_REQUESTS)	Number of times enqueuing (locking) was requested[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Lock Timeouts (LOCK_TIMEOUTS)	Number of times an enqueuing (locking) request was not permitted within the allocated time[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Lock Waits (LOCK_WAITS)	Number of times an enqueuing (locking) request was placed in wait status. The difference between the enqueuing requests count and the enqueuing waits count is the number of times the request was not treated as an enqueuing request. [#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Logical Reads (LOGICAL_READS)	Number of logical read operations in read consistency mode and number of requests to the current copy of blocks[#2]	AVG	double	Yes	All	db block gets + consistent gets
Memory Sorts (SORTS_MEMORY)	Number of sort operations in memory[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Non-Index Lookups % (NON_INDEX_LOOKUPS)	Percentage ratio of full table scans that do not involve caching[#2]	AVG	double	Yes[#6]	All	(table scans (long tables) / (table scans (short tables) + table scans (long tables))) * 100
Physical Reads (PHYSICAL_READS)	Number of physical read operations on database block from disk[#2]	AVG	double	Yes	All	physical reads - physical reads direct - physical reads direct (lob)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations on the disk by DBWR[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Ping Write % (PING_WRITE_PERCENTAGE)	A large value means that lock conversion occurred frequently. In the case of Oracle 11g R2 or later, this field is always set to 0.[#2]	AVG	double	Yes[#6]	All	(DBWR cross instance writes / physical writes) * 100
Read/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	physical reads / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Type (INPUT_RECORD_TYPE)	Record name (always PI)[#1]	COPY	string(4)	No	All	Agent Collector
Recursive Calls (RECURSIVE_CALLS)	Number of user calls processed[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Recursive To User Call % (RECURSIVE_TO_USER_CALL_PERCENTAGE)	Correct values cannot be collected in this field. Percentage indicating overhead[#2]	AVG	double	Yes[#6]	All	(recursive calls / user calls) * 100
Redo Alloc Immediate % (REDO_ALLOC_IMMEDIATE_HIT_PERCENTAGE)	Success rate of immediately acquiring REDO allocation latch[#2]	AVG	double	Yes[#6]	All	(1 - (V $LATCH.IMMEDIATE_MISSES / (V $LATCH.IMMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES))) * 100 where V $LATCH.NAME = 'redo allocation'
Redo Alloc Willing to Wait % (REDO_ALLOC_WILLING_TO_WAIT_HIT_PERCENTAGE)	Success rate of acquiring REDO allocation latch from cache[#2]	AVG	double	Yes[#6]	All	(1 - (V $LATCH.MISSES / V $LATCH.GETS)) * 100 where name = 'redo allocation'
Redo Copy Immediate % (REDO_COPY_IMMEDIATE_HIT_PERCENTAGE)	Success rate of immediately acquiring REDO copy latch[#2]	AVG	double	Yes[#6]	All	(1 - (V $LATCH.IMMEDIATE_MISSES / (V $LATCH.IMMEDIATE_GETS + V $LATCH.IMMEDIATE_MISSES))) * 100 where V $LATCH.NAME = 'redo copy'
Redo Copy Willing to Wait % (REDO_COPY_WILLING_TO_WAIT_HIT_PERCENTAGE)	Success rate of acquiring REDO copy latch from cache[#2]	AVG	double	Yes[#6]	All	(1 - (V $LATCH.MISSES / V $LATCH.GETS)) * 100 where name = 'redo copy'
Redo Log Buffer Alloc Retries (REDO_LOG_BUFFER_ALLOC_RETRIES)	Number of waits that occurred when an attempt was made to allocate the REDO log buffer	AVG	double	Yes	All	V$SYSSTAT.VALUE
Redo Log Buffer Wait % (REDO_LOG_BUFFER_WAIT_PERCENTAGE)	Percentage of waits that occurred when an attempt was made to allocate the REDO log buffer	AVG	double	Yes[#6]	All	(redo buffer allocation retries / redo entries) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Redo Log Space Requests (REDO_LOG_SPACE_REQUESTS)	Number of times Oracle must wait for disk spaces to be allocated to REDO log entry because the active log file is full. #2	AVG	double	Yes	All	V$SYSSTAT.VALUE
Redo Log Space Wait % (REDO_LOG_SPACE_WAIT_PERCENTAGE)	Percentage of waiting time until disk space is allocated for the REDO log entry#2	AVG	double	Yes#6	All	(redo log space requests / redo entries) * 100
Row Source % (ROW_SOURCE_PERCENTAGE)	Percentage ratio of rows obtained by full-table scan#2	AVG	double	Yes#6	All	(table scan rows gotten / (table fetch by rowid + table scan rows gotten)) * 100
SQL Executing (SQL_EXECUTING)	This field is not supported. Number of current SQL executions	AVG	double	No	Not supported	Agent Collector
SQL Net Bytes Rcvd (SQL_NET_BYTES_RECEIVED)	Amount of data received from clients via SQL*Net#2	AVG	double	Yes	All	V$SYSSTAT.VALUE
SQL Net Bytes Sent (SQL_NET_BYTES_SENT)	Amount of data sent to clients via SQL*Net#2	AVG	double	Yes	All	V$SYSSTAT.VALUE
Session CPU Usage (SESSION_CPU_USAGE)	If the value of the TIMED_STATISTICS parameter in the init.ora file is TRUE, the CPU time used (in 1/100 seconds); if the parameter value is FALSE, this field contains 0.#2	AVG	double	Yes	All	V$SYSSTAT.VALUE
Session Cursor Cache Count (SESSION_CURSOR_CACHE_COUNT)	Number of analysis calls for the session's cursor calls#2	AVG	double	Yes	All	V$SYSSTAT.VALUE
Session Cursor Cache Hit % (SESSION_CURSOR_CACHE_HIT_PERCENTAGE)	Hit rate of analysis calls in session's cursor cache#2	AVG	double	No	All	(session cursor cache hits / session cursor cache count) * 100
Session Cursor Cache Hits (SESSION_CURSOR_CACHE_HITS)	Number of times a cursor was found in the session's cursor cache by analysis call#2	AVG	double	Yes	All	V$SYSSTAT.VALUE
Session PGA Memory (SESSION_PGA_MEMORY)	Size of PGA memory currently being used in bytes#2	AVG	double	No	All	SUM(V$SESSTAT.VALUE)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Session UGA Memory (SESSION_UGA_MEMORY)	Size of used session memory in bytes[#2]	AVG	double	No	All	SUM(V$SESSTAT.VALUE)
Sort Overflow % (SORT_OVERFLOW_PERCENTAGE)	Percentage ratio of sort operations using a temporary segment [#2]	AVG	double	Yes[#6]	All	(sorts (disk) / (sorts (memory) + sorts (disk))) * 100
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Sys Undo Blk Wait Events (SYSTEM_UNDO_BLOCK_WAIT_EVENTS)	Number of block wait events for the system's rollback segment [#2]	AVG	double	Yes	All	V$WAITSTAT.COUNT where class = 'system undo block'
Sys Undo Hdr Wait Events (SYSTEM_UNDO_HEADER_WAIT_EVENTS)	Number of header block wait events for the system's rollback segment [#2]	AVG	double	Yes	All	V$WAITSTAT.COUNT where class = 'system undo header'
Total Logons (TOTAL_LOGONS)	Number of logins to the server [#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
Total SQL Executions (TOTAL_SQL_EXECUTIONS)	Total number of SQL executions [#2]	AVG	double	Yes	All	execute count
Trans/sec (TRANSACTION_RATE)	Number of transactions per second[#2]	AVG	double	No	All	user commits / seconds in interval
Undo Blk Wait Events (UNDO_BLOCK_WAIT_EVENTS)	Number of block wait events for other rollback segments[#2]	AVG	double	Yes	All	V$WAITSTAT.COUNT where class = 'undo block'
Undo Hdr Wait Events (UNDO_HEADER_WAIT_EVENTS)	Number of header block wait events for other rollback segments[#2]	AVG	double	Yes	All	V$WAITSTAT.COUNT where class = 'undo header'
User Calls (USER_CALLS)	Number of user calls processed[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
User Calls / Parse (USER_CALLS_PER_PARSE)	Application management status in the context area [#2]	AVG	double	Yes[#6]	All	user calls / parse count (total)
User Commits (USER_COMMITS)	Number of transactions[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE
User Rollback % (USER_ROLLBACK_PERCENTAGE)	Failure rate of application transactions [#2]	AVG	double	No	All	(user rollbacks / (user commits + user rollbacks)) * 100
User Rollbacks (USER_ROLLBACKS)	Number of rollbacks[#2]	AVG	double	Yes	All	V$SYSSTAT.VALUE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations #2	AVG	double	No	All	(physical writes / (physical reads + physical writes)) * 100
Writes/sec (WRITES_RATE)	Number of write operations per second#2	AVG	double	No	All	physical writes / seconds in interval

#6

The field value is calculated based on the delta of the amount of collected data from Oracle Database.

System Statistics (PD_PDST)

Function

The System Statistics (PD_PDST) record stores performance data indicating the status of all sessions in the system at a specific point in time. PFM - Agent for Oracle creates one record for each statistical value. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	300	Y
Collection Offset	5	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDST_NAME

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 96 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Class (CLASS)	Statistic class	--	string(20)	No	All	V$SYSSTAT.CLASS
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDST)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Statistic # (STATISTIC_NUM)	Statistic number	--	short	No	All	V$SYSSTAT.STATISTIC#

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Statistic Name (NAME)	Statistic name	--	string(64)	No	All	V$SYSSTAT.NAME
Value (VALUE)	Statistical value	--	double	No	All	V$SYSSTAT.VALUE

Table Access (PD_PDTA)

Function

The Table Access (PD_PDTA) record stores performance data indicating the status of a table accessed by sessions during data collection at a specific point in time. PFM - Agent for Oracle creates one record for each table accessed by sessions. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	130	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDTA_SID
- PD_PDTA_OBJECT

Lifetime

From the locking to the unlocking of an object

Record size

- Fixed part: 678 bytes
- Variable part: 250 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Object (OBJECT)	Object name	--	string(100)	No	All	V$ACCESS.OBJECT
Owner (OWNER)	Object's owner	--	string(64)	No	All	V$ACCESS.OWNER
Program (PROGRAM)	Name of the program being executed	--	string(48)	No	All	V $SESSION.PROGRAM where V $ACCESS.SID = V $SESSION.SID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Type (INPUT_RECORD_TYPE)	Record name (always PDTA)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID accessing the object whose name is specified in the Object field	--	ulong	No	All	V$ACCESS.SID where V$ACCESS.SID = V$SESSION.SID
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME where V$ACCESS.SID = V$ACCESS.SID

Tablespace (PD_PDTS)

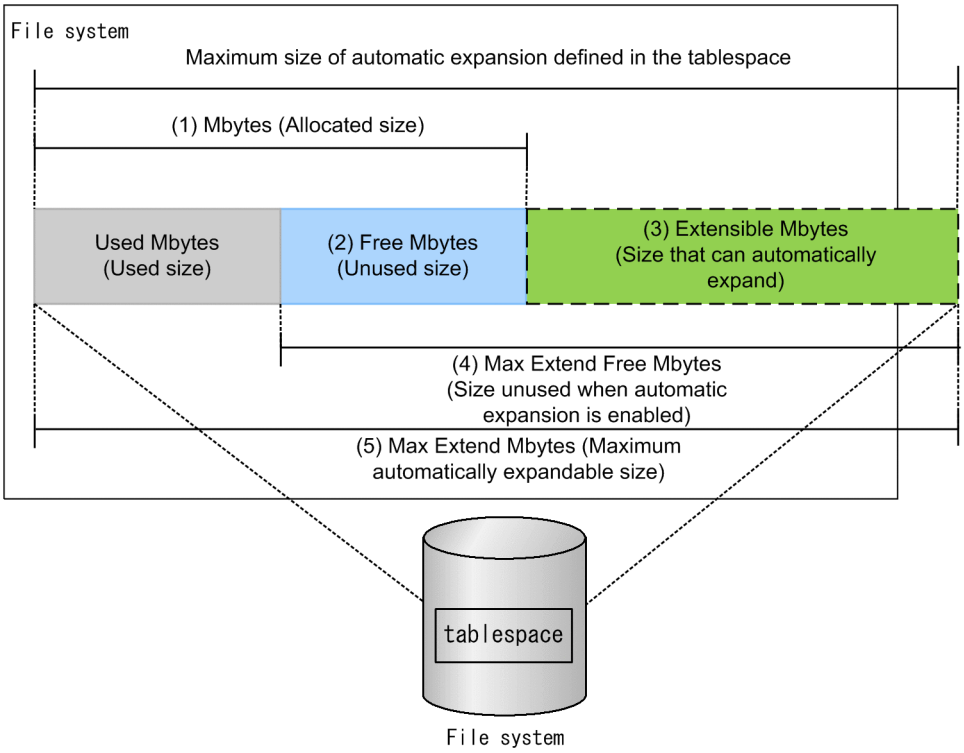
Function

The Tablespace (PD_PDTS) record stores performance data indicating the status (at a specific point in time) of tablespaces in a database. PFM - Agent for Oracle creates one record for each tablespace in a database. This is a multi-instance record.

For details about the difference between the Tablespace (PD_PDTS) record and the CDB Tablespace (CD_CDTS) record, see the information provided in the *CDB Tablespace (PD_CDTS)* record.

The following figure shows the relationship between the fields associated with tablespace sizes.

Figure 6–5: Relationship between the fields associated with tablespace sizes



Expressions for calculating the values of fields related to the space size percentage:
The same expressions are used for all tablespaces.

- Free % (percentage of free space)
  ([Free Mbytes (2)] / [Mbytes (1)]) × 100
- Max Extend Free % (percentage of unused size in relation to the maximum size of automatic expansion)
  ([Max Extend Free Mbytes (4)] / [Max Extend Mbytes (5)]) × 100
- Extensible Mbytes % (percentage of remaining size that can automatically expand)
  ([Extensible Mbytes (3)] / [Max Extend Mbytes (5)]) × 100

The following fields related to the automatic expansion of tablespaces are valid when Oracle Database 11g R2 or later is monitored. When Oracle Database 11g R1 or earlier is monitored, these fields are either always set to 0 or empty.

No.	Field	Value set when Oracle Database 11g R1 or earlier is monitored
1	Auto Extensible	Empty

No.	Field	Value set when Oracle Database 11g R1 or earlier is monitored
2	Extensible Mbytes	0
3	Extensible Mbytes %	0
4	Max Extend Free %	0
5	Max Extend Free Mbytes	0
6	Max Extend Mbytes	0

If the size of automatic expansion is unlimited, the Max Extend Free Mbytes field displays the value based on the theoretical maximum size that can be set for Oracle Database. The values of the other fields are calculated based on the value of the Max Extend Free Mbytes field.

The values set for `localtemp_option` and `undospace_option` in the instance information affect only values of the Used Mbytes field and Free Mbytes field. The following figures show effects of the values of `localtemp_option` and `undospace_option`.

Figure 6–6: Effects of the value of localtemp_option

Temporary tablespace	100MB
Space that was allocated once	80MB
Space in use	40MB

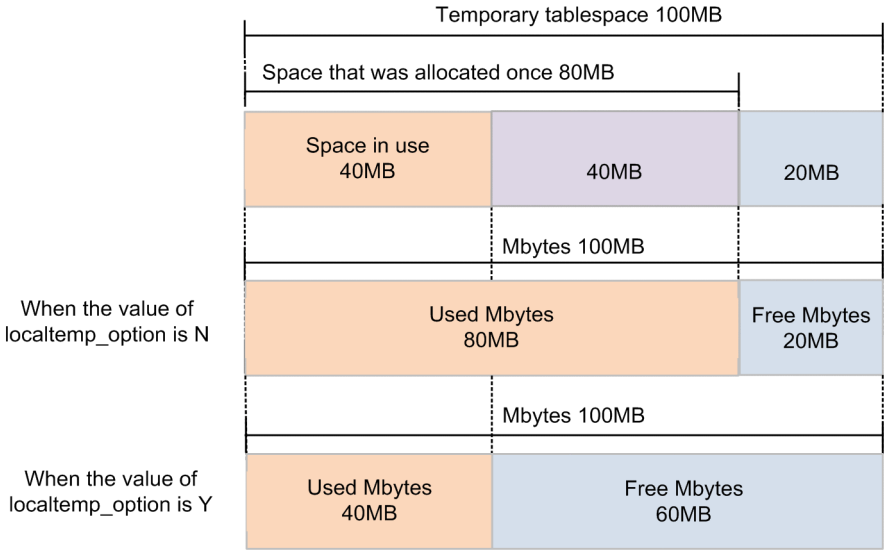
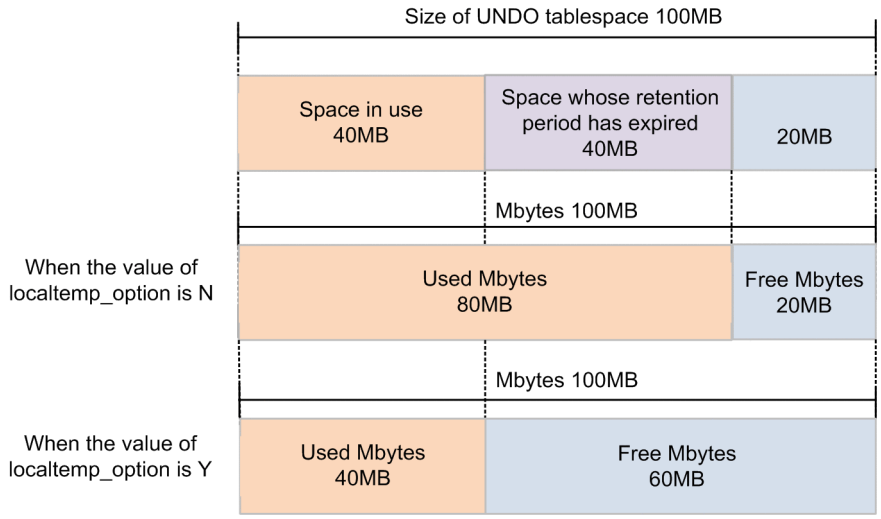


Figure 6–7: Effects of the value of undospace_option

Size of UNDO tablespace	100MB
Space whose retention period has expired	40MB
Space in use	40MB



Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	30	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDTS_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a tablespace

Record size

- Fixed part: 678 bytes
- Variable part: 161 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Auto Extensible (AUTO_EXTENSIBLE)	Indicates whether automatic expansion of the tablespace is enabled when the	--	string(3)	No	All	DBA_DATA_FILES.AUTOEXTENSIBLE

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Auto Extensible (AUTO_EXTENSIBLE)	monitoring target is Oracle Database 11g R2 or later. YES is set if automatic expansion is enabled. NO is set if automatic expansion is disabled. YES is set if automatic expansion is enabled for at least one of the data files which comprise the tablespace. A blank is set if the monitoring target is Oracle Database 11g R1 or earlier.	--	string(3)	No	All	DBA_DATA_FILES.AUTOEXTENSIBLE
Blocks (BLOCKS)	Size of tablespace in Oracle blocks	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_DATA_FILES.BLOCKS) • For locally managed temporary tablespaces: SUM(DBA_TEMP_FILES.BLOCKS)
Data Files (DATAFILES)	Number of data files in use by the tablespace	--	ushort	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_DATA_FILES) • For locally managed temporary tablespaces: COUNT(DBA_TEMP_FILES)
Extensible Mbytes (EXTENSIBLE_BYTES)	Size remaining in megabytes for automatic expansion when the monitoring target is Oracle Database 11g R2 or later.	--	double	No	All	MAX_BYTES - BYTES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Extensible Mbytes (EXTENSIBLE_BYTES)	0 is set if automatic expansion is disabled, the maximum size has already been reached, or the monitoring target is Oracle Database 11g R1 or earlier.	--	double	No	All	MAX_BYTES - BYTES
Extensible Mbytes % (PERCENT_EXTENSIBLE_BYTES)	Percentage of space remaining for automatic expansion if the monitoring target is Oracle Database 11g R2 or later. 0 is set if automatic expansion is disabled, the maximum size has already been reached, or the monitoring target is Oracle Database 11g R1 or earlier.	--	double	No	All	(EXTENSIBLE_BYTES / MAX_BYTES) * 100
Extents (EXTENTS)	Number of extents	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or directory managed temporary tablespaces: SUM(DBA_SEGMENTS.EXTENTS) • For locally managed temporary tablespaces when the value of localtemp_option is Y: SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) • For locally managed temporary tablespaces when the value of localtemp_option is N: SUM(V$SORT_SEGMENT.TOTAL_EXTENTS)
Free % (PERCENT_FREE)	Percentage ratio of free space	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces,

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free % (PERCENT_FREE)	Percentage ratio of free space	--	double	No	All	dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_FREE_SPACE.BYTES) / DBA_DATA_FILES.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is Y: ((SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / DBA_TEMP_FILES.BYTES) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is N: (SUM(V$TEMP_SPACE_HEADER.BYTES_FREE) / (DBA_TEMP_FILES.BYTES)) * 100 • For the UNDO tablespaces when the value of undospace_option is Y: ((SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / DBA_DATA_FILES.BYTES) * 100
Free Extents (FREE_EXTENTS)	Number of free extents.	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Extents (FREE_EXTENTS)	Number of free extents.	--	ulong	No	All	dictionary managed temporary tablespaces: COUNT(DBA_FREE_SPACE) • For locally managed temporary tablespaces when the value of localtemp_option is Y: SUM(DBA_TEMP_FILES.BYTES / V$TEMP_EXTENT_MAP.BYTES) - V$SORT_SEGMENT.USED_EXTENTS • For locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(V$TEMP_SPACE_HEADER)
Free Mbytes (FREE_BYTES)	Size of free space in megabytes.	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: SUM(DBA_FREE_SPACE.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Free Mbytes (FREE_BYTES)	Size of free space in megabytes.	--	double	No	All	SUM(V$TEMP_SPACE_HEADER.BYTES_FREE) / (1024 * 1024) • For the UNDO tablespaces when the value of undospace_option is Y: (SUM(DBA_FREE_SPACE.BYTES) + SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 **x** 1024)
Increase % (PCT_INCREASE)	Default rate of increase in the extent size	--	short	No	All	DBA_TABLESPACES.PCT_INCREASE
Initial Extent (INITIAL_EXTENT)	Default size of the initial extent	--	double	No	All	DBA_TABLESPACES.INITIAL_EXTENT
Max Extend Free % (MAX_PERCENT_FREE)	Percentage of the unused size in relation to the maximum automatically expandable size if the monitoring target is Oracle Database 11g R2 or later. The value is the same as the Free % field when automatic expansion is disabled or the maximum size has already been reached. The value is 0 when the monitoring target is Oracle Database 11g R1 or earlier.	--	double	No	All	(MAX_FREE_BYTES / MAX_BYTES) * 100
Max Extend Free Mbytes (MAX_FREE_BYTES)	Size unused when automatic expansion is enabled and the monitoring target is Oracle Database 11g R2 or later. This size is the total of the unused size and the automatically expandable size in megabytes. The value is the same as the Free Mbytes field	--	double	No	All	MAX_BYTES - USED_BYTES

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Max Extend Free Mbytes (MAX_FREE_BYTES)	when automatic expansion is disabled or the maximum size has already been reached. The value is 0 when the monitoring target is Oracle Database 11g R1 or earlier.	--	double	No	All	MAX_BYTES - USED_BYTES
Max Extend Mbytes (MAX_BYTES)	Maximum automatic expandable size (in megabytes) when the monitoring target is Oracle Database 11g R2 or later. The value is the same as the Mbytes field when automatic expansion is disabled or the maximum size has already been reached. The value is 0 when the monitoring target is Oracle Database 11g R1 or earlier.	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_DATA_FILES.MAXBYTES) / (1024 * 1024) • For locally managed temporary tablespaces: SUM(DBA_TEMP_FILES.MAXBYTES) / (1024 * 1024)
Max Extents (MAX_EXTENTS)	Default maximum number of extents	--	ulong	No	All	DBA_TABLESPACES.MAX_EXTENTS
Mbytes (BYTES)	Size of the tablespace in megabytes #4	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_DATA_FILES.BYTES) / (1024 * 1024) • For locally managed temporary tablespaces: SUM(DBA_TEMP_FILES.BYTES) / (1024 * 1024)
Min Extents (MIN_EXTENTS)	Default minimum number of extents	--	long	No	All	DBA_TABLESPACES.MIN_EXTENTS
Next Extent (NEXT_EXTENT)	Default size of the incremental extent	--	double	No	All	DBA_TABLESPACES.NEXT_EXTENT
Record Time (RECORD_TIME)	Collection termination time for	--	time_t	No	All	Agent Collector

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Record Time (RECORD_TIME)	the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDTS)	--	string(4)	No	All	Agent Collector
Segments (SEGMENTS)	Number of segments. Always 1 for locally managed temporary tablespaces when the value of localtemp_option is Y.	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_SEGMENTS) • For locally managed temporary tablespaces when the value of localtemp_option is Y: Agent Collector • For locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(V$SORT_SEGMENT)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	Time_t	No	All	Agent Collector
Status (STATUS)	Tablespace status. Valid values are INVALID (tablespace was deleted), OFFLINE, and ONLINE.	--	string(9)	No	All	DBA_TABLESPACES STATUS
Tablespace Name (TABLESPACE_NAME)	Tablespace name	--	string(30)	No	All	DBA_TABLESPACES.TABLESPACE_NAME
Used Mbytes (USED_BYTES)	Size of used area in megabytes. #4	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, dictionary managed temporary tablespaces, or the UNDO tablespaces when the value of undospace_option is N: (SUM(DBA_DATA_FILES.BYTES) -

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Used Mbytes (USED_BYTES)	Size of used area in megabytes. #4	--	double	No	All	SUM(DBA_FREE_SPACE.BYTES)) / (1024 *1024) • For locally managed temporary tablespaces when the value of localtemp_option is Y: (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES)) / (1024 * 1024) • For locally managed temporary tablespaces when the value of localtemp_option is N: (SUM(DBA_TEMP_FILES.BYTES) - SUM(V$TEMP_SPACE_HEADER.BYTES_FREE)) / (1024 *1024) • For the UNDO tablespaces when the value of undospace_option is Y: (DBA_DATA_FILES.BYTES - SUM(DBA_FREE_SPACE.BYTES) - SUM(DBA_UNDO_EXTENTS.BYTES) WHERE STATUS='EXPIRED') / (1024 * 1024)

Tablespace Fragmentation (PD_PDTF)

Function

The Tablespace Fragmentation (PD_PDTF) record stores performance data indicating the status (at a specific point in time) of fragmentation of tablespaces. PFM - Agent for Oracle creates one record for each tablespace in a database. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	135	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDTF_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a tablespace

Record size

- Fixed part: 678 bytes
- Variable part: 79 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Fragment (AVERAGE_FRAGMENT)	Average fragment size in bytes. The value of this field is free space when this field displays locally managed temporary tablespaces and the value of localtemp_option is Y, because fragment to the locally managed temporary tablespace is 1.	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_FREE_SPACE.BYTES) / COUNT(DBA_FREE_SPACE) where DBA_TABLESPACES.TABLESPACE_NAME = DBA_FREE_SPACE.TABLESPACE_NAME(+) • For locally managed temporary tablespaces when the value of localtemp_option is Y:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Avg Fragment (AVERAGE_FRAGMENT)	Average fragment size in bytes. The value of this field is free space when this field displays locally managed temporary tablespaces and the value of localtemp_option is Y, because fragment to the locally managed temporary tablespace is 1.	--	double	No	All	SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V&TEMP_EXTENT_MAP.BYTES)) • For locally managed temporary tablespaces when the value of localtemp_option is N: SUM(V$TEMP_SPACE_HEADER.BYTES_FREE) / COUNT(V$TEMP_SPACE_HEADER) where DBA_TEMP_FILES.FILE_ID = V$TEMP_SPACE_HEADER.FILE_ID(+)
Extents (EXTENTS)	Number of extents	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(DBA_SEGMENTS.EXTENTS) • For locally managed temporary tablespaces when the value of localtemp_option is N: SUM(V$SORT_SEGMENT.TOTAL_EXTENTS)
Fragments (FRAGMENTS)	Number of fragments. Always 1 for locally managed temporary tablespaces when the value of localtemp_option is Y.	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_FREE_SPACE) where DBA_TABLESPACES.TABLESPACE_NAME = DBA_FREE_SPACE.TABLESPACE_NAME(+) • For locally managed temporary tablespaces when the value of localtemp_option is Y: Agent Collector • For locally managed temporary tablespaces when the value of localtemp_option is N:

