Progress®



Implementing Roll-Forward Recovery Using After-Imaging

OpenEdge®

Copyright

© 2021 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.

DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Defrag This, Deliver More Than Expected, DevReach (and design), Icenium, Inspec, Ipswitch, iMacros, Kendo UI, Kinvey, MessageWay, MOVEit, NativeChat, NativeScript, OpenEdge, Powered by Chef, Powered by Progress, Progress, Progress Software Developers Network, SequeLink, Sitefinity (and Design), Sitefinity, Sitefinity (and design), SpeedScript, Stylus Studio, Stylized Design (Arrow/3D Box logo), Styleized Design (C Chef logo), Stylized Design of Samurai, TeamPulse, Telerik, Telerik (and design), Test Studio, WebSpeed, WhatsConfigured, WhatsConnected, WhatsUp, and WS_FTP 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, Chef Automate, Chef Compliance, Chef Desktop, Chef Habitat, Chef WorkStation, Corticon.js, Corticon Rules, Data Access, DataDirect Autonomous REST Connector, DataDirect Spy, DevCraft, Fiddler, Fiddler Everywhere, FiddlerCap, FiddlerCore, FiddlerScript, Hybrid Data Pipeline, iMail, JustAssembly, JustDecompile, JustMock, KendoReact, NativeScript Sidekick, OpenAccess, PASOE, Pro2, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, Push Jobs, SafeSpaceVR, Sitefinity Cloud, Sitefinity CMS, Sitefinity Digital Experience Cloud, Sitefinity Feather, Sitefinity Insight, Sitefinity Thunder, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, Supermarket, SupportLink, Unite UX, 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.

July 2021

Product version: Progress OpenEdge 12.4

Updated: 2021/06/28

Table of Contents

Preface....7

OpenEdge database disaster recovery methods....9

Roll-forward recovery....10
OpenEdge replication....12
Failover clusters....13

Implementing roll-forward recovery using after-imaging....15

Sizing fixed AI extents....16
How AI extents work....16
Procedure: Sizing fixed AI extents using PROMON....17
Procedure: Sizing fixed AI extents using OpenEdge Management....18
Example of sizing AI extents....19
Creating fixed AI extents....20
Enabling after-imaging and automatic AI file management offline....21
Enabling after-imaging offline....22
Enabling automatic AI file management for a database offline....22
Starting a database with automatic AI file management startup parameters....23
Procedure: Configuring AI management parameters for a database using OpenEdge Explorer.24
Enabling after-imaging and automatic AI file management online....25

Managing AI extents....27

Managing archived AI files....29
Adjusting automatic AI file management parameters....30

Performing roll-forward recovery....31

Identifying a roll-forward recovery set....33
Example of a roll-forward recovery set....34
Identifying busy or full AI extents....35
Procedure: Creating a roll-forward recovery batch file....36
Example of a roll-forward recovery batch file....37
Procedure: Performing roll-forward recovery using a batch file....37

Disabling after-imaging....39

Preface

Purpose

The *Implementing Roll-Forward Recovery Using After-Imaging* guide describes <placeholder>

Audience

This guide is aimed at <placeholder>.

Organization

This book is organized into the following chapters::

- **OpenEdge database disaster recovery methods**—
- **Implementing roll-forward recovery using after-imaging**—
- **Managing AI extents**—
- **Performing roll-forward recovery**—
- **Disabling after-imaging**—

Documentation conventions

See Documentation Conventions for an explanation of the terminology, format, and typographical conventions used throughout the OpenEdge content library.

