Progress®



OpenEdge® Management:
Database Management

OpenEdge®

Copyright

© **2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.**

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.

Corticon, DataDirect (and design), DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Deliver More Than Expected, Icenium, Kendo UI, Kinvey, NativeScript, OpenEdge, Powered by Progress, Progress, Progress Software Developers Network, Rollbase, SequeLink, Sitefinity (and Design), Sitefinity, SpeedScript, Stylus Studio, TeamPulse, Telerik, Telerik (and Design), Test Studio, and WebSpeed are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries. Analytics360, AppServer, BusinessEdge, DataDirect Autonomous REST Connector, DataDirect Spy, SupportLink, DevCraft, Fiddler, JustAssembly, JustDecompile, JustMock, Kinvey Chat, NativeChat, NativeScript Sidekick, OpenAccess, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, and WebClient are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

February 2019

Last updated with new content: Release 12.0.0

Updated: 2019/03/01

Table of Contents

Preface....9
Purpose....9
Audience....10
Organization....10
Using ABL documentation....10
References to ABL compiler and run-time features....10
References to ABL data types....11
Typographical conventions....11
Examples of syntax descriptions....12
Long syntax descriptions split across lines....14
Complex syntax descriptions with both required and optional elements....14
OpenEdge messages....14
Obtaining more information about OpenEdge messages....15

Chapter 1: Database Resource Monitoring....17
Resource monitoring terms....18
Managing databases using OpenEdge Management....18
Managed versus scripted databases....18
Migrating a database....18
Starting a remote managed database agent....19
Monitoring managed databases....20
Default database monitoring plans....20
Modifying database monitoring plans....21
Monitoring scripted databases....22
Starting a scripted database agent....23
Understanding and using database rules....24
Viewing rules associated with a monitoring plan....25
Working with database rule sets....26
Creating a database rule set....27
Editing a database rule set....27
Copying a database rule set....27
Deleting a database rule set....28
Rule sets with one or more rules in common....28
Associating a rule set with a database monitoring plan....28
Using the Configuration Advisor....28
Initiating the Configuration Advisor....29
Understanding the recommended threshold settings....31
Comparing and selecting threshold settings....32

Submitting your threshold setting selections....32
Determining the effectiveness of your selections....33

Chapter 2: Graphical Displays of Database Data....35

Interpreting graphical database data....35
General graph information....35

Chapter 3: Examining Data from an OpenEdge Database....37

Accessing database information....38
Changing database control settings....39
Database log file monitors....40
Log file monitor default values....40
Log file monitor considerations....40
Customizing a database log file monitor....41
Using the log file viewer....41
Understanding OpenEdge database views....43
Database Configuration view....44
Performance Summary view....45
Vital Signs view....45
File Systems view....46
Storage Areas view....47
User Activity views....49
Record and Index Activity view....50
Locks and Latches views....50
Transactions view....50
Memory Resources view....51
Page Writers view....51
Advanced views....52

Chapter 4: Database Maintenance Job Templates....53

Database maintenance using job templates....53
Displaying database job templates....54
General considerations....54
Creating database maintenance job instances....55
Setting up Database Backup jobs....56
Setting up a Database Restore job....58
Setting up a Truncate BI job....59
Setting up a Grow BI job....60
Setting up a Database Analysis job....61
Setting up an Index Compaction job....62
Setting up a Backup Configuration job....63
Editing advanced job information....64
Environment variables for database maintenance jobs....65

Index.....69

Preface

For details, see the following topics:

- Purpose
- Audience
- Organization
- Using ABL documentation
- Typographical conventions
- Examples of syntax descriptions
- OpenEdge messages

Purpose

This guide describes how to use OpenEdge® Management to manage and monitor the condition of your OpenEdge databases. The guide describes scripted and managed databases, database monitoring plans, database views, and database maintenance jobs.

For details about setting database configuration properties, see *OpenEdge Management and OpenEdge Explorer: Configuration*.

For details about working with databases enabled for multi-tenancy, see *OpenEdge Management and OpenEdge Explorer: Configuring Multi-tenancy*.

Audience

This guide is designed for users of the OpenEdge Management product. Typical users are OpenEdge database administrators and any others responsible for the daily management of a database. Before reading this guide, you should be familiar with the OpenEdge Management console and the basic concepts of resource monitoring.

For details about the OpenEdge Management console and an introduction to resource monitoring, see *OpenEdge Management: Resource Monitoring*.

Organization

Database Resource Monitoring on page 17

Provides details about how OpenEdge Management recognizes a database's existence, how to set up default database monitoring, how to use rules with a database, and how to update database monitoring plans. Use the information in this chapter in conjunction with the information provided in *OpenEdge Management: Resource Monitoring*.

Graphical Displays of Database Data on page 35

Describes the graphical display of database data, including pinup graphs.

Examining Data from an OpenEdge Database on page 37

Presents information about database log file monitors, database log file monitoring plans and rules, and database views.

Database Maintenance Job Templates on page 53

Describes the OpenEdge Management-provided job templates you use to set up and schedule routine maintenance activities for your OpenEdge database.

Using ABL documentation

OpenEdge provides a special purpose programming language for building business applications. In the documentation, the formal name for this language is *ABL (Advanced Business Language)*. With few exceptions, all keywords of the language appear in all `UPPERCASE`, using a font that is appropriate to the context. All other alphabetic language content appears in mixed case.

For the latest documentation updates go to https://docs.progress.com.

References to ABL compiler and run-time features

ABL is both a compiled and an interpreted language that executes in a run-time engine. The documentation refers to this run-time engine as the *ABL Virtual Machine (AVM)*. When the documentation refers to ABL source code compilation, it specifies *ABL* or *the compiler* as the actor that manages compile-time features of the language. When the documentation refers to run-time behavior in an executing ABL program, it specifies *the AVM* as the actor that manages the specified run-time behavior in the program.

For example, these sentences refer to the ABL compiler's allowance for parameter passing and the AVM's possible response to that parameter passing at run time: "ABL allows you to pass a dynamic temp-table handle as a static temp-table parameter of a method. However, if at run time the passed dynamic temp-table schema does not match the schema of the static temp-table parameter, the AVM raises an error." The following sentence refers to run-time actions that the AVM can perform using a particular ABL feature: "The ABL socket object handle allows the AVM to connect with other ABL and non-ABL sessions using TCP/IP sockets."

References to ABL data types

ABL provides built-in data types, built-in class data types, and user-defined class data types. References to built-in data types follow these rules:

- Like most other keywords, references to specific built-in data types appear in all `UPPERCASE`, using a font that is appropriate to the context. No uppercase reference ever includes or implies any data type other than itself.
- Wherever *integer* appears, this is a reference to the `INTEGER` or `INT64` data type.
- Wherever *character* appears, this is a reference to the `CHARACTER`, `LONGCHAR`, or `CLOB` data type.
- Wherever *decimal* appears, this is a reference to the `DECIMAL` data type.
- Wherever *numeric* appears, this is a reference to the `INTEGER`, `INT64`, or `DECIMAL` data type.

References to built-in class data types appear in mixed case with initial caps, for example, `Progress.Lang.Object`. References to user-defined class data types appear in mixed case, as specified for a given application example.

Typographical conventions

This documentation uses the following typographical and syntax conventions:

Convention	Description
Bold	Bold typeface indicates commands or characters the user types, provides emphasis, or the names of user interface elements.
Italic	Italic typeface indicates the title of a document, or signifies new terms.
SMALL, BOLD CAPITAL LETTERS	Small, bold capital letters indicate OpenEdge key functions and generic keyboard keys; for example, **GET** and **CTRL**.
KEY1+KEY2	A plus sign between key names indicates a **simultaneous** key sequence: you press and hold down the first key while pressing the second key. For example, **CTRL+X**.
KEY1 KEY2	A space between key names indicates a **sequential** key sequence: you press and release the first key, then press another key. For example, **ESCAPE H**.
Syntax:	
`Fixed width`	A fixed-width font is used in syntax, code examples, system output, and file names.

Convention	Description
Fixed-width italics	Fixed-width italics indicate variables in syntax.
Fixed-width bold	Fixed-width bold italic indicates variables in syntax with special emphasis.
UPPERCASE fixed width	ABL keywords in syntax and code examples are almost always shown in upper case. Although shown in uppercase, you can type ABL keywords in either uppercase or lowercase in a procedure or class.
Period (.) or colon (:)	All statements except DO, FOR, FUNCTION, PROCEDURE, and REPEAT end with a period. DO, FOR, FUNCTION, PROCEDURE, and REPEAT statements can end with either a period or a colon.
[ ]	Large brackets indicate the items within them are optional.
[ ]	Small brackets are part of ABL.
{ }	Large braces indicate the items within them are required. They are used to simplify complex syntax diagrams.
{ }	Small braces are part of ABL. For example, a called external procedure must use braces when referencing arguments passed by a calling procedure.
|	A vertical bar indicates a choice.
. . .	Ellipses indicate repetition: you can choose one or more of the preceding items.

Examples of syntax descriptions

In this example, ACCUM is a keyword, and *aggregate* and *expression* are variables:

Syntax

```
ACCUM aggregate expression
```

FOR is one of the statements that can end with either a period or a colon, as in this example:

```
FOR EACH Customer NO-LOCK:
  DISPLAY Customer.Name.
END.
```

In this example, STREAM *stream*, UNLESS-HIDDEN, and NO-ERROR are optional:

Syntax

```
DISPLAY [ STREAM stream ] [ UNLESS-HIDDEN ] [ NO-ERROR ]
```

In this example, the outer (small) brackets are part of the language, and the inner (large) brackets denote an optional item:

Syntax

```
INITIAL [ constant [ , constant ] ]
```

A called external procedure must use braces when referencing compile-time arguments passed by a calling procedure, as shown in this example:

Syntax

```
{ &argument-name }
```

In this example, EACH, FIRST, and LAST are optional, but you can choose only one of them:

Syntax

```
PRESELECT [ EACH | FIRST | LAST ] record-phrase
```

In this example, you must include two expressions, and optionally you can include more. Multiple expressions are separated by commas:

Syntax

```
MAXIMUM ( expression , expression [ , expression ] ... )
```

In this example, you must specify MESSAGE and at least one *expression* or SKIP [(*n*)], and any number of additional *expression* or SKIP [(*n*)] is allowed:

Syntax

```
MESSAGE { expression | SKIP [ ( n ) ] } ...
```

In this example, you must specify {*include-file*, then optionally any number of *argument* or *&argument-name = "argument-value"*, and then terminate with }:

Syntax

```
{ include-file
    [ argument | &argument-name = "argument-value" ] ... }
```

Long syntax descriptions split across lines

Some syntax descriptions are too long to fit on one line. When syntax descriptions are split across multiple lines, groups of optional and groups of required items are kept together in the required order.

In this example, WITH is followed by six optional items:

Syntax

```
WITH [ ACCUM max-length ] [ expression DOWN ]
  [ CENTERED ] [ n COLUMNS ] [ SIDE-LABELS ]
  [ STREAM-IO ]
```

Complex syntax descriptions with both required and optional elements

Some syntax descriptions are too complex to distinguish required and optional elements by bracketing only the optional elements. For such syntax, the descriptions include both braces (for required elements) and brackets (for optional elements).

In this example, ASSIGN requires either one or more *field* entries or one *record*. Options available with *field* or *record* are grouped with braces and brackets:

Syntax

```
ASSIGN   { [ FRAME frame ] { field [ = expression ] }
           [ WHEN expression ] } ...
       | { record [ EXCEPT field ... ] }
```

OpenEdge messages

OpenEdge displays several types of messages to inform you of routine and unusual occurrences:

- **Execution messages** inform you of errors encountered while OpenEdge is running a procedure; for example, if OpenEdge cannot find a record with a specified index field value.

- **Compile messages** inform you of errors found while OpenEdge is reading and analyzing a procedure before running it; for example, if a procedure references a table name that is not defined in the database.
- **Startup messages** inform you of unusual conditions detected while OpenEdge is getting ready to execute; for example, if you entered an invalid startup parameter.

After displaying a message, OpenEdge proceeds in one of several ways:

- Continues execution, subject to the error-processing actions that you specify or that are assumed as part of the procedure. This is the most common action taken after execution messages.
- Returns to the Procedure Editor, so you can correct an error in a procedure. This is the usual action taken after compiler messages.
- Halts processing of a procedure and returns immediately to the Procedure Editor. This does not happen often.
- Terminates the current session.

OpenEdge messages end with a message number in parentheses. In this example, the message number is `200`:

```
** Unknown table name table. (200)
```

If you encounter an error that terminates OpenEdge, note the message number before restarting.