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Fragments (FRAGMENTS)	Number of fragments. Always 1 for locally managed temporary tablespaces when the value of localtemp_option is Y.	--	ulong	No	All	COUNT(V $TEMP_SPACE_HEADER) where DBA_TEMP_FILES.FILE_ID = V $TEMP_SPACE_HEADER.FILE_ID(+)
High Max Extents (HIGH_MAX_EXTENTS)	Number of segments whose PCT_MAX_EXTENTS value exceeds 90%	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_SEGMENTS) where EXTENTS > MAX_EXTENTS * 0.9 • For locally managed temporary tablespaces: COUNT(V $SORT_SEGMENT) where EXTENT_SIZE > TOTAL_EXTENTS * 0.9
Largest Fragment (LARGEST_FRAGMENT)	Largest fragment size in bytes The value of this field is free space when this field displays locally managed temporary tablespaces and the value of localtemp_option is Y, because fragment to the locally managed temporary tablespace is 1.	--	double	No	All	• For dictionary managed permanent tabelspaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: MAX(DBA_FREE_SPACE.BYTES) where DBA_TABLESPACES.TABLESPACE_NAME = DBA_FREE_SPACE.TABLESPACE_NAME(+) • For locally managed temporary tablespaces when the value of localtemp_option is Y: SUM(DBA_TEMP_FILES.BYTES) - (V $SORT_SEGMENT.USED_EXTENTS * AVG(V $TEMP_EXTENT_MAP.BYTES)) • For locally managed temporary tablespaces: MAX(V $TEMP_SPACE_HEADER.BYTES_FREE) where DBA_TEMP_FILES.FILE_ID = V $TEMP_SPACE_HEADER.FILE_ID(+)
Largest Fragment % (LARGEST_FRAGMENT_PERCENT)	Percentage ratio of tablespace in the largest fragment	--	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Largest Fragment % (LARGEST_FRAGMENT_PERCENT)	Percentage ratio of tablespace in the largest fragment	--	double	No	All	tablespaces, or dictionary managed temporary tablespaces: (MAX(DBA_FREE_SPACE.BYTES) / SUM(DBA_DATA_FILES.BYTES)) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is Y: ((SUM(DBA_TEMP_FILES.BYTES) - (V$SORT_SEGMENT.USED_EXTENTS * AVG(V$TEMP_EXTENT_MAP.BYTES))) / SUM(DBA_TEMP_FILES_BYTES)) * 100 • For locally managed temporary tablespaces when the value of localtemp_option is N: (MAX(V$TEMP_SPACE_HEADER.BYTES_FREE) / SUM(DBA_TEMP_FILES.BYTES)) * 100
Next Alloc Fails (NEXT_ALLOC_FAILS)	Indicates whether the following extent allocation failed. For failure: 1 For success: 0 These results take effect for the following conditions: • The tablespace is a locally managed tablespace. • Uniform extent management is performed for the extent. 0 is returned for any other conditions.	--	ulong	No	All	• For locally managed permanent tablespaces: MAX(DBA_SEGMENTS.NEXT_EXTENT) > MAX(DBA_FREE_SPACE.BYTES)
Overextended (OVEREXTENDED)	Number of segments with more than five extents	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_SEGMENTS) where EXTENTS > 5

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Overextended (OVEREXTENDED)	Number of segments with more than five extents	--	ulong	No	All	• For locally managed temporary tablespaces: COUNT(V$SORT_SEGMENT) where TOTAL_EXTENTS > 5
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDTF)	--	string(4)	No	All	Agent Collector
Segments (SEGMENTS)	Number of segments Always 1 for locally managed temporary tablespaces when the value of localtemp_option is Y.	--	ulong	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_SEGMENTS) • For locally managed temporary tablespaces when the value of localtemp_option is Y: Agent Collector • For locally managed temporary tablespaces when the value of localtemp_option is N: COUNT(V$SORT_SEGMENT)
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Tablespace name	--	string(30)	No	All	DBA_TABLESPACES.TABLESPACE_NAME

Tablespace Interval (PI_PITS)

Function

The Tablespace Interval (PI_PITS) record stores performance data, taken at specific intervals, about tablespaces in a database. PFM - Agent for Oracle creates one record for each tablespace in a database. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	3600	Y
Collection Offset	50	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PI_PITS_TABLESPACE_NAME

Lifetime

From the creation to the deletion of a tablespace

Record size

- Fixed part: 678 bytes
- Variable part: 291 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Datafiles (DATAFILES)	Number of data files in use by the tablespace[#2]	AVG	ushort	No	All	• Dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: COUNT(DBA_DATA_FILES) • For locally managed temporary tablespaces: COUNT(DBA_TEMP_FILES)
I/O Ops/sec (IO_RATE)	Number of I/O operations per second[#2]	AVG	double	No	All	(SUM(V$FILESTAT.PHYRDS) + SUM(V$FILESTAT.PHYWRTS)) / seconds in interval

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Physical Blocks Read (PHYSICAL_BLOCKS_READ)	Number of physical blocks read[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYBLKRD) • For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYBLKRD)
Physical Blocks Written (PHYSICAL_BLOCKS_WRITTEN)	Number of physical blocks written[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYBLKWRT) • Locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYBLKWRT)
Physical Reads (PHYSICAL_READS)	Number of physical read operations that were completed[#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYRDS) • For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYRDS)
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed [#2]	AVG	double	Yes	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYWRTS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Physical Writes (PHYSICAL_WRITES)	Number of physical write operations that were completed [#2]	AVG	double	Yes	All	• For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYWRTS)
Reads/sec (READ_RATE)	Number of read operations per second[#2]	AVG	double	No	All	SUM(V$FILESTAT.PHYRDS) / seconds in interval
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PITS)[#1]	COPY	string(4)	No	All	Agent Collector
Rollback Segments (ROLLBACK_SEGMENTS)	Number of rollback segments. Performance data about the locally managed tablespaces is not collected.[#2]	AVG	ulong	No	All	COUNT(DBA_ROLLBACK_SEGS)
Rollback Segments Hit % (ROLLBACK_SEGMENTS_HIT_PERCENTAGE)	Percentage ratio of the HIT value to the GET value. Performance data about locally managed tablespaces is not collected.[#2]	AVG	double	No	All	((SUM(V$ROLLSTAT.GETS) - SUM(V$ROLLSTAT.WAITS)) / SUM(V$ROLLSTAT.GETS)) * 100
Rollback Segments Trans (ROLLBACK_SEGMENTS_TRANS)	Number of transactions that were active during data collection. Performance data about locally managed temporary tablespace is not collected.[#2]	AVG	long	No	All	SUM(V$ROLLSTAT.XACTS)
Sort Segments (SORT_SEGMENTS)	Number of sort segments. Performance data about locally managed permanent tablespaces is not collected.[#2]	AVG	ulong	No	All	COUNT(V$SORT_SEGMENT)
Sorting Users (SORTING_USERS)	Number of users that were active in the sort segment during data collection. Performance data about locally managed permanent tablespaces is not collected.[#2]	AVG	long	No	All	SUM(V$SORT_SEGMENT.CURRENT_USERS)

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Tablespace Name (TABLESPACE_NAME)	Tablespace name[#1]	COPY	string(30)	No	All	DBA_TABLESPACES.TABLESPACE_NAME
Write % (WRITE_PERCENTAGE)	Percentage ratio of write operations to all physical I/O operations [#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: (SUM(V$FILESTAT.PHYWRTS) / (SUM(V$FILESTAT.PHYRDS) + SUM(V$FILESTAT.PHYWRTS))) * 100 • For locally managed temporary tablespaces: (SUM(V$TEMPSTAT.PHYWRTS) / SUM(V$TEMPSTAT.PHYWRTS) + SUM(V$TEMPSTAT.PHYRDS)) * 100
Writes/sec (WRITES_RATE)	Number of write operations per second[#2]	AVG	double	No	All	• For dictionary managed permanent tablespaces, locally managed permanent tablespaces, or dictionary managed temporary tablespaces: SUM(V$FILESTAT.PHYWRTS) / seconds in interval • For locally managed temporary tablespaces: SUM(V$TEMPSTAT.PHYWRTS) / seconds in interval

Transaction (PD_PDTR)

Function

The Transaction (PD_PDTR) record stores performance data indicating the status (at a specific point in time) of transactions. PFM - Agent for Oracle creates one record for each transaction. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	145	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PD_PDTR_SID
- PD_PDTR_ADDRESS

Lifetime

From the start to the end of a transaction

Record size

- Fixed part: 678 bytes
- Variable part: 240 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Address (ADDRESS)	Address of the transaction status object	--	string(16)	No	All	V $TRANSACTION.ADDR
Cache Hit % (CACHE_HIT_PERCENTAGE)	Cache hit rate	--	double	No	All	(((V $TRANSACTION.LOG_IO + V $TRANSACTION.CR_GET) - V $TRANSACTION.PHY_IO) / (V $TRANSACTION.LOG_IO + V $TRANSACTION.CR_GET)) * 100
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Percentage indicating the extents used for consistency in	--	double	No	All	(V $TRANSACTION.CR_CHANGE / V

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	transaction read operations	--	double	No	All	$TRANSACTION.CR_GET) * 100
Consistent Changes (CONSISTENT_CHANGES)	Number of consistent changes	--	double	No	All	V $TRANSACTION.CR_CHANGE
Consistent Gets (CONSISTENT_GETS)	Number of consistent acquisitions	--	double	No	All	V $TRANSACTION.CR_GET
Locks (LOCKS)	Number of transaction locks	--	double	No	All	COUNT(V $LOCKED_OBJECT)
Logical I/O (LOGICAL_IO)	Logical I/O	--	double	No	All	V $TRANSACTION.LOG_IO
No Undo (NO_UNDO)	Identifier of a non-UNDO transaction. For a non-UNDO transaction, the value of this field is Yes. For an UNDO transaction, the value is No.	--	string(3)	No	All	V $TRANSACTION.NOUNDO
Physical I/O (PHYSICAL_IO)	Physical I/O	--	double	No	All	V $TRANSACTION.PHY_IO
Previous XID (PREVIOUS_XID)	Parent transaction ID	--	string(30)	No	All	V $TRANSACTION.PRV_XIDUSN + V $TRANSACTION.PRV_XIDSLT + V $TRANSACTION.PRV_XIDSQN
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDTR)	--	string(4)	No	All	Agent Collector
Recursive (RECURSIVE)	Identifier of a recursive transaction. For a recursive transaction, the value of this field is Yes. For a non-recursive transaction, the value is No.	--	string(3)	No	All	V $TRANSACTION.RECURSIVE
SID (SID)	Session ID	--	ulong	No	All	V$SESSION.SID where V $TRANSACTION.SES_ADDR = V $SESSION.ADDR

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Space (SPACE)	Identifier of a space transaction. For a space transaction, the value of this field is Yes. For a non-space transaction, the value is No.	--	string(3)	No	All	V$TRANSACTION.SPACE
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Transaction status	--	string(16)	No	All	V$TRANSACTION.STATUS
Tran Secs (TRANS_SECS)	Number of seconds since the start time	--	ulong	No	All	V$TRANSACTION.START_TIME
Tran Start (TRANS_START)	Start time	--	string(20)	No	All	V$TRANSACTION.START_TIME
Used Undo Blocks (USED_UNDO_BLOCKS)	Number of UNDO blocks used	--	double	No	All	V$TRANSACTION.USED_UBLK
Used Undo Records (USED_UNDO_RECORDS)	Number of UNDO records used	--	double	No	All	V$TRANSACTION.USED_UREC
User (USERNAME)	Oracle user name	--	string(30)	No	All	V$SESSION.USERNAME
XID (XID)	UNDO segment number, slot number, and sequence number (not applicable for an inactive transaction)	--	string(30)	No	All	V$TRANSACTION.XIDUSN + V$TRANSACTION.XIDSLOT + V$TRANSACTION.XIDSQN

Transaction Interval (PI_PITR)

Function

The Transaction Interval (PI_PITR) record stores performance data, taken at specific intervals, about transactions. PFM - Agent for Oracle creates one record for each transaction. This is a multi-instance record.

When Log is set to Yes, if you collect history over a long period of time, because of the short lifetime, it is not summarized in units of years or months. All instances are retained, resulting in a bloated store database. In addition, when the collected history is summarized, more memory is used than necessary. The memory shortage might cause monitoring to stop. If you want to collect history over a long period of time, use the Transaction (PD_PDTR) record for monitoring.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	145	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

- PI_PITR_SID
- PI_PITR_ADDRESS

Lifetime

From the start to the end of a transaction

Record size

- Fixed part: 678 bytes
- Variable part: 348 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Address (ADDRESS)	Address of the transaction status object[#1]	COPY	string(16)	No	All	V $TRANSACTION.ADDR
Cache Hit % (CACHE_HIT_PERCENTAGE)	Cache hit rate[#2]	AVG	double	No	All	((V $TRANSACTION.LOG_IO - V $TRANSACTION.PHY_IO) / V $TRANSACTION.LOG_IO) * 100

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Consistent Change % (CONSISTENT_CHANGE_PERCENTAGE)	Percentage indicating the extents used for consistency in transaction read operations[#2]	AVG	double	No	All	(V $TRANSACTION.CR_CHANGE / V $TRANSACTION.CR_GET) * 100
Consistent Changes (CONSISTENT_CHANGES)	Number of consistent changes[#2]	AVG	double	Yes	All	V $TRANSACTION.CR_CHANGE
Consistent Gets (CONSISTENT_GETS)	Number of consistent acquisitions[#2]	AVG	double	Yes	All	V $TRANSACTION.CR_GET
Locks (LOCKS)	Number of locks caused by the transaction[#2]	AVG	double	Yes	All	COUNT(V $LOCKED_OBJECT)
Logical I/O (LOGICAL_IO)	Logical I/O[#2]	AVG	double	Yes	All	V $TRANSACTION.LOG_IO
No Undo (NO_UNDO)	Identifier of a non-UNDO transaction (if this is a non-UNDO transaction, the value of this field is Yes; otherwise, the value is No)[#1]	COPY	string(3)	No	All	V $TRANSACTION.NOUNDO
Physical I/O (PHYSICAL_IO)	Physical I/O[#2]	AVG	double	Yes	All	V $TRANSACTION.PHY_IO
Previous XID (PREVIOUS_XID)	Parent transaction ID[#1]	COPY	string(30)	No	All	V $TRANSACTION.PRV_XIDUSN + V $TRANSACTION.PRV_XIDSLT + V $TRANSACTION.PRV_XIDSQN
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PITR)[#1]	COPY	string(4)	No	All	Agent Collector
Recursive (RECURSIVE)	Identifier of a recursive transaction (if this is a recursive transaction, the value of this field is Yes; otherwise, the value is No)[#1]	COPY	string(3)	No	All	V $TRANSACTION.RECURSIVE
SID (SID)	Session ID[#1]	COPY	ulong	No	All	V$SESSION.SID where V $TRANSACTION.SES_

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
SID (SID)	Session ID[#1]	COPY	ulong	No	All	ADDR = V $SESSION.ADDR
Space (SPACE)	Identifier of a space transaction (if this is a space transaction, the value of this field is Yes; otherwise, the value is No)[#1]	COPY	string(3)	No	All	V $TRANSACTION.SPACE
Start Time (START_TIME)	Collection start time for the performance data stored in the record[#1]	COPY	time_t	No	All	Agent Collector
Status (STATUS)	Transaction status[#1]	COPY	string(16)	No	All	V $TRANSACTION.STATUS
Tran Secs (TRANS_SECS)	Number of seconds since the start time[#1]	COPY	ulong	No	All	V $TRANSACTION.START_TIME
Tran Start (TRANS_START)	Start time[#1]	COPY	string(20)	No	All	V $TRANSACTION.START_TIME
Used Undo Blocks (USED_UNDO_BLOCKS)	Number of UNDO blocks used[#2]	AVG	double	Yes	All	V $TRANSACTION.USED_UBLK
Used Undo Records (USED_UNDO_RECORDS)	Number of UNDO records used[#2]	AVG	double	Yes	All	V $TRANSACTION.USED_UREC
User (USERNAME)	Oracle user name[#1]	COPY	string(30)	No	All	V $SESSION.USERNAME
XID (XID)	UNDO segment number, slot number, and sequence number (this field is not applicable to an inactive transaction)[#1]	COPY	string(30)	No	All	V $TRANSACTION.XIDUSN + V $TRANSACTION.XIDSLOT + V $TRANSACTION.XIDSQN

Transaction Lock (PD_PDTL)

Function

The Transaction Lock (PD_PDTL) record stores performance data indicating the status (at a specific point in time) of transaction locks. PFM - Agent for Oracle creates one record for each lock held by each transaction. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	140	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDTL_XID

Lifetime

From the locking to the unlocking of an object

Record size

- Fixed part: 678 bytes
- Variable part: 180 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Locked Mode (LOCKED_MODE)	Lock mode	--	string(20)	No	All	V$LOCKED_OBJECT.LOCKED_MODE
Object Name (OBJECT_NAME)	Name of the locked object	--	string(30)	No	All	DBA_OBJECTS.OBJECT_NAME where DBA_OBJECTS.OBJECT_ID = V$LOCKED_OBJECT.OBJECT_ID
Object Type (OBJECT_TYPE)	Object type	--	string(30)	No	All	DBA_OBJECTS.OBJECT_TYPE where DBA_OBJECTS.OBJECT_ID = V$LOCKED_OBJECT.OBJECT_ID

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Owner (OWNER)	Owner of the object	--	string(30)	No	All	DBA_OBJECTS.OWNER where DBA_OBJECTS.OBJECT_ID = V$LOCKED_OBJECT.OBJECT_ID
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDTL)	--	string(4)	No	All	Agent Collector
SID (SID)	Session ID	--	ulong	No	All	V$LOCKED_OBJECT.SESSION_ID
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
User (USER_NAME)	Oracle user name	--	string(30)	No	All	V$LOCKED_OBJECT.ORACLE_USERNAME
XID (XID)	Undo segment number, slot number, and sequence number. When a transaction is not running, 000 is set for this field.	--	string(30)	No	All	V$LOCKED_OBJECT.XIDUSN + V$LOCKED_OBJECT.XIDSLOT + V$LOCKED_OBJECT.XIDSQN

Version (PD_PDV)

Function

The Version (PD_PDV) record stores performance data, taken at a specific point in time, indicating the version number of a core component on the Oracle server. PFM - Agent for Oracle creates one record for each core component. This is a multi-instance record.

Default and changeable values

Item	Default value	Changeable
Collection Interval	600	Y
Collection Offset	150	Y
Log	No	Y
LOGIF	(Blank)	Y
Over 10 Sec Collection Time	No	N

ODBC key fields

PD_PDV_COMPONENT

Lifetime

From the creation to the deletion of an Oracle instance

Record size

- Fixed part: 678 bytes
- Variable part: 195 bytes

Fields

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Component (COMPONENT)	Component name	--	string(64)	No	All	PRODUCT_COMPONENT_VERSION.PRODUCT
Record Time (RECORD_TIME)	Collection termination time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Record Type (INPUT_RECORD_TYPE)	Record name (always PDV)	--	string(4)	No	All	Agent Collector
Start Time (START_TIME)	Collection start time for the performance data stored in the record	--	time_t	No	All	Agent Collector
Status (STATUS)	Component status	--	string(64)	No	All	PRODUCT_COMPONENT_VERSION.STATUS

PFM - View name (PFM - Manager name)	Description	Summary	Format	Delta	Supported version	Data source
Version (VERSION)	Component's version number	--	string(64)	No	All	PRODUCT_COMPONENT_VERSION.VERSION

7 Messages

This chapter describes the PFM - Agent for Oracle message format, lists the locations to which messages are output, shows which messages are output to `syslog` and the Windows event log, and describes the messages in detail.

7.1 Message format

This section shows the format of messages that are issued by PFM - Agent for Oracle, and the notations used in this manual to explain the messages.

7.1.1 Format of output messages

This section explains the format of the messages issued by PFM - Agent for Oracle. Each message consists of the message ID, followed by the message text. The message format is as follows:

```
KAVFnnnnn-Y message-text
```

The message ID indicates the following:

K
: System identifier.

AVF
: Indicates a PFM - Agent for Oracle message.

nnnnn
: Serial number of the message. PFM - Agent for Oracle message numbers are in the format 12*xxx*.

Y
: Type of message:

- E: Error
  Message issued when the system cancels processing.
- W: Warning
  Message issued when the system resumes processing after message output.
- I: Information
  Message in which the system provides the user with information.
- Q: Query
  Message in which the system prompts the user for a response.

The following is the correspondence of the message types to the syslog priority levels:

-E
- Level: LOG_ERR
- Description: Error message

-W
- Level: LOG_WARNING
- Description: Warning message

-I
- Level: LOG_INFO
- Description: Additional information message

-Q

(Not output)

The following is the correspondence of the message types to the Windows event log types:

-E

- Level: Error
- Description: Error message

-W

- Level: Warning
- Description: Warning message

-I

- Level: Information
- Description: Additional information message

-Q

(Not output)

7.1.2 Format of message explanations

This section describes the format used to explain messages in this manual. The portion of a message text that is shown in italics represents information that is variable depending on the situation. The manual lists the messages in the order of the message IDs. The following illustrates the format of a message explanation:

message-ID

message-text

Explanation of the message

(S)

Explains the processing performed by the system.

(O)

Explains the action the operator should take when the message is displayed.

> **Note**
>
> Reference note:
>
> When contacted by an operator, see *8. Error Handling Procedures*, collect the log information, and conduct initial checking. When you conduct initial checking to determine the cause of a problem, examine all applicable log information, such as the log information for the OS (Windows event log for Windows and `syslog` for UNIX) and the log information output by PFM - Agent for Oracle. This log information enables you to understand the details of the problem, take appropriate action, and prevent the problem from occurring in the future. You should also make a record of the operations that led to the problem and determine whether or not the problem is likely to recur.

7.2 Message output destinations

This section shows the output destinations of the messages issued by PFM - Agent for Oracle.

In Table 7-1, Y and N have the following meanings:

Legend:

Y: Message is output.

N: Message is not output.

Table 7–1: PFM - Agent for Oracle message output destinations

Message ID	Output destination								
	syslog	Windows event log	Common message log	Standard output	Standard error output	Agent log		JP1 system event#1	Agent event#2
						Normal log	Error log		
KAVF12001	Y	Y	Y	N	N	N	N	N	N
KAVF12002	Y	Y	Y	N	N	N	N	N	N
KAVF12003	Y	Y	Y	N	N	N	N	N	N
KAVF12004	Y	Y	Y	N	N	N	N	N	N
KAVF12009	N	N	Y	N	N	N	N	N	N
KAVF12010	Y	Y	Y	N	N	N	N	N	N
KAVF12011	Y	Y	Y	N	N	N	N	N	N
KAVF12014	Y	Y	Y	N	N	N	N	N	N
KAVF12015	Y	Y	Y	N	N	N	N	N	N
KAVF12017	Y	Y	Y	N	N	N	N	N	N
KAVF12018	Y	Y	Y	N	N	N	N	N	N
KAVF12019	N	N	Y	N	N	N	N	N	N
KAVF12020	Y	Y	Y	N	N	N	N	N	N
KAVF12021	Y	Y	Y	N	N	N	N	N	N
KAVF12022	Y	Y	Y	N	N	N	N	N	N
KAVF12023	Y	Y	Y	N	N	N	N	N	N
KAVF12301	N	N	Y	N	N	N	N	N	N
KAVF12302	N	N	Y	N	N	N	N	N	N
KAVF12303	Y	Y	Y	N	N	N	N	N	N
KAVF12304	Y	Y	Y	N	N	N	N	N	N
KAVF12305	Y	Y	Y	N	N	N	N	N	N
KAVF12306	N	N	Y	N	N	N	N	N	N
KAVF12401	N	N	Y	N	N	N	N	Y	N
KAVF12402	Y	Y	Y	N	N	N	N	N	N
KAVF12411	N	N	Y	N	N	N	N	N	N

Message ID	Output destination								
	syslog	Windows event log	Common message log	Standard output	Standard error output	Agent log		JP1 system event#1	Agent event#2
						Normal log	Error log		
KAVF12412	N	N	Y	N	N	N	N	N	N
KAVF12413	N	N	Y	N	N	N	N	N	N
KAVF12501	N	N	N	Y#3	N	N	N	N	N
KAVF12502	N	N	N	Y#3	N	N	N	N	N
KAVF12504	N	N	N	Y#4	N	N	N	N	N
KAVF12505	N	N	N	Y#5	N	N	N	N	N
KAVF12506	N	N	N	Y#5	N	N	N	N	N
KAVF12507	N	N	N	Y#5	N	N	N	N	N
KAVF12508	N	N	N	Y#5	N	N	N	N	N
KAVF12509	N	N	N	Y#5	N	N	N	N	N
KAVF12510	N	N	N	Y#5	N	N	N	N	N
KAVF12511	N	N	N	Y#5	N	N	N	N	N
KAVF12512	N	N	N	Y#5	N	N	N	N	N
KAVF12513	N	N	N	Y#5	N	N	N	N	N
KAVF12514	N	N	N	Y#5	N	N	N	N	N
KAVF12515	N	N	N	Y#5	N	N	N	N	N
KAVF12516	N	N	N	Y#5	N	N	N	N	N
KAVF12517	N	N	N	Y#5	N	N	N	N	N
KAVF12518	N	N	N	Y#5	N	N	N	N	N
KAVF12519	N	N	Y	N	N	N	N	N	N
KAVF12600	N	N	N	N	N	N	Y	N	N
KAVF12601	N	N	N	N	N	Y	N	N	N
KAVF12602	N	N	N	N	N	Y	N	N	N
KAVF12603	N	N	N	N	N	Y	N	N	N
KAVF12604	N	N	N	N	N	Y	N	N	N
KAVF12605	N	N	N	N	N	Y	N	N	N
KAVF12606	N	N	N	N	N	Y	N	N	N
KAVF12607	N	N	N	N	N	Y	N	N	N
KAVF12608	N	N	N	N	N	Y	N	N	N
KAVF12609	N	N	N	N	N	Y	N	N	N
KAVF12610	N	N	N	N	N	Y	N	N	N

Message ID	Output destination								
	syslog	Windows event log	Common message log	Standard output	Standard error output	Agent log		JP1 system event#1	Agent event#2
						Normal log	Error log		
KAVF12611	N	N	N	N	N	Y	N	N	N
KAVF12612	N	N	N	N	N	Y	N	N	N
KAVF12613	N	N	N	N	N	N	Y	N	N
KAVF12614	N	N	N	N	N	N	Y	N	N
KAVF12615	N	N	N	N	N	N	Y	N	N
KAVF12616	N	N	N	N	N	N	Y	N	N
KAVF12617	N	N	N	N	N	N	Y	N	N
KAVF12618	N	N	N	N	N	N	Y	N	N
KAVF12619	N	N	N	N	N	N	Y	N	N
KAVF12620	N	N	N	N	N	N	Y	N	N
KAVF12621	N	N	N	N	N	N	Y	N	N
KAVF12622	N	N	N	N	N	N	Y	N	N
KAVF12623	N	N	N	N	N	N	Y	N	N
KAVF12624	N	N	N	N	N	N	Y	N	N
KAVF12625	N	N	N	N	N	N	Y	N	N
KAVF12626	N	N	N	N	N	N	Y	N	N
KAVF12627	N	N	N	N	N	N	Y	N	N
KAVF12628	N	N	N	N	N	N	Y	N	N
KAVF12629	N	N	Y	N	N	N	N	N	N
KAVF12630	N	N	Y	N	N	N	N	N	N
KAVF12631	N	N	N	N	N	N	Y	N	N
KAVF12632	N	N	N	N	N	N	Y	N	N
KAVF12633	N	N	Y	N	N	N	Y	N	N
KAVF12634	N	N	N	N	N	N	Y	N	N
KAVF12635	N	N	N	N	N	Y	N	N	N
KAVF12636	N	N	N	N	N	Y	N	N	N
KAVF12638	N	N	N	N	N	N	Y	N	N

#1

JP1 system events are the events to notify JP1/IM of status change of the agent. For details about JP1 system events, see the chapter on how to coordinate Performance Management programs and JP1/IM programs and conduct operation monitoring in the *JP1/Performance Management User's Guide*.

Table 7-2 shows the prerequisite programs to issue JP1 system events.

Table 7–2: Prerequisite programs to issue JP1 system events

Hosts	Prerequisite program	Version
PFM - Manager host	PFM - Manager	09-00 or later
PFM - Web Console host	PFM - Web Console	08-00 or later
PFM - Agent host	PFM - Agent for Oracle	09-00 or later
	PFM - Manager or PFM - Base	09-00 or later
	JP1/Base	08-50 or later

#2

Agent events are the events to notify PFM - Manager of status change of the agent. For details about agent events, see the chapter on displaying the events in the *JP1/Performance Management User's Guide*.

Table 7-3 shows the prerequisite programs to issue agent events.