1

OpenEdge database disaster recovery methods

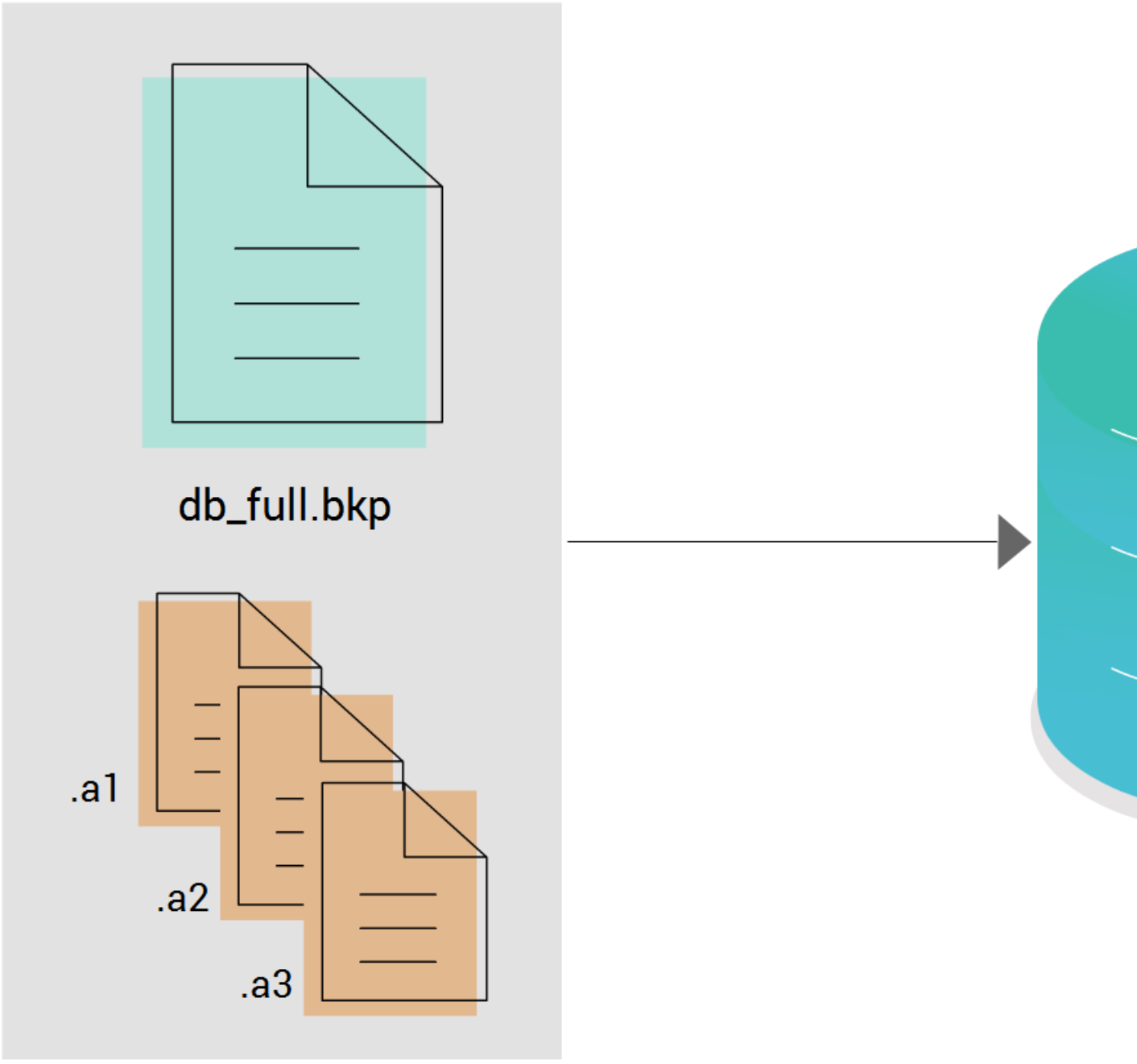
OpenEdge offers three methods to implement database disaster recovery. They are listed in ascending order of high availability and cost:

- Roll-forward recovery
- OpenEdge replication
- Failover clusters

For details, see the following topics:

- Roll-forward recovery
- OpenEdge replication
- Failover clusters

Roll-forward recovery



Roll-forward recovery is a process in which you use a combination of database backups and associated AI files to recover committed data up to the point of failure.

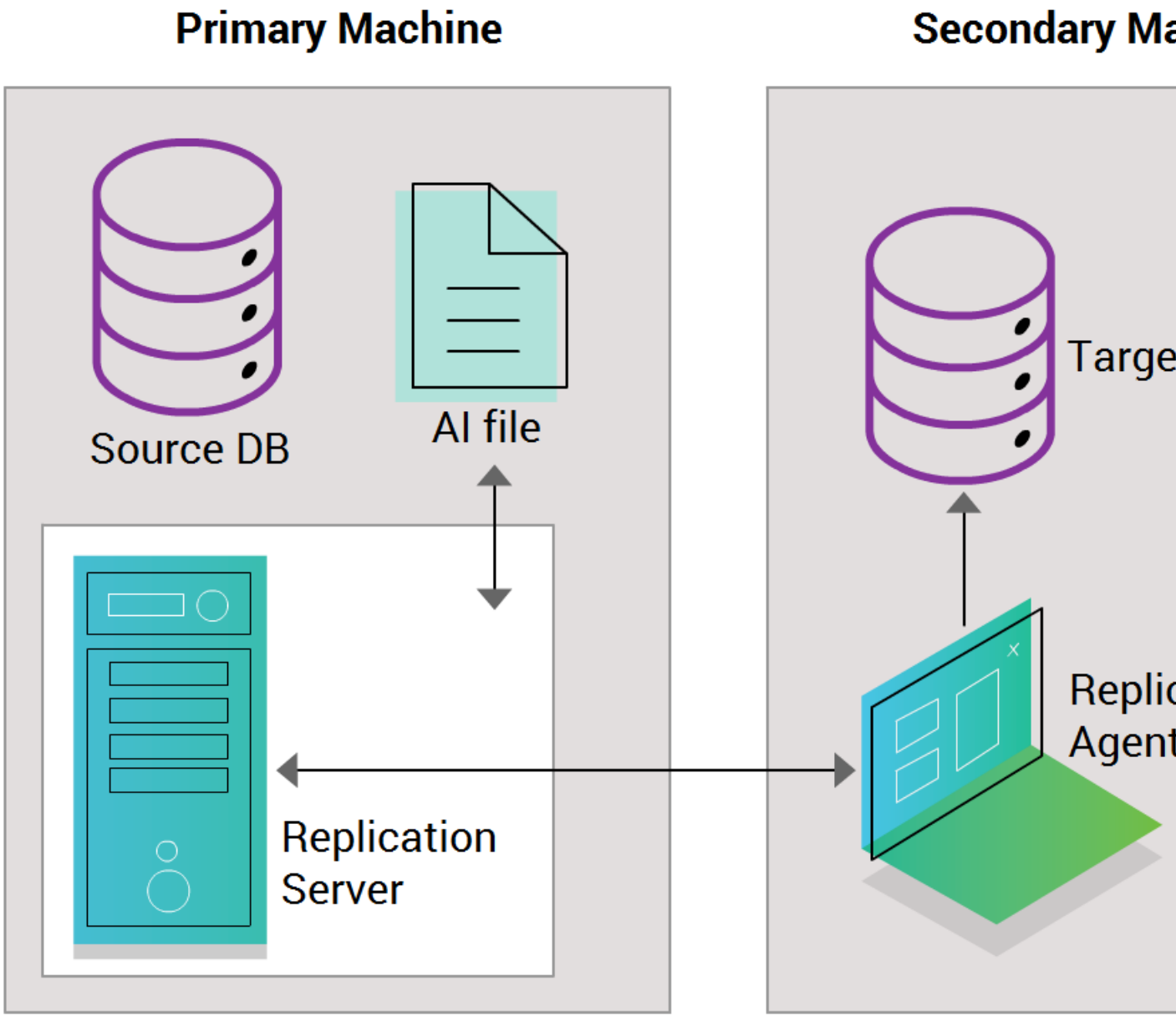
If a database crashes or is damaged because of disk failure, database corruption, or user errors, you can recover the database by restoring it from its most recent full and incremental backups and then applying the associated AI files to the restored database. In this way, you can recover most, or all, committed transactions up to the point when the database is lost or damaged.