Obtaining more information about OpenEdge messages

In Windows platforms, use OpenEdge online help to obtain more information about OpenEdge messages. Many OpenEdge tools include the following Help menu options to provide information about messages:

- Choose **Help** > **Recent Messages** to display detailed descriptions of the most recent OpenEdge message and all other messages returned in the current session.
- Choose **Help** > **Messages** and then type the message number to display a description of a specific OpenEdge message.
- In the Procedure Editor, press the **HELP** key or **F1**.

On UNIX platforms, use the OpenEdge `pro` command to start a single-user mode character OpenEdge client session and view a brief description of a message by providing its number.

To use the pro command to obtain a message description by message number:

1. Start the Procedure Editor:

   ```
   OpenEdge-install-dir/bin/pro
   ```

2. Press **F3** to access the menu bar, then choose **Help** > **Messages**.
3. Type the message number and press **ENTER**. Details about that message number appear.
4. Press **F4** to close the message, press **F3** to access the Procedure Editor menu, and choose **File** > **Exit**.

1

Database Resource Monitoring

OpenEdge® Management helps you better manage your database resources by monitoring the databases in your environment and alerting you to issues of potential concern. This chapter introduces database resource monitoring terminology and explains how OpenEdge Management uses rules and schedules to manage database resources.

For details about setting database configuration properties, see *OpenEdge Management and OpenEdge Explorer: Configuration*.

For details about working with databases enabled for multi-tenancy, see *OpenEdge Management and OpenEdge Explorer: Configuring Multi-tenancy*.

For details, see the following topics:

- Resource monitoring terms
- Managing databases using OpenEdge Management
- Monitoring managed databases
- Monitoring scripted databases
- Understanding and using database rules
- Working with database rule sets
- Using the Configuration Advisor

Resource monitoring terms

Before you begin monitoring database resources using OpenEdge Management, it is important to understand the different pieces of resource monitoring and how they work together. A resource is a specific component of your configuration.

OpenEdge Management uses the following components in monitoring resources:

- **Alerts** — Notifications sent when rules are broken.
- **Actions** — Activities triggered in response to alerts. For example, you might indicate that you want to receive an e-mail if a database crashes.
- **Resource monitoring plan** — A plan that identifies the rules against which a resource is monitored and the schedule for that monitoring. OpenEdge Management cannot monitor a resource without a monitoring plan.
- **Rules** — Criteria against which a resource's performance is measured.
- **Schedule** — The block of time during which a resource is polled.

For information about monitoring file, network, and system resources, see *OpenEdge Management: Resource Monitoring*.

For information about monitoring OpenEdge servers, see *OpenEdge Management: Servers, DataServers, Messengers, and Adapters*.

For information about setting configuration properties for databases and other resources, see *OpenEdge Management and OpenEdge Explorer: Configuration*.

Managing databases using OpenEdge Management

OpenEdge Management runs as a process or thread in the AdminServer. Therefore, the type of relationship that you can establish between a database and OpenEdge Management depends on the relationship that currently exists between a database and the AdminServer that OpenEdge Management is running in.

Managed versus scripted databases

A database recognized and managed by the AdminServer is called a managed database.

A database not managed by an AdminServer is called a scripted database. Scripted databases are administered outside the AdminServer using parameter files (`.pf`) and operating system-dependent scripts.

You can elect to migrate a scripted database to a managed database. To migrate a scripted database to a managed database, perform the steps in Migrating a database on page 18.

For details about encryption in general, and starting and monitoring encrypted databases, see *OpenEdge Data Management: Database Administration*.

Migrating a database

After you initially configure OpenEdge Management, you can use the Database Migration utility at any time to migrate a scripted database to a managed database.

To migrate a database:

1. On the OpenEdge management console menu, click **Resources** > **New** > **Database**.

   The **Database Migration Utility** page appears.

2. Enter the database information, and click **Submit**.

Note: For additional information about migrating a database, see *OpenEdge Management and OpenEdge Explorer: Getting Started*.

The AdminServer now controls the migrated database. Use the AdminServer or the DBMAN utility to start and stop the database. If you managed the database with scripts before its migration, do not use these scripts after the migration. If you use these scripts, OpenEdge Management will not recognize your database. OpenEdge Management can only recognize databases run by the AdminServer.

Starting a remote managed database agent

If the AdminServer managing the remote database is configured as an AdminServer (a named instance of an AdminServer that is running OpenEdge Management or has been configured to be monitored by OpenEdge Management), you do not need to take any steps to start the database agent. It will autostart whenever the broker starts. However, if the AdminServer is not configured as an AdminServer, you must edit the `conmgr.properties` file to start the database agent for a remote, managed database.

To enable a remote managed database agent:

1. Shut down the AdminServer.
2. Open the database's `conmgr.properties` file. This file is located in *`OpenEdge-install-dir`*`/properties`.
3. Add the following to the `conmgr.properties` file:

```
[agentremoteconnection]
agentremotesupport=true
```

4. Save the `conmgr.properties` file.
5. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in a grid frame.

6. Filter or search for, and select the remote database agent you want to enable monitoring.
7. Click **Configuration** in the **Command and Control** section.

   The **Configuration** page appears.

8. Click the link for the default configuration in the **Configuration and Server Group Links** section.
9. Click **Edit**.
10. In the **Agent** category, select the **Monitored** option.
11. Click **Save**.

Note: If the machine hosting OpenEdge Management is rebooted after the AdminServer is shut down, remote database agents do not reconnect to OpenEdge Management . You must restart remote database agents after rebooting the machine hosting OpenEdge Management

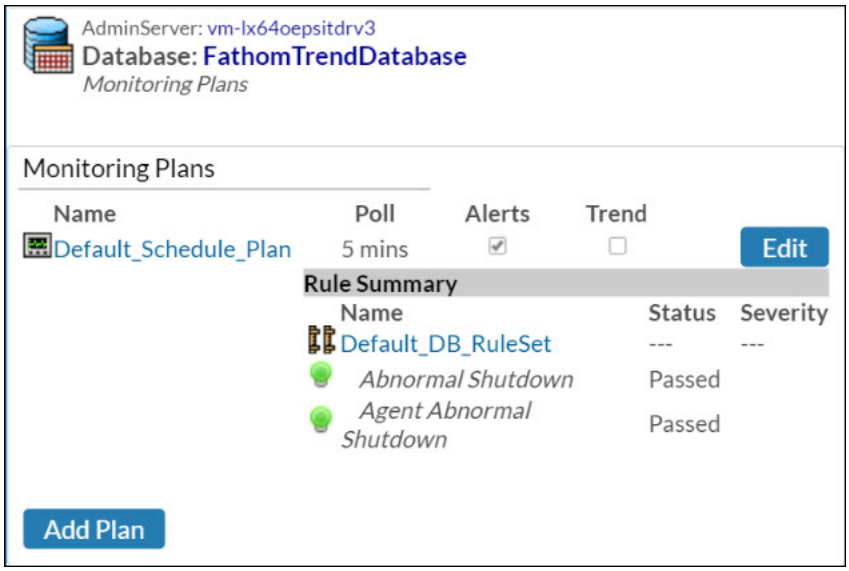
Monitoring managed databases

OpenEdge Management needs a monitoring plan before it can monitor a database. OpenEdge Management automatically creates a default monitoring plan for each managed database. The default monitoring plan contains a default rule set and schedule. You can change the default values at any time, or you can create a new monitoring plan.

Default database monitoring plans

OpenEdge Management provides a default monitoring plan for each managed database. The following figure shows a sample default database monitoring plan.

Figure 1: Sample default database monitoring plan



Note that the name of the database resource is displayed preceded by the AdminServer name. In this example, the **FathomTrendDatabase** is contained by **vm-lx64oepsitdrv2**.

The default monitoring plan shown above consists of the values described in the following table.

Table 1: Monitoring plan default values

Field	Default value	Description
Name	**Default Schedule Plan**	Identifies the name of the system-defined 24/7 default schedule that is used when the plan is active.

Field	Default value	Description
Poll	**5 mins**	Identifies the polling cycle set up for this database resource monitor. The polling cycle is the frequency at which the resource's rules are checked.
Alerts	**Enabled**	Indicates whether alerts are active and if they will be generated when the plan is active.
Trend	**Enabled**	Indicates whether the data monitored while this plan is active will be stored in the OpenEdge Management Trend Database. (This option is not initially set as a default value unless you chose it when you added the database to OpenEdge Management.)
Rule summary	**Default DB RuleSet**	Consists of two rules: **Abnormal Shutdown** and **Agent Abnormal Shutdown**.

Note: OpenEdge Management prevents the assignment of schedules that share overlapping time periods. For example, if you have a Default Schedule set up for a resource monitor, you cannot set up an additional plan because the Default_Schedule is defined for 7 days a week, 24 hours a day. In order to add other plans you must modify or remove the Default_Schedule from the plan.

Modifying database monitoring plans

You can modify the default database monitoring plan or other database monitoring plans you create.

To modify a database monitoring plan:

1. Click **Resources** > **Go to Resources** in OpenEdge Management console menu.

   All resources managed by your console appears in the grid frame.

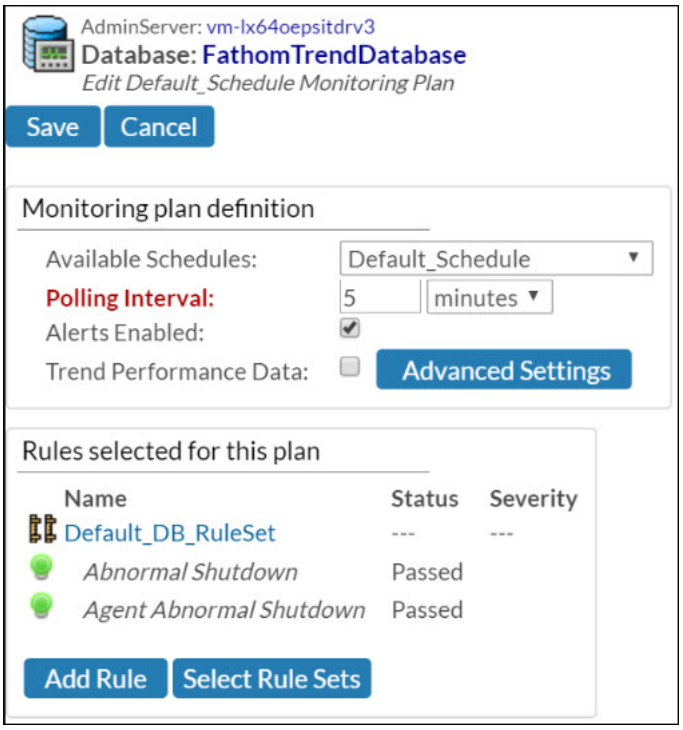
2. Filter or search for the database whose monitoring plan you want to update. The **Database** details page appears.

3. Click **Monitoring Plans** in the **Command and control** section of the **Database** details page.

   The monitoring plans for the specified database appear.

4. Click **Edit** for the schedule associated with the plan that you want to update.

   When the **Default Schedule Plan** is selected, the following **Edit Default_Schedule Monitoring Plan** page appears:



5. Edit the values in the **Monitoring plan definition** section as needed.

   To see or modify current trend value settings, click **Advanced Settings**. Then click **Save**.

6. Click **Add Rule** in the **Rules selected for this plan** section of the page. The **Select Database Rules** page appears.
7. Click the rule you want to add and update the values you want to edit. Click **Save**. The **Select Database Rules** page reappears.

   Note: Any rules you define and add are associated only with this particular plan. If you create another plan and add the same rules, you can select values that are appropriate for that particular plan.

   ---

8. Repeat Step 7 on page 22 for each additional rule you want to apply to this plan. After you add and define the criteria for each rule you want to add, click **Done Adding Rules** on the **Select Database Rules** page. The **Edit** page for your schedule reappears.

   You can also add rule sets to your plan by clicking **Select Rule Sets** in the **Rules selected for this plan** section of the page (shown in Step 4 on page 21). See Associating a rule set with a database monitoring plan on page 28 for information about adding rule sets.

9. Click **Save**. The updated monitoring plan appears in the monitoring plan definition at the top of the **Monitoring Plan** summary page.

Monitoring scripted databases

To create a monitoring plan for a scripted database, you must do the following:

- Add a resource monitor for a scripted database.

- Start a dbagent for the scripted database. For details about how to enable a scripted database agent, see Starting a scripted database agent on page 23.

To create a resource monitoring plan for a scripted database:

1. Click **Resources** > **New** > **Scripted Database** in the OpenEdge Management console.

   The **Create Scripted Database Monitor** page appears.

2. Provide the following information:

   - The display name for the database. The name must be unique among all scripted databases you are monitoring.
   - The description of the database (optional).
   - The host name. This name must be a valid name that can be resolved.
   - The host's IP address (optional).
   - The TCP/IP version (IPv4 or IPv6).
   - The name of, and the full path to the database.