Table 7–3: Prerequisite programs to issue agent events

Hosts	Prerequisite programs	Version
PFM - Manager host	PFM - Manager	09-00 or later
PFM - Web Console host	PFM - Web Console	08-00 or later
PFM - Agent host	PFM - Manager or PFM - Base	09-00 or later

#3

This message is output during the execution of the `sp_inst.sql` script.

#4

This message is output during the execution of the `sp_inst.sql` or the `mk_user.sql` script.

#5

This message is output during the execution of the `mk_user.sql` script.

7.3 List of messages output to the Windows event log and syslog

This section lists the messages that PFM - Agent for Oracle outputs to `syslog` and to the Windows event log.

When the OS is Windows, the Windows event log is displayed in the application log of the Event Viewer window.

> **Note**
>
> Reference Note:
>
> To open the Event Viewer window, from the Windows **Start** menu, choose **Administrative Tools** and then **Event Viewer**.

For an event issued by PFM - Agent for Oracle, the identifier `PFM-Oracle` is displayed in the **Source** column of the Event Viewer window.

When the OS is UNIX, the syslog information is output to the `syslog` file. For the installation location of the `syslog` file, see the syslog daemon configuration file (default path is `/etc/syslogd.conf`).

The following table lists the messages that PFM - Agent for Oracle outputs to syslog and to the Windows event log.

Table 7–4: Messages output to syslog and to the Windows event log

Message ID	syslog		Windows event log	
	Facility	Level	Event ID	Type
KAVF12001-I	LOG_DAEMON	LOG_INFO	12001	Information
KAVF12002-E	LOG_DAEMON	LOG_ERR	12002	Error
KAVF12003-I	LOG_DAEMON	LOG_INFO	12003	Information
KAVF12004-E	LOG_DAEMON	LOG_ERR	12004	Error
KAVF12010-E	LOG_DAEMON	LOG_ERR	12010	Error
KAVF12011-E	LOG_DAEMON	LOG_ERR	12011	Error
KAVF12014-E	LOG_DAEMON	LOG_ERR	12014	Error
KAVF12015-E	LOG_DAEMON	LOG_ERR	12015	Error
KAVF12017-E	LOG_DAEMON	LOG_ERR	12017	Error
KAVF12018-E	LOG_DAEMON	LOG_ERR	12018	Error
KAVF12020-E	LOG_DAEMON	LOG_ERR	12020	Error
KAVF12021-E	LOG_DAEMON	LOG_ERR	12021	Error
KAVF12022-E	LOG_DAEMON	LOG_ERR	12022	Error
KAVF12023-E	LOG_DAEMON	LOG_ERR	12023	Error
KAVF12303-E	LOG_DAEMON	LOG_ERR	12303	Error
KAVF12304-E	LOG_DAEMON	LOG_ERR	12304	Error
KAVF12305-E	LOG_DAEMON	LOG_ERR	12305	Error
KAVF12402-E	LOG_DAEMON	LOG_ERR	12402	Error

7.4 Messages

This section explains the messages issued by PFM - Agent for Oracle and the corresponding actions to be taken.

KAVF12001-I

Agent Collector has stopped. (host= *host-name*, service= *host-name*<*Oracle*>)

The Agent Collector service stopped normally.

(S)

Stops Agent Collector service processing.

KAVF12002-E

Agent Collector failed to start.

An attempt to start the Agent Collector service failed.

(S)

Stops Agent Collector service processing.

(O)

Check the immediately preceding message that was issued to the common message log and take appropriate action.

KAVF12003-I

Agent Collector has started. (host= *host-name*, service= *host-name*<*Oracle*>)

Agent Collector service startup was completed.

(S)

Starts collecting performance data for the Agent Collector service.

KAVF12004-E

Agent Collector stopped abnormally.

The Agent Collector service stopped abnormally.

(S)

Stops Agent Collector service processing.

(O)

Check the immediately preceding message that was output to the common message log and take appropriate action.

KAVF12009-W

The object to be monitored is not available. (host=*host-name*, service= *host-name* <*Oracle*>)

PFM - Agent for Oracle was unable to establish connection with the Oracle Database to be monitored.

(S)

Continues Agent Collector service processing.

(O)

Check to see if Oracle Database is active. Also check for errors in the following information specified during instance environment setup:

- `oracle_home`
- `oracle_sid`
- `oracle_user`
- `oracle_passwd`

KAVF12010-E

An attempt to read the initialization file failed.

An attempt to read the service startup initialization file failed during startup processing for the Agent Collector service.

(S)

Stops Agent Collector service processing.

(O)

Check to see if the service startup initialization file (`jpcagt.ini`) is stored in either of the following directories:

- For Windows
  installation-folder`\agto\agent\`*instance-name*
- For UNIX
  `/opt/jp1pc/agto/agent/`*instance-name*

If you do not find the service startup initialization file, copy the contents of the `jpcagt.ini.model` file to the `jpcagt.ini` file. If the cause of the error is unknown, collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12011-E

Initialization of interprocess communication failed.

Preparations could not be made to enable communication between the Agent Collector service and the performance data collection program.

(S)

Stops Agent Collector service processing.

(O)

Check the immediately preceding message that was output to the common message log and take appropriate action.

If there is no such message, collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12014-E

The [Agent | Collector] semaphore could not be obtained.

Semaphore acquisition failed.

(S)

Stops Agent Collector service processing.

(O)

Check the kernel parameters for semaphores and specify valid information. For details about semaphore values, see *B. Kernel Parameters*.

KAVF12015-E

The Collector process could not start.

Startup of the performance data collection program failed.

(S)

Stops Agent Collector service processing.

(O)

Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12017-E

The environment variable [*environment-variable*] could not be set.

PFM - Agent for Oracle was unable to set the indicated environment variable.

(S)

Stops Agent Collector service processing.

(O)

Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12018-E

An attempt to start the collector failed. (GetProgram=*program-name*)

Startup of the performance data collection program failed because of invalid database monitoring settings.

(S)

Stops Agent Collector service processing.

(O)

Check to see if Oracle Database is active. Also check for errors in the following information specified during instance environment setup:

- `oracle_home`
- `oracle_sid`
- `oracle_user`
- `oracle_passwd`

KAVF12019-W

External command could not be called while collecting *record-name*. (Command= *command-line*)

The indicated external command call failed.

(S)

Continues Agent Collector service processing.

(O)

Check that the OS and Oracle are running normally.

Make sure that the command that is output to the command line can be executed.

KAVF12020-E

While executing *function-name* function *called-function-name* failed.

An error occurred during execution of the indicated function.

(S)

Stops Agent Collector service processing.

(O)

Perform the following:

- If Load Library() failed:

  Make sure that the value specified for `oracle_home` when the instance environment was set up is correct.
- For other cases:

  Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12021-E

Error occurred by function *function-name*. (en=*error-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

An error occurred during execution of the indicated function.

(S)

Stops Agent Collector service processing.

(O)

Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12022-E

Processing was interrupted by signal. (signal=*signal-number*)

Processing was interrupted by the indicated signal.

(S)

Stops Agent Collector service processing.

(O)

Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12023-E

Agent Collector is going to stop because the error has occurred.

The Agent Collector service will be stopped because of an error.

(S)

Stops Agent Collector service processing.

(O)

Check the immediately preceding message that was output to the common message log and take appropriate action.

KAVF12301-W

At Agent Collector startup, the system could not connect to Oracle. (rc=*return-code*)

An attempt to establish connection with the Oracle Database failed.

(S)

Continues Agent Collector service processing.

(O)

Take the appropriate action indicated in the message that follows this message.

KAVF12302-W

An OCI call failed because of function *function-name*. (rc=*return-code*, errcode= *error-number*)

An OCI call resulted in an error during connection establishment with the Oracle Database.

(S)

Continues Agent Collector service processing.

(O)

Take one of the actions listed below on the basis of the Oracle error number:

- `errcode` value is `942`

  Check if you are attempting to collect a record that is not supported by the current configuration. Alternatively, make sure that the required system privileges have been granted to the Oracle account specified for `oracle_user`. When granting permissions as a role, grant the system privileges explicitly using `GRANT` *privileges*.

- `errcode` value is `1017` or `1031`

  An attempt to log in to the Oracle Database to be monitored failed because the user name or password was invalid. Check whether the user name and the password you specified during setup of the instance environment are correct. For details about how to check the instance environment, see *2.4.2 Updating an instance environment* (for Windows), or *3.4.2 Updating an instance environment* (for UNIX).

- `errcode` value is `6550`

  You need to execute the `sp_inst.sql` script for the Oracle Database subject to data collection. For details about how to execute the script, see *2.1.4(3) Set up an instance environment* (for Windows), or *3.1.4(4) Set up an instance environment* (for UNIX).

  Note that you must execute the `sp_inst.sql` script after connecting to the Oracle Database with the account that was specified for `oracle_user` during setup of the instance environment. Check the account that was used to execute the `sp_inst.sql` script.

- `errcode` value is `1013`

  Performance data collection may be canceled due to the cancellation facility. In this case, check whether the `KAVF12636-I` message has been output to the common log of the agent log. To prevent collection data from being canceled, change the timeout value.

- `errcode` value is `12546`

  If another user (OTHER user) lacks execution permissions for the files in the Oracle home directory (same value as the `ORACLE_HOME` environment variable), a connection to the Oracle Database may be unable to be established. You can solve this problem by specifying Y for sqlnet when setting up an instance environment. For details about checking the instance environment, see *2.4.2 Updating an instance environment* (for Windows), or *3.4.2 Updating an instance environment* (for UNIX).

- When any other value is displayed for `errcode`

  See the Oracle manual and take appropriate action.

KAVF12303-E

An attempt to allocate memory failed. (RecordName=*record-name*, Size=*size*)

Memory allocation for the indicated record failed.

(S)

Stops Agent Collector service processing.

(O)

Increase the amount of memory space available.

KAVF12304-E

Semaphore is insufficient.

Semaphore is insufficient.

(S)

Stops Agent Collector service processing.

(O)

Increase the semaphore value in the kernel parameter. For details about semaphore values, see *B. Kernel Parameters*.

KAVF12305-E

exception-name exception raised. (Detail: *detailed-information*)

The indicated exception occurred.

(S)

Stops Agent Collector service processing.

(O)

Collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12306-W

An attempt to allocate memory failed. (RecordName=*record-name*, Size=*size*)

An attempt to allocate memory failed due to insufficient memory.

(S)

Continues Agent Collector service processing.

(O)

Increase the amount of memory space available.

KAVF12401-W

An attempt to collect a record failed. (RecordName=*record-name*)

Collection of the indicated record failed.

(S)

Continues Agent Collector service processing.

(O)

If this message is issued repeatedly, check the system environment settings for the program being monitored. If you cannot identify the cause of the error, collect maintenance information and contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12402-E

An unexpected abnormality occurred during the collection of records. (RecordName= *record-name*)

Record collecting has been canceled due to an unexpected error.

(S)

Stops Agent Collector service processing.

(O)

Collect the maintenance data and then contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12411-W

The listener's information cannot be collected. (ListenerName= *listener-name*)

The system was unable to collect the listener information indicated by *listener-name.*

(S)

Continues Agent Collector service processing.

(O)

An open or write error may have occurred on a work file. Check to see if there is enough unused capacity on the disk.

If there is enough unused capacity on the disk, collect the maintenance information and then contact the system administrator. For details about how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12412-W

The listener does not exist. (ListenerName=*listener-name*)

The system was unable to collect the information, because the listener indicated by *listener-name* does not exist.

(S)

Continues Agent Collector service processing.

(O)

Check *listener_name* specified during the instance environment setup for any error.

KAVF12413-W

The listener is not running. (ListenerName=*listener-name*)

The system cannot collect the information because the listener indicated by *listene-rname* is not active.

(S)

Continues Agent Collector service processing.

(O)

Start the listener.

KAVF12501-E

This Oracle Version is not supported.

The Oracle Database version is not supported.

(S)

Cancels the script execution.

(O)

Make sure that the version of the Oracle Database to be monitored is supported. An unsupported version of the Oracle Database cannot be monitored.

KAVF12502-E

The permission for monitoring the Oracle Database is insufficient.

The privilege is insufficient for monitoring the Oracle Database.

(S)

Cancels the script execution.

(O)

If the user executing the script is not the one specified for `oracle_user`, have the user specified for `oracle_user` reexecute the script.

If the user executing the script is the one specified for `oracle_user`, set the privileges for referencing and executing SYS schema objects for this user and then reexecute the script.

KAVF12504-E

An unexpected error occurred.

An unexpected error has occurred.

(S)

Cancels the script execution.

(O)

Make sure that the privileges have been granted properly. If there is no problem with the privileges, collect the executed script and maintenance data and then contact the system administrator. For details on how to collect maintenance information, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

KAVF12505-I

The script ended normally.

The script terminated normally.

(S)

Terminates the script execution.

KAVF12506-E

Script processing will now stop because an error occurred.

Script processing will now stop because an error occurred.

(S)

Cancels the script execution.

(O)

Take the appropriate action indicated in the message that follows this message.

KAVF12507-E

The number of entered characters exceeded the maximum.

The number of characters in the entered string exceeded the maximum length. You can enter a character string of 30 or fewer bytes.

(S)

Cancels the script execution.

(O)

Check the entered value, and then reexecute the script.

KAVF12508-E

The entered value is invalid.

The entered value is invalid.

(S)

Cancels the script execution.

(O)

Check the entered value, and then reexecute the script. For details about character strings that can be specified for entered values, see *2.1.4(3) Table 2-4 Information required to create an Oracle account* (for Windows), or *3.1.4(3) Table 3-5 Information required to create an Oracle account* (for UNIX).

KAVF12509-E

A user with the same name already exists.

The same account name already exists in the database.

(S)

Cancels the script execution.

(O)

Check the account name, and then reexecute the script.

KAVF12510-E

The specified user name is invalid.

The specified account name is invalid. The specified account name contains a symbol that cannot be specified.

(S)

Cancels the script execution.

(O)

Check the account name, and then reexecute the script.

KAVF12511-E

The password is not specified or is invalid.

The password is not specified or is invalid.

(S)

Cancels the script execution.

(O)

If the password is not specified, specify it and then reexecute.

If the password is specified, the password contains a symbol that cannot be specified. Check the password, and then reexecute the script.

KAVF12512-E

The password did not meet the necessary complexity specifications.

The password does not meet the necessary complexity specifications.

(S)

Cancels the script execution.

(O)

The specified password does not meet the level of complexity required by Oracle. After asking the database administrator about the required complexity for passwords, check the password and then reexecute the script.

KAVF12513-E

The tablespace name is not specified.

The default tablespace name or default temporary tablespace name is not specified.

(S)

Cancels the script execution.

(O)

Specify the default tablespace name or default temporary tablespace name, and then reexecute the script.

KAVF12514-E

The specified tablespace name is invalid.

The specified default tablespace name or default temporary tablespace name is invalid. The specified tablespace name contains a symbol that cannot be specified.

(S)

Cancels the script execution.

(O)

Check the default tablespace name or default temporary tablespace name, and then reexecute the script.

KAVF12515-E

The specified tablespace does not exist.

The specified default tablespace or default temporary tablespace does not exist in the database.

(S)

Cancels the script execution.

(O)

Check the default tablespace or default temporary tablespace, and then reexecute the script.

KAVF12516-E

The specified tablespace cannot be used as the default tablespace.

The specified tablespace name cannot be used as the default tablespace, such as a temporary tablespace.

(S)

Cancels the script execution.

(O)

Check the specification of the default tablespace name, and then reexecute the script.

KAVF12517-E

The specified tablespace cannot be used as the default temporary tablespace.

The specified temporary tablespace name cannot be used as the default temporary tablespace, such as an UNDO tablespace.

(S)

Cancels the script execution.

(O)

Check the specification of the default temporary tablespace name, and then reexecute the script.

KAVF12518-E

The permission for executing the script is insufficient.

The permission is insufficient for executing the script.

(S)

Cancels the script execution.

(O)

Check whether the CREATE USER system privilege and GRANT ANY PRIVILEGE system privilege have been assigned to the Oracle account used for executing the script.

If the account used for executing the script does not have the necessary privileges, assign the necessary privileges or use another account that has sufficient privileges, and then reexecute the script.

KAVF12519-E

Failed to output to Agent log. *OS-function* failed. Error code = *error-code*.

An attempt to output an Agent log has failed. The indicated OS function failed due to the error indicated by *error-code*.

(S)

Continues Agent Collector processing. Subsequent Agent logs will not be collected until the error has been corrected.

(O)

Check the output destination path of the Agent log and access permissions.

KAVF12600-E

An error occurred in OS function *function-name*. (rc=*error-code*)

An error occurred during execution of the function indicated by *function-name*. *error-code* is a value returned by the system call.

(S)

Continues Agent Collector processing.

(O)

Make sure that the OS has sufficient resources and that no errors have occurred in the OS. If this message is issued repeatedly, check the system environment settings for the program being monitored. If you cannot identify the cause of the error, collect maintenance information and contact the system administrator. For details about how to collect maintenance information, see the chapter in the *JP1/Performance Management User's Guide* that describes troubleshooting.

KAVF12601-I

Agent : Started : Collecting records.

Agent has started collecting records.

(S)

Continues Agent Collector processing.

KAVF12602-I

Agent : Started : Sending a request to the collector. (*process-ID*)

Agent has started sending a processing request to the collector with the ID indicated by *process-ID*.

(S)

Continues Agent Collector processing.

KAVF12603-I

Collector : Started : Receive a request.

The collector process has started receiving a processing request from Agent.

(S)

Continues Agent Collector processing.

KAVF12604-I

Agent : Ended : Sending a request to the collector. (*process-ID*)

Agent has finished sending a processing request to the collector with the ID indicated by *process-ID*. Agent will now wait for the results.

(S)

Continues Agent Collector processing.

KAVF12605-I

Collector : Ended : Receive a request. (*record-name*)

The collector process has finished receiving a processing request for the *record-name* from Agent.

(S)

Continues Agent Collector processing.

KAVF12606-I

Agent : Waiting for the results.

Agent is waiting for the results from the collector process.

(S)

Continues Agent Collector processing.

KAVF12607-I

Collector : Started : Sending the results.

The collector process has finished accessing the Oracle database, and has started sending the results to Agent.

(S)

Continues Agent Collector processing.

KAVF12608-I

Agent : Received the results.

Agent has finished receiving the results from the collector process.

(S)

Continues Agent Collector processing.

KAVF12609-I

Collector : Ended : Sending the results.

The collector process has finished sending the results to Agent.

(S)

Continues Agent Collector processing.

KAVF12610-I

Agent : Started : Storing the results to the Store DB. (*record-name*) count= *number-of-records*

Agent has started storing the records indicated by *record-name* in the Store database. The number of records to be stored is indicated by *number-of-records*.

(S)

Continues Agent Collector processing.

KAVF12611-I

Agent : Ended : Storing the results to the Store DB. (*record-name*)

Agent has finished storing the records indicated by *record-name* in the Store database.

(S)

Continues Agent Collector processing.

KAVF12612-I

Agent : Ended : Collecting records.

Agent has finished collecting records.

(S)

Continues Agent Collector processing.

KAVF12613-W

The object to be monitored is not available. (host=*host-name*, service=*host-name*<*Oracle*>)

PFM - Agent for Oracle cannot connect to the Oracle server to be monitored.

(S)

Continues Agent Collector processing.

(O)

Check whether the Oracle Database has started. Also make sure that the following items were correctly specified during setup of the instance environment:

- `oracle_sid`
- `oracle_home`
- `oracle_user`
- `oracle_passwd`

KAVF12614-E

The environment variable [*environment-variable*] could not be set.

PFM - Agent for Oracle was unable to set the indicated environment variable.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12615-W

External command could not be called while collecting *record-name*. (Command= *command-line*)

The indicated external command call failed.

(S)

Continues Agent Collector processing.

(O)

Check that the OS and Oracle are running normally.

Make sure that the command that is output to the command line can be executed.

KAVF12616-E

While executing *function-name* function *called-function-name* failed.

An error occurred during execution of the function indicated by *function-name*.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12617-E

Error occurred by function *function-name*. (en=*error-code*, arg1=*argument-1*, arg2=*argument-2*, arg3=*argument-3*)

An error occurred during execution of the function indicated by *function-name*.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12618-E

Processing was interrupted by signal. (signal=*signal-number*)

Processing was interrupted by the indicated signal.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12619-W

At Agent Collector startup, the system could not connect to Oracle. (rc=*return-code*)

Connection establishment with the Oracle server failed.

(S)

Continues Agent Collector processing.

(O)

Take the appropriate action indicated in the message that follows this message.

KAVF12620-W

An OCI call failed because of function *function-name*. (rc=*return-code*, errcode= *error-number*)

OCI call resulted in an error during connection establishment with the Oracle server.

(S)

Continues Agent Collector processing.

(O)

Take one of the actions listed below on the basis of the Oracle error number:

- `errcode` value is `942`

  Make sure that collection is not being performed for records that cannot be collected under the current configuration. Alternatively, make sure that the required system privileges have been granted to the Oracle account specified for `oracle_user`. When granting privileges as a role, grant the system privileges explicitly using `GRANT` *privileges*.

- `errcode` value is `6550`

The sp_inst.sql needs to be executed on the collection target Oracle Database. For details about how to execute this script, *2.1.4(3) Set up an instance environment* (for Windows), or *3.1.4(4) Set up an instance environment* (for UNIX).

Note that you must execute the sp_inst.sql script after connecting to the Oracle Database with the account that was specified for oracle_user during setup of the instance environment. Check the account that was used to execute the sp_inst.sql script.

- errcode value is 1013

  Performance data collection may be canceled due to the cancellation facility. In this case, check whether the KAVF12636-I message has been output to the common log of the agent log. To prevent collection data from being canceled, change the timeout value.

- errcode value is 12546

  If another user (OTHER user) lacks execution permissions for the files in the Oracle home directory (same value as the ORACLE_HOME environment variable), a connection to the Oracle Database may be unable to be established. You can solve this problem by specifying Y for sqlnet when setting up an instance environment. For details about checking the instance environment, see *2.4.2 Updating an instance environment* (for Windows), or *3.4.2 Updating an instance environment* (for UNIX).

- errcode value is another value

  For details about what to do, see the Oracle documentation.

KAVF12621-E

An attempt to allocate memory failed. (RecordName=*record-name*, Size=*size*)

Memory allocation for the indicated record failed.

(S)

Stops Agent Collector processing.

(O)

Increase the amount of memory space available.

KAVF12622-E

exception-name exception raised. (Detail: *detailed-information*)

The indicated exception occurred.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12623-W

An attempt to allocate memory failed. (RecordName=*record-name*, Size=*size*)

Memory allocation for the indicated record failed.

(S)

Continues Agent Collector processing.

(O)

Increase the amount of memory space available.

KAVF12624-W

An attempt to collect a record failed. (RecordName=*record-name*)

An attempt to collect the record indicated by *record-name* failed.

(S)

Continues Agent Collector processing.

(O)

If this message appears repeatedly, check the settings of the system environment being monitored. If you cannot identify the cause of the error, collect maintenance information and contact the system administrator.

KAVF12625-E

An unexpected abnormality occurred during the collection of records. (RecordName= *record-name*)

Record collecting has been canceled due to an unexpected error.

(S)

Stops Agent Collector processing.

(O)

Collect maintenance information and contact the system administrator.

KAVF12626-W

The listener's information cannot be collected. (ListenerName=*listener-name*)

Information about the listener indicated by *listener-name* cannot be collected.

(S)

Continues Agent Collector processing.

(O)

An open or write error may have occurred on a work file. Check whether there is sufficient unused capacity on the disk.

If there is enough unused capacity on the disk, collect the maintenance data and then contact the system administrator.

KAVF12627-W

The listener does not exist. (ListenerName=*listener-name*)

Information cannot be collected because the listener indicated by *listener-name* does not exist.

(S)

Continues Agent Collector processing.

(O)

Make sure that *listener_name* was specified correctly during setup of the instance environment.

KAVF12628-W

The listener is not running. (ListenerName=*listener-name*)

The system was unable to collect the information because the listener indicated by *listener-name* was not active.

(S)

Continues Agent Collector processing.

(O)

Start the listener.

KAVF12629-I

Agent log. path = *Agent-log-output-folder-path*

The system has recorded that the path for Agent log output is the folder indicated by *Agent-log-output-folder-path*.

(S)

Continues Agent Collector processing.

KAVF12630-W

Agent property *property-name* is outside injustice or the range. (Range: *range-value*)

An invalid value or a value outside the valid range is specified for a property of the Agent Collector service.

(S)

Invalidates the specified value and continues Agent Collector processing. The value of the indicated property remains unchanged. If a value from 1 to 9 is specified for the `TIMEOUT` property, it is replaced with `10`.

(O)

Check the specified value. If there is a problem with the specified value, specify an appropriate value.

KAVF12631-E

An error occurred: *error-details*

An error occurred during Agent Collector service processing.

(S)

Stops Agent Collector processing.

(O)

Make sure that the OS has sufficient resources and that no errors have occurred in the OS. If this message is issued repeatedly, check the system environment settings for the program being monitored. If you cannot identify the cause of the error, collect maintenance information and contact the system administrator. For details about how to collect maintenance information, see the chapter in the *JP1/Performance Management User's Guide* that describes troubleshooting.

KAVF12632-W

A warning-level error occurred: *warning-error-details*

A warning error occurred during Agent Collector service processing.

(S)

Continues Agent Collector processing.

(O)

Make sure that:

- The OS has sufficient resources.
- No errors have occurred in the OS.
- The system environment to be monitored is set up correctly.

KAVF12633-W

Can't get data from Oracle.

Information about Oracle could not be collected.

(S)

The system cannot connect to Oracle.

(O)

Check whether Oracle is running normally.

Also check the information that was specified during setup of the instance.

KAVF12634-W

Getting record error(*record-name*). This record is only in drilldown reports.

The record indicated by *record-name* could not be obtained. This record can be obtained only in drilldown reports.

(S)

Continues Agent Collector processing.

(O)

Set a drilldown for records that have the ODBC key field specific to this record. For details about drilldown reports, see the chapter on creating reports used for operation analysis in the *JP1/Performance Management User's Guide*.

KAVF12635-I

Agent : Elapsed time *required-time* sec. (*record-name*)

The number of seconds required for collecting the record indicated in *record-name* will be recorded.

(S)

Continues Agent Collector processing.

KAVF12636-I

The cancellation of the record collection (*record-name*) by the time-out was accepted.

A collection cancellation was accepted due to a timeout for the record indicated by *record-name*.

(S)

Continues Agent Collector processing.

KAVF12638-W

The mismatch of ORACLE_SID was detected. (*SID of the monitoring target Oracle Database*)

The oracle_sid you specified when you set up the instance environment is different from the SID of the monitoring target Oracle Database.

(S)

Continues Agent Collector processing.

(O)

Check whether the `oracle_sid` you specified when you set up the instance environment is different from the SID of the monitoring target Oracle Database.

Alternatively, check whether the `net_service_name` you specified when you set up the instance environment is correct.

8 Error Handling Procedures

This chapter describes how to troubleshoot problems in Performance Management operation, focusing mainly on problems that occur in PFM - Agent. For details about how to deal with issues affecting the Performance Management system as a whole, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

8.1 When an error occurs

If an error occurs in Performance Management, follow the steps below.

Check events

Check the following:

- What events occur when the problem is encountered
- The content of messages (if output)
- Log information such as the common message log

For details about the causes of messages and the action to take, see *7. Messages*. For details about the logs that are output by Performance Management, see *8.3 Log information*.

Collect data

You need to collect data to determine the cause of the problem. See *8.4 Required troubleshooting information* and *8.5 Collecting troubleshooting information* to collect the necessary information.

Investigate the problem

Identify the cause of the problem based on the collected data. Isolate the problem or all the areas affected by the problem.

8.2 Troubleshooting procedures

This section describes how to troubleshoot Performance Management. If a problem occurs while you are using Performance Management, first check for any of the events described in this section.

The following table shows the main types of problems that may occur in Performance Management.

Table 8–1: Problems that occur in Performance Management

Category	Description	Reference
Problems relating to the start and setup of services	• A Performance Management program service does not start. • The Oracle Database does not stop.	*8.2.1*
	• There is a delay from the time the start request is issued until the service starts. • Communication fails when another program starts a service immediately after a Performance Management program service is stopped. • The following message is output and the Master Store or Agent Store service stops: `The disk capacity is insufficient.` • The PFM - Agent Collector service does not start.	See the chapter that explains troubleshooting in the *JP1/Performance Management User's Guide*.
Problems relating to command execution	• The name of an inactive service is output when you execute the `jpctool service list` command. • The data output by the `jpctool db dump` command is not the Store data you specified.	
Problems relating to report definitions	• During a certain period, the collected data is not shown in the historical report.	
Problems relating to alarm definitions	• A program defined in an action does not operate correctly. • Alarm events are not displayed. • Although the alarm threshold has been exceeded, the color of the alarm icon shown in the Alarm Status window of the Agents tree remains green.	
Problems relating to collection and management of performance data	• The PFM - Agent Store database remains large despite setting a short data retention period. • The following message is output to the common message log: `Illegal data was detected in the Store database.`	
	• Performance data is not collected after PFM - Agent startup.	*8.2.2*

8.2.1 Problems relating to the start and setup of services

This subsection describes how to correct errors related to the start and setup of services. For details about how to correct other types of errors, see the *JP1/Performance Management User's Guide*.