Note: You can also use roll-forward recovery to maintain a copy of a production database for the following purposes:

- To minimize business disruption at little cost by creating a warm standby database for your production database. In the event of a system failure on the production database, you apply the last AI files from the production database to the warm standby database, and start using the warm standby database as the new production database.
- To reduce the work load that resource-intensive reporting applications would incur on your production database.

In this section, you will learn how to implement roll-forward recovery for a database using after-imaging.

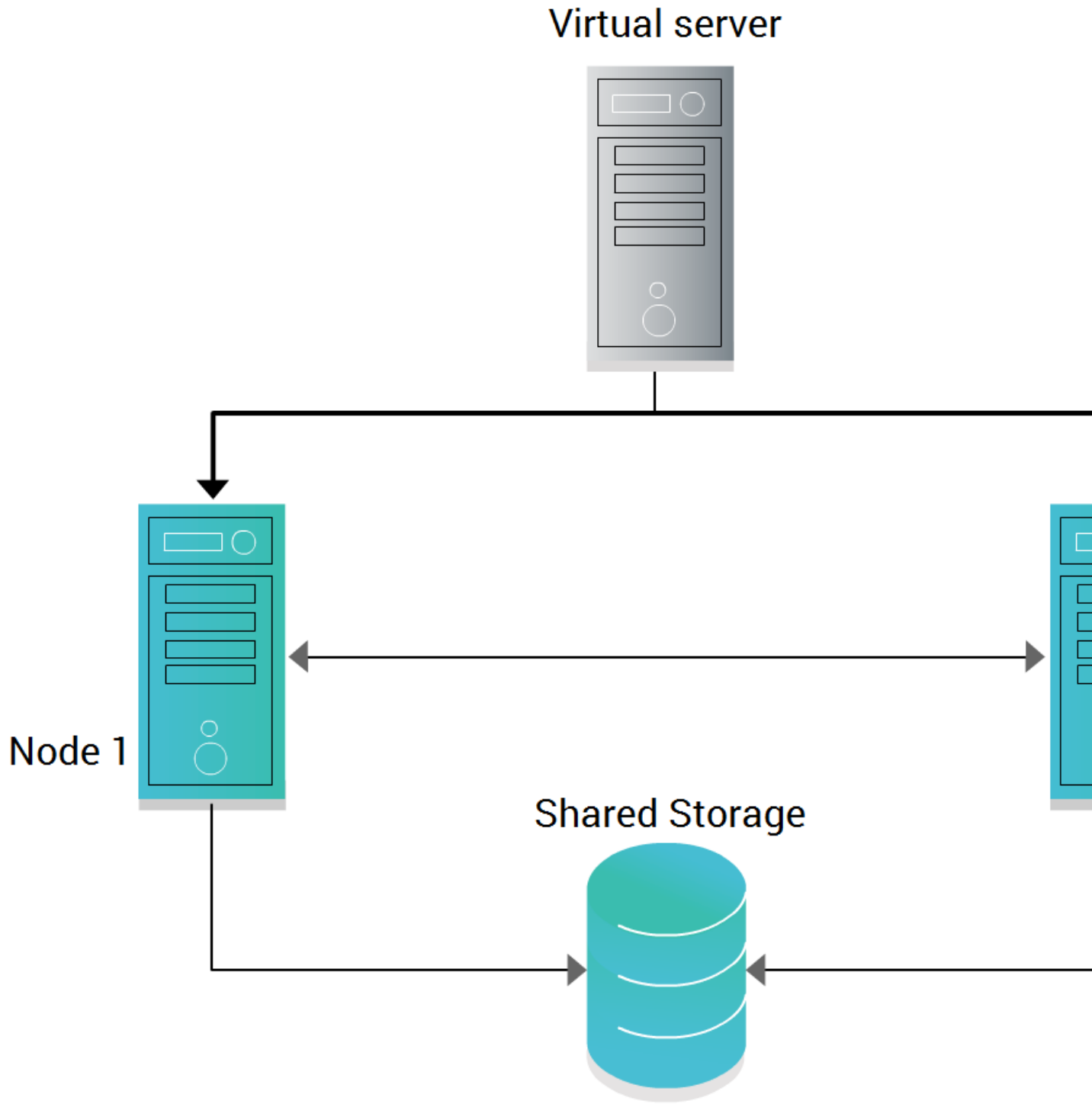
OpenEdge replication



OpenEdge replication is a process in which data is continuously replicated from a source database to a target database using AI files. If the source database is non-operational, the target database can be brought online to ensure availability of data. Replication eliminates the database as a single point of failure, provides high data availability, and minimizes business disruption.

Progress provides a product called OpenEdge Replication to enable you to implement database replication. With OpenEdge Replication, you can create and maintain a hot standby database. It is a copy of a production database and is readily available for use if the production database is damaged, lost, or unavailable.

Failover clusters



A failover cluster consists of two or more computer nodes that are tightly integrated through hardware and software to work together as one virtual computer. If a computer node fails or has to be taken offline, the cluster can fail over to another node to ensure continuous data access.

A cluster eliminates the database computer as a single point of failure and virtually eliminates business disruption.