3. Select the **Enabled** check box to enable the resource monitor.
4. Click **Save**.

   The default monitoring plan page appears. You can modify the settings if you choose. The steps for editing scripted database monitoring plans are the same as for managed database monitoring plans. See Monitoring managed databases on page 20 for the procedure.

Until you start the monitoring agent, the database's status appears as **Not Running**. You can use the available buttons to edit, copy, or delete the scripted database monitor from OpenEdge Management. For details about how to enable a scripted database agent, see Starting a scripted database agent on page 23.

Starting a scripted database agent

Before you can monitor a scripted database, you must start the Monitoring Agent.

To enable a scripted database agent:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the database whose monitoring agent you want to start.

   The **Database** details page appears.

3. Click **Control**.
4. Copy the command line provided. The following sample from the **Scripted Database** page shows the command line values to copy:



   The command line includes the following information:

   - The name and the path of the scripted database (`-db <database-name>`)
   - The host machine on which OpenEdge Management is running (`-H <host-name>`)

- The port on which OpenEdge Management is listening (`-S <port-number>`)
- The TCP/IP version number (`-ipver <ipversion-number>`)

5. Paste the command line at the **proenv** prompt on the machine hosting the database. You can now access the scripted database's control information and views.

Note: Using the command line supplied this way, add the start of your dbagent to your database start script. Each time your database starts up, the dbagent is automatically started and OpenEdge Management automatically begins monitoring your database.

Because the database is scripted, the link to the **File Systems** view that is normally available on a local database's details page does not appear.

Understanding and using database rules

OpenEdge Management supplies each managed database with a default rule set and also provides over 20 different database rules that you can define for a monitoring plan. These rules allow you to customize your database monitoring plans so that OpenEdge Management monitors your databases for specific conditions.

The following figure shows the **Select Database Rules** page.

Figure 2: Select Database Rules page



Only the rules that are not already part of the monitoring plan appear in the list. When you select any of these rules, the specific criteria associated with each rule appear. You can modify the default values associated with each rule.

Viewing rules associated with a monitoring plan

To see a list of rules associated with a monitoring plan:

- Click **Edit** on the **Monitoring Plan** page. The **Edit Monitoring Plan** page appears.

  For each rule shown in the **Rules selected for this plan** section, the following details are provided:

  - A colored dot, preceding the rule name, that indicates the status associated with each rule. See *OpenEdge Management: Resource Monitoring* for general resource status information.
  - The status represented by the colored dot.
  - The alert severity for each rule that has failed.

The following table lists possible rule status values.

Table 2: Resource status legend

Status	Dot color	Description
Pass	Green	The resource monitor is currently working.
Fail	Red	The most recent test involving the associated resource failed. This includes statuses such as tardy, timeout, or unreachable, for example. Check the **Alert Summary** page or the specific monitor for possible alert details. This status can also identify an internal error that prevents the resource from being monitored.
Not checked	Yellow	The resource monitor's status is currently unknown. For example, if system startup just occurred, it is possible that the resource has not been polled yet.
Not running	Blue	The resource is currently not running. Watch resources such as databases and servers for this status, as they must be running before you can monitor them.
Disabled	Dark gray	The resource monitor is disabled and is not currently monitoring a resource.
Inactive	White	There is no active monitoring plan.
Offline	Light gray	The resource is currently offline.

Working with database rule sets

A database rule set is a set of rules that you associate with one or more database resources through a monitoring plan. Database rule sets are stored in the **OpenEdge Management Component Library** and provide a way for you to manage many databases by sharing rule definitions. In this way, you create a common set of rules that you can set up for multiple databases.

Because each database rule set you create is stored in the **OpenEdge Management Component Library**, the rule set is available for your reuse or for use by others.

You can also add individual rules to a monitoring plan, whether or not the rules are part of any rule set. If you include a rule in a monitoring plan's rule set and then add the same rule individually again with modifications, the rule in the rule set is overridden by the rule with the modifications.

OpenEdge Management provides a default database rule set. When a database resource is added to OpenEdge Management, a default monitoring plan (with the default rule set) is assigned to it.

Creating a database rule set

When associating rules sets with database resources, you can use the default rule set provided by OpenEdge Management or you can create your own rule set.

To create a database rule set from the OpenEdge Management console menu:

1. Choose one:
   - Click **Library** > **New** > **Database Rule Set**.
   - Click **Library** > **Go to Library** > **Create Database Rule Set**.

   The **Create new Database Rule Set** page appears.
2. In the **Name** field, enter the name of the rule set (no spaces allowed).
3. In the **Description** field, enter a brief description of the rule set.
4. Click **Save**. The **Database Rule Set** page appears. (The rule set is also visible in the list frame under **Rule Sets** > **Database**.)
5. Click **Add Rule** to select the rules you want in the rule set. The steps for adding rules to a set are the same for adding rules to a monitoring plan. See Modifying database monitoring plans on page 21 for information about adding rules.

Note the following about rule sets:

- Once you create a rule set, you can edit, copy, or delete it.
- If you add a rule or a rule set to an existing rule set, the change affects **all** databases using that rule set.

Editing a database rule set

You can change the name or description of a rule set.

To edit a rule set:

- Click **Edit** on the **Database Rule Set** page.

Note: You can access the list of existing database rule sets at any time from the **OpenEdge Management Component Library** list frame.

Copying a database rule set

You can copy a database rule set and make modifications to it. You must give the copy a unique name.

To copy a database rule set:

1. From the **Database Rule Set** page, click **Copy**. The **Copy Database Rule Set** page appears.
2. Rename the copy and (optionally) change the description.
3. Click **Save**.

You can now add one or more rules to the copy from either the **Copy Database Rule Set** page or the **Database Rule Set** page.

Deleting a database rule set

You can delete a rule set as long as it is not currently associated with any resource monitoring plans.

To delete a rule set:

1. From the **Database Rule Set** page, click **Delete**.
2. Click **Yes** in the confirmation dialog box to confirm deletion.

Rule sets with one or more rules in common

You can have multiple rule sets associated with a monitoring plan. If you edit one of the rule sets, evaluation of only the first occurrence of any identically named rules takes place when the resource is polled. Which occurrence is considered "first" is determined by the alphabetic order of the rule set.

Associating a rule set with a database monitoring plan

You create a database rule set to associate and use it with one or more monitoring plans. Once you establish the association, the rule set is active for the database whenever the monitoring plan is active.

To associate a database rule set with a database monitoring plan:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.
2. Filter or search for, and select the database for which you want to associate rule set with the monitoring plan.

   The **Database** details page appears.
3. Click **Monitoring Plans** in the **Command and Control** section.

   The **Monitoring Plans** page appears.
4. Click the monitoring plan you want to update and click **Edit**.

   The **Edit Monitoring Plan** page appears
5. Under **Rules selected for this plan**, click **Select Rule Sets**. A list of available rule sets appears.
6. Select one or more rule sets you want to associate with the plan.
7. Click **Save** when you finish. The monitoring plan is updated, and the **Monitoring Plan** page reappears.

Using the Configuration Advisor

The Configuration Advisor calculates suggested threshold values for rules by analyzing trend data. When you apply a recommended rule threshold setting, the alerts triggered as a result of rule violations provide a more meaningful indication of your resource's performance than if you were to arbitrarily set rule values.

The Configuration Advisor analyzes a rule's past performance for a specified period of time and, based on this data, indicates a baseline value. A *baseline value* is a number that serves as the foundation for calculating a set of possible threshold settings based on your system's past activity for a specific rule.

It is recommended that you use the Configuration Advisor-related default values for a set period of time (for example, one week) to capture rule data to the OpenEdge Management Trend Database. This initial step will provide you sufficient data with which to perform the comparison.

Keep the following points in mind before you run the Configuration Advisor for the first time:

- Each rule you want to analyze must have at least 32 data samples stored in the OpenEdge Management Trend Database. This sampling provides sufficient data from which the Configuration Advisor can determine a baseline value and subsequently perform a successful analysis of each rule's data.
- A polled rule must currently be associated with an active monitoring plan for it to be a candidate for processing by the Configuration Advisor.
- If your monitoring plan contains after-imaging (`AI`) rules but you do not have after-imaging enabled for the database, do not select `AI` rules for analysis. When a monitoring plan contains `AI` rules and `AI` is not enabled for the database, the value stored in the OpenEdge Management Trend Database is zero. The Configuration Advisor works by contrasting the OpenEdge Management Trend Database data against the rule's range of values. Since zero falls outside the range of values, the Configuration Advisor's analysis becomes meaningless and it returns a baseline value that is meaningless.

Initiating the Configuration Advisor

To initiate the Configuration Advisor:

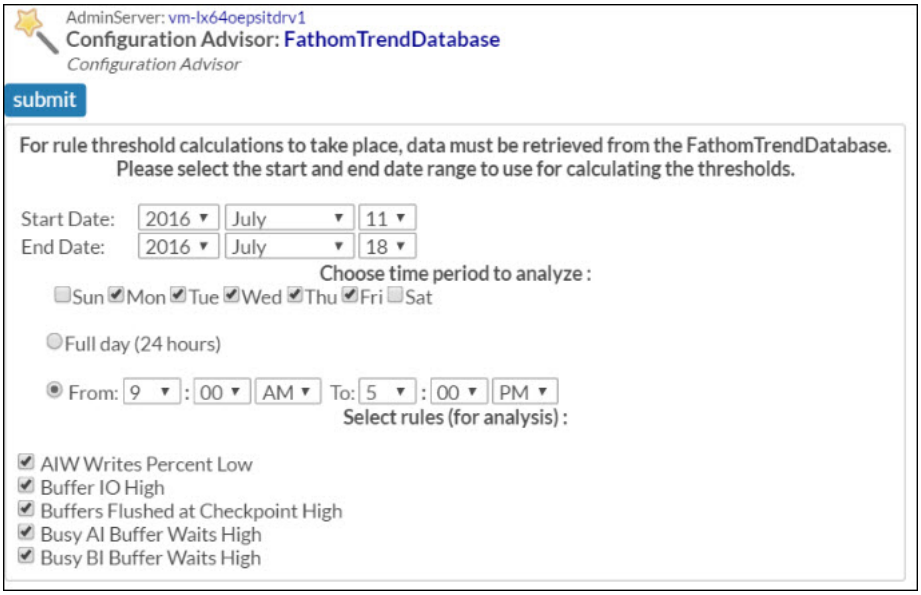
1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the database for which you want to initiate the Configuration Advisor.

   The **Database** details page appears.

3. Click **Configuration Advisor** in the **Command and Control** section. The **Configuration Advisor** page appears:



4. In the **Start Date** and **End date** fields, define a date range that OpenEdge Management will use to collect data from the OpenEdge Management Trend Database. (The default date range is one week.)

5. In the **Choose time period to analyze** field, choose a time frame that defines a representative period of time in which the rules are generally active, or being used. This time frame is the period against which you want OpenEdge Management to calculate your baseline activity. (The default time period is Monday through Friday, 9:00 AM to 5:00 PM.)

   Note: Your monitoring plan schedules are not necessarily the best choice for a time frame. A schedule defines a period of time in which rules are in effect; it does not focus on time periods in which your resource usage is highest. For example, you may use the default schedule (24 hours a day, 7 days a week) to monitor your system, but you may want to select Monday through Friday from 8:00 AM to 6:00 PM for calculating your baseline settings.

6. In the **Select rules (for analysis)** field, deselect any rules you do not want to analyze. The Configuration Advisor can analyze the following database resource rules:

   - AIW Writes Percent Low
   - BIW Writes Percent Low
   - Buffer IO High
   - Buffers Flushed at Checkpoint High
   - Busy AI Buffer Waits High
   - Busy BI Buffer Waits High
   - Checkpoint Length Short
   - Database Commits Low
   - Empty AI Buffer Waits High
   - Empty BI Buffer Waits High

   Only rules associated with a monitoring plan appear in the **Select rules (for analysis)** list.