(1) Performance Management program service does not start

Potential causes and actions to take are as follows:

- Oracle Database is not installed

For the Agent Collector service to start, Oracle Database must be installed. Install Oracle Database on the PFM - Agent host.

- The Oracle Database has not started

  The Agent Collector service may not start, depending on whether the Oracle Database is running. Make sure that the Agent Collector service is started after the Oracle Database.

- An Oracle service is running under an account other than the local system account

  If both of the following conditions are satisfied, an error occurs in Oracle and you might not be able to start the PFM - Agent Collector service.

  - N was specified in the `sqlnet` item of the instance information for PFM - Agent for Oracle.
  - Any Oracle service is running under an account other than the local system account.

- The instance environment settings are specified incorrectly

  The Agent Collector service cannot start if any of the following is specified incorrectly during instance environment setup:

  - `oracle_sid`
  - `oracle_home`
  - `oracle_user`
  - `oracle_passwd`

  Execute the `jpcconf agent setup` command to specify the correct settings. For details about the `jpcconf agent setup` command, see the chapter on the commands in the manual *JP1/Performance Management Reference*.

(2) The Oracle Database does not stop

When you attempt to stop the monitored Oracle Database before stopping PFM - Agent for Oracle, a normal shutdown may not stop the Oracle Database. In this case, use an immediate shutdown to stop the Oracle Database.

8.2.2 Problems relating to the collection and management of performance data

This subsection describes how to correct errors related to the collection and management of performance data in Performance Management. For details about how to correct other types of errors, see the *JP1/Performance Management User's Guide*.

(1) Performance data is not collected after PFM - Agent startup

Take the following action:

- Check the startup status of the Oracle Database and start it if it is inactive.
- Check the instance environment settings.

  Execute the `jpcconf inst setup` command to specify the correct value for each item. For details about the `jpcconf inst setup` command, see the chapter on the commands in the manual *JP1/Performance Management Reference*.

8.2.3 Other problems

Check what events occur when the problem is encountered. If a message has been output, check the contents of the message. For details about the log information output by Performance Management, see *8.3 Log information*.

If the actions described in chapter on troubleshooting in the *JP1/Performance User's Guide* and sections *8.2.1 Problems relating to the start and setup of services* through *8.2.2 Problems relating to the collection and management of performance data* do not resolve the problem, or the problem arises in a different scenario, collect information to determine the cause of the problem, and contact the system administrator.

For details about the data you need to collect and how to collect it, see *8.4 Required troubleshooting information* and *8.5 Collecting troubleshooting information*.

8.3 Log information

You can determine what action to take when a problem occurs in Performance Management by reviewing the log information. Five types of log information are output during Performance Management operation:

- System log
- Common message log
- Operation status log
- Trace log
- Agent log

This section explains each type of log information.

8.3.1 Types of log information

(1) System log

A system log contains log information that describes the system status and reports errors in the system. System logs are output to the following log files:

- In Windows
  Event log file
- In UNIX
  syslog file

For details about the output format, see the chapter on log information in the manual *JP1/Performance Management Reference*.

Cautionary note on logical host use

In addition to the system log for Performance Management, logs for the cluster software are required. Use these logs to check the instructions issued to Performance Management by the cluster software.

(2) Common message log

The common message log contains log information that describes the system status and reports errors in the system. The common message log contains information that is more detailed than that in the system log. For details about the output file name and file size of the common message log, see *8.3.2 List of log files and directories*. For details about the output format of the common message log, see the chapter on log information in the manual *JP1/Performance Management Reference*.

Cautionary note on logical host use

When Performance Management is set up for logical host use, the common message log is output to a shared disk and inherited at failover. This means that the messages are recorded in the same log file before and after failover.

(3) Operation status log

The operation status log contains log information output by PFM - Web Console. For details about the output file name and file size of the operation status log, see the chapter on troubleshooting in the *JP1/Performance User's Guide*. For

details about the output format of the operation status log, see the chapter on log information in the manual *JP1/Performance Management Reference*.

(4) Trace log

A trace log contains log information that helps you investigate the status of the system leading up to the problem, and measure the processing time for each process.

Trace logs are output to the log files belonging to each Performance Management service.

Cautionary note on logical host use:

When Performance Management is set up for logical host use, trace logs are output to the shared disk and inherited at failover. This means that the messages are recorded in the same log file before and after failover.

(5) Agent log

An agent log that is output by PFM - Agent for Oracle contains log information about the processing executed to acquire records. If a problem occurs, collect agent logs to acquire detailed information about the processing.

Agent logs are output to separate files according to its type: normal log or error log. For details about output destinations, see *8.3.2(3) Agent log*.

Format:

The format of an output agent log is as follows:

```
yyyy/mm/dd hh:mm:ss.sss agto PID inf1 inf2 inf3 MessageID Message

The following explains the output items.
```

Table 8–2: Agent log items

Item	Description
yyyy/mm/dd	Date on which the log was output (*yyyy*: year, *mm*: month, and *dd*: day)
hh:mm:ss.sss	Local time at which the log was output (*hh*: hour, *mm*: minute, *ss*: second, and *sss*: millisecond)
agto	Name of the process that output the log (*agto* is the process name of PFM - Agent for Oracle).
PID	Output process ID
inf1 to inf3	Maintenance information
MessageID	Message ID#
Message	Message#

#

For details about the message contents, see *7. Messages*.

Notes

- Do not change the time set on the Agent host or the update time of an agent log file. Since information about the last update date and time is used to output agent logs, the agent logs may not be output correctly if these times are changed.
- When Performance Management is set up for logical host use, specify a path on the shared disk so that the agent log output destination is the same for both the executing node and the standby node.

8.3.2 List of log files and directories

This subsection describes the log information output by a Performance Management program. Performance Management outputs the following log information:

For details about the output file name and file size of the operation status log, see the chapter on 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 subsection describes the trace logs output by Performance Management. The following tables list the name of the service or command that outputs trace logs for PFM - Agent and the directory where the logs are stored, for each OS.

Table 8–3: Trace log storage folders (in Windows)

Type of log	Output source	Folder name
Trace log	Action Handler service	*installation-folder*\bin\action\log\
	Performance Management command	*installation-folder*\tools\log\
	Agent Collector service	*installation-folder*\agto\agent*instance-name*\log\
	Agent Store service	*installation-folder*\agto\store*instance-name*\log\
	Status Server service	*installation-folder*\bin\statsvr\log\
Trace log (logical host use)	Action Handler service	*environment-directory*#\jp1pc\bin\action\log\
	Performance Management command	*environment-directory*#\jp1pc\tools\log\
	Agent Collector service	*environment-directory*#\jp1pc\agto\agent*instance-name*\log\
	Agent Store service	*environment-directory*#\jp1pc\agto\store*instance-name*\log\

#

The environment directory is the directory you specified on the shared disk when creating the logical host.

Table 8–4: Trace log storage directories (in UNIX)

Type of log	Output source	Directory name
Trace log	Action Handler service	/opt/jp1pc/bin/action/log/
	Performance Management command	/opt/jp1pc/tools/log/

Type of log	Output source	Directory name
Trace log	Agent Collector service	/opt/jp1pc/agto/agent/*instance-name*/log/
	Agent Store service	/opt/jp1pc/agto/store/*instance-name*/log/
	Status Server service	/opt/jp1pc/bin/statsvr/log/
Trace log (logical host use)	Action Handler service	*environment-directory*#/jp1pc/bin/action/log/
	Performance Management command	*environment-directory*#/jp1pc/tools/log/
	Agent Collector service	*environment-directory*#/jp1pc/agto/agent/*instance-name*/log/
	Agent Store service	*environment-directory*#/jp1pc/agto/store/*instance-name*/log/

#

The environment directory is the directory you specified on the shared disk when creating the logical host.

(3) Agent log

This subsection describes the agent logs output by Performance Management. The following tables list the name of the service or command that outputs agent logs for PFM - Agent for Oracle, the name of the log file, and the disk space used by each file.

Table 8–5: Agent log files

Type of log	Output source	Default output destination#1	File name	Default Disk usage#1 (MB)
Normal log	PFM - Agent for Oracle	In Windows *installation-folder*\agto\agent*instance-name*\log\ In UNIX /opt/jp1pc/agto/agent/*instance-name*/log/	Agtoinf{01|02}#2	16
Error log			Agtoerr{01|02}#2	
Normal log (logical host use)	PFM - Agent for Oracle	In Windows *environment-directory*#3\jp1pc\agto\agent*instance-name*\log\ In UNIX *environment-directory*#3/jp1pc/agto/agent/*instance-name*/log/	Agtoinf{01|02}#2	16
Error log (logical host use)			Agtoerr{01|02}#2	

#1

You can use the following methods to check and change the output destination of agent logs and the maximum file size:

- jpcconf inst setup command
- Agent Configuration property in the PFM - Web Console window

For details about how to use the `jpcconf inst setup` command to change the settings, see *2.4.2 Updating an instance environment* (for Windows) or *3.4.2 Updating an instance environment* (for UNIX).

#2

Agent logs are output using two sequential files. The file names are suffixed with `01` or `02`, which have the following meanings:

- `01`: Current file
- `02`: Backup file

For details about sequential files, see *When using sequential files (jpclog)* in *(1) Common message log*.

#3

The environment directory is on the shared disk specified when the logical host was created.

8.4 Required troubleshooting information

If the actions described in *8.2 Troubleshooting procedures* do not resolve the problem, collect information to determine the cause of the problem, and then contact the system administrator. This section describes the information you need to collect when an error occurs.

Performance Management provides the `jpcras` command to collect the required information in a batch. Use this command to collect information about PFM - Agent. In the following tables, the information that can be collected by the `jpcras` command is indicated as such.

Note:

The data collected by the `jpcras` command depends on the options specified when the command was executed. For details about the command options and the data that can be collected, see the chapter on commands in the manual *JP1/Performance Management Reference*.

Cautionary notes on logical host use:

- When running in a logical host environment, Performance Management outputs logs to a shared disk. If the shared disk is online (Windows) or mounted (UNIX) when you execute the `jpcras` command, the logs on the shared disk are also collected.
- To investigate the cause of a problem that occurred during failover, you will need information from before and after the failover. For this reason, you must collect information from both the executing node and the standby node.
- When Performance Management is running in a logical host environment, you must also collect information for the cluster software. Because Performance Management is started and stopped by the cluster software in a logical host environment, collecting this information allows you to check the behavior of Performance Management against the behavior of the cluster software.

8.4.1 In Windows

(1) Log information about the OS

Collect the information about the OS. The following table lists the information about the OS:

Table 8–6: Information about the OS

Type of information	Details	Default file name	Collected by jpcras command
System log	Windows event log	--	Y
Process information	List of processes	--	Y
System file	`hosts` file	*system-folder*`\system32\drivers\etc\hosts`	Y
	`services` file	*system-folder*`\system32\drivers\etc\services`	Y
OS information	System information	--	Y

Type of information	Details	Default file name	Collected by jpcras command
OS information	Network status	--	Y
	Host name	--	Y
	Windows firewall information	--	Y
Dump information	Problem Reports and Solutions log file	*user-mode-process-dump-output-folder**program-name.process-ID.dmp* Example: `jpcagto.exe.2420.dmp`	N

Legend:

Y: Can be collected

N: Cannot be collected

--: Not applicable

Note

If you set up Dr. Watson to output log files to a different folder, collect the information from that folder.

(2) Performance Management information

Collect the log information about Performance Management. If the problem relates to a network connection, also collect information from the machine that is the connection target. The following table lists the log information about the Performance Management:

Table 8–7: Log information about Performance Management

Type of information	Details	Default file name	Collected by jpcras command
Common message log	Message log output by Performance Management (sequential files)	*installation-folder*\log\jpclog{01|02}[#1]	Y
	Message log output by Performance Management (wraparound files)	*installation-folder*\log\jpclogw{01|02}[#1]	Y
Configuration information	Configuration information files	--	Y
	Output of `jpctool service list` command	--	Y
Version information	Product versions	--	Y
	Log information	--	Y
Database information	Agent Store service	For Store 1.0 *installation-folder*\agto\store*instance-name**.DB *installation-folder*\agto\store*instance-name**.IDX For Store 2.0 *installation-folder*\agto\store*instance-name*\STPD The following files under the	Y

Type of information	Details	Default file name	Collected by jpcras command
Database information	Agent Store service	*installation-folder*\agto\store*instance-name*\STPI folder: • *.DB • *.IDX	Y
Trace log	Trace information for Performance Management services	--[#2]	Y
Agent log	Normal log for processing related to acquisition of PFM - Agent for Oracle records	*installation-folder*\agto\agent*instance-name*\log\agtoinf{01|02}[#3]	Y[#4]
	Error log for processing related to acquisition of PFM - Agent for Oracle records	*installation-folder*\agto\agent*instance-name*\log\agtoerr{01|02}[#3]	Y[#4]
Install log[#5]	Message logs from installation	The following files under the *system-folder* \TEMP\HCDINST folder: • HCDMAIN.LOG and HCDMAINn.LOG[#6] • HCDINST.LOG and HCDINSTn.LOG[#6] • *product-model-name*.LOG	N

Legend:

Y: Can be collected

N: Cannot be collected

--: Not applicable

#1

For details about the output format of the log files, see the chapter on detecting errors in Performance Management in the *JP1/Performance Management User's Guide*.

#2

For details about the storage folders for trace logs, see *8.3.2(2) Trace log*.

#3

For details about the output format of an agent log and how to change the storage folder, see *8.3.2(3) Agent log*.

#4

The jpcras command collects agent log information only from the currently specified output destination folder. If you change the output destination folder for agent logs, you need to manually collect data from the agent log files that were output before the change.

#5

Collect this information if installation failed.

#6

n indicates number

(3) Operation information

Collect the following information about the operation that was being performed when the problem occurred:

- Details of the operation

- Time when the error occurred
- Machine configuration (version of each OS, host name, configuration of PFM - Manager and PFM - Agent)
- Whether the error is reproducible
- Name of any Performance Management user who logged in from PFM - Web Console
- Arguments specified in the command when an error occurs during command execution

(4) Error information on the screen

Collect hard copies of the following:

- Windows on the screen when the application error occurred
- The error dialog boxes (Also copy the detailed information if a dialog box contains a **Details** button.)
- Information in the Command Prompt window or [Administrator Console] window when an error occurs during command execution

(5) User mode process dump

If a Performance Management process stops due to an application error, obtain a user mode process dump.

(6) Collecting problem reports

If a Performance Management process stops due to an application error, obtain a problem report.

(7) Information about performance data

Collect the following information about performance data. You must also collect files located on the machine, command results, and registry information.

The following table shows information about performance data that is specially collected by PFM - Agent for Oracle in an environment in which PFM - Agent for Oracle is installed.

Table 8–8: Performance data information collected by PFM - Agent for Oracle

Type of information	Details	Default file name[#1]	Collected by jpcras command
PDNL record information	Output result of lsnrctl (Oracle command)	*installation-folder*\agto\agent*instance-name*\pdnl.out	Y
PDNL record information	Output result of lsnrctl (Oracle command)	*installation-folder*\agto\agent*instance-name*\lsnrctl.status.pdnl.out.err.txt	Y
PDNL record information	Execution result of hostname	*installation-folder*\agto\agent*instance-name*\sqlnet.tmp	Y
PDLS record information	Output result of lsnrctl (Oracle command)	*installation-folder*\agto\agent*instance-name*\pdls.out	Y
PDLS record information	Output result of lsnrctl (Oracle command)	*installation-folder*\agto\agent*instance-name*\lsnrctl.status.pdls.out.err.txt	Y

Type of information	Details	Default file name[#1]	Collected by jpcras command
Error information during Oracle connection[#2]	Error code when an error occurs for the Oracle connection	*installation-folder*\agto\agent*instance-name*\pdia_xxxxx_status.db[#3]	Y
Internal maintenance information	Log file for OCI functions	*installation-folder*\agto\agent*instance-name*\ocilog.txt	Y

Legend:

Y: Can be collected

#1

When a logical host is used, replace *installation-folder* with *environment-folder*\jp1pc

#2

This file may be created when an error occurs during an Oracle connection.

#3

xxxxx is replaced with any string.

(8) Other information

Also collect the following information:

- Windows system information
- Windows system and application logs (of Windows event logs)

8.4.2 In UNIX

(1) Log information about the OS

Collect the information about the OS. The following table lists the information about the OS:

Table 8–9: Information about the OS

Type of information	Details	Default file name	Collected by jpcras command
System log	syslog	In HP-UX /var/adm/syslog/syslog.log In Solaris /var/adm/messages In AIX -- In Linux /var/log/messages	Y[#1]
Process information	List of processes	--	Y
System file	hosts file	/etc/hosts	Y

Type of information	Details	Default file name	Collected by jpcras command
System file	`hosts` file	`/etc/inet/ipnodes`[#2]	Y[#3]
	`services` file	`/etc/services`	Y
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

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 08-50 or later, or PFM - Base 08-50 or later, can collect this data

#4

In HP-UX 11i V3 (IPF), 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.

(2) Information about Performance Management

Collect the information about Performance Management. If the problem relates to a network connection, also collect information from the machine that is the connection target.

The following table lists the information about Performance Management:

Table 8–10: Information about Performance Management

Type of information	Details	Default file name	Collected by jpcras command
Common message log	Message log output by Performance Management (sequential files)	`/opt/jp1pc/log/jpclog{01|02}`[#1]	Y

Type of information	Details	Default file name	Collected by jpcras command
Common message log	Performance Management (wraparound files)	/opt/jp1pc/log/jpclogw{01|02}[#1]	Y
Configuration information	Configuration information files	--	Y
	Output of jpctool service list command	--	Y
Version information	Product version	--	Y
	Log information	--	Y
Database information	Agent Store service	For Store 1.0 /opt/jp1pc/agto/store/*instance-name*/*.DB /opt/jp1pc/agto/store/*instance-name*/*.IDX For Store 2.0 /opt/jp1pc/agto/store/*instance-name*/STPD The following files under the /opt/jp1pc/agto/store/*instance-name*/STPI directory: *.DB *.IDX	Y
Trace log	Trace information for Performance Management services	--[#2]	Y
Agent log	Normal log for processing related to acquisition of PFM - Agent for Oracle records	/opt/jp1pc/agto/agent/*instance-name*/log/agtoinf{01|02}[#3]	Y[#4]
	Error log for processing related to acquisition of PFM - Agent for Oracle records	/opt/jp1pc/agto/agent/*instance-name*/log/agtoerr{01|02}[#3]	Y[#4]
Install log[#5]	Standard log for Hitachi Program Product 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

N: Cannot be collected

--: Not applicable

#1

For details about the output format of the log files, see the chapter on detecting errors in Performance Management in the *JP1/Performance Management User's Guide*.

#2

For details about the storage directories for trace logs, see *8.3.2 List of log files and directories*.

#3

For details about the output format of an agent log and how to change the storage folder, see *8.3.2 List of log files and directories*.

#4

The `jpcras` command collects agent log information only from the currently specified output destination folder. If you change the output destination folder for agent logs, you need to manually collect data from the agent log files that were output before the change.

#5

Collect this information if installation failed.

(3) Operation information

Collect the following information about the operation that was being performed when the problem occurred:

- Details of the operation
- Time when the error occurred
- Machine configuration (version of each OS, host name, configuration of PFM - Manager and PFM - Agent)
- Whether the error is reproducible
- Name of any Performance Management user who logged in from PFM - Web Console

(4) Error information

Collect the following error information:

- Messages output to the console when an error occurs during command execution

(5) Information about performance data

Collect the following information about performance data. You must also collect files located on the machine, command results, and registry information.

The following table shows information about performance data that is specially collected by PFM - Agent for Oracle in an environment in which PFM - Agent for Oracle is installed.

Table 8–11: Performance data information collected by PFM - Agent for Oracle

Type of information	Details	File name[#1]	Collected by jpcras command
PDNL record information	Output result of lsnrctl (Oracle command)	`/opt/jp1pc/agto/agent/`*instance-name*`/pdnl.out`	Y
PDNL record information	Output result of lsnrctl (Oracle command)	`/opt/jp1pc/agto/agent/`*instance-name*`/lsnrctl.status.pdnl.out.err.txt`	Y
PDNL record information	Execution result of hostname	`/opt/jp1pc/agto/agent/`*instance-name*`/sqlnet.tmp`	Y
PDLS record information	Output result of lsnrctl (Oracle command)	`/opt/jp1pc/agto/agent/`*instance-name*`/pdls.out`	Y
PDLS record information	Output result of lsnrctl (Oracle command)	`/opt/jp1pc/agto/agent/`*instance-name*`/lsnrctl.status.pdls.out.err.txt`	Y
Error information during Oracle connection[#2]	Error code when an error occurs for the Oracle connection	`/opt/jp1pc/agto/agent/`*instance-name*`/pdia_xxxxx_status.db`[#3]	Y
Internal maintenance information	Log file for OCI functions	`/opt/jp1pc/agto/agent/`*instance-name*`/ocilog.txt`	Y

Legend:

Y: Can be collected

#1

When a logical host is used, replace `/opt/jp1pc` with *environment-directory*`/jp1pc`

#2

This file may be created when an error occurs during an Oracle connection.

#3

xxxxx is replaced with any string.

(6) Other information

Also collect the following information:

- Arguments specified in the command when an error occurs during command execution

8.5 Collecting troubleshooting information

This section describes how to collect information when an error occurs.

8.5.1 In Windows

(1) Collecting dump information

To collect dump information in a Windows environment:

1. Open Task Manager.
2. Select the process tab.
3. Right-click the process name for which you want to collect dump information, and then select Create Dump File.
   Dump files are stored in the following folder:

   ```
   system-drive\Users\user-name\AppData\Local\Temp
   ```

4. Obtain the dump file from the folder created in step 3.
   If you have changed the environment variable settings so that dump files are output to a different folder, obtain the dump file from that folder.

(2) Execute the data collection command

Use the `jpcras` command to collect the information needed to investigate an error. The following procedure shows how to execute the data collection command. Note that you must perform these operations as an OS user who belongs to the Administrators group.

1. Log on to the host on which the service whose data you are collecting is installed.
2. At the command prompt, execute the following command to enable command extensions in the command interpreter.

   ```
   cmd /E:ON
   ```

3. Execute the `jpcras` command, specifying the data to be collected and the folder in which to store it.
   For example, to have the `jpcras` command store all collectible data in the folder `c:\tmp\jpc\agt`, specify the command as follows:

   ```
   installation-folder\tools\jpcras c:\tmp\jpc\agt all all
   ```

When you execute `jpcras` command, `jpctool service list -id * -host *` command is executed to acquire the PFM service list and check the startup status of these services. If the execution-host-name and other Performance Management System host communicate across a firewall or if the scale of the system is large, it may take time to execute `jpctool service list -id * -host *` command. In that case, by setting JPC_COLCTRLNOHOST environment variable to 1 `jpctool service list -id * -host *` command is not executed so that `jpcras` command can take shorter time.

For details about the `jpcras` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(3) Execute the data collection command (for logical host use)

When you run Performance Management in a logical host environment, the data is stored on a shared disk. In this case, collect data from both the executing node and the standby node.

Use the `jpcras` command to collect the information needed to investigate an error. The following procedure shows how to execute the data collection command. Note that you must perform these operations as an OS user who belongs to the Administrators group.

To execute the data collection command in a logical host environment:

1. Bring the shared disk online.

   For logical hosts, data is stored on a shared disk. On the executing node, make sure that the shared disk is online before collecting data.

2. Execute the `jpcras` command on both the executing node and the standby node, specifying the data to collect and the folder in which to store it.

   For example, to have the `jpcras` command store all collectible data in the folder `c:\tmp\jpc\agt`, specify the command as follows:

   ```
   installation-folder\tools\jpcras c:\tmp\jpc\agt all all
   ```

   If you execute the `jpcras` command without specifying the `lhost` argument, data relating to Performance Management is collected from all physical and logical hosts on that node. If any Performance Management programs are running in a logical host environment, the log files will be collected from the shared disk.

   If the shared disk is offline for the node on which you execute the `jpcras` command, you will be unable to acquire the files on the shared disk. However, the command will end normally without generating an error.

   Note

   You need to collect data from both the executing node and standby node by executing the data collection command on each node. Data from both nodes is required to investigate the status of the system leading up to and following failover.

   When you execute `jpcras` command, `jpctool service list -id * -host *` command is executed to acquire the PFM service list and check the startup status of these services. If the execution-host-name and other Performance Management System host communicate across a firewall or if the scale of the system is large, it may take time to execute `jpctool service list -id * -host *` command. In that case, by setting JPC_COLCTRLNOHOST environment variable to 1 `jpctool service list -id * -host *` command is not executed so that `jpcras` command can take shorter time.

   For details about the `jpcras` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

3. Collect data about the cluster software.

   This data is required to determine whether a problem is caused by Performance Management or the cluster software. Collect data from which you can determine the results of any instructions, such as start and stop requests that the cluster software issued to Performance Management.

(4) Collect the Windows event log

Contents of **System** and **Application** in the Windows Event Viewer window

(5) Check the operation information

If an error occurs while an operation is being performed, check and record information about the operation. Check the following information:

- Details of the operation
- The time when the error occurred
- Machine configuration (version of each OS, host name, configuration of PFM - Manager and PFM - Agent)
- Whether the error is reproducible
- Name of any Performance Management user who logged in from PFM - Web Console
- Arguments specified in the command when an error occurs during command execution

(6) Collecting the error information on the window

Collect hard copies of the following items:

- Hard copy of the window operation if an application error occurred
- Hard copy of the error message dialog box
  If detailed information is available, also copy its content.
- Hard copy of the Command Prompt window or Administrator Console window if a problem occurred during command execution
  To obtain a hard copy of the Command Prompt window or Administrator Console window in Windows, specify the following in the Command Prompt Properties window:
  - **Edit Options** under the **Options** tab
    Select **Quick Edit Mode**.
  - **Layout** tab
    Set **Height** under **Screen Buffer Size** to `500`.

(7) Collect other information

Collect the needed information in the following contents:

- Contents of **System** and **Application** in the Windows Event Viewer window
- Content of **System Information** under **System Tools** under **Accessories**

8.5.2 In UNIX

(1) Execute the data collection command

Use the `jpcras` command to collect the information needed to investigate an error. The following procedure shows how to execute the data collection command. Note that you must perform these operations as an OS user who has root privileges.

To execute the data collection command:

1. Log in to the host on which the service whose data you are collecting is installed.

2. Execute the jpcras command, specifying the data to collect and the directory in which to store it.

   For example, to have the jpcras command store all collectible data in the directory /tmp/jpc/agt specify the command as follows:

   ```
   /opt/jp1pc/tools/jpcras /tmp/jpc/agt all all
   ```

   The data collected by the data collection command is compressed with the tar and compress commands and stored in the specified directory. The file is given the following name:

   ```
   jpcras YYMMDD.tar.Z
   ```

   YYMMDD indicates the date.

When you execute jpcras command, jpctool service list -id * -host * command is executed to acquire the PFM service list and check the startup status of these services. If the execution-host-name and other Performance Management System host communicate across a firewall or if the scale of the system is large, it may take time to execute jpctool service list -id * -host * command. In that case, by setting JPC_COLCTRLNOHOST environment variable to 1 jpctool service list -id * -host * command is not executed so that jpcras command can take shorter time.

For details about the jpcras command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

(2) Execute the data collection command (for logical host use)

When you run Performance Management in a logical host environment, the data is stored on a shared disk. In this case, collect data from both the executing node and the standby node.

Use the jpcras command to collect the information needed to investigate an error. The following procedure shows how to execute the data collection command. Note that you must perform these operations as an OS user who has root privileges.

To execute the data collection command in a logical host environment:

1. Mount the shared disk.

   For logical hosts, data is stored on a shared disk. On the executing node, make sure that the shared disk is mounted before collecting data.

2. Execute the jpcras command on both the executing node and the standby node, specifying the data to collect and the directory in which to store it.

   For example, to have the jpcras command store all collectible data in the directory /tmp/jpc/agt specify the command as follows:

   ```
   /opt/jp1pc/tools/jpcras /tmp/jpc/agt all all
   ```

   The data collected by the data collection command is compressed with the tar and compress commands and stored in the specified directory. The file is given the following name:

   ```
   jpcras YYMMDD.tar.Z
   ```

   YYMMDD indicates the date.