Progress provides a feature called Failover Clusters with OpenEdge Enterprise RDBMS to enable you to implement failover clusters. You also need additional cluster hardware and software to make failover clusters work in your environment.

2

Implementing roll-forward recovery using after-imaging

Roll-forward recovery uses the most recent full and incremental database backups and all the AI extents that have been archived since the most recent backup until the point of failure.

In addition to a database backup plan, to implement roll-forward recovery using after-imaging, you need to:

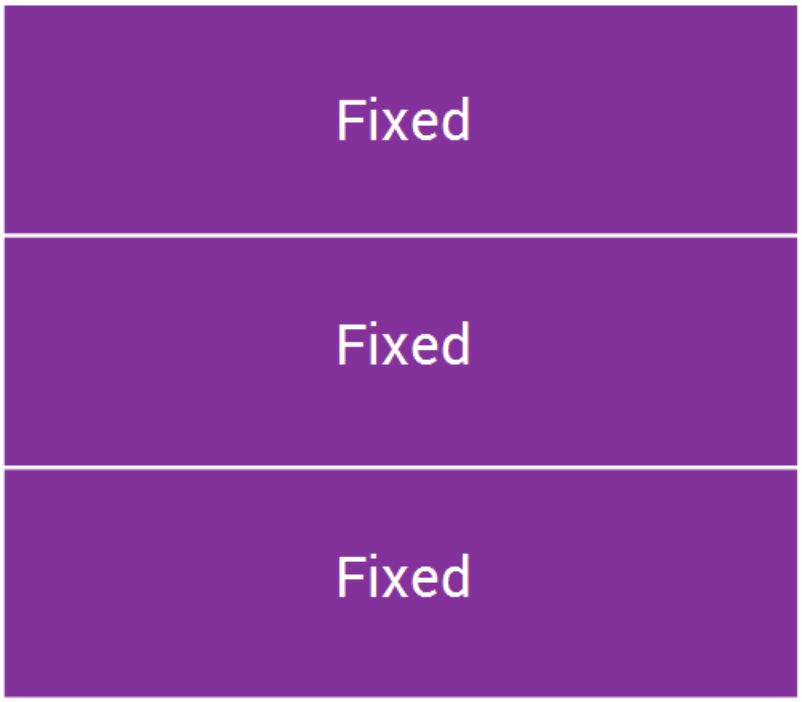
1. Size fixed AI extents.
2. Create fixed AI extents offline or online.
3. Enable after-imaging and automatic AI file management offline or online.

You already know how to create extents offline and online. Next, you will learn how to size fixed AI extents and enable after-imaging and automatic AI file management offline and online.

For details, see the following topics:

- Sizing fixed AI extents
- Creating fixed AI extents
- Enabling after-imaging and automatic AI file management offline
- Enabling after-imaging and automatic AI file management online

Sizing fixed AI extents



OpenEdge RDBMS uses AI extents to store AI notes about database transactions. Properly sized AI extents enable it to perform this task efficiently.

The AI extent factors that affect database performance are:

- The type of AI extents—Fixed AI extents are better for performance than variable AI extents. Because fixed AI extents are preformatted, OpenEdge just writes to the fixed AI extents as transactions occur. On the other hand, variable AI extents are formatted as needed, so OpenEdge must format and write to the variable AI extents as transactions occur.
- The size of fixed AI extents—The size of a fixed AI extent affects performance because it determines how much AI data OpenEdge can write to the extent and how frequently OpenEdge must switch and write to the next AI extent.