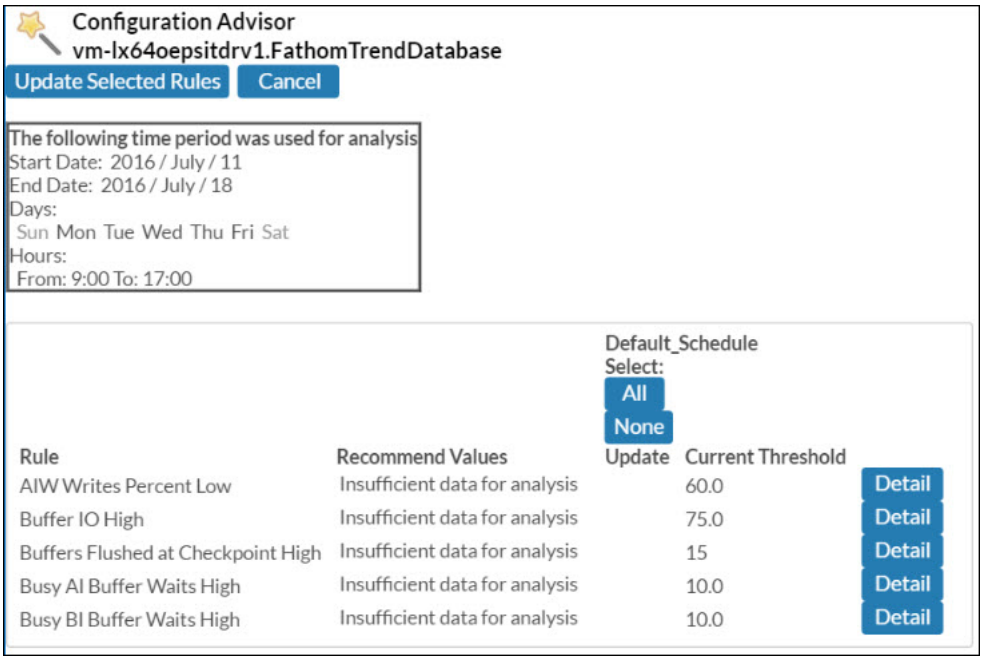
   Note: If you added `AI` rules to a monitoring plan but do not have `AI` enabled on your database, do not select the `AI` rules for analysis.

7. Click **Submit**.

   Note: Once you click **Submit**, you can browse to other OpenEdge Management pages and perform other tasks. You can return to the Configuration Advisor later to check the status and result details.

   The Configuration Advisor calculates the threshold settings and reports the progress of each calculation it is performing. Depending upon the criteria that you set on the initial **Configuration Advisor** page, the number of rules you selected, and other factors such as your machine's speed, this calculation process can take time.

   When the Configuration Advisor completes its calculations, the following page appears:



The following time period was used for analysis section of this page summarizes the values defined on the initial Configuration Advisor page. These values are displayed here to remind you of the time period criteria you set.

The **Rule** section contains all the rule-related calculated data. You can now review the Configuration Advisor's recommendations and choose whether or not to update your threshold values. See Understanding the recommended threshold settings on page 31 for more information about the Configuration Advisor's calculated values.

Note: The Configuration Advisor page displays the calculated results data until you click either **Update Selected Rules** or **Cancel**.

Understanding the recommended threshold settings

When you view the Configuration Advisor's analysis, each analyzed rule appears as an individual line item in the **Rule** section. Associated with each rule is a **Recommended Values** drop-down list. The list contains either of the following entries:

- Numeric values that identify the recommended rule threshold settings. This list can contain up to several different numeric items. Collectively, these values comprise the range of recommended threshold settings.
- An **Insufficient data for analysis** message. The Configuration Advisor displays this message when the criteria to perform the data analysis successfully is not met.

Reviewing recommended values

The Configuration Advisor displays a range of possible values from which to select. The initial value that appears in the **Recommended Values** field indicates the primary recommended threshold setting, based on the data analysis process.

Each recommended value is expressed as a set of two numbers. The first number specifies the recommended threshold setting. The second number, displayed in brackets, identifies the number of times this threshold value, if used with the collected data, would have broken the rule and triggered an alert. As you review the recommended threshold settings, keep in mind the rule behavior and alert notification frequency you want to establish for a resource.

Using the Detail button

Each row has an associated **Detail** button that displays information about the rule's analysis. The **Detail** page for a rule for which there is insufficient data for analysis identifies the number of samples found. This number is always lower than the minimum of 32 data samples required. Review this data to help you decide if you need to expand the time period to try and capture more samples and rerun the Configuration Advisor for a given rule.

Comparing and selecting threshold settings

By default, the Configuration Advisor assumes that you are going to select and submit one of the recommended threshold settings; it provides a check mark in the **Update** field for each rule row. However, you have options concerning the selection process. As you compare the existing and recommended values, you can elect to change none, some, or all values for a rule and each individual monitoring plan.

This section provides the procedure you can use to compare the current rule setting with the recommended threshold settings, and to update each schedule with your specific selections. Perform this comparison to help you determine your final selection.

To compare and select threshold settings:

1. For a specific row, note the value that displays in the **Current Threshold** field under a specific schedule.
2. Click **Recommended Values** to display the range of recommended values for the associated rule.
3. Compare the possible **Recommended Values** that display with the value in the **Current Threshold** field. As you determine the best threshold rule setting, keep your goals for this rule in mind. Also, consider any additional selection criteria as you compare values.
4. Repeat these steps for each rule and its associated monitoring plan.

Additional selection criteria

Some of the reasons for selecting one value over another are:

- How often you want alerts generated
- Any factors unique to your resource's performance that you want to consider when making your selection
- Your knowledge of your system's operational needs and goals

Submitting your threshold setting selections

When you click **Update Selected Rules**, OpenEdge Management applies all of your selections at the same time. There is no undo option associated with this group submission. To reset any values back to a previously defined setting, you must access the resource's monitoring plan, display the individual rule, and override the displayed value.

Note: The Configuration Advisor page displays the calculated results data until you click either **Update Selected Rules** or **Cancel**.

Determining the effectiveness of your selections

The most effective way to determine if your threshold adjustments are meeting your needs is to review your alert notifications. Strive for a threshold setting that is consistent with your resource and business needs. If you receive alerts too frequently or infrequently for your operational needs, you might want to further refine your threshold settings.

2

Graphical Displays of Database Data

OpenEdge Management provides a graphical display of database data in several different areas of the product. Graphical displays of data allow you to check the status of your database resources at a glance. This chapter provides details about OpenEdge Management's graphical display of data.

For details, see the following topics:

- Interpreting graphical database data
- General graph information

Interpreting graphical database data

Several of the database views provided by OpenEdge Management include both a text-based and a graphical interpretation of data. Graphs appearing in OpenEdge Management database views belong to the historical family and the graphical interpretation is in the form of line chart, area chart, column chart, stacked area chart, or stacked column chart. Historical graphs show values over a period of time and use dashed lines to indicate thresholds.

Each graph includes a legend that defines what the different graph colors represent and shows maximum, minimum, average, and last graph values. You can click the legend to show or hide the graph values.

General graph information

When looking at graphs, keep the following in mind:

- The production of graphs is CPU-intensive. If you are monitoring CPU usage, an alert might fire when the graph is generated. To avoid firing such an alert, increase the number of failed polls before OpenEdge Management throws an alert.
- Only resources that have defined monitors produce information that is graphed.
- A yellow header in the viewlet graph on the **Dashboard** page indicates that the resource with which the viewlet is attached is not checked in. A red header in the viewlet graph indicates that the resource is offline. For more information about viewlets and the **Dashboard** page, see *OpenEdge Management: Resource Monitoring*.
- Data averaging is displayed on all graphs and appears in the graph's legend.
- Any discontinuity in the caching of a database resource's data will be reflected in the graphs. Caching discontinuity occurs when a resource goes off line or a resource monitor is disabled. Discontinuities are obvious in line graphs, as they appear as a break in the line. Note that caching discontinuities are harder to distinguish on area or column graphs, since gaps in area or column graphs can indicate either a zero value or caching discontinuity.
- All graph data is stored in the OpenEdge Management work directory's `cachedb` folder (by default, in `$OEMgmt\work`). If OpenEdge Management is not running, you can delete the database. Doing so, however, will cause you to lose all of your cached graph data.
- By default, 15 days of graph data is stored for each database resource. You can choose a different setting on the **Graph Cache Database Configure** page.

  You can access the **Graph Cache Database Configure** page by clicking **Options** > **Graph Cache** on the OpenEdge Management console. See *OpenEdge Management and OpenEdge Explorer: Getting Started* for more details about setting the graph cache.
- All graphs except viewlets have three options in the top-right corner to customize the graphs. Using these options you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation. For information about customizing viewlets on the **Dashboard** page, see *OpenEdge Management: Resource Monitoring*.

3

Examining Data from an OpenEdge Database

Your database is one of the most critical operational resources you maintain at your site. OpenEdge Management provides you with several ways to view and analyze data collected from your database resources so you can better manage them.

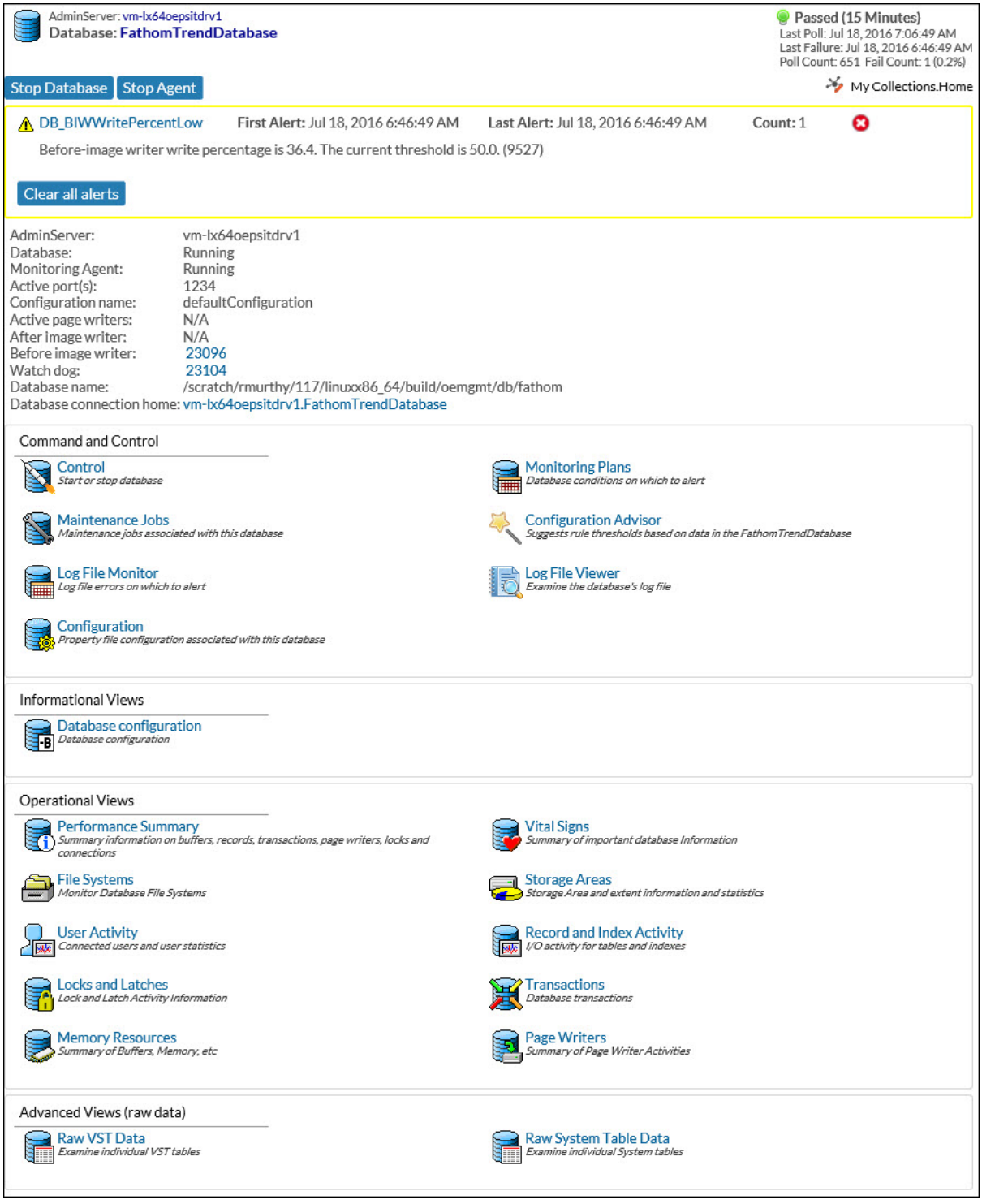
For details, see the following topics:

- Accessing database information
- Changing database control settings
- Database log file monitors
- Using the log file viewer
- Understanding OpenEdge database views

Accessing database information

The **Database** details page, shown in the following figure, provides access to detailed information about an individual database.

Figure 3: Database details page



To access the **Database** details page:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the database whose details you want to view. The **Database** details page appears.

   The following information is available on the **Database** details page:

   - Five of the most recent alerts are displayed on the page. For more information about alerts on a resource home page, see *OpenEdge Management: Alerts Guide and Reference*.
   - Summary information of the database including a **Database connection home** link to access the **Database Administration Console** page and manage the database.
   - Different sections to control, configure, monitor, and view the database and its properties.
   - Links in each section of the page to access relevant details about the database. A brief description of each link on the **Database** details page appears below the link itself.