   If you execute the jpcras command without specifying the lhost argument, data relating to Performance Management is collected from all physical and logical hosts on that node. If any Performance Management programs are running in a logical host environment, their log files will be collected from the shared disk.

If the shared disk is not mounted on the node on which you execute the `jpcras` command, you will be unable to acquire the files on the shared disk. However, the command will end normally without generating an error.

Note

Collect data from both the executing node and standby node by executing the data collection command on each node. Data from both nodes is required to investigate problems leading up to and following failover.

When you execute `jpcras` command, `jpctool service list -id * -host *` command is executed to acquire the PFM service list and check the startup status of these services. If the execution-host-name and other Performance Management System host communicate across a firewall or if the scale of the system is large, it may take time to execute `jpctool service list -id * -host *` command. In that case, by setting JPC_COLCTRLNOHOST environment variable to 1 `jpctool service list -id * -host *` command is not executed so that `jpcras` command can take shorter time.

For details about the `jpcras` command, see the chapter on commands in the manual *JP1/Performance Management Reference*.

3. Collect data about the cluster software.

This data is required to determine whether a problem is caused by Performance Management or the cluster software. Collect data from which you can determine the results of any instructions, such as start and stop requests, that the cluster software issued to Performance Management.

(3) Check the operation information

If an error occurs while an operation is being performed, check and record information about the operation. Check the following information:

- Details of the operation
- The time when the error occurred
- Machine configuration (version of each OS, host name, configuration of PFM - Manager and PFM - Agent)
- Whether the error is reproducible
- Name of any Performance Management user who logged in from PFM - Web Console

(4) Collect error information

Collect the following error information:

- Messages output to the console when an error occurs during command execution

(5) Collect other information

Also collect the following information:

- Arguments specified in the command when an error occurs during command execution

8.6 Detecting errors in Performance Management

Performance Management provides a status management facility that allows you to check the status of Performance Management itself when an error occurs. This facility monitors the operating status of PFM - Agent and PFM - Agent host and reports these operating statuses on the PFM - Web Console.

By using PFM service automatic restart facility, you can automatically restart PFM services when PFM services abnormally end, or you can regularly restart PFM services.

When you monitor the operating status of PFM - Agent or automatically restart PFM services, use the status management facility, which checks the detailed status of Performance Management services. As a result, it is required that the version number of PFM - Agent you are using supports the status maintenance facility and that you enable it. Note that there are no prerequisites for monitoring the PFM - Agent hosts.

Alternatively, you can use JP1/Base, an integrated system monitor, to monitor the log file of Performance Management so that you can detect errors in Performance Management. By using these facilities, the system administrator can detect errors when they occur, identify the cause of them, and then take the appropriate action to recover from them.

For details about detecting errors in Performance Management itself, see the chapter on error detection in Performance Management in the *JP1/Performance Management User's Guide*.

8.7 Recovering from errors in Performance Management

When an error occurs on a Performance Management server, you must use backup files to restore the system to a normal state before the error occurred.

For details about how to do so, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

Appendixes

A. Estimating System Requirements

Hitachi recommends that you estimate in advance whether the computer to be configured in your PFM - Agent for Oracle system has sufficient processing capacity to run the program.

This appendix describes the system requirements you need to consider.

A.1 Memory requirements

Memory requirements depend on how PFM - Agent for Oracle is set up and used.

For details about the formula for estimating memory requirements, see the *Release Notes*.

A.2 Disk space requirements

Disk space requirements depend on the number of records used to collect performance data. To estimate the disk space requirements, you need to estimate the disk space requirements of the entire system and the Store database of Store version 1.0 or of Store version 2.0. For details about the formula for estimating these requirements, see the *Release Notes*.

A.3 Disk space requirements for operation in a cluster system

The disk space requirements for operation in a cluster system are estimated in the same way as for operation on a single node. For details about the disk space requirements, see the *Release Notes*.

B. Kernel Parameters

To use PFM - Agent for Oracle, you must adjust the OS kernel parameters to allocate resources required for execution processing. This appendix describes the kernel parameters that must be adjusted.

For details about adjusting the kernel parameters for using PFM - Manager and PFM - Web Console in a UNIX environment, see the list of kernel parameters in the appendixes in the *JP1/Performance Management Planning and Configuration Guide*.

B.1 HP-UX

(1) System resources required for PFM - Agent for Oracle

The following table shows the kernel parameters that must be adjusted in HP-UX environment.

Table B–1: System resources required for PFM - Agent for Oracle (HP-UX)

System resource	Parameter	Estimate
File system	`nfile`	--
	`nflocks`	--
	`maxfiles`	--
Common memory	`shmmni`	--
	`shmmax`	--
	`shmseg`	--
Semaphore	`semmni`	2 * *number-of-instances-of-PFM-Agent-for-Oracle*
	`semmns`	2 * *number-of-instances-of-PFM-Agent-for-Oracle*
	`semume`	--
	`semmnu`	--
Process	`nproc`	--
Memory	`maxdsiz`	--
	`maxssiz`	--

Legend:

--: Indicates a parameter that does not require adjustment.

B.2 Solaris

(1) System resources required for PFM - Agent for Oracle

The following table shows the kernel parameters that must be adjusted in Solaris environment.

Table B–2: System resources required for PFM - Agent for Oracle (Solaris)

System resource	Parameter	Estimate
Common memory	shmsys:shminfo_shmmni	--
	shmsys:shminfo_shmmax	--
	shmsys:shminfo_shmseg	--
Semaphore	semsys:seminfo_semmni (project.max-sem-ids)#1	2 * *number-of-instances-of-PFM-Agent-for-Oracle*
	semsys:seminfo_semmns#2	2 * *number-of-instances-of-PFM-Agent-for-Oracle*
	semsys:seminfo_semume	--
	semsys:seminfo_semmnu	--
Process	maxuprc or max_nprocs	--

Legend:

--: Indicates a parameter that does not require adjustment.

#1

Specify this parameter when performing adjustment using Solaris 10 resource controls.

#2

For Solaris 10, this parameter is not required.

B.3 AIX

For AIX, kernel parameters are not required to be adjusted.

B.4 Linux

(1) System resources required for PFM - Agent for Oracle

The following table shows the kernel parameters that must be adjusted in Linux environment.

Table B–3: System resources required for PFM - Agent for Oracle (Linux)

System resource	Parameter	Estimate
Semaphore	SEMMNI	2 * *number-of-instances-of-PFM-Agent-for-Oracle*
	SEMMNS	2 * *number-of-instances-of-PFM-Agent-for-Oracle*

C. List of Identifiers

When you operate PFM - Agent for Oracle or extract performance data from its Store database, you may require an identifier of PFM - Agent for Oracle. The following table describes the PFM - Agent for Oracle identifiers.

Table C–1: List of PFM - Agent for Oracle identifiers

Use	Name	Identifier	Description
Command	Product ID	O	The product ID is part of the service ID required when using a command to check the Performance Management system configuration or to back up performance data. For details about service IDs, see the naming rules described in the appendix in the *JP1/Performance Management Planning and Configuration Guide*.
	Service key	agto or Oracle	A service key is required when using a command to start or stop PFM - Agent for Oracle. For details about service keys, see the naming rules described in the appendix in the *JP1/Performance Management Planning and Configuration Guide*.
Help	Help ID	pcao	The help ID indicates that the help is for PFM - Agent for Oracle.

D. List of Processes

This appendix describes the processes of PFM - Agent for Oracle.

The following table lists the PFM - Agent for Oracle process. The value following the process name is the number of processes that can be started concurrently.

Note:

The process and limit numbers are identical whether PFM - Agent on a physical host or logical host.

Table D–1: Processes of PFM - Agent for Oracle (for Windows)

Process name (Process count)	Function
`jpcagto.exe` (*n*)	The process of the Agent Collector service. One process is started for each instance of PFM - Agent for Oracle.
`jpcOcollect.exe` (*n*)[#1]	The process for collecting performance data. One process is started for each instance.
`jpcsto.exe` (*n*)	The process of the Agent Store service. One process is started for each instance of PFM - Agent for Oracle.
`stpqlpr.exe` (1)[#2]	The process for backup or export of the Store database

#1

This process is a child process of the `jpcagto.exe` process.

#2

This process is a child process of the `jpcsto.exe` process.

Table D–2: Processes of PFM - Agent for Oracle (for UNIX)

Process name (Process count)	Function
`jpcagto` (*n*)	The process of the Agent Collector service. One process is started for each instance of PFM - Agent for Oracle.
`jpcOcollect_10` (*n*)[#1]	The process for collecting performance data. One process is started for each instance. Monitored database: Oracle10g[#2], Oracle11g[#3] or later
`jpcOcollect_11` (*n*)[#1]	The process for collecting performance data. One process is started for each instance. Monitored database: Oracle11g[#4] or later
`agto/jpcsto` (*n*)	The process of the Agent Store service. One process is started for each instance of PFM - Agent for Oracle.
`stpqlpr` (1)[#5]	The process for backup or export of the Store database

#1

This process is a child process of the `jpcagto` process.

#2

This process runs if you monitor Oracle10g in HP-UX.

#3

This process runs if you monitor Oracle11g or later in Linux or AIX.

#4

This process runs if you monitor Oracle11g or later in HP-UX or Solaris.

#5

This process is a child process of the `jpcsto` process.

E. List of Port Numbers

This appendix lists the port numbers used by PFM - Agent for Oracle.

For details about the port numbers and firewall routing in PFM - Manager and PFM - Base, see the appendixes in the manual *JP1/Performance Management Reference*.

The port numbers can be changed to suit the user environment.

For details about changing a port number, see the chapter on installation and setup in the *JP1/Performance Management Planning and Configuration Guide*. The TCP/IP protocol is used.

Note:

- Performance Management supports network address translation in static mode (Basic NAT), which provides one-to-one address mapping.
- Performance Management does not support dynamic NAT or NAPT containing port translations (IP Masquerade and NAT+).

E.1 numbers for PFM - Agent for Oracle

The following table shows the port numbers used by PFM - Agent for Oracle.

Table E–1: Port numbers used by PFM - Agent for Oracle

Port number	Service name	Parameter	Use
Automatic#1	Agent Store service	`jp1pcstoo` [*nnn*] #2	Used for recording performance data or acquiring a historical report
Automatic#1	Agent Collector service	`jp1pcagto` [*nnn*]#2	Used for binding an alarm or acquiring a real-time report

#1

A port number not already being used by the system is assigned automatically every time the service is restarted.

#2

When multiple instances are created, a sequential number (*nnn*) is appended to the second and subsequent instances. The first instance does not have a sequential number.

E.2 Routing through a firewall

If PFM - Manager and PFM - Agent for Oracle communicate across a firewall, set fixed port numbers for all PFM - Manager and PFM - Agent ports.

For more details, see the section describing the firewall passage direction in the manual *JP1/Performance Management Reference*.

F. PFM - Agent for Oracle Properties

This appendix lists the properties of the following PFM - Agent for Oracle services that are displayed in PFM - Web Console:

- Agent Store service
- Agent Collector service

F.1 List of Agent Store service properties

The following table lists the properties for the Agent Store service of PFM - Agent for Oracle.

Table F–1: List of Agent Store service properties of PFM - Agent for Oracle

Directory name	Property name	Description
--	First Registration Date	Displays the date and time on which the service was first recognized by PFM - Manager.
	Last Registration Date	Displays the date and time on which the service was last recognized by PFM - Manager.
General	--	Stores information such as the host name and directories. The properties in this folder cannot be changed.
	Directory	Displays the name of the current directory where the service runs.
	Host Name	Displays the name of the physical host on which the service runs.
	Process ID	Displays the process ID of the service.
	Physical Address	Displays the IP address and port number of the host on which the service runs when IPv6 communication is disabled.
	Physical Address(IPv4)	Displays the IP address (IPv4) of the host on which the service runs when IPv6 communication is enabled.
	Physical Address(IPv6)	Displays the IP address (IPv6) of the host on which the service runs when IPv6 communication is enabled.
	Port Number	Displays the port number on which the service runs when IPv6 communication is enabled.
	User Name	Displays the name of the user who executed the service process.
	Time Zone	Displays the time zone in which the service was used.
System	--	Stores information about the OS under which the service runs. The properties in this folder cannot be changed.
	CPU Type	Displays the CPU type.
	Hardware ID	Displays the hardware ID.
	OS Type	Displays the type of OS.
	OS Name	Displays the name of the OS.
	OS Version	Displays the version of the OS.

Directory name		Property name	Description
Network Services		--	Stores information about the common library for Performance Management communication. The properties in this folder cannot be changed.
		Build Date	Displays the date on which the Agent Store service was created.
		INI File	Displays the directory containing the `jpcns.ini` file.
Network Services	Service	--	Stores information about the service. The properties in this folder cannot be changed.
		Description	Displays the host name in the following format: *instance-name_host-name*
		Local Service Name	Displays the service ID.
		Remote Service Name	Displays the service ID of the Master Manager service on the connection-target PFM - Manager host.
		EP Service Name	Displays the service ID of the Correlator service on the connection-target PFM - Manager host.
Retention		--	Sets the data retention period when the Store version is 1.0. For details, see the chapter on management of operation monitoring data in the *JP1/Performance Management User's Guide*.
		Product Interval - Minute Drawer	Sets the retention period for records of the PI record type per minute. 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	Sets the retention period for records of the PI record type per hour. 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	Sets the retention period for records of the PI record type per day. One of the following can be selected: • Day • 2 Days • 3 Days

Directory name		Property name	Description
Retention		Product Interval - Day Drawer	• 4 Days • 5 Days • 6 Days • Week • Month • Year
		Product Interval - Week Drawer	Sets the retention period for records of the PI record type per week. One of the following can be selected: • Week • Month • Year
		Product Interval - Month Drawer	Sets the retention period for records of the PI record type per month. One of the following can be selected: • Month • Year
		Product Interval - Year Drawer	The retention period for records of the PI record type per year is displayed. The specifiable value is fixed to Year.
		Product Detail - *record-ID-of-PD-record-type*	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.
Retention Ex		--	Sets the data retention period when the Store version is 2.0. For details, see the chapter on management of operation monitoring data in the *JP1/Performance Management User's Guide*.
Retention Ex	Product Interval - *record-ID-of-PI-record-type*	--	Sets the retention period for records of the PI record type.
		Period - Minute Drawer (Day)	Sets the retention period for records of the PI record type per minute. Values from 0 to 366 days can be specified, on a daily basis.
		Period - Hour Drawer (Day)	Sets the retention period for records of the PI record type per hour. Values from 0 to 366 days can be specified, on a daily basis.
		Period - Day Drawer (Week)	Sets the retention period for records of the PI record type per day. Values from 0 to 266 weeks can be specified, on a weekly basis.
		Period - Week Drawer (Week)	Sets the retention period for records of the PI record type per week. Values from 0 to 266 weeks can be specified, on a weekly basis.
		Period - Month Drawer (Month)	Sets the retention period for records of the PI record type per month. Values from 0 to 60 months can be specified, on a monthly basis.
		Period - Year Drawer (Year)	Sets the retention period for records of the PI record type per year. This is fixed at 10.
	Product Detail - *record-ID-of-PD-record-type*	Period (Day)	Sets the retention period for each ID for records of the PD record type. Retention periods can be set as an integer from 0 to 366, in days.
Disk Usage		--	This folder contains information for disk space used by each database. The values in this folder are those current at the time the properties are displayed. The properties in this folder cannot be changed.
		Product Interval	Displays the disk space used by the records of PI record type

Directory name	Property name	Description
Disk Usage	Product Detail	Displays the disk space used by the records of PD record type
	Product Alarm	Displays the disk space used by the records of PA record type. This property is not used in PFM - Agent for Oracle.
	Product Log	Displays the disk space used by the records of PL record type. This property is not used in PFM - Agent for Oracle.
	Total Disk Usage	Displays the total disk space used by the entire database.
Configuration	--	Displays the property 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

F.2 List of Agent Collector service properties

The following table lists the properties for the Agent Collector service of PFM - Agent for Oracle.

Table F–2: List of Agent Collector service properties of PFM - Agent for Oracle

Directory name	Property name	Description
--	First Registration Date	Displays the date and time on which the service was first recognized by PFM - Manager.
	Last Registration Date	Displays the date and time on which the service was last recognized by PFM - Manager.
	Data Model Version	Displays the version of the data model.
General	--	Stores information such as the host name and directories. The properties in this folder cannot be changed.
	Directory	Displays the name of the current directory where the service runs.
	Host Name	Displays the name of the physical host on which the service runs.
	Process ID	Displays the process ID of the service.
	Physical Address	Displays the IP address and port number of the host on which the service runs when IPv6 communication is disabled.

Directory name		Property name	Description
General		Physical Address(IPv4)	Displays the IP address (IPv4) of the host on which the service runs when IPv6 communication is enabled.
		Physical Address(IPv6)	Displays the IP address (IPv6) of the host on which the service runs when IPv6 communication is enabled.
		Port Number	Displays the port number on which the service runs when IPv6 communication is enabled.
		User Name	Displays the name of the user who executed the service process.
		Time Zone	Displays the time zone in which the service is used.
System		--	Stores information about the OS under which the service runs. The properties in this folder cannot be changed.
		CPU Type	Displays the CPU type.
		Hardware ID	Displays the hardware ID.
		OS Type	Displays the type of OS.
		OS Name	Displays the name of the OS.
		OS Version	Displays the version of the OS.
Network Services		--	Stores information about the common library for Performance Management communication. The properties in this folder cannot be changed.
		Build Date	Displays the date on which the Agent Collector service was created.
		INI File	Displays the name of the directory containing the `jpcns.ini` file.
Network Services	Service	--	Stores information about the service. The properties in this folder cannot be changed.
		Description	Displays the host name in the following format: *instance-name_host-name*
		Local Service Name	Displays the service ID.
		Remote Service Name	Displays the service ID of the Agent Store service to which the Agent Collector service connects.
		AH Service Name	Displays the service ID of the Action Handler service on the same host.
JP1 Event Configurations		--	Specify the condition under which JP1 event is issued.
		Each service	The user selects **Yes** or **No** from a list (Agent Collector service, Agent Store service, Action Handler service, and Status Server service) to specify whether each service issues JP1 system event.
		JP1 Event Send Host	Specify the connection target JP1/Base event server. Note that you can specify the event server on the same logical or physical host that the Action Handler service is running. The value you specify must consist of alphanumeric characters, ".", and "-", and must not exceed 255 bytes. If you exceed 255 bytes, the value you specify is discarded. If you exceed 255 bytes or do not specify this value, the host on which Action Handler service is running is used as the eventissuing host. If you specify **localhost**, the physical host is set to this property.
		Monitoring Console Host	Specify the PFM - Web Console host, if you open a PFM - Web Console login page by using JP1/IM - Manager monitor startup

Directory name		Property name	Description
JP1 Event Configurations		Monitoring Console Host	function. The value you specify must consist of alphanumeric characters, ".", and "-", and must not exceed 255 bytes. If you exceed 255 bytes, the value you specify is discarded. If you exceed 255 bytes or do not specify this value, the connection target PFM - Manager host is set to this value.
		Monitoring Console Port	Specify the port number (HTTP request port number). The range of the value is from 1 to 65535. If the value you specify is out of the range, the value is discarded. If the value is out of the range or you do not specify this value, 20358 is set to this value.
		Monitoring Console Https	Specifies whether to use HTTPS-encrypted communication to access PFM - Web Console when PFM - Web Console is started by JP1/IM - Manager monitor startup. By default, this property is set to No. • Yes: Use encrypted communication. • No: Do not use encrypted communication.
JP1 Event Configurations	Alarm	JP1 Event Mode	Specify which type of events to issue when the alarm status changes. • JP1 User Event: issuing JP1 user Event. • JP1 System Event: issuing JP1 system event.
Detail Records		--	
Detail Records	*record-ID*[#1]	--	Stores the properties of a record.
		Description	Displays a description of the record. This property cannot be changed.
		Log	The user selects **Yes** or **No** from a list to specify whether to save the record to the Store database. The record is saved when this value is **Yes** and the value of **Collection Interval** is greater than zero.
		Log(ITSLM)	Displays **Yes** or **No** to indicate whether to save the records to the Store database of PFM - Agent for Oracle from JP1/SLM - Manager. For this property, **No** (fixed value) is displayed. This property is read-only and cannot be changed.
		Monitoring(ITSLM)	Displays **Yes** or **No** to indicate the JP1/SLM - Manager setting for whether to send records to JP1/SLM - Manager. For this property, **No** (fixed value) is displayed. This property is read-only and cannot be changed.
		Collection Interval	Specifies the data collection interval. The value is in seconds, and can be from 0 to 2,147,483,647. When zero is specified, no data is collected.
		Collection Offset	Specifies the offset value to apply before the first collection cycle. The value is in seconds, and can be from 0 to 32,767, but must be less than the value specified in **Collection Interval**. The time at which the collected data is recorded matches the collection interval time, regardless of the offset value.
		Over 10 Sec Collection Time	This property is only displayed if collection of historical data takes precedence over the display processing of real-time reports (if the functionality that prioritizes the collection of historical data is enabled).[#2] Whether record collection might require 10 seconds or more is indicated by **Yes** or **No**. • **Yes**: Might require 10 seconds or more. • **No**: Does not require 10 seconds. The value of this property cannot be changed.

Directory name		Property name	Description
Detail Records	*record-ID*[#1]	Realtime Report Data Collection Mode	Specifies the display mode of real-time reports. • 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 conditions for saving the record to the database. Only records that satisfy the conditions are saved. This property shows the conditional expression (character string) specified in the LOGIF Expression Editor window, which opens when the user clicks **LOGIF** in the bottom frame of the **Properties** pane in the PFM - Web Console's **Services** page.
Interval Records		--	Stores the properties of a record of PI record type. The record ID of the collected record is shown in bold type.
Interval Records	*record-ID*[#1]	--	Stores the properties of the record.
		Description	Displays a description of the record. This property cannot be changed.
		Log	The user selects **Yes** or **No** from a list to specify whether to save the record to the Store database. The record is saved when this value is **Yes** and the value of **Collection Interval** is greater than zero.
		Log(ITSLM)	Displays **Yes** or **No** to indicate whether to save the records to the Store database of PFM - Agent for Oracle from JP1/SLM - Manager. For this property, **No** (fixed value) is displayed. This property is read-only and cannot be changed.
		Monitoring(ITSLM)	Displays **Yes** or **No** to indicate the JP1/SLM - Manager setting for whether to send records to JP1/SLM - Manager. For this property, **No** (fixed value) is displayed. This property is read-only and cannot be changed.
		Collection Interval	Specifies the data collection interval. The value is in seconds, and can be from 0 to 2,147,483,647. When zero is specified, no data is collected.
		Collection Offset	Specifies the offset value to apply before the first collection cycle. The value is in seconds, and can be from 0 to 32,767, but must be less than the value specified in **Collection Interval**. The time at which the collected data is recorded matches the collection interval time, regardless of the offset value.
		Over 10 Sec Collection Time	This property is only displayed if collection of historical data takes precedence over the display processing of real-time reports (if the functionality that prioritizes the collection of historical data is enabled).[#2] Whether record collection might require 10 seconds or more is indicated by **Yes** or **No**. • **Yes**: Might require 10 seconds or more. • **No**: Does not require 10 seconds. The value of this property cannot be changed.
		Realtime Report Data Collection Mode	Specifies the display mode of real-time reports. • 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.

Directory name		Property name	Description
Interval Records	*record-ID*[#1]	LOGIF	Specifies conditions for saving the record to the database. Only records that satisfy the conditions are saved. This property shows the conditional expression (character string) specified in the LOGIF Expression Editor window, which opens when the user clicks **LOGIF** in the bottom frame of the **Properties** pane in the PFM - Web Console's **Services** page.
Log Records		--	Stores the properties of a record of PL record type. This folder is not used because PFM - Agent for Oracle does not use this record.
Restart Configurations		--	Specifies the conditions for automatically restarting the PFM services. The condition can be set when the version of PFM - Manager or PFM - Base is 08-50 or later. For details about the PFM service Auto-restart functionality, see the chapter on Performance Management functionality in the *JP1/Performance Management Planning and Configuration Guide*.
		Restart when Abnormal Status	Specifies whether to automatically restart a service when the Status Server service cannot obtain the status of the Action Handler service, Agent Collector service, and Agent Store service in a normal state.
		Restart when Single Service Running	Specifies whether to automatically restart a service when only either the Agent Collector service or the Agent Store service is running.
Restart Configurations	Action Handler	Auto Restart	Specifies whether to use automatic restart for the Action Handler service.
		Auto Restart - Interval (Minute)	Specifies the interval for checking the operating status of a service when automatic restart is used. You can specify a value from 1 through 1,440 (minutes).
		Auto Restart - Repeat Limit	Specifies the number of consecutive times restart is attempted when automatic restart is used. You can specify a value from 1 through 10.
		Scheduled Restart	Select **Yes** or **No** from the list items to specify whether to use the normal restart procedure for the Action Handler service.
		Scheduled Restart - Interval	Specifies the restart interval when the normal restart procedure is used. You can specify an integer value from 1 through 1000.
		Scheduled Restart - Interval Unit	Selects **Hour**, **Day**, **Week**, or **Month** from the list items to specify the unit for the restart interval when the normal restart procedure is used.
		Scheduled Restart - Origin - Year	Specifies the year when restart is performed. You can specify an integer value from 1971 through 2035.
		Scheduled Restart - Origin - Month	Specifies the month when restart is performed. You can specify an integer value from 1 through 12.
		Scheduled Restart - Origin - Day	Specifies the day when restart is performed. You can specify an integer value from 1 through 31.
		Scheduled Restart - Origin - Hour	Specifies the time (hour) when restart is performed. You can specify an integer value from 0 through 23.

Directory name		Property name	Description
Restart Configurations	Action Handler	Scheduled Restart - Origin - Minute	Specifies the time (minute) when restart is performed. You can specify an integer value from 0 through 59.
	Agent Collector	Auto Restart	Specifies whether to use automatic restart for the Agent Collector service.
		Auto Restart - Interval (Minute)	Specifies the interval for checking the operating status of a service when automatic restart is used. You can specify a value from 1 through 1,440 (minutes).
		Auto Restart - Repeat Limit	Specifies the number of consecutive times restart is attempted when automatic restart is used. You can specify an integer value from 1 through 10.
		Scheduled Restart	Select **Yes** or **No** from the list items to specify whether to use the normal restart procedure for the Agent Collector service.
		Scheduled Restart - Interval	Specifies the restart interval when the normal restart procedure is used. You can specify an integer value from 1 through 1,000.
		Scheduled Restart - Interval Unit	Selects **Hour**, **Day**, **Week**, or **Month** from the list items to specify the unit for the restart interval when the normal restart procedure is used.
		Scheduled Restart - Origin - Year	Specifies the year when restart is performed. You can specify an integer value from 1971 through 2035.
		Scheduled Restart - Origin - Month	Specifies the month when restart is performed. You can specify an integer value from 1 through 12.
		Scheduled Restart - Origin - Day	Specifies the day when restart is performed. You can specify an integer value from 1 through 31.
		Scheduled Restart - Origin - Hour	Specifies the time (hour) when restart is performed. You can specify an integer value from 0 through 23.
		Scheduled Restart - Origin - Minute	Specifies the time (minute) when restart is performed. You can specify an integer value from 0 through 59.
	Agent Store	Auto Restart	Specifies whether to use automatic restart for the Agent Store service.
		Auto Restart - Interval (Minute)	Specifies the interval for checking the operating status of a service when automatic restart is used. You can specify a value from 1 through 1,440 (minutes).
		Auto Restart - Repeat Limit	Specifies the number of consecutive times restart is attempted when automatic restart is used. You can specify a value from 1 through 10.
		Scheduled Restart	Select **Yes** or **No** from the list items to specify whether to use the normal restart procedure for the Agent Store service.
		Scheduled Restart - Interval	Specifies the restart interval when the normal restart procedure is used. You can specify an integer value from 1 through 1000.