Because of these factors, if you size fixed AI extents properly, you will enable OpenEdge to write AI data efficiently.

Next, you will learn about how AI extents work and how to size fixed AI extents.

How AI extents work

To work with AI extents effectively, you need to know how AI extents are used:

- The AI extents are written in the order defined in the .st file.
- For each fixed AI extent, OpenEdge RDBMS writes to it until it becomes full, then automatically switches over and writes to the next empty AI extent.
- When the last AI extent in the AI extent sequence fills up, OpenEdge RDBMS loops back and uses the first empty extent in the sequence in a round-robin fashion.

- If all AI extents are full and transactions are still occurring, OpenEdge RDBMS would shut down the database.

In general, AI extents can grow very rapidly because—unlike the before-image (BI) and application data files—an AI extent cannot be reused until it is emptied.

At any given time, an AI extent can be in one of the following modes:

- Empty—The extent is available for use.
- Busy—The extent is currently in use.
- Full—The extent has filled up and should be archived.

It is recommended that you use the automatic AI file management utility to manage AI extents. The utility ensures that full AI extents are archived and empty AI extents are available. All you need to do is to ensure there is enough disk space for the automatic AI file management utility to write to.

Procedure: Sizing fixed AI extents using PROMON

You can size fixed AI extents using BI file information because the amount of data written to a BI file is approximately equal to the amount of data written to an AI file in any given interval.

Important: In addition, you should size fixed AI extents during peak periods, for example, at the end of the month or end of the quarter, to ensure that you calculate sufficient space to store AI notes during the heaviest workload.

To size a fixed AI extent using PROMON, first you look up the BI cluster size. A BI file contains multiple BI clusters. When a database is started, OpenEdge RDBMS by default creates and formats each BI extent into a chain of four clusters. You also look up the number of checkpoints during a sample period. When a BI cluster fills up, a checkpoint occurs, and OpenEdge RDBMS writes all modified data referenced by the cluster to the data files on disk. Checkpointing is a process in which OpenEdge RDBMS makes the data on disk consistent with the updated data that exists only in memory. Next, you calculate the average number of checkpoints, and determine the required number of checkpoints. Finally, you calculate the size of a fixed AI extent.

Assume that your database is running. Follow these steps to size a fixed AI extent using PROMON:

Step	Action
1	Invoke PROMON on your running database.
2	At the “Enter your selection:” prompt, enter R&D (for Advanced options).
3	In the Main (Top) Menu, enter 1 (for Status Displays).
4	In the Status Displays Menu, enter 9 (for BI Log). Status information about the BI log such as BI cluster size, number of BI buffers, and BI block size is displayed.
5	Make a note of the BI cluster size.
6	Enter p twice to return to the Main (Top) Menu.

Steps	Action
7	In the Main (Top) Menu, enter 3 (for Other Displays).
8	In the Other Displays Menu, enter 4 (for Checkpoints). Information about checkpoints is displayed. **Note:** Make sure the report has eight checkpoints. This is the maximum number of checkpoints that PROMON displays during a sample period.
9	Look at the “Len” (for length) column, which indicates the length of time it takes to write modified data associated with the previous BI cluster to disk. **Note:** This represents the approximate interval between checkpoints.
10	To calculate the average interval in minutes between checkpoints during a two-hour period of peak activity, first add up all the Len values and divide the sum by the number of Len values. Then, convert the result from seconds to minutes (by dividing by 60).
11	Determine your required interval in minutes between checkpoints. Best practice: For optimal performance, the interval in minutes between checkpoints should be between 5 and 10 minutes.
21	Use this formula to calculate a fixed AI extent: `Fixed AI extent size = BI cluster size * required interval in minutes / average interval in minutes between checkpoints * 1.25` **Note:** In this formula, each fixed AI extent size is increased by 25 percent to account for unanticipated growth.

Procedure: Sizing fixed AI extents using OpenEdge Management

If you have Progress® OpenEdge® Management, you can use it to find the same information that PROMON provides in order to size fixed AI extents.

Follow these steps to size a fixed AI extent using OpenEdge Management:

Steps	Action
1	Launch OpenEdge Management. The Authentication page opens.

Steps	Action
2	Enter the username and password of the admin user and then click Login. The OpenEdge Management page opens.
3	Click **Resources > Go to Resources** and then click the name of the database for which you want to size fixed AI extents. The Database page appears.
4	In the **Operational Views** of the running database, click **Page Writers**. The Page Writers page appears.
5	In the **BI summary** section, make a note of the BI cluster size.
6	In the **Checkpoint summary** section, look at the **Duration** column, which displays values in minutes and seconds. **Note:** This represents the approximate interval between checkpoints.
7	To calculate the average interval in minutes between checkpoints during a two-hour period of peak activity, first convert the Duration values to seconds. Then, add up all the Duration values and divide the sum by the number of Duration values. Last, convert the result back to minutes (by dividing by 60).
8	Determine your required interval in minutes between checkpoints. Best practice: For optimal performance, the interval in minutes between checkpoints should be between 5 and 10 minutes.
9	Use this formula to calculate a fixed AI extent: `Fixed AI extent size = BI cluster size * required interval in minutes / average interval in minutes between checkpoints * 1.25` **Note:** In this formula, each fixed AI extent size is increased by 25 percent to account for unanticipated growth.