   Note: If the dbagent is connected to an AdminServer running on a different machine from the dbagent itself, the **Log File Monitor** link on a scripted database's **Database** details page is disabled.

   The following information is available in the upper-right corner:

   - The database's current status, and how long it has been at that status
   - The date and time of the last poll
   - The number of polls taken
   - The number and percent of failed polls

Changing database control settings

From the **Database** details page, click **Control** in the **Command and Control** section to access the **Database Control** page.

The **Database Control** page provides information about the current database and database agent settings and allows you to start and stop a managed database and agent. For a scripted database, the page allows you to see the database status and the remote monitoring agent status, and to copy the command line that you use to start the dbagent on the remote machine. You can also stop the remote monitoring agent from a scripted database's **Database Control** page.

You can perform the following from the **Database Control** page of a managed database:

- Stop the database by clicking **Stop Database**; the status is updated to **Not Running**. To restart the database, click **Start Database**. (This is the same button—the text changes to display the opposite status of the one currently set.)

  Note: The database agent, or monitoring agent, works only if the database is running. Therefore, when you stop a database, the monitoring agent status automatically changes to **Not Running**. However, when you start the local database, the agent does not automatically start.

- Change the status of a database monitor from **Enabled** to **Disabled** by clicking **Edit** and clearing the **Enabled** check box. If you disable database monitors, the database still appears under **Databases** in the list frame. However, the database is no longer being monitored by OpenEdge Management.

  If you delete a database from within OpenEdge Management, you have permanently removed it from the console. It is preferable to disable the database (rather than delete it) because you have more options to use the resource at a later time. For example, you could copy and reuse the database's configuration for a new database you create, or you could migrate the database using the Database Migration Utility. You can also delete it and add it again in the future if necessary.

  Note that you can still continue to run historical reports against a database you delete from OpenEdge Management.

Database log file monitors

OpenEdge Management automatically creates a database log file monitor for each managed database started through OpenEdge Management.

The database log file monitor can help you:

- Ensure the integrity of database log files by monitoring files for errors and allowing you to define actions to trigger when errors occur.
- Use predefined database-related search criteria, or create your own criteria, to run against the data in your database file. You create and maintain search criteria in the OpenEdge Management Component Library.

Specifically, the predefined search criteria provide detailed data about the recorded operations of your database and offer you a means to extract this data. This data can help you effectively respond to and manage your database.

Note: You can also create a log file monitor for use with files such as log files, text files, or any non-OpenEdge log file. The search criteria feature works the same in each of these resource types. For information about the log file monitor and detailed information about search criteria that pertain to these resource types, see *OpenEdge Management: Resource Monitoring*.

Log file monitor default values

The predefined database log file monitor that OpenEdge Management creates for each database contains several default values. The only default database log file monitor property you can modify is the **Enabled** property (and its description). See Customizing a database log file monitor on page 41 for additional information.

The default values are as follows:

- The **Bookmark** is unique, and it is set to **Last Line**.
- The database default log file monitor is disabled until the database is first started.
- The **On First Poll** property is set to **Search From End**.

Log file monitor considerations

Before you customize a database log file monitor, review the following points:

- The database log file monitor is not enabled until the database for which it was created is started. When the database log file monitor first starts monitoring a database, it always starts at the end of the log file.
- The database log file provides predefined search criteria that address common database-related events. You can use these searches as defined, or you can copy and customize them.
- You can also create your own search criteria. For example, if there is a particular database error for which you want to monitor a database, you can use the promsg number as the search text.
- The OpenEdge Management Trend Database is a managed, OpenEdge database. Therefore, it has a predefined database log file monitor that you can customize to meet your needs.

Customizing a database log file monitor

The database log file monitor that OpenEdge Management creates for each database contains default values. You can use these values, or you can customize the log file monitor to suit your needs.

Note: This guide presents the basic steps needed to customize a log file monitor and does not describe search criteria in detail. See *OpenEdge Management: Resource Monitoring* for detailed information about creating and changing search criteria.

To customize a database log file monitor:

1. From the **Database** details page of the selected database, click **Log File Monitor** in the **Command and Control** section.

   The **Log File Monitor for: <database name>** page appears.
2. In the **Monitoring Plans** section, choose one:
   - Click **Add Plan** to add a monitoring plan to this resource monitor.

     Note: OpenEdge Management prevents the assignment of schedules that share days or have times that overlap. For example, if you have a Default_Schedule set up for a resource monitor, you cannot set up an additional plan because the Default_Schedule is defined for 7 days a week, 24 hours a day. You must modify or remove the Default_Schedule to set up additional plans.

   - Click **Edit** against an existing plan to customize.
3. Set or change the polling interval.
4. Select the **Alerts Enabled** check box to enable alerts.
5. For a new plan, click **Save**.
6. Click either **Add Rule** or **Select Rule Sets** in the **Rules selected for this plan** section.
7. After you either add a rule or select from the list of predefined rule sets, click **Save** to return to the monitoring plan detail page.

Using the log file viewer

The log file viewer allows you to examine various log files through an HTML interface. This section describes viewing database log files. You can use the log file viewer from OpenEdge Management or OpenEdge Explorer.

To access and use the database log file viewer:

1. Click **Log File Viewer** in the **Command and Control** section of the **Database** details page. The viewer appears.
2. You can work with the Log File Viewer in the following ways:

   - Use the **Show** field to control how many database log file entries display at one time. The number entered into the **Show** field cannot be less than 10.
   - Use the **Overlap** field to control how many entries are repeated from screen to screen.

     Note: The value in the **Overlap** field cannot be more than the number in the **Show** field minus one. For example, if you show 30 entries, you can overlap only 29 or fewer of them.

   - Click **Reload** after changing the values in the **Show** field or **Overlap** field. If you do not reload, the viewer continues to display the previous values.
   - Click **Go To** to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the **Go To** field will begin the display from the tenth log file entry.

     Note: You must click **Go To** after entering a value in the **Go To** field or the viewer will not update its display.

   - The default display of entries is in ascending order; choose **Descending** to change the display. Note that the **Show** field dictates the number of entries shown, regardless of whether they display in ascending or descending order.
   - Click **First** to display the first *x* entries, where *x* is the value in the **Show** field.
   - Click **Previous** to display the previous *x* entries, where *x* is the value in the **Show** field.
   - Click **Next** to display the next *x* entries, where *x* is the value in the **Show** field.
   - Click **Last** to display the last *x* entries, where *x* is the value in the **Show** field.
   - To view additional log file entries without changing your current starting log file entry, leave the **Go To** field blank, change the value in the **Show** field, and click **Reload**.
   - If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the **Log file status** field.
   - OpenEdge Management considers a viewer that has been inactive for more than four hours stale and at that point releases ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management automatically reloads the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.
   - OpenEdge Management considers a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management releases all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.

You can also use the log file viewer with resources other than the database, as listed in the following table.

Table 3: Log file viewer documentation

For information about . . .	See the . . .
AppServer broker and server logs	*OpenEdge Management: Servers, DataServers, Messengers, and Adapters*

For information about . . .	See the . . .
NameServer broker log	*OpenEdge Management: Servers, DataServers, Messengers, and Adapters*
WebSpeed broker and server logs	*OpenEdge Management: Servers, DataServers, Messengers, and Adapters*
DataServer broker and server logs	*OpenEdge Management and OpenEdge Explorer: Configuration*
AppServer Internet Adapter log	*OpenEdge Management and OpenEdge Explorer: Configuration*
WebSpeed Messenger log	*OpenEdge Management and OpenEdge Explorer: Configuration*
Web Services Adapter log	*OpenEdge Management and OpenEdge Explorer: Configuration*
OE Web Server log	*OpenEdge Management and OpenEdge Explorer: Configuration*
SonicMQ Adapter broker and server log	*OpenEdge Management and OpenEdge Explorer: Configuration*

Understanding OpenEdge database views

The various database views display data about key database components so that you can assess what is happening with the databases. The database agent gathers data for the views by polling the database's Virtual System Tables (VSTs). (View data does not come from the OpenEdge Management Trend Database.) For more information on VSTs, see *OpenEdge Data Management: Database Administration*.

Views capture information at a specific point in time. Because database status can change from moment to moment, refresh the detail frame to keep the view's data current. All views display the date and time they were built.

Different database views appear in different formats; some include graphical displays of data. Views created for scripted databases do not contain graphical displays of data. For more information about graphical displays of data, see Graphical Displays of Database Data on page 35.

Some views allow you to narrow the scope of data by clicking on any underlined subject. Note that underlined subjects behave differently than the underlined column headings. Clicking an underlined column heading does not bring you to a more detailed view of the subject; instead, it changes the column's sort order. If the default sort order for a column is ascending, clicking the heading will change the sort to descending. If the default sort order is descending, clicking the heading will make it ascending. Note that when you change the sort order, the page automatically refreshes.

Database views are organized into three overall categories: **Informational** views, **Operational** views, and **Advanced** views. The following table lists and describes all database views, and indicates whether the view can be created for both managed and scripted databases or only for managed databases.

To access the database views, click **Resources** > **Go to Resources** and select the required database using the filtering or search options. You can find the following database views on the **Database** details page:

Table 4: Database views

View name	Database type	Description
Database configuration		Configuration details of the database
Performance Summary		Summary information of buffers, records, transactions, page writers, locks, and connections
Vital Signs	Managed and scripted	Summary of important database information
File Systems	Managed	List of file systems and associated files that the database uses (not available for remote databases)
Storage Areas	Managed and scripted	Storage area and extent information and statistics
User Activity	Managed and scripted	Information about connected users and user statistics
Record and Index Activity	Managed and scripted	I/O activity for tables and indexes
Locks and Latches	Managed and scripted	Information about lock and latch activity
Transactions	Managed and scripted	Information about database transactions
Memory Resources	Managed and scripted	Summary of buffers, memory, etc.
Page Writers	Managed and scripted	Summary of page writer activities
Start Parameters	Managed and scripted	Detail about the database start parameters
General Details	Managed and scripted	General information about the database, such as start, backup, etc.
Raw VST Data	Managed and scripted	Examination of individual VST tables
Raw System Table Data	Managed and scripted	Examination of individual system tables

Database Configuration view

The **Database configuration** view provides general information such as details, features, and other parameters of your database.

From the **Database** details page, click **Database configuration** in the **Informational Views** section. The **Database configuration** view appears and you can find more information in the following sections:

- **Database Startup Parameters**
- **Database Features**
- **SQL Broker Parameters**
- **Internationalization Parameters**
- **Statistics**
- **Schema Information**
- **Performance Parameters**

Note: The **Database configuration** view does not allow you to modify your database configuration properties. For modifying your database configuration properties, see *OpenEdge Management and OpenEdge Explorer: Configuration*.

Performance Summary view

The **Performance Summary** view provides summarized information about your database and all other operational views associated with it.

From the **Database** details page, click **Performance Summary** in the **Operational Views** section. The **Performance Summary** view appears and you can find more information in the following sections:

- **Database Summary**
- **Buffer Summary**
- **Record and Transaction Summary**
- **Writer Summary**
- **Lock Summary**
- **Connection Summary**
- **Replication Summary**
- **Area Summary**

Use the links in these sections to find historical data for particular information with time periods of 1 hour, 1 day, 1 week, and 4 weeks. To see the most recent information on the page, click **Refresh** or set an auto refresh interval.

Vital Signs view

The **Vital Signs** view shows a page of graphical information relevant to the critical operations of the database.

From the **Database** details page, click **Vital Signs** in the **Operational Views** section. The **Vital Signs** view appears and displays the following graphs:

- Transaction activity
- Checkpoint activity
- Buffer hit rate
- Record wait/read ratio
- Record activity
- AIW writes percent
- APW writes percent

These graphs indicate activity over time. For more information about graphs, see Graphical Displays of Database Data on page 35.

The data populating the **Vital Signs** view comes from many VSTs, including _ActAILog, _ActPWLog, _ActBuffer, _ActSummary, _UserIO, and _ActIOType. If the database is also configured for alternate buffer pools, polling retrieves the alternate buffer pool values from the VST _ActBuffer. Therefore, the **Buffer hit rate** details shown in the page reflect the total buffer hits for both primary and alternate buffer pools.

File Systems view

The **File Systems** view displays a list of file systems and associated files on a local machine that a specific database is using.

From the **Database** details page, click **File Systems** in the **Operational Views** section. The **File Systems** view appears and displays the following general file system and detailed storage area information:

- **Monitor** — The name of a file system monitor. Only those file systems that have a defined monitor display a specific monitor name. File systems that do not have a defined file system monitor currently set up display the entry **- define -** under the **Monitor** heading. See Creating file resource monitors on page 47 for more information.
- **Capacity** — The byte capacity for each file system.
- **Free** — The number of bytes that are currently free.
- **Used** — The number of bytes currently used.
- **Utilization** — The number of bytes currently used, expressed as a percent of the capacity. Percent amounts display one decimal place of precision.
- **DB Size** — The size of all extent files on that file system in bytes.
- **DB%** — The DB size as a percent of the capacity.
- **Extent** — The filename and path of the database's storage area extents.
- **Database** — The name of the database.
- **Area #** — The database storage area number.
- **Size** — The size in bytes.
- **% of file system** — The size as a percentage of the file system capacity.
- **Reads** and **Writes** — The number of block read/writes that have occurred since the database was started.