Directory name		Property name	Description
Restart Configurations	Agent Store	Scheduled Restart - Interval Unit	Selects **Hour**, **Day**, **Week**, or **Month** from the list items to specify the unit for the restart interval when the normal restart procedure is used.
		Scheduled Restart - Origin - Year	Specifies the year when restart is performed. You can specify an integer value from 1971 through 2035.
		Scheduled Restart - Origin - Month	Specifies the month when restart is performed. You can specify an integer value from 1 through 12.
		Scheduled Restart - Origin - Day	Specifies the day when restart is performed. You can specify an integer value from 1 through 31.
		Scheduled Restart - Origin - Hour	Specifies the time (hour) when restart is performed. You can specify an integer value from 0 through 23.
		Scheduled Restart - Origin - Minute	Specifies the time (minute) when restart is performed. You can specify an integer value from 0 through 59.
ITSLM Connection Configuration		--	Displays information about the linked JP1/SLM - Manager.
ITSLM Connection Configuration	ITSLM Connection	--	Displays information about the connection-target JP1/SLM - Manager.
		ITSLM Host	Displays the host name of the connected JP1/SLM - Manager. If a connection with JP1/SLM - Manager has not been established, this property is not displayed.
		ITSLM Port	Displays the port number of the connected JP1/SLM - Manager. If a connection with JP1/SLM - Manager has not been established, this property is not displayed.
	MANAGE ITSLM CONNECTION	--	Specifies whether to end the connection with JP1/SLM - Manager.
		DISCONNECT ITSLM CONNECTION	Selects from the list items the JP1/SLM - Manager host name that is to be disconnected. If **(empty string)** is selected from the list items, nothing is done. If a connection with JP1/SLM - Manager has not been established, only **(empty string)** is displayed in the list items.
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 for settings specific to PFM - Agent for Oracle.
Agent Configuration	Agent	--	Displays an overview of the Agent Collector service. The properties stored in this folder cannot be changed.
		Product	Displays product ID O.
		Instance	Displays the instance name specified with the `jpcconf inst setup` command.
		Description	Displays the description of the service.

Directory name		Property name	Description
Agent Configuration	Agent	Version	Displays the Oracle version being monitored.
Agent Configuration	jpcOcollect	--	Displays the properties of the data collection program of the Agent Collector service.
		ORACLE_HOME	Displays the value of oracle_home that was specified when the instance environment was set up. This property cannot be changed.
		ORACLE_SID	Displays the value of oracle_sid that was specified when the instance environment was set up. This property cannot be changed.
		USER	Displays the value of oracle_user that was specified when the instance environment was set up. This property cannot be changed.
		HOST	Displays the name of the physical host on which the monitored Oracle database is installed. This property cannot be changed.
		SQLNET	Displays **Y** (connect) or **N** (do not connect) to indicate whether the Oracle database is to be connected via SQL*NET. This property cannot be changed.
		CONNECT_DESCRIPTOR	Displays the alias to be used when the Oracle database is to be connected via SQL*NET (when **Y** is displayed in the SQLNET property). This property indicates the value that was specified for the net_service_name property when an instance environment was set. This property cannot be changed.
		NUMBER_EXTENTS	Displays the threshold value for the Overextended (OVEREXTENDED) field for the following records: • Database (PD_PDDB) • Database Interval (PI_PIDB) • Segment Detail (PD_PDSM) • Tablespace Fragmentation (PD_PDTF) This property cannot be changed.
		PCT_MAX_EXTENTS	Displays the threshold value for the High Max Extents (HIGH_MAX_EXTENTS) field for the following records: • Database (PD_PDDB) • Database Interval (PI_PIDB) • Tablespace Fragmentation (PD_PDTF) This property cannot be changed.
		TOPN_SQL	Specifies the maximum number of SQL statements to be displayed when they are displayed in descending order (largest values first) of the values in the Disk Reads/Exec fields or Logical Reads/Exec fields, which are in the SQL Text - Performance Bases (PD_PDES) records. This property cannot be changed.
		LOGICAL_READS_PER_EXECUTION	Displays the threshold value for the number of logical read operations executed by SQL statements that is stored in the SQL Text - Performance Based (PD_PDES) record. PFM- Agent for Oracle collects performanc e data only on SQL statements for which

Directory name		Property name	Description
Agent Configuration	jpcOcollect	LOGICAL_READS_PER_EXECUTION	the value of the Logical Reads/Exec field is greater than this threshold value. This property cannot be changed.
		DISK_READS_PER_EXECUTION	Displays the threshold value for the number of disk read operations executed by SQL statements that is stored in the SQL Text - Performance Based (PD_PDES) record. PFM - Agent for Oracle collects performance data only on SQL statements for which the value of the Disk Reads/Exec field is greater than this threshold value. This property cannot be changed.
		LISTENER_HOME	Displays the value of listener_home specified when the instance environment was set up. This property cannot be changed.
		LISTENER_NAME	Displays the value of listener_name specified when the instance environment was set up. This property cannot be changed.
		RETRY_TIME	Displays the number of seconds that reconnection is attempted when an authentication error occurs while connection is being established with Oracle. In UNIX, the value is always set to 0. This property cannot be changed.
		LOG_PATH	Displays the value of log_path specified when the instance environment was set up. This property can be changed.[#3]
		LOG_SIZE	Displays the value of log_size specified when the instance environment was set up. This property can be changed.[#3]
		TIMEOUT	Displays the value of timeout specified when the instance environment was set up. If this is updated with a value of 1 to 9 specified, it is not changed. This property can be changed.[#3]
		SQL_OPTION	Displays the value of sql_option specified when the instance environment was set up. This property can be changed.[#3]
		NUMERIC_10	Displays the value of numeric_10 specified when the instance environment was set up. This property can be changed.[#3]
		STARTUP_ALWAYS	Displays the value of startup_always specified when the instance environment was created. This property can be changed.[#3]
		LOCALTEMP_OPTION	Displays the value of localtemp_option that you specify when you set up the instance environment. This property can be changed.[#3]
		UNDOSPACE_OPTION	Displays the value of undospace_option specified when the instance environment was created. This property can be changed.[#3]
		NLS_LANG	Displays the value of nls_lang specified when the instance environment was created. This property cannot be changed.

Legend:

--: Not applicable.

#1

The dictionary name is shown as the record ID excluding the database ID. For details about the record ID of each type of record, see *6. Records*.

#2

For details, see the chapter on troubleshooting in the *JP1/Performance Management User's Guide*.

#3

To reflect updated values, restart the Agent Collector service.

G. List of Files and Directories

This appendix lists the files and directories of PFM - Agent for Oracle for each OS.

The Performance Management installation directory for each OS is as follows.

In Windows:

system-drive\Program Files(x86)\Hitachi\jp1pc\

In UNIX:

/opt/jp1pc/

G.1 PFM - Agent for Oracle files and directories

(1) Windows

The following table lists the files and folders for the Windows version of PFM - Agent for Oracle.

Table G–1: List of files and folders for PFM - Agent for Oracle (Windows version)

Folder name	File name	Description
installation-folder\	instagto.ini	Relay file for internal processing
installation-folder\agto\	--	Base folder of PFM - Agent for Oracle
	Readme_ja.txt	README file (Japanese)
	Readme_en.txt	README file (English)
	insrules.dat	Relay file for internal processing
	PATCHLOG.TXT	Relay file for internal processing
	jpcagtras.bat	Maintenance information collection program
	jpcagtras.exe	
	VERSION.txt	Version file
	multilingual.dat	Maintenance data file
installation-folder\agto\agent\	--	Base folder of the Agent Collector service
	agtlist.ini	Relay file for internal processing
	inssetup.bat.instmpl	Relay file for internal processing
	jpcagt.ini.instmpl	Relay file for internal processing

Folder name	File name	Description
installation-folder\agto\agent\	jpcagto.exe	Service executing program of Agent Collector
	jpcOcollect_9.exe	Performance data collection program of Agent Collector service (for Oracle11g or later)
installation-folder\agto\agent\sql\	--	SQL script storage folder
	sp_drop.sql	SQL script file for deleting an Oracle Database object
	sp_inst.sql	SQL script file for registering an Oracle Database object
	mk_user.sql	Script file for creating an Oracle account used by PFM - Agent for Oracle
installation-folder\agto\agent*instance-name*\	--	Base folder of Agent collector service (for each instance)[#1]
	COSLMMI.DB	Data file for the JP1/SLM linkage setting
	COSLMMI.IDX	Index file for the data file for the JP1/SLM linkage setting
	COSLMMI.LCK	Lock file for the data file for the JP1/SLM linkage setting
	jpcagt.ini	Service startup initialization file of Agent Collector (for each instance)[#1]
	jpcagt.ini.model	Model file for the service startup initialization file of Agent Collector service (for each instance) [#1]
	jpcOcollect.exe	Performance data collection program of Agent Collector service (for each instance) [#1]
	inssetup.bat	Relay file for internal processing[#1]
	status.dat	Relay file for internal processing[#3]
	pdls_*xxxxx*_status.db[#4]	Relay file for internal processing[#3]
	pdnl_listener_status.db	Relay file for internal processing[#3]
	pdia_*xxxxx*_status.db[#4]	Relay file for internal processing[#3]
	ocilog.txt	Maintenance information file
installation-folder\agto\agent*instance-name*\log\	--	Storage folder for internal log file of the Agent Collector service (for each instance)[#1]
	agtoinf01.log agtoinf02.log	PFM- Agent for Oracle agent log file

Folder name	File name	Description
installation-folder\agto\agent*instance-name*\log\	agtoerr01.log agtoerr02.log	PFM- Agent for Oracle agent log error file
	agtoinf.lck	Relay file for internal processing[#3]
	agtoerr.lck	Relay file for internal processing[#3]
	msglog01 msglog02	Internal log file[#2]
	msgdat01 msgdat02	Internal log file[#2]
	nslog01 nslog02	Internal log file[#2]
installation-folder\agto\lib\	--	Message catalog installation folder
installation-folder\agto\store\	--	Base folder of Agent Store service
	jpcsto.ini.instmpl	Relay file for internal processing
	stolist.ini	Relay file for internal processing
	*.DAT[#4]	Data model definition file
installation-folder\agto\store*instance-name*\	--	Base folder of Agent Store service (for each instance)[#1]
	*.DB[#4]	Performance data file (for each instance)[#2]
	*.IDX[#4]	Performance data index file (for each instance)[#2]
	*.LCK[#4]	Performance data lock file (for each instance)[#2]
	jpcsto.ini	Agent Store (for each instance)[#1]
	jpcsto.ini.model	Model file for the service startup initialization file of Agent Store (for each instance)[#1]
	*.DAT[#4]	Data model definition file (for each instance)[#1]
	status.dat	Relay file for internal processing[#3]
installation-folder\agto\store*instance-name*\backup\	--	Default database backup destination folder (for each instance)[#1]
installation-folder\agto\store*instance-name*\partial\	--	Partial backup destination folder for standard databases (per instance)[#1]
installation-folder\agto\store*instance-name*\dump\	--	Default database export destination folder (for each instance)[#1]
installation-folder\agto\store*instance-name*\import\	--	Database import destination folder for standard databases (per instance)[#1]

Folder name	File name	Description
installation-folder\agto\store*instance-name*\log\	--	Storage folder for internal log file of Agent Store service (for each instance)[#1]
	msglog01 msglog02	Internal log file[#2]
	msgdat01 msgdat02	Internal log file[#2]
	nslog01 nslog02	Internal log file[#2]
installation-folder\agto\store*instance-name*\STPD\	--	PD database-specific folder
installation-folder\agto\store*instance-name*\STPI\	--	PI database-specific folder
installation-folder\auditlog\	--	Action log file output folder
	jpcaudit*n*.log[#5]	Action log file
installation-folder\patch_files\agto\	--	Folder for storing batch files (for agents)
installation-folder\setup\	--	Setup file storage folder
	jpcagtou.Z	Archive file for PFM - Agent for Oracle setup (UNIX)
	jpcagtow.EXE	Archive file for PFM - Agent for Oracle setup (Windows)

Legend:

--: Not applicable

#1

Created by execution of the `jpcconf inst setup` command.

#2

Created when the Agent Store service is started.

#3

This may be temporarily created.

#4

Any string may be used in place of *xxxxx* and `*`.

#5

n is numeric value. The number of log files can be changed in the `jpccomm.ini` file.

(2) UNIX

The following table lists the files and directories for the UNIX version of PFM - Agent for Oracle.

Notes

- If you change the file permission, the product will no longer operate normally. Do not change the file permission.
- If you change the `umask` setting, the file permission might be changed during operation. Do not change the `umask` setting.

Table G–2: List of files and directories for PFM - Agent for Oracle (UNIX version)

Directory name	File name	Permission	Description
/opt/jp1pc/	instagto.ini	644	Relay file for internal processing
/opt/jp1pc/agto/	--	755	Base directory of PFM - Agent for Oracle
	jpcagtras	555	Maintenance information collection program
	insrules.dat	640	Relay file for internal processing
	PATCHLOG.TXT	644	Relay file for internal processing
	patch_history	644	Relay file for internal processing
	multilingual.dat	444	Maintenance data file
/opt/jp1pc/agto/agent/	--	755	Base directory of Agent Collector service
	jpcagto	555	Service executing program of Agent Collector
	jpcOcollect_10	555	Performance data collection program of Agent Collector service (for Oracle10g/Oracle11g or later) #1
	jpcOcollect_11 #2	555	Performance data collection program of Agent Collector service (for Oracle11g or later) #3
	agtlist.ini	644	Relay file for internal processing
	inssetup.instmpl	755	Relay file for internal processing
	jpcagt.ini.instmpl	444	Relay file for internal processing
/opt/jp1pc/agto/agent/sql/	--	755	SQL SCRIPT STORAGE DIRECTORY
	sp_drop.sql	555	SQL script file for deleting an Oracle Database object
	sp_inst.sql	555	SQL script file for registering an Oracle Database object
	mk_user.sql	555	Script file for creating an Oracle account used by PFM - Agent for Oracle
/opt/jp1pc/agto/agent/*instance-name*/	--	755	Base directory of Agent Collector service (for each instance)#4
	COSLMMI.DB	644	Data file for the JP1/SLM linkage setting
	COSLMMI.IDX	644	Index file for the data file for the JP1/SLM linkage setting
	COSLMMI.LCK	644	Lock file for the data file for the JP1/SLM linkage setting

Directory name	File name	Permission	Description
/opt/jp1pc/agto/agent/*instance-name*/	jpcagt.ini	600	Agent Collector service startup initialization file (for each instance)[#4]
	jpcagt.ini.lck	777	Agent Collector service startup initialization file lock file (for each instance)[#5]
	jpcagt.ini.model	444	Model file for the service startup initialization file of Agent Collector (for each instance)[#4]
	inssetup	755	Relay file for internal processing
	status.dat	600	Relay file for internal processing[#6]
	FILEMAP_*xxxxx*[#7]	666	Relay file for internal processing[#8]
	pdls_*xxxxx*_status.db[#7]	--[#9]	Relay file for internal processing[#6]
	pdnl_listener_status.db	--[#9]	Relay file for internal processing[#6]
	pdia_*xxxxx*_status.db[#7]	--[#9]	Relay file for internal processing[#6]
	ocilog.txt	644	Maintenance information file
/opt/jp1pc/agto/agent/*instance-name*/log/	--	777	Storage folder for internal log file of Agent Collector service (for each instance)[#4]
	agtoinf01.log agtoinf02.log	644	PFM- Agent for Oracle agent log file[#5]
	agtoerr01.log agtoerr02.log	644	PFM- Agent for Oracle agent log error file[#5]
	agtoinf.lck	644	Relay file for internal processing[#6]
	agtoerr.lck	644	Relay file for internal processing[#6]
	msglog01 msglog02	666	Internal log file[#8]
	nslog01 nslog02	666	Internal log file[#8]
/opt/jp1pc/agto/nls/	--	755	Message catalog installation directory For details about the files and directories in this directory, see *G.2 List of files and directories in the message catalog storage directory*.
/opt/jp1pc/agto/store/	--	755	Base directory of Agent Store service
	jpcsto.ini.instmpl	444	Relay file for internal processing
	stolist.ini	644	Relay file for internal processing
	*.DAT[#7]	444	Data model definition file

Directory name	File name	Permission	Description
/opt/jp1pc/agto/store/ *instance-name*/	--	755	Base directory of Agent Store service (for each instance)[#4]
	*.DB[#7]	644	Performance data file (for each instance)[#8]
	*.IDX[#7]	644	Performance data index file (for each instance)[#8]
	*.LCK[#7]	666	Performance data lock file (for each instance)[#8]
	jpcsto.ini	644	Agent Store service startup initialization file (for each instance)[#4]
	jpcsto.ini.model	444	Model file for the service startup initialization file of Agent Store service (for each instance)[#4]
	*.DAT[#7]	444	Data model definition file (for each instance)[#4]
	status.dat	600	Relay file for internal processing[#6]
/opt/jp1pc/agto/store/*instance-name*/backup/	--	755	Default database backup destination directory (for each instance)[#4]
/opt/jp1pc/agto/store/*instance-name*/partial/	--	755	Partial backup destination directory for standard databases (per instance)[#4]
/opt/jp1pc/agto/store/ *instance-name*/dump/	--	777	Default database export destination folder (for each instance)[#4]
/opt/jp1pc/agto/store/*instance-name*/import/	--	755	Database import destination directory for standard databases (per instance)[#4]
/opt/jp1pc/agto/store/*instance-name*/STPD/	--	755	PD database-specific directory
/opt/jp1pc/agto/store/*instance-name*/STPI/	--	755	PI database-specific directory
/opt/jp1pc/agto/store/ *instance-name*/log/	--	777	Storage directory for internal log file of Agent Store service (for each instance)[#4]
	msglog01 msglog02	666	Internal log file[#8]
	msgdat01 msgdat02	666	Internal log file[#8]
	nslog01 nslog02	666	Internal log file[#8]
/opt/jp1pc/patch_files/agto	--	755	Folder for storing batch files (for agents)
/opt/jp1pc/auditlog/	--	700	Action log file output directory
	jpcaudit*n*.log[#10]	600	Action log file
/opt/jp1pc/setup/	--	755	Setup file storage directory

Directory name	File name	Permission	Description
/opt/jp1pc/setup/	jpcagtou.Z	444	PFM - Agent for Oracle setup archive file (UNIX)
	jpcagtow.EXE	444	PFM - Agent for Oracle setup archive file (Windows)
/opt/jp1pc/tools/log/	agto_sp_inst.log	644[#11]	Internal log file[#12]
	agto_mk_user.log	644[#11]	Internal log file[#13]

Legend:

--: Not applicable

#1

This includes the HP-UX program file for Oracle10g and the Linux program file and AIX program file for Oracle11g or later.

#2

This file exist in HP-UX and Solaris

#3

This process runs if you monitor Oracle 11g or later in HP-UX and Solaris

#4

Created by execution of the `jpcconf inst setup` command.

#5

Do not change or delete this file. This file is used internally by PFM - Agent.

#6

This may be temporarily created.

#7

Any string may be used in place of *xxxxx* and `*`.

#8

Created when the Agent Store service is started.

#9

No file permissions are specified, because this is a temporary file.

#10

n is numeric value. The number of log files can be changed in the `jpccomm.ini` file.

#11

This depends on the application executing the script (such as Sql*Plus).

#12

This is created when the `sp_inst.sql` script is executed.

#13

This is created when the `mk_user.sql` script is executed.

G.2 List of files and directories in the message catalog storage directory

The following explains the configuration of the files and directories in the message catalog storage directory (`/opt/jp1pc/agto/nls/`).

(1) HP-UX

The following table lists the files and directories in the message catalog storage directory for the HP-UX version of PFM - Agent for Oracle.

Table G–3: List of files and directories in the message catalog storage directory for PFM - Agent for Oracle (HP-UX version)

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/C/	755	LANG=C message catalog storage directory
/opt/jp1pc/agto/nls/C/jpcagtomsg.cat	444	LANG=C message catalog
/opt/jp1pc/agto/nls/ja_JP.SJIS/	755	SJIS message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.SJIS/jpcagtomsg.cat	444	SJIS message catalog
/opt/jp1pc/agto/nls/ja_JP.eucJP/	755	EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.eucJP/jpcagtomsg.cat	444	EUC message catalog
/opt/jp1pc/agto/nls/japanese	777	Symbolic link to SJIS message catalog storage directory
/opt/jp1pc/agto/nls/japanese.euc	777	Symbolic link to EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.UTF-8/	755	UTF-8 message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.UTF-8/jpcagtomsg.cat	444	UTF-8 message catalog

(2) AIX

The following table lists the files and directories in the message catalog storage directory for the AIX version of PFM - Agent for Oracle.

Table G–4: List of files and directories in the message catalog storage directory for PFM - Agent for Oracle (AIX version)

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/C/	755	LANG=C message catalog storage directory
/opt/jp1pc/agto/nls/C/jpcagtomsg.cat	444	LANG=C message catalog
/opt/jp1pc/agto/nls/Ja_JP/	755	SJIS message catalog storage directory
/opt/jp1pc/agto/nls/Ja_JP/jpcagtomsg.cat	444	SJIS message catalog
/opt/jp1pc/agto/nls/Ja_JP.IBM-932	777	Symbolic link to SJIS message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP/	755	EUC message catalog storage directory

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/ja_JP/jpcagtomsg.cat	444	EUC message catalog
/opt/jp1pc/agto/nls/ja_JP.IBM-eucJP	777	Symbolic link to EUC message catalog storage directory
/opt/jp1pc/agto/nls/JA_JP/	755	UTF-8 message catalog storage directory
/opt/jp1pc/agto/nls/JA_JP/jpcagtomsg.cat	444	UTF-8 message catalog
/opt/jp1pc/agto/nls/JA_JP.UTF-8	777	Symbolic link to UTF-8 message catalog storage directory

(3) Solaris

The following table lists the files and directories in the message catalog storage directory for the Solaris version of PFM - Agent for Oracle.

Table G–5: List of files and directories in the message catalog storage directory for PFM - Agent for Oracle (Solaris version)

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/C/	755	LANG=C message catalog storage directory
/opt/jp1pc/agto/nls/C/jpcagtomsg.cat	444	LANG=C message catalog
/opt/jp1pc/agto/nls/ja/	755	EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja/jpcagtomsg.cat	444	EUC message catalog
/opt/jp1pc/agto/nls/ja_JP.eucJP/	777	Symbolic link to EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.PCK/	755	SJIS message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.PCK/jpcagtomsg.cat	444	SJIS message catalog
/opt/jp1pc/agto/nls/japanese	777	Symbolic link to EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.UTF-8/	755	UTF-8 message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.UTF-8/jpcagtomsg.cat	444	UTF-8 message catalog

(4) Linux

The following table lists the files and directories in the message catalog storage directory for the Linux version of PFM - Agent for Oracle.

Table G–6: List of files and directories in the message catalog storage directory for PFM - Agent for Oracle (Linux version)

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/C/	755	LANG=C message catalog storage directory
/opt/jp1pc/agto/nls/C/jpcagtomsg.cat	444	LANG=C message catalog

File or directory name	Permission	Description
/opt/jp1pc/agto/nls/ja_JP.UTF-8/	755	UTF-8 message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.UTF-8/jpcagtomsg.cat	444	UTF-8 message catalog
/opt/jp1pc/agto/nls/ja_JP.eucJP/	755	EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.eucJP/jpcagtomsg.cat	444	EUC message catalog
/opt/jp1pc/agto/nls/ja_JP.ujis	777	Symbolic link to EUC message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.utf8	777	Symbolic link to UTF-8 message catalog storage directory
/opt/jp1pc/agto/nls/ja_JP.SJIS	755	Message catalog storage directory for SJIS
/opt/jp1pc/agto/nls/ja_JP.sjis	777	Symbolic link to the message catalog storage directory for SJIS
/opt/jp1pc/agto/nls/ja_JP.SJIS/jpcagtomsg.cat	444	Message catalog for SJIS

H. Migration Procedure and Notes on Migration

To upgrade PFM - Agent for Oracle, you need to perform overwrite installation on PFM - Agent for Oracle. For details about the installation procedure, see the following chapters.

In Windows:

See *2. Installation and setup (Windows)*.

In UNIX:

See *3. Installation and setup (UNIX)*.

For details about notes on upgrading the versions of Performance Management programs, see the section describing the notes on version upgrading in the chapter and appendix that explain installation and setup in the *JP1/Performance Management Planning and Configuration Guide*.

This appendix shows the notes on upgrading the version of PFM - Agent for Oracle.

- Do not uninstall the old version of PFM - Agent for Oracle during upgrading. If you uninstall it, performance data created in the old version is deleted and will no longer be available in the new version.
- When you perform overwrite installation on a PFM - Agent for Oracle program, the following information is updated automatically:
  - Store database files of Agent Store service
  - `ini` file
  - Instance environment of PFM - Agent for Oracle
- In Performance Management programs of version 08-00 or newer, the locations of the Store executing programs (jpcsto.exe and stpqlpr.exe) have been changed. When PFM - Manager and PFM - Agent are upgraded to version 08-00 or newer, the Store executing programs at the older locations are deleted.
- After you have installed PFM - Agent for Oracle as an overwrite installation, be careful when you perform the setup procedures described in *Create an Oracle account to be used in PFM - Agent for Oracle*. You need to perform that procedure only when the Oracle account being used has been changed or deleted.
- When you use Windows or Linux, for monitoring of the Oracle database, the 32-bit Oracle Client has been the required product in PFM - Agent for Oracle 10-50 or earlier. However, in PFM - Agent for Oracle 11-00 or later, the Oracle Client 64-bit library is used for monitoring.

  To upgrade PFM - Agent for Oracle 10-50 or earlier to 11-00 or later, instance information is subject to change, and must therefore be updated before starting the PFM - Agent for Oracle service. The following table shows the instance information for which you need to specify the environment settings, and the settings themselves.

  Table H–1: Items in instance information needed to update when upgrading the PFM - Agent for Oracle version

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
`oracle_home`	Specify the Oracle home folder for Oracle Client 32-bit.	Specify the Oracle home folder for Oracle Database.
`oracle_version`	Specify the version of Oracle Client 32-bit.	Specify the version of Oracle Database. When monitoring target is Oracle Database 12c or later (about Oracle Database 18c), `12` is set.

Item	PFM - Agent for Oracle	
	10-50 or earlier	11-00 or later
`sqlnet`	Specify `Y`	Specify the settings according to the explanation in *Table 2-14* (for Windows) or *Table 3-18* (for UNIX).
`net_service_name`	Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Client 32-bit.	If you specify `Y` for `sqlnet`, specify this item. Specify the net service name that can be used to connect to the monitoring-target Oracle Database specified in Oracle Database.

Note

Note

- PFM - Agent for Oracle 11-00 or later uses the Oracle Client 64-bit library included with the Oracle database, so you do not need to install the 64-bit Oracle Client.
- Oracle Client 32-bit, which was used for monitoring in version 10-50 or earlier, is not needed and can therefore be uninstalled if it is not used by any program other than PFM - Agent for Oracle.

I. Precautions Regarding Permissions

The permissions needed for using PFM - Agent for Oracle differ depending on what is being used.

The following lists the permissions required for each operation target.

I.1 When the sp_inst.sql script is executed

The following table lists the system privileges needed for the Oracle account executing the `sp_inst.sql` script.

Table I–1: Oracle account privileges needed to execute the sp_inst.sql script

Privileges required	Description
`CREATE SESSION`	Required when a session is established for the monitored Oracle Database.
`CREATE TABLE`	Required when a table[#1] required to monitor the monitored Oracle Database is registered.
`CREATE PROCEDURE`	Required when a procedure[#1] required to monitor the monitored Oracle Database is registered.
`SELECT ANY DICTIONARY`	Required when information[#1] required to monitor the monitored Oracle Database is registered.
`UNLIMITED TABLESPACE`	Required when information[#1] required to monitor the monitored Oracle Database is registered[#2].

#1

See the following tables.

In Windows

In *2.1.4(3)(b) Registering objects in the Oracle Database*, see *Table 2-10*

In UNIX

In *3.1.4(4)(b) Registering objects in the Oracle Database*, see *Table 3-14*

For details about the `CREATE TABLE` privilege, see *Table 2-10* or *Table 3-14*. For details about the `CREATE PROCEDURE` privilege, see the corresponding package.

#2

This privilege is not needed when the assignment limit that allows writing to the default tablespace of the account used for monitoring has been set.

I.2 When the sp_drop.sql script is executed

The following table lists the system privileges needed for the Oracle account executing the `sp_drop.sql` script.

Table I–2: Oracle account privileges needed to execute the sp_drop.sql script

Privileges required	Description
`CREATE SESSION`	Required when a session is established for the monitored Oracle Database.

I.3 When the mk_user.sql script is executed

The following table lists the system privileges needed for the Oracle account executing the `mk_user.sql` script.

Table I–3: Oracle account privileges needed to execute the mk_user.sql script

Privileges required	Description
`CREATE SESSION`	Required when a session is established for the monitored Oracle Database.
`CREATE USER`	Required when a user is created for the monitored Oracle Database.
`GRANT ANY PRIVILEGE`	Required when system privileges are granted for the monitored Oracle Database.

I.4 When performance information is collected

The following table lists the Oracle account system privileges needed for PFM - Agent for Oracle to collect performance information.

Table I–4: Oracle account privileges needed by PFM - Agent for Oracle to collect performance information

Privileges required	Description
`CREATE SESSION`	Required when a session is established for the monitored Oracle Database.
`SELECT ANY DICTIONARY`	Required to obtain performance information from the monitored Oracle Database.
`UNLIMITED TABLESPACE`	Required to obtain the Explain Plan field for the `PD_PDSQ` records.#
`SELECT ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.
`INSERT ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.
`UPDATE ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.
`DELETE ANY TABLE`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.
`CREATE ANY INDEX`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.
`ALTER ANY INDEX`	Required to obtain the Explain Plan (`EXPLAIN_PLAN`) field for SQL Text (`PD_PDSQ`) records.