Example of sizing AI extents

Let’s look at an example of how to size fixed AI extents for a database.

Assume that using the procedures shown previously, you determine the following values for your database:

- The current BI cluster size is 16MB.
- The average interval in minutes between checkpoints during a two-hour period of peak activity is 2.5 minutes.

- You choose 10 minutes as the required interval between checkpoints.
- Therefore:

Fixed AI size = 16MB * (10 minutes / 2.5 minutes) * 1.25 = 80MB

- 80MB is the size you use to create each fixed AI extent for your database.

Creating fixed AI extents

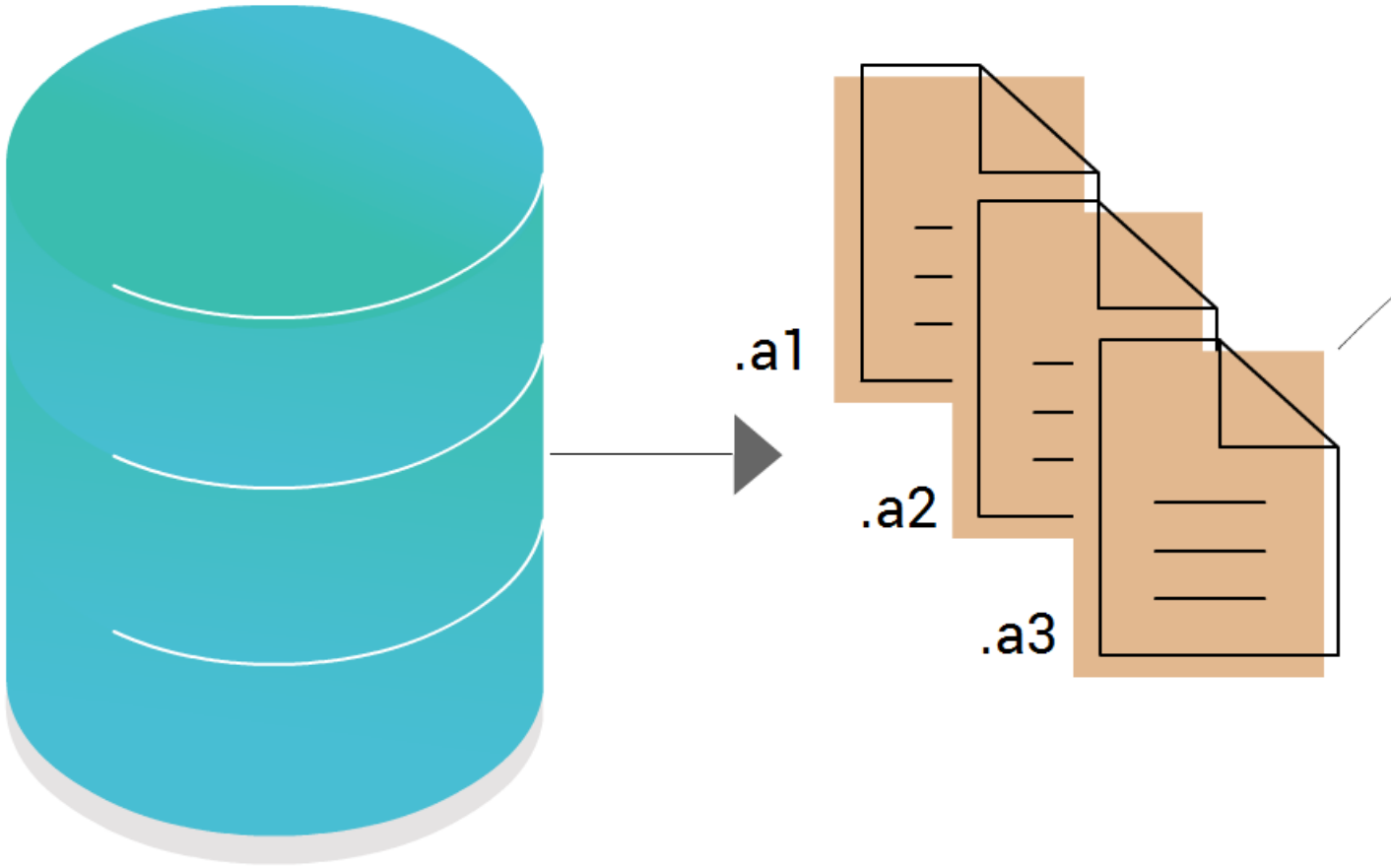
After you size fixed AI extents for a database, you need to create them. Recall that each AI storage area contains only one extent. Follow these guidelines:

- Create at least three fixed AI extents using the size you calculated—one that can be archived, one that is being written to, and one that is empty and ready for use.
- Place the AI extents on a disk that is different from the disks that contain the BI and application data extents.If the disk that contains the database files crashes, you would still be able to use your most recent full and incremental backups and the associated AI extents to recover most, or all, of your committed transactions, up to the point of the media failure.