The database extent data on the **File Systems** view originates primarily from the _FileList VST. Data appearing in the **Size** column originates in the _AreaExtent system VST.

Note: If you access the **File System** view from the **OpenEdge Management Resources** page instead of from the **Database** Details page, OpenEdge Management displays identical data. The only exception is that when you access the **File System** view from the **OpenEdge Management Resources** page, OpenEdge Management shows **all** databases for a system.

Creating file resource monitors

You can set up or access an existing file resource monitor for any file system listed on the **File System** view page.

To set up a monitor:

- Select **-define-** next to the file system. The **Create new File System** page appears. Once you create a file system resource monitor, the resource name replaces the **-define-** label.

You create monitoring plans and rules for a resource monitor by following the procedures outlined in *OpenEdge Management: Resource Monitoring*.

Storage Areas view

The **Storage Areas** view allows you to view the status of your database storage areas.

From the **Database** details page, click **Storage Areas** in the **Operational Views** section. The **Storage Areas** view appears and you can find more information in the following sections:

- **System areas**
- **Application data areas**
- **Before imaging areas**
- **After imaging areas**.

Additionally, you can find graphs displaying the percentage of use for each area, the size of each area, and an area's percentage of reads and writes relative to the monitored database's reads and writes. For more information about graphs, see Graphical Displays of Database Data on page 35

While interpreting the data presented on the **Storage Areas** view page, remember the following points:

- Reads and writes are in database blocks.
- The percentage listed in the **Reads** column and the **Writes** column is the percentage of the database's total reads or writes.

  The absence of an entry in the **Reads** column or the **Writes** column indicates that no reads or writes have taken place.

- An entry of less than one percent in the **Reads** column or the **Writes** column indicates that reads or writes took place, but not enough to reach one percent.
- The **High Water Mark** (**HWM**) column indicates the greatest number of blocks used in a storage area. The following equation produces the percentage displayed in the **HWM** column: x/y, where x is the number of the HWM block and y is the total number of blocks existing in the storage area. For example, if your database's HWM block is 5 and its total number of existing blocks in the storage area is 8, the data displayed in the **HWM** column would be 63% (or 5/8=.625 rounded).

- The **Used** column indicates the amount of space used on a storage area's free chain. The following equation produces the percentage displayed in the **Used** column: $(x-z)/y$, where x is the number of the HWM block, z is the total number of free blocks in the area's extents, and y is the total number of blocks existing in the storage area.
- Double question marks in a data column indicate that the database agent could not access the needed information. See After Imaging areas on page 49 for more information about double question marks.
- Double exclamation points in a data column indicate that the table was unavailable at the time the **Storage Areas** page was built. For example, if the database agent is down and therefore unable to poll the database at the time of the view's creation, double exclamation points appear in the data columns.

System areas

The **System areas** section contains information about the control area and the schema area.

Click either **Control Area** or **Schema Area** to see more information in the following sections: **Summary details**, **Extent file details**, and **Extent file I/O details**.

Note: Although it is possible to create application extents in the schema area, OpenEdge Management shows the schema area in the **System area** view. When best practices are followed and all application extents are placed in their own areas, the schema area stores only system information.

The data presented in the **Control Area** and **Schema Area** views come from the _AreaStatus, _FileList, and _ActIOFile VSTs.

Application data areas

Names of the areas listed in the **Application data areas** section will vary according to the database being viewed.

Click an area to see more information in the following sections: **Summary details**, **Extent file details**, and **Extent file I/O details**.

Note: The statistical data displayed in the **Table Details** and **Index Details** sections are affected if the database being polled was started with the `-indexrangesize` and/or the `-tablerangesize` parameter. Note that both parameters have a default value of 50. If your database has 60 indexes but the `-indexrangesize` parameter is set for the default, the **Index Details** view will only be able to display data for the first 50 indexes. When the **Index Details** view tries to retrieve data from the _IndexStat VST and does not find any, it will return a value of double question marks (??). See *OpenEdge Deployment: Startup Command and Parameter Reference* for more information about the `-indexrangesize`and `-tablerangesize`parameters.

The application area data presented in the above view come from the _AreaStatus, _FileList, _ActIOFile, _TableStat, and _IndexStat VSTs.

Before Imaging areas

The **Before Imaging areas** section contains information about the primary recovery area.

Click **Primary Recovery Area** to see more information in the following sections: **Summary details**, **Extent file details**, and **Extent file I/O details**.

The data displayed in the **BI Storage Area** view come from the _AreaStatus, _FileList, and _ActIOFile VSTs.

After Imaging areas

The **After Imaging areas** section contains information about any defined after image areas.

Click an area to see more information in the following sections: **Summary details**, **Extent file details**, and **Extent file I/O details**.

Note: Because only one `AI` area can be active at a time, additional listed `AI` areas will have some data columns populated by two question marks.

The data displayed in the **AI Storage Area** view come from the _AreaStatus, _FileList, and _ActIOFile VSTs.

User Activity views

The **User Activity** view provides information about list of database connections, statistics of the connection, and user details and activities.

From the **Database** details page, click **User Activity** in the **Operational Views** section. The **User Activity** view appears and has the following tabs:

- **Database Connections**
- **User IO Activity**
- **User Lock Activity**
- **Connection Statistics**
- **User Details**

  In the **User Details** tab:

  - Provide a user number in the **User number** field and click **Show User** to find the details of an individual user.

    The details displayed vary depending on the user type. If the user is a self-service user, the tab allows you to view transactions and locks for that user.

  - To disconnect a user from the database, click **Disconnect User** and click **Yes** in the confirmation dialog box.

In each tab of the **User Activity** view, you can:

- Click the user number of a user to find its details in the **User Details** tab.
- Use the filtering options to filter user details by any given parameter.
- Click the column heading to choose the order of the filtered results (ascending, descending, or alphabetically).
- Move the page forward or backward, and specify the number of users per page.
- Use the **Refresh** icon to find the most recent data.

The data found in the **User Activity** view comes from the _UserIO and _UserLock VSTs.

Record and Index Activity view

The **Record and Index Activity** view presents an overview of your records and indexes.

From the **Database** details page, click **Record and Index Activity** in the **Operational Views** section. The **Record and Index Activity** view appears and you can find more information in the following sections:

- **Record summary**
- **Index summary**
- **Writes to DB**

  Click **Record** or **Index** for a more detailed view.

- **Reads from DB**

  Click **Record** or **Index** for a more detailed view.

The information displayed in the **Record and Index Activity** view comes from the _TableStat and _IndexStat VSTs.

Locks and Latches views

The **Locks and Latches** view provides summary of locks and latches activity for the selected database.

From the **Database** details page, click **Locks and Latches** in the **Operational Views** section. The **Locks and Latches** view appears and provides activity information in the **Lock summary** and **Latch summary** sections.

The information found in this view comes from the _ActLock and _Latch VSTs.

Transactions view

The **Transactions** view displays all the data associated with transactions currently being processed within the database.

From the **Database** details page, click **Transactions** in the **Operational Views** section. The **Transactions** view appears, and displays a graph indicating transaction activity over time and the **DB transactions summary** section. For more information about graphs, see Graphical Displays of Database Data on page 35.

The **DB transactions summary** section has the following fields and columns:

- **Allocated transactions** — The number of transactions allocated.
- **Display allocated** — Whether to display the allocated transactions (No or Yes).
- **Trans #** — The transaction number.
- **User** — The user who entered the transaction. This information reflects the information available in the **User Activity** view.
- **User #** — The number of the user listed in the **User** column. This information reflects the information available in the **User Activity** view.
- **State** — The state of the transaction (**Allocated**, **Active**, or **Committing**). Allocated means the transactions have been "announced" to the database, but not yet started.

  The **Active** state is read to determine the data for the **Duration** and **Start Time**.

- **Duration** — How much time has passed since the transaction started. Duration is displayed for active transactions.
- **Start Time** — When the transaction started. Start Time is displayed for active transactions.
- **Coord DB** — The name of the coordinator database. Coord DB appears if two-phase commit is in use.
- **Coord Tx** — The number of the transaction on the coordinator database. Coord Tx appears if two-phase commit is in use.

The information for this view comes from the _Trans VST table.

Memory Resources view

The **Memory Resources** view provides summary of buffers, memory, and other relevant information of your database.

From the **Database** details page, click **Memory Resources** in the **Operational Views** section. The **Memory Resources** view appears and you can find more information in the following sections:

- **Buffers summary**
- **Share memory summary**
- **Space management summary**
- **Resources summary**

Data populating the **Memory Resources** view comes from the _ActBuffer, _BuffStatus, _Segments, _ActSpace, and _Resrc VSTs.

If the database is also configured for alternate buffer pools, polling retrieves the alternate buffer pool values from the VST _ActBuffer. Therefore, the details illustrated in the **Buffer Hits** section of the **Memory Resources** view, as shown in , reflect the total buffer hits for both primary and alternate buffer pools.

Page Writers view

The **Page Writers** view provides data about your Before-image, After-image, and Asynchronous Page Writers, as well as checkpoint information.

From the **Database** details page, click **Page Writers** in the **Operational Views** section. The **Page Writers** view appears and you can find more information in the following sections:

- **BI summary**
- **AI summary**
- **Asynchronous page summary**
- **Checkpoint summary**

  The **Checkpoint summary** section contains the following columns:

  - **Start Time** — The time the checkpoint started
  - **Duration** — The length of time the checkpoint lasted
  - **# Pending** — The number of database buffers not yet written to disk
  - **# Written** — The number of buffers written to disk
  - **# Scan** — The number of buffers written during the scan
  - **# APW** — The number of buffers written from the APW queue
  - **# Flushed** — The number of buffers flushed at the checkpoint

The data displayed in the **Page Writers** view comes from the following VSTs:

- _ActBILog (BI Summary View)
- _ActAILog (AI Summary View)
- _ActPW (APW Summary View)
- _Checkpoint (Checkpoint Summary View)

Advanced views

There are two advanced views:

- **Raw VST Data** — Enables you to examine individual VST tables.

  From the **Database** details page, click **Raw VST Data** in the **Advanced Views (raw data)** section. The **Raw VST Data** view appears, and allows you to choose a VST table and the format in which you want the data to be displayed. Click **Submit** to see the data.

- **Raw System Table Data** — Enables you to examine individual System tables.

  From the **Database** details page, click **Raw System Table Data** in the **Advanced Views (raw data)** section. The **Raw System Table Data** view appears, and allows you to choose a System table and the format in which you want the data to be displayed. Click **Submit** to see the data.

Note: Advanced views are provided for users who understand and have worked with raw data.

4

Database Maintenance Job Templates

OpenEdge Management supplies several database maintenance job templates to assist you with setting up and scheduling routine maintenance activities for your OpenEdge database. This chapter describes these templates and shows how to set up and run them.

Note: This chapter assumes that you are familiar with the job information presented in *OpenEdge Management: Resource Monitoring*.

For details, see the following topics:

- Database maintenance using job templates
- Creating database maintenance job instances
- Editing advanced job information

Database maintenance using job templates

OpenEdge Management includes several specialized database maintenance job templates. You can use these templates to create individual job instances and:

- Ensure that you can quickly and easily set up and initiate maintenance routines for your OpenEdge databases
- Standardize the information you want used when these routine activities are performed
- Define schedules for the various types of maintenance routines you perform for your OpenEdge databases

Using these job templates allows you to create an OpenEdge Management "best practices" approach to ensuring specific database activities are accomplished routinely for your OpenEdge database.

Note: OpenEdge Management also provides two database maintenance job templates for use only with the OpenEdge Management Trend Database. The job templates are named **Trend Data Compaction** (dataCompaction) and **Trend Data Compaction Unlock** (UnlockCompaction). See *OpenEdge Management: Database Management* for information about these job templates.

Displaying database job templates

To view the database maintenance job templates:

1. From the OpenEdge Management console menu, click **Jobs** > **Go to Jobs**.

   The **OpenEdge Management Jobs** page appears.

2. In the **View Existing Jobs and Job Templates** section, click **View Job Templates**.

   The **Jobs.Job Templates** page appears and displays the following:

   - Predefined maintenance job templates
   - The **Trend Data Compaction** template (displayed as **DataCompaction** in the list frame) and the **Trend Data Compaction Unlock** template (displayed as **Unlock Compaction** in the list frame)
   - Any user-defined job templates

3. To display the predefined job template pages associated with each of these database management activities, select a template from those listed.