#

This privilege is not needed when the assignment limit that allows writing to the default tablespace of the account used for monitoring has been set.

The Oracle user account needs all of the privileges listed in Table I-4 to obtain `PD_PDSQ` records.

When `PD_PDSQ` records are not obtained, the `CREATE SESSION` and `SELECT ANY DICTIONARY` privileges are needed for the Oracle user account.

J. Version Compatibility

For PFM - Agent, there are different data model versions as well as products.

The data model version remains unchanged. Because data models are upward-compatible, the report and alarm definitions created by an older version are available in a newer data model version.

The table below shows the correspondence between the versions of PFM - Agent for Oracle.

Table J–1: Correspondence between PFM - Agent for Oracle versions

PFM - Agent for Oracle version	Data model version	Version of the monitoring template alarm table
06-70	3.0	6.70
07-00	4.0	7.00
07-10	4.0	7.10
07-50	4.0	7.50
08-00	5.0	8.00
08-10	6.0	8.10
08-50	7.0	8.50
09-00	7.0	09.00
09-10	7.0	09.00
10-00	8.0	10.00
10-50	9.0	10.50
11-00	9.0	10.50
12-00	9.1	12.00
12-10	9.2	12.10

For details about version compatibility, see the information in the appendixes of the *JP1/Performance Management Planning and Configuration Guide*.

K. Outputting Action Log Information

Action log information of Performance Management is output in conjunction with the alarm function related to an exceeded threshold for information, such as system load.

For example, when PFM services start or stop or when the connection status with PFM - Manager changes, action logs are output.

Action logs are output if the version of PFM - Manager or PFM - Base is 08-10 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 logs is specified in the `jpccomm.ini` file. This appendix describes the contents of the action log output by PFM - Agent and PFM - Base, and how to specify the setting for outputting the action log.

K.1 Event types output to the action log

The following table shows the event types output to the action log and the times at which PFM - Agent and PFM - Base output information to the action log. The event types are the identifiers used within the action log to classify the events output to the action log.

Table K–1: Event types 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.

K.2 Format for saving the action log files

This section explains the format for saving the action log files.

Action log information is output to a specified file (current output file). When the current output file becomes full, the action log information in that file is saved to another file (shift file). The procedure for switching the file for storing action log information is as follows:

1. Action log information is output sequentially to the current output file `jpcaudit.log`.
2. When the current output file becomes full, the action log information is saved in a shift file. The name of a shift file is the current output file name suffixed with a number. Each time the current output file becomes full, each shift file

is renamed by incrementing the 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 information is written.

Whether action log information 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 *K.4 Settings for outputting action logs*.

K.3 Action log output format

Information related to audit events is output to the Performance Management action log. One action log information file is output for one host (physical host and logical host). The action log file is output to either of the following hosts:

- When a service is executed: The file is output to the host on which the service runs.
- When a command is executed: The file is output to 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

On physical hosts

- In Windows
  installation-folder`\auditlog\`
- In UNIX
  `/opt/jp1pc/auditlog/`

On logical hosts

- In Windows
  environment-folder`\jp1pc\auditlog\`
- In UNIX
  environment-directory`/jp1pc/auditlog/`

The action log output destination can be changed in the `jpccomm.ini` file. For details about how to specify the `jpccomm.ini` file, see *K.4 Settings for outputting action logs*.

(3) Output items

There are two types of output items:

- Common output item
  An item that is always output by all JP1 products that output action logs
- Fixed output item
  An item that is optionally output by a JP1 product that outputs action logs

(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 K–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.sss*TZD[#]	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

No.	Output item		Value	Explanation
	Item name	Output attribute name		
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 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 K–3: Fixed output items in action logs

No.	Output item		Value	Explanation
	Item name	Output attribute name		
1	Object information	`obj`	• *PFM - Agent for Oracle-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`	Information about the action that caused the event

No.	Output item		Value	Explanation
	Item name	Output attribute name		
2	Action information	op	• 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
6	Free description	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 is output. The following describes the message ID and output information for each case.

■ A PFM service is started or stopped (StartStop)

- Output host: The host on which the service is running
- Output component: The service that was 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 is started or terminated (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

Note:

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, a PFM - Agent service starts in stand-alone mode. In this mode, only part of the service's functionality, such as the collection of operating information, is enabled. At the same time, KAVE03002-I is output to indicate that the service has started in stand-alone mode. 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 An event was resent to PFM - Manager: KAVE03301-I

Note:

1. No fixed output items are output.
2. When sending of an event to PFM - Manager fails, Agent Store service starts queuing events. The maximum capacity of the queue is 3 events. KAVE03300-I is output when sending of an event to PFM - Manager fails and queuing starts. After the connection with PFM - Manager restores and the queued events are resent, KAVE03301-I is output. From this sequence of the log, you can judge that the period when an event-sending to PFM - Manager is not real time is specifiable.
3. Agent Collector service normally sends events to PFM - Manager through Agent Store service. Agent Collector service directly sends events to PFM - Manager only when Agent Store Service stops for any reason. When Agent Collector Service fails to send events directly to PFM - Manager, KAVE03300-I is output. In this case, KAVE03301-I is no output because the queuing does not start. From this sequence of the log, you can judge that there are events that are not 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. E-mail was send successfully: KAVE03502-I. Sending of e-mail failed: KAVE03503-W
Free description	msg	Command execution: cmd=*executed-command-line*.

Item name	Attribute name	Value
Free description	msg	E-mail sending: mailto=*destination-email-address*.

Note:

KAVE03500-I is output when the command execution process is successfully created. After KAVE03500-I is output, whether the command is successfully executed or not and the execution result are not output to the action log.

(4) Output example

The following is an example of action log output.

```
CALFHM 1.0, seqnum=1, msgid=KAVE03000-I, date=2007-01-18T22:46:49.682+09:00,
progid=JP1PFM, compid=OA1host01, pid=2076,
ocp:host=host01, ctgry=StartStop, result=Occurrence,
subj:pid=2076,op=Start,
```

K.4 Settings for outputting action logs

The settings for outputting action logs are defined in the jpccomm.ini file. If no settings are specified, no action logs are output. The following describes the settings required to output the action logs, and how to specify the settings.

(1) Setting procedure

To specify the settings for outputting action log information:

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

In Windows

installation-folder

In UNIX

/opt/jp1pc/

(b) Format

In the jpccomm.ini file, define the following information:

- Whether or not to output action log information
- 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 specify.

Table K–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 or not to output action log information. You must specify this item. • Initial value 0 (Information not output) • Specifiable value 0 (Information not output) or 1 (Information output) If any other value is specified, an error message is output and action log information will not be output.
3	`Action Log Dir`#	Specify the action log output destination. In a logical host environment, specify a directory on the shared disk. If the directory you specify is not on the shared disk, Performance Management will output action logs to each physical host that forms the basis for the logical host. 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 information will not be output. • Initial value None set • Default value used when no specification is made On physical hosts: Windows: *installation-folder*`\auditlog\` UNIX: `/opt/jp1pc/auditlog/` On logical hosts: Windows: *environment-folder*`\jp1pc\auditlog\` UNIX: *environment-directory*`/jp1pc/auditlog/` • Specifiable value A character string of 1 to 185 bytes
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 file 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 an 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

No.	Item	Explanation
5	Action Log Size	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 an integer nearest the specified value in the range from 512 to 2096128 is set.

#

When you use the `jpcconf ha setup` command to set up a logical host after you configure action log output on the physical host, the settings in the jpccomm.ini file of the physical host apply to the logical host. When using Performance Management on a logical host and a physical host at the same time, make sure that both hosts do not output action logs to the same directory.

L. Linking with JP1/SLM

The capability of PFM - Agent for Oracle to monitor operating status can be enhanced through linkage with JP1/SLM.

PFM - Agent for Oracle provides default monitoring items specific to JP1/SLM for PFM - Manager to enable monitoring on JP1/SLM.

The default monitoring items provided by PFM - Agent for Oracle for PFM - Manager are described in the following table.

In addition, PFM - Agent for Oracle collects records corresponding to the value specified as the key for multi-instance records. For the corresponding collection key, see the collection result for each record.

Table L–1: Default monitoring items provided by PFM - Agent for Oracle for PFM - Manager

Display name in JP1/SLM	Description	Record (Record ID)	Key (PFM-Manager name)	Field name
Cache miss rate	Monitors the percentage of data requests issued due to cache misses.	System Stat Summary Interval (PI)	--	DICTIONARY_CACHE_GET_MISSES_PERCENTAGE
Disk sorts execution rate	Monitors the percentage of all sorts executed on the disk when memory or disk I/O is used.	System Stat Summary Interval (PI)	--	SORT_OVERFLOW_PERCENTAGE
Buffer cache usage rate	Monitors buffer and cache usage.	System Stat Summary Interval (PI)	--	CACHE_HIT_PERCENTAGE
Buffer busy rate	Monitors the percentage of rollback or data conflicts in a database.	System Stat Summary Interval (PI)	--	BUFFER_BUSY_WAIT_PERCENTAGE
Library cache miss rate	Monitors the percentage of the objects in the library cache that are reloaded.	System Stat Summary Interval (PI)	--	LIBRARY_CACHE_MISS_PERCENTAGE

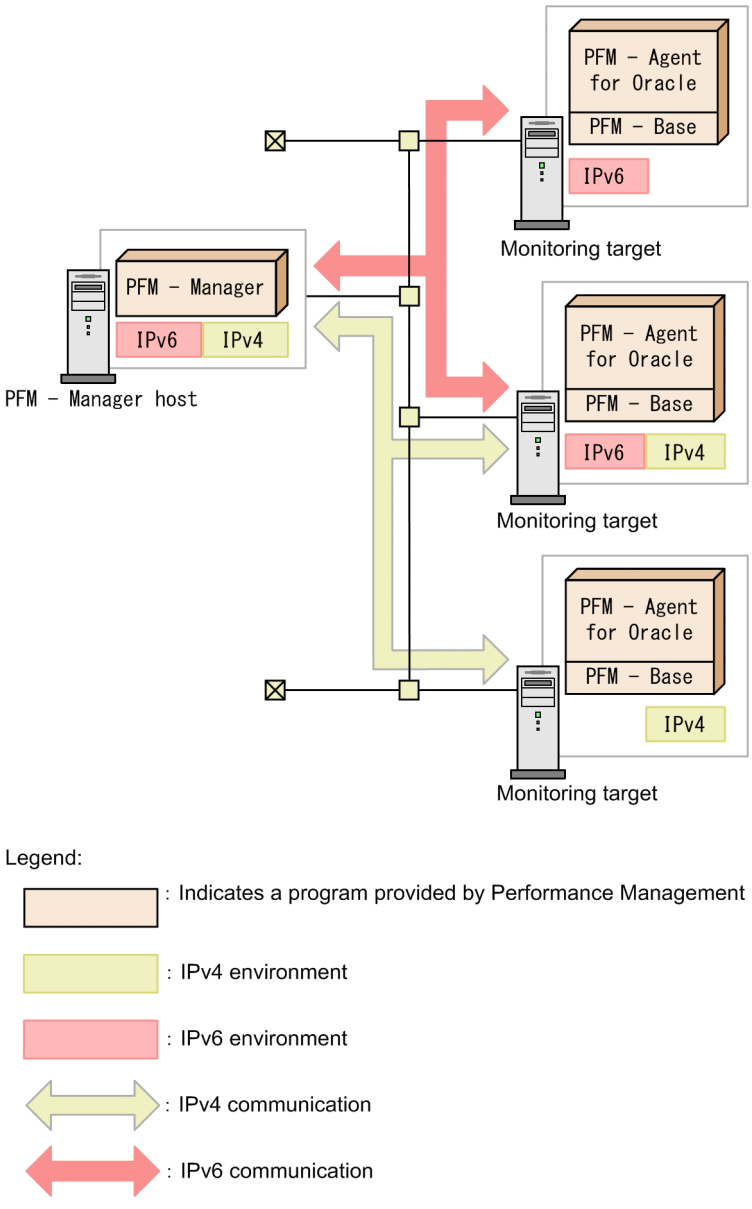
To provide the default monitoring items for PFM - Manager, you need to copy the setup file and execute the setup command. For details, see *2.1.4(1) Register PFM - Agent for Oracle (for Windows)* or *3.1.4(2) Register PFM - Agent for Oracle (for UNIX)*.

M. About Communication in IPv4 Environments and IPv6 Environments

Performance Management supports IPv6 environments in addition to IPv4 environments as a network configuration. Therefore, Performance Management can operate even in a network configuration in which both an IPv4 environment and IPv6 environment are used.

Note that this explanation applies only when the OS of a host on which PFM - Agent for Oracle and PFM - Manager are installed is Windows or Linux.

Figure M–1: Scope of communication when an IPv4 environment and an IPv6 environment are used



To enable communication in an IPv6 environment, you must execute the `jpcconf ipv6 enable` command. For details about the `jpcconf ipv6 enable` command, see the chapter that describes commands in the manual *JP1/Performance Management Reference*. For the conditions and timing for executing the `jpcconf ipv6 enable` command, see the chapter that describes network configuring examples in an IPv6 environment in the *JP1/Performance Management Planning and Configuration Guide*.

N. Version Revisions

This appendix shows the changes that have been made to each version of the manual.

N.1 Revisions in 12-10

- A change was made so that the multitenant container database (CDB) configuration allows all pluggable databases to be monitored from the root container (`CDB$ROOT`) to which they belong. In response to this, the following reports were added so that PDB information inside the CDB can be monitored:
  - CDB Audit Size Status
  - CDB Data File Size Status
  - CDB Diagnostics Size Status
  - CDB Status
  - CDB Tablespace Status
- Changes were made to the procedure for registering PFM - Agent Option for Oracle.
- The version of the data model has been changed from 9.1 to 9.2, and the version of the alarm table has been changed from 12.00 to 12.10.
- The following report was added to monitor the fast recovery area:
  - Fast Recovery Area Status
- The following records were added to collect PDB information from the CDB:
  - CDB Status (PD_CDCS)
  - CDB Tablespace (PD_CDTS)
- A field for monitoring free space available for the disk group was added to the following record:
  - ASM Disk Group Interval (PI_PIDG)
- A field for monitoring the fast recovery area was added to the following record:
  - Collection Instance 2 (PD_PCI)

N.2 Revisions in 12-00

- Windows Server 2019 was added to the list of OSs that can run PFM - Agent for Oracle.
- Oracle Database 19c was added to the list of monitoring target programs.
- For the following OSs were deleted from the OSs on which PFM - Agent for Oracle runs:
  - Windows Server 2008
  - AIX 6.1
  - Solaris 10
- Oracle Database 10g was deleted to the list of monitoring target programs except HP-UX.
- Windows Server 2016 was added to the list of OSs that can run PFM - Agent for Oracle.
- Oracle Database 12c Release 2 was added to the list of monitoring target programs. This enabled PFM - Agent for Oracle to operate with a multitenant container database (CDB) configuration.

- The version of the data model has been changed from 9.0 to 9.1, and the version of the alarm table has been changed from 10.50 to 12.00.
- With the change of data model, the versions of the following reports that use fields for which the type of their data model was changed in version 9.1 or that refer to such reports were changed:
  - Blocking Locks
  - Disk Sorts - Top 10 Sessions
  - Error Log
  - I/O Activity - Top 10 Datafiles
  - Lock Usage - Top 10 Sessions
  - Locked Objects
  - Longest Transactions - Top 10 Sessions
  - Memory Usage - Top 10 Sessions
  - Physical I/O - Top 10 Sessions
  - Session Detail
  - SGA Status
  - SGA Status Summary
  - System Overview
- The sizes of the following records were changed:
  - Errorlog Detail (PD_PDEL)
  - Session Detail (PD_PDS)
  - SGA Components (PD_PDSG)
- Fields were added to the following records:
  - Errorlog Detail (PD_PDEL)
  - Session Detail (PD_PDS)
  - SGA Components (PD_PDSG)
- A description about effects of the values of the following instance information was added to the Tablespace (PD_PDTS) record:
  - `localtemp_option`
  - `undospace_option`

N.3 Revisions in 11-00

- Windows Server 2003 was deleted from the OSs on which PFM - Agent for Oracle runs.
- A Linux distribution on which PFM - Agent for Oracle runs was added.
- In Windows environment, the prerequisite product was changed from Oracle Client 32-bit to Oracle Client 64-bit.
- For monitoring of the Oracle Database 11g Release 2 or Oracle Database 12c Release 1 in a Linux environment, the prerequisite product was changed from Oracle Client 32-bit to Oracle Client 64-bit.
- The default value of following instance information was changed.
  - `startup_always`

    - undospace_option
- nls_lang has been added to instance information
- The available values of the LANG environment variable for PFM - Agent for Oracle have been added.
- The following languages can now be used with Performance Management:
    - Chinese (Simplified)
    - Korean
    - Spanish
    - German
    - French
    - Russian
- Monitoring Console Https is now included as one of the Agent Collector service properties.
- The product name was changed from PFM - Agent for OpenTP1 to PFM - Agent for Transaction System.
- The product name was changed from JP1/ITSLM to JP1/SLM.

N.4 Revisions in 10-50

- UTF-8 is now supported in HP-UX environment and AIX environment
- The following ASM-related records were added:
    - ASM Disk (PD_PDDK)
    - ASM Disk Group Interval (PI_PIDG)
- The following fields associated with automatic expansion were added to the tablespace (PD_PDTS) record:
    - Auto Extensible
    - Extensible Mbytes
    - Extensible Mbytes %
    - Max Extend Free %
    - Max Extend Free Mbytes
    - Max Extend Mbytes
- The following fields associated with the Redo log buffer busy wait rate were added to the System Stat Summary (PD) record and the System Stat Summary Interval (PI) record:
    - Redo Log Buffer Alloc Retries
    - Redo Log Buffer Wait %
- The Over 10 Sec Collection Time property was added as a condition for collecting performance data for each record.
- The data model version was changed from 8.0 to 9.0 and the alarm table version was changed from 10.00 to 10.50.

N.5 Revisions in 10-00

- Windows Server 2012 was added to the list of OSs that can run PFM - Agent for Oracle.

- The non-CDB environment for Oracle Database 12c Release 1 was added to the list of programs that can be monitored.
- Information about monitoring items for monitoring services through linkage with JP1/IT Service Level Management has been added. In addition, information about the following monitoring items has been added:
  - Cache miss rate
  - Disk sorts execution rate
  - Buffer cache usage rate
  - Buffer busy rate
  - Library cache miss rate
- The collection of performance data is now supported in an IPv6 environment if the OS of the host on which PFM - Agent for Oracle is installed is Windows Server 2008 R2, Windows Server 2012, or Linux.
- UTF-8 is now supported in Solaris environment and Linux environment
- An option for switching UNDO tablespace monitoring has been added.
- The `Pool` field has been added to the SGA Components (`PD_PDSG`) record.
- The data model version was changed from 7.0 to 8.0 and the alarm table version was changed from 09.00 to 10.00.

N.6 Revisions in 09-10

- The following has been added as databases that can be monitored. In addition to it, notes on setting instance information and others have been added.
- Oracle Database Release 2
- Setup commands can now be executed non-interactively.
- The following detailed items have been added to alarm properties in the monitoring template:
  - Product
  - Alarm message
  - Enable alarm
  - Evaluate all data
  - Monitoring time range
  - E-mail
  - Command

N.7 Revisions in 09-00

- The name solution set has been changed monitoring template.
- Due to the addition of commands with a new format that is compatible with the commands for version 08-11 or earlier, the commands for version 09-00 or later are now referred to as follows:

  Commands for 09-00 or later (commands for 08-11 or earlier)
- `localtemp_option` has been added to instance information
- The following packages have been added to the `LSC_13_PLAN_TABLE` table:

    - LSC_13_PDDB2
    - LSC_13_PIDB3
- The description of the following records were changed.
    - Collection Tablespace 2 (PD_PCTS)
    - Data File (PD_PDDF)
    - Data File Interval (PI_PIDF)
    - Database (PD_PDDB)
    - Database Interval (PI_PIDB)
    - Tablespace (PD_PDTS)
    - Tablespace Fragmentation (PD_PDTF)
- The following properties has been added to the list of properties for the Agent Collector service:
    - LOCALTEMP_OPTION

O. Reference Material for This Manual

This appendix provides reference information, including various conventions, for this manual.

O.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):

Manuals associated with JP1/Performance Management:

- *JP1 Version 12 Performance Management: Getting Started* (3021-3-D75(E))
- *JP1/Performance Management Planning and Configuration Guide* (3021-3-D76(E))
- *JP1/Performance Management User's Guide* (3021-3-D77(E))
- *JP1/Performance Management Reference* (3021-3-D78(E))

Manuals associated with JP1:

- *Job Management Partner 1/Software Distribution Administrator's Guide Volume 1* (3020-3-S81(E)), for Windows systems
- *Job Management Partner 1/Software Distribution Manager Description and Administrator's Guide* (3000-3-841(E))
- *Job Management Partner 1/Software Distribution SubManager Description and Administrator's Guide* (3020-3-L42(E)), for UNIX systems
- *Job Management Partner 1/Software Distribution Client Description and User's Guide* (3020-3-S85(E)), for UNIX systems

O.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)
IPF		Itanium(R) Processor Family
JP1/IM	JP1/IM - Manager	JP1/Integrated Management - Manager
		JP1/Integrated Management 2 - Manager
	JP1/IM - View	JP1/Integrated Management - View
		JP1/Integrated Management 2 - View
JP1/ITSLM (10-50 and earlier)	JP1/ITSLM - Manager	JP1/IT Service Level Management - Manager
	JP1/ITSLM - UR	JP1/IT Service Level Management - User Response
JP1/SLM	JP1/SLM - Manager	JP1/Service Level Management - Manager
	JP1/SLM - UR	JP1/Service Level Management - User Response

Abbreviation			Full name or meaning
JP1/Software Distribution			Job Management Partner 1/Software Distribution Client
			Job Management Partner 1/Software Distribution Manager
			JP1/Software Distribution SubManager
Linux	CentOS	CentOS 6 (x64)	CentOS 6.1 (x64) or later
		CentOS 7	CentOS 7.1 or later
	Linux 6 (x64)		Red Hat Enterprise Linux(R) Server 6.1 (64-bit x86_64) or later
	Linux 7		Red Hat Enterprise Linux(R) Server 7.1 or later
	Oracle Linux	Oracle Linux 6 (x64)	Oracle Linux(R) Operating System 6.1 (x64) or later
		Oracle Linux 7	Oracle Linux(R) Operating System 7.1 or later
	SUSE Linux	SUSE Linux 12	SUSE Linux(R) Enterprise Server 12
		SUSE Linux 15	SUSE Linux(R) Enterprise Server 15
Oracle	Oracle10g		Oracle Database 10g
	Oracle11g		Oracle Database 11g
	Oracle12c		Oracle Database 12c
	Oracle18c		Oracle Database 18c
	Oracle19c		Oracle Database 19c
Performance Management			JP1/Performance Management
PFM - Agent	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 IIS#		JP1/Performance Management - Agent Option for Microsoft(R) Internet Information Server
	PFM - Agent for JP1/AJS#	PFM - Agent for JP1/AJS2	JP1/Performance Management - Agent Option for JP1/AJS2
		PFM - Agent for JP1/AJS3	JP1/Performance Management - Agent Option for JP1/AJS3
	PFM - Agent for Microsoft SQL Server		JP1/Performance Management - Agent Option for Microsoft(R) SQL Server

Abbreviation			Full name or meaning
PFM - Agent	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 (UNIX)
		PFM - Agent for Platform(Windows)	JP1/Performance Management - Agent Option for Platform (Windows)
	PFM - Agent for Service Response		JP1/Performance Management - Agent Option for Service Response
	PFM - Agent for Transaction System#		JP1/Performance Management - Agent Option for Transaction System
	PFM - Agent for WebLogic Server#		JP1/Performance Management - Agent Option for BEA 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 - Agent for WebSphere MQ#		JP1/Performance Management - Agent Option for IBM WebSphere MQ
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	PFM - RM for Platform(UNIX)	JP1/Performance Management - Remote Monitor for Platform (UNIX)
		PFM - RM for Platform(Windows)	JP1/Performance Management - Remote Monitor for Platform (Windows)
	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 only runs in a Japanese environment.

O.3 Conventions Acronyms

This manual also uses the following acronyms:

Acronym	Full name or meaning
CPU	Central Processing Unit

Acronym	Full name or meaning
DDL	Data Define Language
DHCP	Dynamic Host Configuration Protocol
DML	Data Manipulation Language
FQDN	Fully Qualified Domain Name
HTML	Hyper Text Markup Language
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
LAN	Local Area Network
NAPT	Network Address Port Translation
NAT	Network Address Translation
ODBC	Open Database Connectivity
OS	Operating System
SNMP	Simple Network Management Protocol
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
UAC	User Account Control
URL	Uniform Resource Locator
Web	World Wide Web
WOW64	Windows On Windows 64

O.4 Conventions: Product names, service IDs, and service keys

Performance Management version 09-00 or later can display the product name as the service ID and service key by enabling the product name display functionality.

Identifiers	Product name display functionality	
	Disabled	Enabled
Service ID	OS1 `hostname`	`hostname` <Oracle>(Store)
	OA1 `hostname`	`hostname` <Oracle>
Service Key	agto	Oracle

Hereafter in this manual, service IDs and service keys are shown in the format when the product name display functionality is enabled.

Note that you can enable the product name display functionality only when you satisfy the two conditions listed below:

- The version number of the prerequisite programs (PFM - Manager or PFM - Base) is 09-00 or later.

- The version number of PFM - Web Console and connection-target PFM - Manager is 09-00 or later.

O.5 Conventions: Installation folder

In this manual, the installation folder for the Windows version of Performance Management is indicated by *installation-folder*. The installation directory for the UNIX version of Performance Management is indicated by *installation-directory*.

The default installation folder for the Windows version of Performance Management is as follows:

- Default installation folder for PFM - Base:

  ```
  system-drive\Program Files(x86)\Hitachi\jp1pc
  ```

  Note

This manual uses the term *installation-folder* for the PFM - Base installation folder.

- Default installation folder for PFM - Management:

  ```
  system-drive\Program Files(x86)\Hitachi\jp1pc
  ```

- Default installation folder for PFM - Web Console:

  ```
  system-drive\Program Files(x86)\Hitachi\jp1pcWebCon
  ```

The default installation directory for the UNIX version of Performance Management is 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.6 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.

P. Glossary

action

An action executed automatically by Performance Management when the data being monitored reaches a threshold value. The following actions are supported:

- Sending an email
- Executing a command
- Issuing an SNMP trap
- Issuing a JP1 event

Action Handler

A PFM - Manager or PFM - Base service that executes actions.

Agent

A PFM - Agent service that collects performance data.

Agent Collector

A PFM - Agent service that collects performance data and evaluates the data according to the threshold values set in alarms.

Agent Store

A PFM - Agent service that stores performance data in a database. The Agent Store service uses a database for recording performance data. Each PFM - Agent has its own Agent Store service.

alarm

Information that defines an action or event message that is triggered when the data being monitored reaches a threshold value.

alarm table

A table containing the following definition information about one or more alarms:

- Monitored object (process, TCP, Web service, and so on)
- Monitored information (CPU usage, number of bytes received per second, and so on)
- Monitored condition (threshold value)

binding

The process of associating alarms with an agent. Binding enables the user to be notified when the performance data collected by the agent reaches a threshold value defined in an alarm.

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.

→ *HA cluster system*

→ *Load-balancing cluster system*

Correlator

A PFM - Manager service that controls event distribution between services. This service evaluates the alarm status, and sends an alarm event or agent event to the Trap Generator service and to PFM - Web Console if the alarm status exceeds a threshold value.

database ID

An ID attached to each record in PFM - Agent, indicating the database in which the record is stored and the record type. The database ID may be either of the following:

- `PI`

  Indicates that the database contains records of the PI record type.

- `PD`

  Indicates that the database contains records of the PD record type.

data model

A generic term for the records and fields contained in a PFM - Agent. Data models are versioned.

drilldown report

A report related to another report or to the fields in the report. A drilldown report can be used to display detailed information or related information for 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 operation information entries in a record. Each field serves as a monitoring item for Performance Management.

Function ID

A one-byte identifier indicating the function type of a service of Performance Management programs. This is part of the service ID.

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 that tracks the status of an object being monitored from a point in the past to the present.

instance

In this manual, the term *instance* is used as follows.

- To indicate the format of a record:

  A record written on one line is known as a *single-instance record*. A record spanning multiple lines is known as a *multi-instance record*, each line of which is known as an *instance*.