You can add extents to a database online or offline.

- If your business environment allows downtime for performing database maintenance tasks, then you can add fixed AI extents offline, using the PROSTRCT ADD utility.
- If your database must be available 24x7, then you can add fixed AI extents online, using the PROSTRCT ADDONLINE utility.

Enabling after-imaging and automatic AI file management offline



If your business environment allows downtime to perform database maintenance tasks, you can enable after-imaging and automatic AI file management offline.

To enable after-imaging and automatic AI file management offline, you perform the following tasks:

1. Enable after-imaging offline.
2. Enable automatic AI file management offline.
3. Start your database with automatic AI file management startup parameters. Alternatively, you can configure automatic AI management startup parameters for a database using OpenEdge Explorer.
4. You will learn how to perform each of these tasks next.

Enabling after-imaging offline

Before you can enable after-imaging for a database offline, you must:

1. Shut down the database.
2. Perform a full backup of the required files of the database using PROBKUP.
3. Back up the optional database files using operating system (OS) or third-party backup tools.

Note: If your backup strategy requires that you back up a database using OS or third-party backup tools, you should mark the database as backed up using the RFUTIL utility with the MARK BACKEDUP qualifier.

Then, you use the RFUTIL utility with the AIMAGE BEGIN qualifier to enable after-imaging for the database offline. When after-imaging is enabled, OpenEdge RDBMS continuously writes AI notes about committed transactions to the AI extents.

The command to enable after-imaging for a database offline is:

```
rfutil db-name –C aimage begin
```

Suppose you have brought the sports database offline and performed its full backup.

Here is an example of how to enable after-imaging for the sports database offline:

```
rfutil sports –C aimage begin
```

Enabling automatic AI file management for a database offline

After you enable after-imaging for a database, you can enable automatic AI file management.

The automatic AI file management utility makes managing AI extents easy for you. The AI archiver represents the AI file management utility, and is a daemon process that is launched by the broker of a running database. The AI archiver does the following:

- Archives full AI extents to a user-specified location.
- Empties full AI extents for reuse.
- Maintains an archive log file to help with the roll forward recovery process.
- Is tightly integrated with OpenEdge Replication.

You use the RFUTIL utility with the AIARCHIVER ENABLE qualifier to enable automatic AI file management for a database offline.

The command to enable automatic AI file management for a database offline is:

```
rfutil db-name –C aiarchiver enable
```

Once the AI archiver is enabled for a database, you can start up the database in multi-user mode. When you do so, the broker is launched, and it in turn launches the AI archiver.

Assume that you have enabled after-imaging for the sports database.

Here is an example of how to enable automatic AI file management for the **sports** database offline:

```
rfutil sports –C aiarchiver enable
```

Starting a database with automatic AI file management startup parameters

After you enable after-imaging and automatic AI file management for a database offline, you can start up the database with automatic AI file management startup parameters using PROSERVE. As a best practice, you should add the automatic AI file management parameters to the parameter file and use the file to start up the database.

The command to start up a database using automatic AI file management parameters is:

```
proserve db-name –aiarcdir dirlist [–aiarcinterval number-of-seconds]
```

Where:

`–aiarcdir dirlist`	Specifies one or more AI directories for the archived AI extents of your database. This parameter is required every time you use PROSERVE. When you specify multiple directories: • In UNIX, list the directories using a comma but without embedded spaces. • In Windows, list the directories using a comma but without embedded spaces, and enclose the directory names in double quotes (“ ”) Before you use this parameter, ensure that the directories have adequate space for future archived AI extents.
`–aiarcinterval number-of-seconds`	Specifies the archive interval in seconds. The AI archiver operates in two modes: • Timed—The AI archiver backs up the busy AI extent at a specified interval or when the extent is full, and empties it for reuse.If you do not want to lose data for more than a specific time interval, then you should use the timed mode. Specify a value between 120 (2 minutes) and 86400 (24 hours). • On-demand—The AI archiver backs up AI extents only when they are full, and empties them for reuse. If you want to use the on-demand mode, omit this parameter. Note: In either mode, the AI archiver checks for full AI extents every 5 seconds.

Assume that you have enabled after-imaging and automatic AI file management for the sports database offline. Further, suppose that you do not want to lose more than 5 minutes of data.

Here is an example of how to start up the **sports** database—with full AI extents set to be archived to the ai directory at most every 5 minutes (5x60 = 300 seconds):

```
proserve sports –aiarcdir ..\ai –aiarcinterval 300
```

Procedure: Configuring AI management parameters for a database using OpenEdge Explorer

If you use OpenEdge Explorer to start up a database, you can configure automatic AI file management startup parameters in the database configuration properties.

Note: OpenEdge Explorer only allows you to configure the timed AI archival mode. You cannot use OpenEdge Explorer to configure the on-demand mode.