   A summary of the current template values appears in the detail frame. You can review or edit the values defined for the database maintenance templates as you would any user-created job template.

Note: Template jobs can be run only against managed databases. In order to run a job against a scripted database, you must create your own job, without a template. See *OpenEdge Management: Resource Monitoring* for more information on creating jobs without using templates.

General considerations

You work with database maintenance jobs just as you do any jobs you create yourself. For example, once an instance of a database maintenance job is created, you can run it immediately or schedule it to run at a later date.

Note the following when working with job instances:

- By default, the database management jobs are enabled to run as job actions. This option supports job chaining. See *OpenEdge Management: Resource Monitoring* for a detailed explanation of job chaining.
- Database job instances use environment variables common to all jobs, and some that are defined specifically for the type of database maintenance activity you are performing. See Environment variables for database maintenance jobs on page 65 for details about environment variables specific to each database maintenance job. For additional information about common environmental variables, see *OpenEdge Management: Database Management*.
- All predefined default field values are editable, including the template menu categories.

- Note that the name of your job cannot contain spaces or these special characters: %, &, $, or #.
- Required fields on the job's property page appear in red.

You can perform the following tasks for job instances derived from a database maintenance job template:

- Access the database maintenance job template from the **Jobs.Job Templates** page. You can review and edit the template; you cannot set up a job instance of a given template.
- Set up the **Job Completion Actions and Alerts** page.
- Review specific job values on the **Job Summary** page. This page automatically appears once you complete either the **Job Properties** page or the **Job Completion Actions and Alerts** page.

Creating database maintenance job instances

To create a job instance from one of the database maintenance templates, you must access the specific template details from the **Jobs.Job Templates**page.

To access job templates for setting up a job instance:

1. In the OpenEdge Management console menu, choose one:
   - Click **Jobs** > **New** > **Job from a Template**.
   - Click **Jobs** > **Go to Jobs** > **Create Job from a Template**.

   The **Create Custom Job** page appears.

   The **Database Maintenance**, **OpenEdge Management Configuration**, and **OpenEdge Management Trend Database Maintenance** categories include all the OpenEdge Management-supplied jobs. You can create additional templates to add to these categories, or you can create additional categories. For details about creating user-defined templates, see *OpenEdge Management: Resource Monitoring*.

2. Select the database maintenance job from which you want to create a job instance. The **Job Properties** page appears.

   Note: When the **Job Properties** page appears, you might find that several fields already display values. This is one of the main benefits of the template; you need to supply only the data unique to this job instance.

3. Complete the **Job Properties** page. Use the information in the following table to find details about each database maintenance job template's **Properties** page:

For details about the . . .	See . . .
OfflineBackup template	Setting up Database Backup jobs on page 56
OnlineBackup template	Setting up Database Backup jobs on page 56
DatabaseRestore template	Setting up a Database Restore job on page 58
TruncateBI template	Setting up a Truncate BI job on page 59
GrowBI template	Setting up a Grow BI job on page 60

For details about the . . .	See . . .
DatabaseAnalysis template	Setting up a Database Analysis job on page 61
IndexCompaction template	Setting up an Index Compaction job on page 62
Backup Configuration template	Setting up a Backup Configuration job on page 63
DataCompaction template	*OpenEdge Management: Trend Database Guide and Reference*
DataCompactionUnlock template	*OpenEdge Management: Trend Database Guide and Reference*

Setting up Database Backup jobs

The job page and the options that you use to set up and run an online database backup are identical to an offline backup.

To complete the Database Backup Properties page:

1. Provide values for the **Name** and **Description** fields.
2. From the list of available databases, select the database for which you want to perform the backup.
3. Identify the backup device you are using:
   - If you select **Tape**, you must also define the absolute path for the tape device.
   - If you select **Disk**, you must also specify the directory and the filename. If you want to overwrite an existing backup at this location, select the **Overwrite file** option.
4. In the **Volume size (-vs)** field, identify the size, in database blocks, that you want the backup to be before prompting for another volume.

   If you select the **Volume size (**`-vs`) parameter, the following happens:

   - The Backup job looks for a file in your work directory with the name of `<resource name>-backup.txt`. This file should contain names for each backup file. If this file exists, the Backup job uses it as input to the OpenEdge Backup Utility (PROBKUP).

- The following shows an example <*resource name*>-backup.txt file:

```
e:\dbwork/ftd.bak1
e:\dbwork/ftd.bak2
e:\dbwork/ftd.bak3
e:\dbwork/ftd.bak4
e:\dbwork/ftd.bak5
e:\dbwork/ftd.bak6
e:\dbwork/ftd.bak7
e:\dbwork/ftd.bak8
e:\dbwork/ftd.bak9
e:\dbwork/ftd.bak10
e:\dbwork/ftd.bak11
e:\dbwork/ftd.bak12
e:\dbwork/ftd.bak13
e:\dbwork/ftd.bak14
e:\dbwork/ftd.bak15
e:\dbwork/ftd.bak16
e:\dbwork/ftd.bak17
e:\dbwork/ftd.bak18
e:\dbwork/ftd.bak19
e:\dbwork/ftd.bak20
e:\dbwork/ftd.bak21
```

- If this file does not exist, the Backup job creates the file in the working directory. The file is the name of the backup (entered in Step 1 on page 56) with a number from 1 to 99 appended to it. The Backup job uses this file as input to PROBKUP.
- Any files created by the Backup job remain after the program ends. The Backup job does not delete them.

5. Complete the following fields:
   - **Blocking factor (-bf)** — Specifies the blocking factor size used to flush buffers out of the backup device.
   - **Redundancy factor (-red)** — Specifies the redundancy count to create redundancy in the backup.
   - **Incremental overlap (-io)** — Identifies how many previous backups you want included in this incremental backup. Use this field simultaneously with the **Incremental** check box.
   - **Incremental** — If selected, this backs up the blocks that have changed since the last backup. Use this field simultaneously with the **Incremental overlap (-io)** parameter field.
   - **Estimate size** — If selected, this generates a report that estimates the size of the backup.

     Note: This setting **does not** back up the database.

   - **Verbose listing** — If selected, this provides output every 10 seconds concerning the status of the backup.
   - **Compression** — If selected, this indicates you want to compress data blocks as they are backed up.
   - **No recovery** — If selected, this prevents the database from rolling forward an `AI` file. This option is used for mirrored backups.
6. Verify the name of the command's current working directory in the **Working Directory** field. If specified, this directory must exist. This property defaults to the working directory defined at installation.
7. To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** field indicates where messages will be retained.

8. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. For example, you can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
9. Select the **Indicate if the job can be used as an action** option if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
10. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
11. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately. You can also edit advanced information, as detailed in Editing advanced job information on page 64.

Setting up a Database Restore job

The Database Restore job restores an OpenEdge database backup to the location specified in the existing structure for that database. If the file structure does not exist, then the restore is performed to the existing directory for all of the areas and extents.

To complete the Database Restore Properties page:

1. Provide values for the **Name** and **Description** fields.
2. Identify the restore device you are using:
   - If you select **Tape**, you must also define the absolute path for the tape device.
   - If you select **Disk**, you must also specify the directory and file name.

Note: If you used the volume size (`-vs`) parameter when backing up your database to disk, you must specify the file that contains the names of each volume. The Restore job will pass this file to the OpenEdge Restore (PROUTIL) utility. The name of the file should be `<database name>-restore.txt`. The contents of the restore file should be the same as the contents of the backup file passed to PROBKUP. See Step 4 on page 56 in Setting up Database Backup jobs on page 56 for more information about the backup file.

---

3. In the **Restore to** field, select an AdminServer name or type the full path and name of the database to which you want to restore. If you want to overwrite an existing database at this location, select the **Overwrite database** option.
4. Complete the following fields:
   - **Partial verify (-vp)** — Select this option if you want to ensure that a partial verification of the completed backup occurs. Choosing this option ensures only that the backup can be read.

     Note: Choosing this option **does not**restore the database.

   - **Full verify (-vf)** — Select this option if you want a full verification of the completed backup which ensures that a block-by-block comparison is performed between the backup and the database.

     Note: Choosing this option **does not**restore the database.

   - **List structure (-list)** — Select this option to create a structure file from the backup that was completed.

Note: Choosing this option **does not**restore the database.

- **No verification** — Clear this option if you want verification. Note that **No verification** is the default selection.

5. Verify the name of the command's current working directory in the **Working Directory** field. If specified, this directory must exist. This property defaults to the working directory defined at installation.
6. To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** field indicates where messages will be retained.
7. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. For example, you can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
8. Select the **Indicate if the job can be used as an action** option if you want the job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
9. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the optional **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
10. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or run it immediately. You can also edit advanced information. See Editing advanced job information on page 64 for additional details.

Setting up a Truncate BI job

The Truncate BI (before image) job allows you to truncate the `BI` file, shrinking it to zero bytes. This job can be used in processing, if needed. Truncation is typically bundled, however, with Grow BI for performance reasons.

To complete the Truncate BI Properties page:

1. Provide values for the **Name** and **Description** fields.
2. From the **Resources** list, select the database whose `BI` file you want to truncate.
3. Identify the operating system user account in the **User name** or **Group** field. (This account is not necessarily the same as your OpenEdge Management user account name.) If specified, the account must be valid on the server machine (or server domain) where the AdminServer and OpenEdge Management are running. In Windows the name can also include a domain.
4. Type the password of the specified user in the **Password** field. If the **User name** or **Group** does not have an associated password, this field must be left blank (or an error message will be generated).
5. Complete the **Job specification** section as follows:
   a) Type the command to be executed in the **Command** field. This can be any user-defined command that would typically execute from an OS shell. The name can include a full or a relative pathname. You can use environment variables such as `%DLC%` or `$DLC`. This value will already be defined if you are creating a job instance from a template.
   b) Define the input parameter to the command in the **Command parameters** field. The list of parameters takes the same format as from an OS shell. Additionally, you can use environment variables, such as $SHELL or %WINDIR%, and Windows registry values.

c) Verify the name of the command's current working directory in the **Working directory** field. If specified, this directory must exist. This property defaults to the OpenEdge Management working directory defined at installation.

d) Enter the name of the **Input file** to be used for read redirection with a job's command. This is typically used for any keyboard input the command might require (optional).

e) To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** fields indicate where messages will be retained.

Note: When you create or edit a job, OpenEdge Management supplies the `.out` and `.err` filenames. The filenames will be the same as the job name with a different suffix. You can change the filename, or you can remove it if you do not want to create the output files. Also, if you edit the job name, OpenEdge Management will not change the filenames to reflect the new job name. You should review these filenames **before** saving the page to verify the job creates the expected output files.

---

6. In the **Environment name=value pairs** field, define environment variables to be set (in the process context of the task that runs to execute the specified job). These values are application-specific, user-definable variables. For a list of environment variables available for the job, run the job with the **Debug log file** option on. The debug log file lists all environment variables and, if applicable, their values. Environment variables are preceded by **env** in the log file.
7. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. For example, you can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
8. Select the **Indicate if the job can be used as an action** option if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
9. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
10. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately.

Setting up a Grow BI job

Grow BI job instances should be used if the `BI` file is truncated. Grow BI jobs preformat the `BI` file and result in better performance.

To complete the Grow BI Properties page:

1. Provide values for the **Name** and **Description** fields.
2. From the **Resources** list, select the database whose `BI` file you want to grow.
3. Identify the operating system user account in the **User name** or **Group** field. (This account is not necessarily the same as your OpenEdge Management user account name.) If specified, the account must be valid on the server machine (or server domain) where the AdminServer and OpenEdge Management are running. In Windows the name can also include a domain.
4. Type the password of the specified user in the **Password** field. If the **User name** or **Group** does not have an associated password, this field must be left blank or an error message will be generated.
5. Complete the **Job specification** section as follows:

a) Enter the command to be executed in the **Command** field. This can be any user-defined command that would typically execute from an OS shell. The name can include a full or a relative pathname. You can use environment variables such as `%DLC%` or `$DLC`. This value will already be defined if you are creating a job instance from a template.

b) Define the input parameter to the command in the **Command parameters** field. The list of parameters takes the same format as from an OS shell. Additionally, you can use environment variables, such as $SHELL or %WINDIR%, and Windows registry values.

c) Verify the name of the command's current working directory in the **Working directory** field. If specified, this directory must exist. This property defaults to the OpenEdge Management working directory defined at installation.

d) Enter the name of the **Input file** to be used for read redirection with a job's command. This is typically used for any keyboard input the command might require (optional).