- To indicate the number of PFM - Agent:

  A single agent that monitors all the target objects on a host is known as a *single-instance agent*. Agents that share the monitoring of target objects on a host are known collectively as a *multi-instance agent*. Each of these agent services of a multi-instance agent is called an *instance*.

instance number

An identifier for management number used for internal processing. An instance number is part of the service ID.

lifetime

The length of time that the consistency of the performance data collected in each record is retained.

JP1/SLM

A product that performs monitoring from the viewpoint of performance as experienced by the service users of a business system, and that supports service-level maintenance. Linkage with JP1/SLM can enhance monitoring of the operating status.

load-balancing cluster system

A system that distributes the processing load over multiple nodes to improve throughput. Because processing switches to another node if an executing node stops due to a failure, this system also improves the availability of the system.

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.

management tool

Any command or GUI-based function used to verify the status of a service or to manipulate performance data. Management tools allow you to:

- Display the configuration and status of a service
- Save and restore performance data
- Export performance data to a text file
- Delete performance data

Master Manager

A PFM - Manager service. This is the main service of PFM - Manager.

Master Store

A PFM - Manager service that manages the alarm events issued from each PFM - Agent. This service uses a database to store the event data.

monitoring template

A set of predefined alarms and reports provided by PFM - Agent. The monitoring template facilitates preparation for monitoring of the PFM - Agent operation status without the user having to enter complex definitions.

multi-instance record

A record spanning multiple lines. This type of record has unique ODBC key fields.

→ *instance*

ODBC key field

These fields display the primary keys that are necessary to use the data retrieved from records stored in the Store database on either PFM - Manager or PFM - Base. Some ODBC key fields are common to all records; others are record-specific.

PD record type

→ *Product Detail record type*

performance data

Data about the operation status of a resource, collected from the system being monitored.

Performance Management

A generic term for a family of software products used to monitor and analyze problems 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 in the Performance Management family. PFM - Agent is responsible for system monitoring. Several types of PFM - Agent are available, depending on the applications, database, and OS to be monitored. PFM - Agent provides the following features:

- Performance monitoring of target objects
- Collection and recording of data from monitored objects

PFM - Base

One of the program products in the Performance Management family. PFM - Base provides the core functionality for operation monitoring in Performance Management. It is a prerequisite product for running PFM - Agent and provides the following features:

- Commands and other management tools
- Common functions for linking Performance Management with another system

PFM - Manager

One of the program products in the Performance Management family. PFM - Manager performs supervisory functions and provides the following features:

- Management of the Performance Management program products
- Event management

PFM - Manager name

A field name that identifies the field in the Store database that stores the reference data. Use this name, for example, when you execute any command with the field name in the Store database.

PFM - View name

Alias name for PFM - Manager name. PFM - View name is more intuitive than PFM - Manager name. For example, "INPUT_RECORD_TYPE" (PFM - Manager name) is "Record Type"(PFM - View name). Use this field name, for example, when you specify the field name in the PFM - Web Console windows.

PFM - Web Console

One of the program products in the Performance Management family. PFM - Web Console operates as a Web application server to enable centralized monitoring of the Performance Management system via a browser. It provides the following features:

- Display in a graphical user interface
- Integrated monitoring and management
- Definition of reports and alarms

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

→ *Product Interval record type*

Product Detail record type

A type of record for storing performance data indicating the system status at a specific point in time, such as detail information about the currently running process. PD records can be used to acquire system statuses such as the following at a specific point in time:

- System operation status
- Amount of file system capacity currently in use

product ID

A one-byte ID indicating the Performance Management program product to which the service of the Performance Management program belongs. A product ID is part of the service ID.

Product Interval record type

A type of record for storing performance data at set intervals, such as a process count every minute. PI records can be used to analyze such time-based changes and trends in the system status such as the following:

- Number of system calls generated within a set time period
- Changes in the amount of file system capacity used

real-time report

A report that shows the current status of an object being monitored.

record

A group of operation information entries categorized by their purpose. A monitoring agent collects operation information from each record. The types of records that can be collected vary depending on the agent program.

report

Information defined for graphical display of the performance data collected by PFM - Agent. The main types of information you can define are as follows:

- The records to be displayed in a report
- The performance data items to be displayed
- The display format of performance data (table, graph, and so on)

service ID

A unique ID assigned to each service of the Performance Management programs. You must specify the service ID when you execute a command to check the Performance Management system configuration or to back up performance data of an agent, for example. The format of the service ID differs depending on the setting of the product name display functionality. For details about the format of the service ID, see the chapter on Performance Management functionalities in the *JP1/Performance Management Planning and Configuration Guide*.

single-instance record

A record written on a single line. This type of record does not have any unique ODBC key fields.

→ *instance*

stand-alone mode

A PFM - Agent activated as a stand-alone program. If either of the PFM - Manager services Master Manager or Name Server is disabled due to a failure or another problem, you can still collect performance data by starting PFM - Agent.

standby node

Of the server systems in a cluster system, a node that is waiting to take over applications if the executing node fails.

Store database

A database containing performance data collected by the Agent Collector service.

Non-interactive (command)

Command execution method in which operator input required for command execution are provided by values in option specifications or in definition files.

Executing a command non-interactively saves work when configuring an operation monitoring system and can reduce user workload.

Index

A

action 21, 734
Action Handler 734
action log
  setting up (UNIX) 145
  setting up (Windows) 69
action log information
  outputting 713
Activity Summary (PD_PDAS) record 322
Agent 734
Agent Collector 734
Agent Collector service
  properties 687
agent log 656
Agent Store 734
Agent Store service
  properties 684
alarm 21, 734
  format of explanation 233
alarm table 21, 734
alias name 37, 104
ALTER USER statement 53, 124
ASM Disk(PD_PDDK) record 326
ASM Disk Group Interval(PI_PIDG) record 331

B

backup (UNIX) 174
backup (Windows) 95
Backup Async IO 335
Backup Sync IO 338
binding 22, 734
Block Contention Interval (PI_PIBC) record 341
Block Contention Statistics (PD_PDBC) record 343
Blocking Locks (9.1) 251
Buffer Cache Usage alarm 235
Buffer Cache Waits alarm 236
Buffer Pool (PD_PDBP) record 345

C

Cache Summary (PD_PDCS) record 348
Cache Summary Interval (PI_PICS) record 352
Cache Usage (8.0) report 252
Cache Usage Status (Multi-Agent) (8.0) report 253
Cache Usage Trend (Multi-Agent) (8.0) report 254
cancellation facility
  for Oracle access during record collection (UNIX)170
  for Oracle access during record collection (Windows) 90
cancellation facility for Oracle access during record collection (UNIX) 170
cancellation facility for Oracle access during record collection (Windows) 90
CDB Audit Size Status(9.2) report 255
CDB Data File Size Status(9.2) 256
CDB Diagnostics Size Status(9.2) report 257
CDB Status(9.2) report 258
CDB Status (PD_CDCS) record 356
CDB Tablespace(PD_CDTS) record 360
CDB Tablespace Status(9.2) report 259
change the size of log files
  in cluster system (UNIX) 207
  in cluster system (Windows) 196
change the storage location of performance data
  in cluster system (UNIX) 207
  in cluster system (Windows) 196
Circuit (PD_PDCI) record 367
cluster software
  checking whether services can be started and stopped (UNIX) 211
  checking whether services can be started and stopped (Windows) 199
  registering PFM - Agent (UNIX) 209
  registering PFM - Agent (Windows) 198
  setting up environment (UNIX) 211
  setting up environment (Windows) 199
  unregistering PFM - Agent for Oracle (UNIX) 224
  unregistering PFM - Agent for Oracle (Windows) 217
cluster system 734
  installation and setup (UNIX) 108
  installation and setup (Windows) 41
  load-balancing cluster system 184
  notes on operating PFM - Agent for Oracle 226
  operating PFM - Agent for Oracle 182
  overview 183
Collection Instance 2 369
Collection Tablespace 2 371
command
  jpcinslist command 71, 147
common message log 655, 657

communication in IPv4 environments and IPv6 environments 723
Control File (PD_PDCF) record 374
conventions
diagrams 6
fonts and symbols 7
mathematical expressions 8
version numbers 9
Correlator 735
Current Sessions Stat Summary (PD_PDS3) record 376

D

Database (PD_PDDB) record 406
Database Activity Status (8.0) report 260
Database Activity Status (Multi-Agent) (8.0) report 262
Database Activity Status Detail (8.0) report 261
Database Activity Trend (Multi-Agent) (8.0) report 263
database ID 735
Database Interval (PI_PIDB) record 415
Database Object Cache (PD_PDDO) record 425
Database Space Overview (8.0) report 264
Database Space Summary (Multi-Agent) (8.0) report 265
Database Space Trend (Multi-Agent) (8.0) report 266
data collection in event of failure, preparations for (UNIX) 108
data collection in event of failure, preparations for (Windows) 41
Data Dictionary Cache (PD_PDDD) record 381
Data Dictionary Cache Interval (PI_PIDD) record 383
Data File (PD_PDDF) record 385
Datafile I/O Activity Detail (8.0) report 267
Datafile I/O Status Detail (Reads) (8.0) report 268
Datafile I/O Status Detail (Writes) (8.0) report 269
Datafile I/O Status Summary (8.0) report 270
Datafile I/O Trend Detail (Reads) (8.0) report 271
Datafile I/O Trend Detail (Writes) (8.0) report 272
Datafile I/O Trend Summary (8.0) report 273
Data File Interval (PI_PIDF) record 394
data model 21, 299, 735
data update processing performance 32
determining baseline 30
diagram conventions 6
Dict. Cache Usage alarm 237
disk monitoring 33
Disk Sorts alarm 238
Disk Sorts - Top 10 Sessions (9.1) report 274
disk space requirements 677
for operation in cluster system 677
Dispatcher (PD_PDDS) record 427
Dispatcher Interval (PI_PIDS) record 430
drilldown report 735
field level 245
report level 245
DROP TABLESPACE statement 73, 149
DROP USER statement 73, 149

E

editing the jpcsto.ini file (UNIX)
procedure 154
editing the jpcsto.ini file (Windows)
procedure 78
error handling, procedures 650
error handling procedure 651
Error Log (9.1) report 275
Errorlog Detail (PD_PDEL) record 433
errors in Performance Management
detecting 674
recovering from 675
event log, Windows
list of 628
example of performance monitoring using PFM - Agent for Oracle 30
executing node 735

F

failover 735
at failure on PFM - Agent for Oracle host 186
processing during 186
Fast Recovery Area Status(9.2) report 276
features of PFM - Agent for Oracle 20
field 21, 245, 735
added only when data is stored in the Store database 310
list of ODBC key fields 303
output when data stored in Store database is exported 311
value 307
files and directories 697
firewall routing 683
font conventions 7
Free List Waits alarm 239
Full Table Scans (8.0) report 277
Full Table Scans alarm 240

function ID 735

G

GCS Stat Summary (PD_PDGC) record 435
GCS Stat Summary Interval (PI_PIGC) record 437

H

HA cluster system 183, 735
historical report 20, 736
host name, in collected performance data 226

I

I/O Activity - Top 10 Datafiles (9.1) report 278
identifiers
  list of 680
installation (in cluster system)
  preparation (UNIX) 202
  preparation (Windows) 190
  procedure (UNIX) 204
  procedure (Windows) 193
  UNIX 200
  Windows 188
  workflow (UNIX) 200
  workflow (Windows) 188
installation (UNIX) 102, 103
  in cluster system 108
  methods 111
  notes on installing multiple Performance Management programs on one host 108
  order 111
  other cautionary notes 109
  preparation 104
  procedure 111
  workflow 103
installation (Windows) 35, 36
  in cluster system 41
  methods 44
  notes on installing multiple Performance Management programs on one host 41
  order 44
  other cautionary notes 42
  preparation 37
  procedure 43
  workflow 36
instance 736
Instance (PD_PDI) record 439
Instance Availability (PD_PDIA) record 441
instance environment
  deleting (UNIX) 147
  deleting (Windows) 71
  setting (in cluster system) (UNIX) 206
  setting (in cluster system) (Windows) 195
  setting up (UNIX) 124
  setting up (Windows) 53
  setup cancellation (UNIX) 147
  setup cancellation (Windows) 71
  updating (in cluster system) 228
  updating (UNIX) 155
  updating (Windows) 79
instance information
  setting up (UNIX) 124
  setting up (Windows) 53
instance number 736
IP address settings
  UNIX 104
  Windows 37
IPv4 environments and IPv6 environments, communication in 723

J

JP1/SLM 736
jpcconf db define command 76, 152
jpcconf mgrhost define command 144
jpcctrl delete command 72, 147
jpcctrl list command 70, 146
jpchosts file 196, 207
jpcinslist command 71, 87, 147, 169
jpcinssetup 63, 139
jpcinssetup command 87, 169
jpcinsunsetup command 71, 147
jpcnshostname command 68
jpcras command 669
jpcwagtsetup command 47

K

kernel parameters 678

L

LANG environment variable 114
Latch (PD_PDLA) record 444
Latch Interval (PI_PILA) record 447
Library Cache (PD_PDLC) record 450

Library Cache Interval (PI_PILC) record 452
Library Cache Usage alarm 241
lifetime 736
linking with JP1/SLM 722
listener_home 54, 126
listener_home 126
listener_name 55
List of Alarms 234
list of data types 306
load-balancing cluster system 184, 736
  configuration of Oracle in 184
localtemp_option 56, 127
Lock (PD_PDLO) record 454
Lock Activity Interval (PI_PIPL) record 457
Locked Objects (9.1) report 280
Lock Interval (PI_PILO) record 459
Lock Usage - Top 10 Sessions (9.1) report 279
Lock Waiters (PD_PDLW) record 462
log_path 55, 126
log_size 55, 126
log file
  changing size (UNIX) 144
  changing size (Windows) 68
log files and directories 657
logical host 736
logical-host environment definition file
  copying to standby node (UNIX) 208, 223
  copying to standby node (Windows) 197, 216
  exporting (UNIX) 208, 223
  exporting (Windows) 196, 215
  exporting and importing (in cluster system) 229
  importing (UNIX) 208, 223
  importing (Windows) 197, 216
log information 655
Longest Transactions - Top 10 Sessions (9.1) report 281

M

management tool 736
Master Manager 737
Master Store 737
mathematical expressions
  conventions 8
memory requirements 677
Memory Usage - Top 10 Sessions (9.1) report 282
message catalog storage directory
  list of files and directories 705
message format 622
messages 621, 629
  explanation format 623
  output destination 624
  output format 622
  output to syslog and Windows event log, list of 628
migration
  notes on 708
  procedure 708
Minimum Database Interval 2 464
Minimum Data File Interval 2 465
Minimum Tablespace Interval 2 469
monitoring target programs
  UNIX 107
  Windows 40
monitoring template 22, 232, 737
  overview 232
Monitoring template 231
Monitoring wait events concerning REDO log file 34
multi-instance record 737
Multi - Threaded Server 472
Multi - Threaded Server Interval (PI_PIMT) record 476

N

net_service_name 54, 126
network environment settings
  IP address settings (UNIX) 104
  IP address settings (Windows) 37
  port number settings (UNIX) 106
  port number settings (Windows) 39
  UNIX 104
  Windows 37
network settings, specifying
  in cluster system (UNIX) 207
  in cluster system (Windows) 196
  UNIX 143
  Windows 67
nls_lang 56, 127
non-interactive (command) 740
Notes regarding operation 99
numeric_10 56, 127

O

object
  registration in Oracle Database (UNIX) 141
  registration in Oracle Database (Windows) 65

ODBC key field 737
online manuals
settings required to read manuals from Web browser 97, 176
Open Cursor (PD_PDOC) record 480
Open Cursors (4.0) report 283
operation
changing (in cluster system) 228
changing (UNIX) 152
changing (Windows) 76
operation status log 655
Options Installed (PD_PDO) record 482
ORACLE_HOME 54, 125
oracle_passwd 54, 125
ORACLE_SID 53, 125
oracle_user 54, 125
oracle_version 54, 125
Oracle account
creating (in cluster system) (UNIX) 206
creating (in cluster system) (Windows) 195
creating (UNIX) 119
creating (Windows) 47
deleting (UNIX) 149
deleting (Windows) 73
Oracle Database
setting up (UNIX) 142
setting up (Windows) 66
Oracle instance operation monitoring 33
OS requirements
UNIX 104
Windows 37
overview of PFM - Agent for Oracle 19

P

Parallel Query Server (PD_PDPQ) record 483
Parallel Query Server Interval (PI_PIPQ) record 486
Parallel Query Statistics (PD_PDPS) record 489
Parameter Values (PD_PDP) record 490
PD 572
PD_CDCS 356
PD_CDTS 360
PD_PCI 369
PD_PCTS 371
PD_PDAS 322, 454
PD_PDBA 335
PD_PDBC 343
PD_PDBP 345
PD_PDBS 338
PD_PDC 541
PD_PDCF 374
PD_PDCI 367
PD_PDCS 348
PD_PDDB 406
PD_PDDD 381
PD_PDDF 385
PD_PDDK 326
PD_PDDO 425
PD_PDDS 427
PD_PDEL 433
PD_PDES 559
PD_PDEV 516
PD_PDGC 435
PD_PDI 439
PD_PDIA 441
PD_PDLA 444
PD_PDLC 450
PD_PDLS 564
PD_PDLW 462
PD_PDMT 472
PD_PDNL 562
PD_PDO 482
PD_PDOC 480
PD_PDOP 492
PD_PDP 490
PD_PDPQ 483
PD_PDPS 489
PD_PDQU 494
PD_PDRL 496
PD_PDRS 498
PD_PDS 508, 510
PD_PDS2 529
PD_PDS3 376
PD_PDSE 566
PD_PDSG 538
PD_PDSH 545
PD_PDSM 504
PD_PDSQ 557
PD_PDSR 551
PD_PDSS 527
PD_PDST 587
PD_PDTA 589
PD_PDTF 602
PD_PDTL 617
PD_PDTR 611

PD_PDTS 591
PD_PDV 619
PD_PDWA 535
PD record type 737
performance data 737
  changing storage location (UNIX) 144, 152
  changing storage location (Windows) 68, 76
  collection method 29
  management method 29
  overview of collecting and managing 29
Performance Management 737
Performance Management programs
  UNIX 107
  Windows 40
permission required to install PFM - Agent
  UNIX 106
  Windows 39
PFM - Agent 737
PFM - Agent for Oracle
  example of performance monitoring 30
  features 20
  files and directories 697
  overview 19
PFM - Base 738
PFM - Manager 738
  effects of stopping 186
  setting as connection target (in cluster system) (UNIX) 206
  setting as connection target (in cluster system) (Windows) 194
  setting as connection target (UNIX) 144
  setting as connection target (Windows) 68
PFM - Manager name 738
PFM - View name 738
PFM - Web Console 738
physical host 738
Physical I/O - Top 10 Sessions (9.1) report 284
PI 579
PI_PIBC 341
PI_PICS 352
PI_PIDB 415
PI_PIDD 383
PI_PIDF 394
PI_PIDG 331
PI_PIDS 430
PI_PIEV 518
PI_PIGC 437
PI_PIIO 520
PI_PILA 447
PI_PILC 452
PI_PILO 459
PI_PIMT 476
PI_PIPL 457
PI_PIPQ 486
PI_PIRS 501
PI_PIS2 522
PI_PISE 568
PI_PISH 548
PI_PISR 554
PI_PIST 570
PI_PITR 614
PI_PITS 607
PI_PMDB 464
PI_PMDF 465
PI_PMTS 469
PI record type 738
port numbers 683
  list of 683
port number settings
  UNIX 106
  Windows 39
precautions regarding environment variables
  UNIX 108
  Windows 41
precautions regarding permissions 710
prerequisite programs
  UNIX 106
  Windows 40
problem reports
  collecting 663
Process Detail (PD_PDOP) record 492
Processes
  list of 681
Product Detail record type 738
product ID 739
Product Interval record type 739
properties 684
  Agent Collector service 687
  Agent Store service 684
purpose of performance monitoring 30

Q

Queue Statistics (PD_PDQU) record 494

R

real host name 37, 104
real-time report 20, 739
record 21, 245, 739
  explanation format 300
  list of 313
records
  notes 312
Records 298
record type
  PD record type 21
  PI record type 21
  Product Detail record type 21
  Product Interval record type 21
Redo Log Buffer Contention (8.0) report 285
Redo Log Contention alarm 242
registering PFM - Agent for Oracle
  UNIX 116
  Windows 45
report 20, 739
  explanation format 245
  folder organization 246
  list of 248
Resource Limit (PD_PDRL) record 496
restoration (UNIX) 175
restoration (Windows) 96
retry_time 55
Rollback Segment (PD_PDRS) record 498
Rollback Segment Interval (PI_PIRS) record 501

S

search processing performance 31
Segment Detail (PD_PDSM) record 504
Server Configuration Status (4.0) report 286
Server Status (PD_STAT) record 508
Server Status alarm 243
service ID 739
Session Detail (9.1) report 287
Session Detail (PD_PDS) record 510
Session Event (PD_PDEV) record 516
Session Event Interval (PI_PIEV) record 518
Session I/O Interval (PI_PIIO) record 520
Session Statistics(PD_PDSS) record 527
Session Statistics Detail (4.0) report 288
Session Statistics Summary (PD_PDS2) record 529
Session Stat Summary Interval (PI_PIS2) record 522
Session Wait (PD_PDWA) record 535
setting items in jpcsto.ini (UNIX) 153
setting items in jpcsto.ini (Windows) 77
setting up the action log
  in cluster system (Windows) 196
setup (in cluster system)
  bringing shared disk online 194
  canceling (Windows) 212
  checking whether services can be started and stopped from cluster software (UNIX) 211
  checking whether services can be started and stopped from cluster software (Windows) 199
  copying logical-host environment definition file to standby node (UNIX) 208
  copying logical-host environment definition file to standby node (Windows) 197
  creating Oracle account (UNIX) 206
  creating Oracle account (Windows) 195
  exporting logical-host environment definition file (UNIX) 208
  exporting logical-host environment definition file (Windows) 196
  importing logical-host environment definition file (UNIX) 208
  importing logical-host environment definition file (Windows) 197
  mounting shared disk 205
  preparation (UNIX) 202
  preparation (Windows) 190
  procedure (UNIX) 205
  procedure (Windows) 193
  registering PFM - Agent for Oracle (UNIX) 205
  registering PFM - Agent for Oracle (Windows) 193
  registering PFM - Agent in cluster software (UNIX) 209
  registering PFM - Agent in cluster software (Windows) 198
  setting connection-target PFM - Manager (UNIX) 206
  setting connection-target PFM - Manager (Windows) 194
  setting logical host environment for PFM - Agent for Oracle (UNIX) 205
  setting logical host environment for PFM - Agent for Oracle (Windows) 194
  setting up environment (in cluster system) (UNIX) 211
  setting up environment (in cluster system) (Windows) 199
  setting up instance environment (UNIX) 206
  setting up instance environment (Windows) 195

setting up logical host environment for other Performance Management programs (UNIX) 207
setting up logical host environment for other Performance Management programs (Windows) 195
specifying network settings (UNIX) 207
specifying network settings (Windows) 196
taking shared disk offline 197
UNIX 200
unmounting shared disk 208
Windows 188
workflow (UNIX) 200
workflow (Windows) 188
setup (UNIX) 102, 103
canceling 146
copying setup files 118
creating Oracle account 119
executing setup command on PFM - Manager host 118
executing setup command on PFM - Web Console host 118
in cluster system 108
notes on setting up multiple Performance Management programs on one host 108
other cautionary notes 109
preparation 104
procedure 114
procedure for canceling 147
registering PFM - Agent for Oracle 116
setting LANG environment variable 114
workflow 103
setup (Windows) 35, 36
canceling 70
copying setup files 46
creating Oracle account 47
executing setup command on PFM - Manager host 47
executing setup command on PFM - Web Console host 47
in cluster system 41
notes on setting up multiple Performance Management programs on one host 41
other cautionary notes 42
preparation 37
procedure 45
procedure for canceling 71
registering PFM - Agent for Oracle 45
workflow 36
setup cancellation (in cluster system)
bringing shared disk online 214
clearing port number settings (UNIX) 221
clearing port number settings (Windows) 214
copying logical-host environment definition file to standby node (UNIX) 223
copying logical-host environment definition file to standby node (Windows) 216
deleting settings from PFM - Manager (UNIX) 224
deleting settings from PFM - Manager (Windows) 217
exporting logical-host environment definition file (UNIX) 223
exporting logical-host environment definition file (Windows) 215
importing logical-host environment definition file (UNIX) 223
importing logical-host environment definition file (Windows) 216
logical host environment for executing node 214
logical host environment for PFM - Agent for Oracle node 221
logical host environments for other Performance Management programs (UNIX) 223
logical host environments for other Performance Management programs (Windows) 215
mounting shared disk 221
procedure (UNIX) 220
procedure (Windows) 213
stopping services from cluster software (UNIX) 221
stopping services from cluster software (Windows) 214
taking shared disk offline 216
UNIX 219
unmounting shared disk 223
unregistering PFM - Agent for Oracle from cluster software (Windows) 217
unregistering PFM - Agent from cluster software (UNIX) 224
workflow (UNIX) 219
workflow (Windows) 212
setup cancellation (UNIX) 146
cautionary notes 146
deleting instance environment 147
deleting objects registered in Oracle Database 148
deleting Oracle account 149
instance environment 147
procedure 147
setup cancellation (UNIX)
deleting tablespaces used by deleted Oracle account 149

setup cancellation (Windows) 70, 212
cautionary notes 70
deleting instance environment 71
deleting objects registered in Oracle Database 72
deleting Oracle account 73
deleting tablespaces used by deleted Oracle account 73
instance environment 71
procedure 71
setup command (UNIX)
PFM - Manager host 118
PFM - Web Console host 118
setup command (Windows)
PFM - Manager host 47
PFM - Web Console host 47
SGA Components (PD_PDSG) record 538
SGA Status (9.1) report 289
SGA Status Summary (9.1) report 290
Shared Cursor Cache (PD_PDC) record 541
shared disk
bringing online 194, 214
mounting 221
mounting (in cluster system) 205
taking offline 197, 216
unmounting 223
unmounting (in cluster system) 208
Shared Server (PD_PDSH) record 545
Shared Server Interval (PI_PISH) record 548
single-instance record 739
Sort Segment (PD_PDSR) record 551
Sort Segment Interval (PI_PISR) record 554
sql_option 55, 127
SQL*Net Listener (PD_PDNL) record 562
SQL*Net Listeners (PD_PDLS) record 564
sqlnet 54, 125
SQL Text (4.0) report 291
SQL Text (PD_PDSQ) record 557
SQL Text - Performance Based (PD_PDES) record 559
stand-alone mode 739
standby node 739
startup_always 56, 127
Store database 21, 740
Store Version 127
Store Version 56
summary rules 304
symbol conventions 7
syntax conventions 8
syslog, UNIX
list of 628
system configuration
changing (in cluster system) 227
changing (UNIX) 151
changing (Windows) 75
System Event (PD_PDSE) record 566
System Event Interval (PI_PISE) record 568
system log 655
System Overview (9.1) report 292, 294
system requirements
estimating 677
System Stat Interval (PI_PIST) record 570
System Statistics (PD_PDST) record 587
System Stat Summary (PD) record 572
System Stat Summary Interval (PI) record 579

T

Table Access (PD_PDTA) record 589
tablespace
deleting (UNIX) 149
deleting (Windows) 73
Tablespace (PD_PDTS) record 591
Tablespace Fragmentation (PD_PDTF) record 602
Tablespace Interval (PI_PITS) record 607
Tablespace Status (4.0) report 296
Tablespace Status Detail (4.0) report 297
Tablespace Usage alarm 244
timeout 55, 127
trace log 656, 657
Transaction (PD_PDTR) record 611
Transaction Interval (PI_PITR) record 614
Transaction Lock (PD_PDTL) record 617
troubleshooting
information required for 660
procedures 652
setup and service startup 652
troubleshooting information
collecting 669

U

undospace_option 56, 128
uninstallation (in cluster system)
procedure (UNIX) 225
procedure (Windows) 217
UNIX 219

Windows 212
workflow (UNIX) 219
workflow (Windows) 212
uninstallation (UNIX)
cautionary notes 146
note on network configuration 146
note on OS user permission 146
notes on programs 146
notes on services 146
other notes 146
procedure 149
uninstallation (Windows) 146
uninstallation (Windows) 70
cautionary notes 70
note on network configuration 70
note on OS user permission 70
notes on programs 70
notes on services 70
other notes 70
procedure 73
updating the Store version to 2.0 92, 171
upgrading PFM - Agent for Oracle
notes (UNIX) 109
notes (Windows) 42
user mode process dump 663

V

Version (PD_PDV) record 619
version compatibility 712
version number conventions 9

Hitachi, Ltd.
6-6, Marunouchi 1-chome, Chiyoda-ku, Tokyo, 100–8280 Japan