Note: When you create or edit a job, OpenEdge Management supplies the `.out` and `.err` filenames. The filenames will be the same as the job name with a different suffix. You can change the filename, or you can remove it if you do not want to create the output files. Also, if you edit the job name, OpenEdge Management will not change the filenames to reflect the new job name. You should review these filenames **before** saving the page to verify the job created the expected output files.

---

e) To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** fields indicate where messages will be retained.

6. In the **Environment name=value pairs** field, define environment variables to be set (in the process context of the task that runs to execute the specified job). These values are application-specific, user-definable variables. For a list of environment variables available for the job, run the job with the **Debug log file** option on. The debug log file lists all environment variables and, if applicable, their values. Environment variables are preceded by **env** in the log file. For environmental variables specific to the Grow BI job, see Table 5: Environmental variables for database maintenance jobs on page 65.
7. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. You can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
8. Select the **Indicate if the job can be used as an action** option if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
9. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
10. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately.

Setting up a Database Analysis job

The Database Analysis job runs against an OpenEdge database and loads the information into the OpenEdge Management Trend Database. This information can be used for various purposes, including index compaction and data analysis.

To complete the Database Analysis Properties page:

1. Provide values for the **Name** and **Description** fields.
2. Select the database for which you want to perform this analysis from the **Resources** list.

   In OpenEdge 12.0, there is no support for the **Job from a Template** functionality to create a job instance in OpenEdge Management that has a 12.0 or higher versioned remote adminserver.

3. Select the **Enable rule validation** option to indicate that you want this job to test your rules. **Utilization %** and **Block number**, two subfields of the **Enable rule validation** option, are active only if you select this option. Complete these subfields as follows:

   a) In the **Utilization %** field, specify the percentage of utilization of the blocks for the index to validate against. If the index utilization percentage goes below this number, you could set up an alert to trigger.

   b) In the **Block number** field, specify the threshold for the number of blocks that the index must contain for the rule to be run.

4. Select the **Run index compaction** option if you want index compaction run for the database during this job. The **Compaction %** field specifies the percentage to which indexes will be packed when index compaction is run.
5. Verify the command's current working directory in the **Working directory** field. If specified, this directory must exist. This property defaults to the OpenEdge Management working directory defined at installation.
6. To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** field indicates where messages will be retained.
7. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. For example, you can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
8. Select the **Indicate if the job can be used as an action** option if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
9. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
10. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately. You can also edit advanced information. See Editing advanced job information on page 64 for additional details.

Setting up an Index Compaction job

The Index Compaction job compacts indexes to optimal compression and can be run online or offline. To determine which indexes to compact, run database analysis against the database whose indexes you intend to compact.

To complete the Index Compaction Properties page:

1. Provide values for the **Name** and **Description** fields.
2. Select the database for which you are performing this activity.
3. Complete the following fields:

   - **Table name** — Identifies the table on which you intend to perform the index compaction activity
   - **Index name** — Identifies the index name for which you intend to perform the index compaction activity

- **Compaction %** — Specifies the percentage to which indexes will be packed when index compaction is run

4. Verify that the **Working directory** field identifies the command's current working directory. If specified, this directory must exist. This property defaults to the OpenEdge Management working directory defined at installation.
5. To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** field indicates where messages will be retained.
6. Select the **Debug log file** option to obtain diagnostic details that help debug job properties when setting up a job. For example, you can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug data available through the debug log file link on the **Job Summary** page.
7. Select the **Indicate if the job can be used as an action** option if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
8. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Resource Monitoring* for more information about the **Job Completion Actions and Alerts** page.
9. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately. You can also edit advanced information. See Editing advanced job information on page 64 for additional details.

Setting up a Backup Configuration job

The Backup Configuration job backs up the OpenEdge Management configuration for a database.

To complete the Backup Configuration Properties page:

1. Provide values for the **Name** and **Description** fields.
2. From the **Resources** menu, select the database whose OpenEdge Management configuration you want to back up.
3. Identify the operating system user account in the **User name** or **Group** field. (The account is not necessarily the same as your OpenEdge Management user account name.) If specified, the account must be valid on the server machine (or server domain) where the AdminServer and OpenEdge Management are running. In Windows the name can also include a domain.
4. Type the password of the specified user in the **Password** field. If the **User name** or **Group** does not have an associated password, this field must be left blank or an error message will be generated.
5. Complete the **Job specification** section as follows:
   a) Enter the command to be executed in the **Command** field. This can be any user-defined command that would typically execute from an OS shell. The name can include a full or a relative pathname. You can use environment variables such as `%DLC%` or `$DLC`. This value will already be defined if you are creating a job instance from a template.
   b) Define the input parameter to the command in the **Command parameters** field. The list of parameters takes the same format as from an OS shell. Additionally, you can use environment variables, such as $SHELL or %WINDIR%, and Windows registry values.
   c) Verify the name of the command's current working directory in the **Working directory** field. If specified, this directory must exist. This property defaults to the OpenEdge Management working directory defined at installation.

d) Enter the name of the **Input file** to be used for read redirection with a job's command. This is typically used for any keyboard input the command might require (optional).

e) To append subsequent messages to the `.out` or `.err` files, select the **Append** option. The **Output file** fields indicate where messages will be retained.

Note: When you create or edit a job, OpenEdge Management supplies the `.out` and `.err` filenames. The filenames will be the same as the job name with a different suffix. You can change the filename, or you can remove it if you do not want to create the output files. Also, if you edit the job name, OpenEdge Management will not change the filenames to reflect the new job name. You should review these filenames **before** saving the page to verify the job created the expected output files.

---

6. In the **Environment name=value pairs** field, define environment variables to be set (in the process context of the task that runs to execute the specified job). These values are application-specific, user-definable variables. For a list of environment variables available for the job, run the job with the **Debug log file** option on. The debug log file lists all environment variables and, if applicable, their values. Environment variables are preceded by **env** in the log file.
7. Choose whether to set the **Debug log file** option. This option allows you to obtain diagnostic details to help debug job properties when setting up a job. You can set this option and use the **Run Now** feature on the **Job Summary** page. Once you have submitted the job, OpenEdge Management makes debug log file data available through the debug log file link on the **Job Summary** page.
8. Select the **Indicate if the job can be used as an action** check box if you want this job to appear in the list of actions that can be selected for execution from the **Alerts and Job completion action** page.
9. If you intend to set up actions and alerts for this job, click the **Edit** button associated with the **Completion Actions and Alerts** field to display the **Job Completion Actions and Alerts** page. See *OpenEdge Management: Database Management* for more information about the **Job Completion Actions and Alerts** page.
10. Click **Save**. The **Job Summary** page appears. You can now schedule the job to run at a later time, or you can run it immediately.

Editing advanced job information

After you create a job instance, you can view and edit advanced information about the job.

Note: The Truncate BI job and the Grow BI job do not allow you to view advanced information.

To edit the advanced information for a database maintenance job:

1. Click **View** (in the **Properties** section) on the **Job Summary** page.
2. Click **Edit**. The **Advanced information** page appears.
3. Enter the account information for the user who will run the job. The default is the user who started the AdminServer.
4. If you want to change the job specifications, enter the appropriate command and command parameters.

   If you are changing job specifications, you can also consider creating a new job template. See *OpenEdge Management: Resource Monitoring* for the steps you use to create custom job templates.

5. Enter applicable environment variables. See Table 5: Environmental variables for database maintenance jobs on page 65 for a list of database maintenance job environment variables.
6. Click **Save**.

Environment variables for database maintenance jobs

If you choose to create custom jobs by modifying the job templates provided, use the information provided in the following table.

Table 5: Environmental variables for database maintenance jobs

Job	OpenEdge Management variable	Description
Database Backup	FM_BKPDEVICE	Identifies the backup device, which is either the tape or complete path to the file.
Database Backup	FM_BKPDEVICETYPE	Indicates the type of device to be used—disk or tape.
Database Backup	FM_BKPTAPEDIR	Identifies the full tape device name. This name can match the value defined for FM_BKPDEVICE.
Database Backup	FM_BKPDIRECTORY	Identifies the directory for file backup.
Database Backup	FM_BKPFILENAME	Identifies the filename for file backup.
Database Backup	FM_BKPOVERWRITEFILE	Indicates whether a file, but not a tape, is overwritten.
Database Backup	FM_BKPONLINE	Identifies the option that determines whether to perform a backup job online or offline. If this option is not set, OpenEdge Management assumes the backup is offline.
Database Backup	FM_BKPINCREMENTAL	Indicates whether data in blocks that have been changed since the last backup will be backed up.
Database Backup	FM_BKPESTIMATESIZE	Indicates whether a report that estimates the size of the backup will be generated.
Database Backup	FM_BKPVERBOSELISTING	Indicates whether OpenEdge Management provides a verbose output every 10 seconds giving the status of the backup.

Job	OpenEdge Management variable	Description
Database Backup	FM_BKPCOMPRESSION	Indicates whether you want data blocks compressed as they are backed up.
Database Backup	FM_BKPNORECOVERY	Used for mirrored backups. Set this option if you do not want the database to be able to roll forward an AI file.
Database Backup	FM_BKPVOLUMESIZE	Identifies the size that you want the backup to be before prompting for another volume.
Database Backup	FM_BKPBLOCKINGFACTOR	Specifies the blocking factor size used to flush buffers out of the backup device.
Database Backup	FM_BKPREDUNDANCYFACTOR	Specifies the count to create redundancy in the backup.
Database Restore	FM_RSTDEVICE	Identifies the device to which the database is restored. The device can be either a tape or a complete path to a file.
Database Restore	FM_RSTDEVICETYPE	Indicates the type of device to be used—disk or tape.
Database Restore	FM_RSTVERIFY	Identifies one of three possible restore options: • Partial verification • Full verification • Creation of a structure file from the backup that was completed
Database Restore	FM_RSTDBNAME	Identifies the full path and name of the database you want to restore.
GrowBI	FM_BICLUSTERNUMBER	The number of clusters to pregrow the `BI` file to when the command is run. The initial size of the `BI` file is determined by taking the BI cluster size times the BICLUSTERNUMBER.
Database Analysis	FM_DBAUTILIZATIONNUMBER	Specifies the percentage of utilization of the blocks for the index to validate against.

Job	OpenEdge Management variable	Description
Database Analysis	FM_COMPACTIONNUMBER	Specifies the percentage to which indexes will be packed when index compaction is run.
Database Analysis	FM_DBABLOCKNUMBER	Specifies the threshold for the number of blocks that the index must contain for the rule to be run.
Database Analysis	FM_DBARULEVALIDATION	Indicates that you want this job to test your rules.
Index Compaction	FM_IDXTABLENAME	Identifies the name of the table to compact.
Index Compaction	FM_IDXINDEXNAME	Identifies the name of the index to compact.
Index Compaction	FM_COMPACTIONNUMBER	Specifies the percentage to which indexes will be packed when index compaction is run.
All database maintenance jobs	FM_ADMINSERVERNAME	Specifies the resource's AdminServer and the resource. For example: Dev01.sports2000.

Index

A

Advanced Views *52*
After Imaging Area view *49*
Application Data Areas view *48*

B

Before Imaging Area view *48*

C

Control Area view *48*

D

Database
  changing control settings *39*
  Details page *38*
  maintenance *53*
  migrating *18*
  remote *23*
  rules *24*
Database Configuration view *44*
Database log file viewer, See Log file viewer
Database maintenance job templates, See Job templates
Database migration utility *18*
Database monitoring plans, See Monitoring plans
Database resource monitors
  defaults *20*
  rules *24*
  updating *21*
Database rules, See Rules
Database views
  Advanced *52*
  Database configuration *44*
  File Systems *46–47*
  Locks and Latches *50*
  Memory Resources *51*
  Page Writer *51*
  Performance summary *45*
  Raw VST data *52*
  Record and Index Activity *50*
  Storage Area *47*
  Transactions *50*
  User Activity *49–50*
  Vital Signs *45*

F

File resource monitors
  creating *47*

G

Graph caching *35*
Graphs
  general information *35*

J

Job templates, See Templates

L

Log file monitors
  customizing *40–41*
  general considerations *40*

M

Monitoring plans
  default values *20*
  modifying *21*
  rule sets *28*

P

Performance Summary view *45*

R

Raw VST Data view *52*
Record and Index Activity view *50*
Resource monitoring plans, See Monitoring plans
Rule set
  adding *28*
  associating *28*
  database
    copying *27*
    creating *27*
    deleting *28*
Rules
  details *25*

S

Schedules
    Default_Schedule *41*
Storage Area view *47*
System Areas view *48*

T

Templates
    accessing *55*
    backup jobs
        unattended *56*
    database analysis *61*
    database index compaction *62*
    database restore *58*
Templates *(continued)*
    definition *53*
    grow BI *60*
    purpose *53*
    truncate BI *59*
    viewing *54*
Transactions view *50*

U

User Activity view *49–50*

V

Views, See Database views
Vital Signs view *45*