Follow these steps to configure automatic AI file management startup parameters for a database using OpenEdge Explorer:

Steps	Action
1	Launch OpenEdge Explorer. The **Authentication** page opens.
2	Enter the username and password of the admin user and then click **Login**. The OpenEdge Explorer page opens.
3	Click **Resources > Go to Resources**. The Resources page opens.
4	Click the name of the database for which you want to configure automatic AI file management startup parameters. The Database page opens.
5	From the **Database** page, click the **Configuration** link, and then click the **configuration.db-name.defaultconfiguration** link. The **Database Configuration** page opens.
6	Click **Edit**, and then specify the following automatic AI file management startup parameters in the **After Image Management** section: • In the **After-image archival directory list** field, specify one or more directories. • Select the **Create after-image archival directories** check box if the specified directories do not exist. • In the **After-image archival interval** field, specify a time interval in seconds.
7	Click **Save** to save your changes.

After you specify automatic AI file management startup parameters, you can start the database using OpenEdge Explorer.

Enabling after-imaging and automatic AI file management online

If your database must be online 24x7, you can enable after-imaging and automatic AI file management online.

You use the PROBKUP utility to perform a full backup of a database and at the same time enable after-imaging and automatic AI file management for the database online.

The command to enable after-imaging and automatic AI file management for a database online is:

```
probkup online db-name enableai backup-filename enableaiarchiver –aiarcdir dirlist
[-aiarcinterval number-of-seconds]
```

Where:

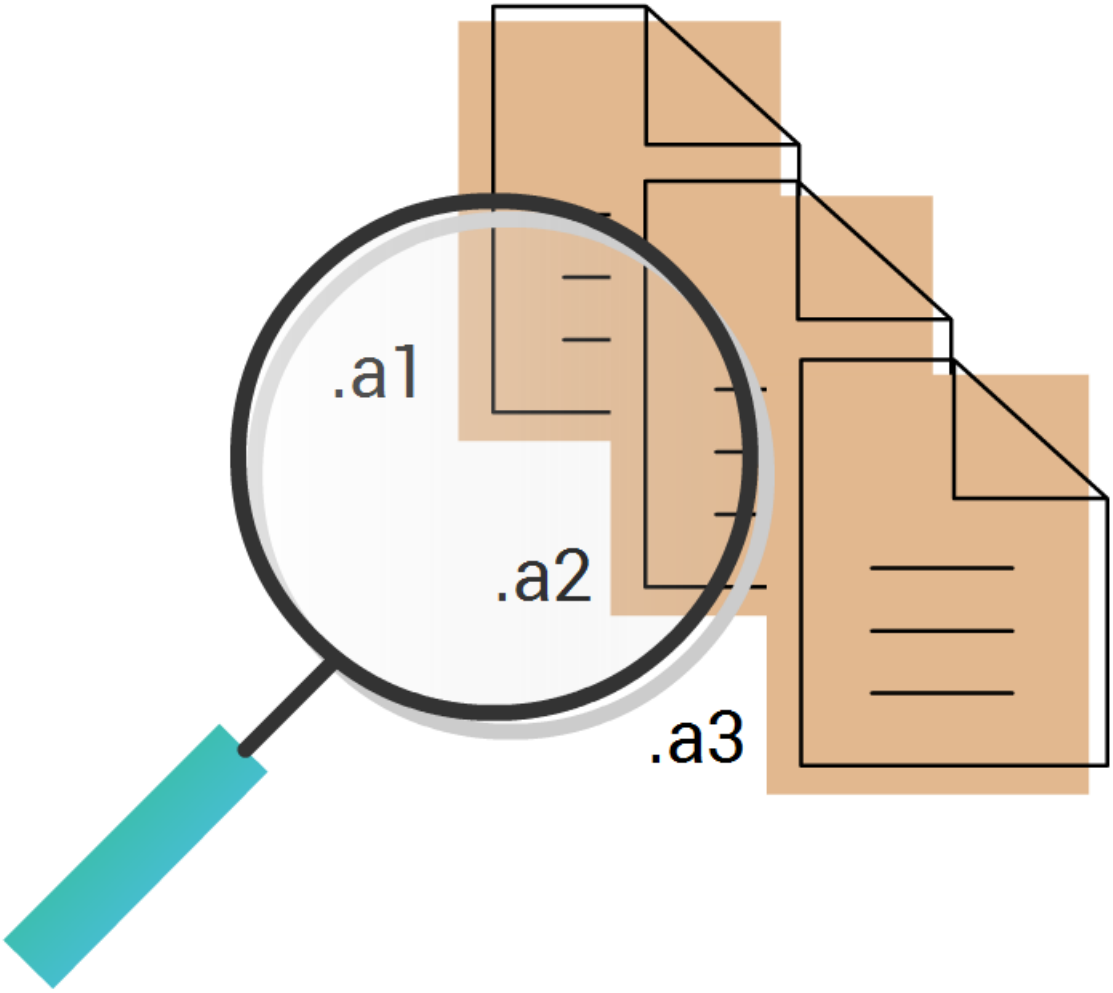
`-aiarcdir dirlist`	Specifies one or more AI directories that are set aside for the archived AI extents of your database. When you specify multiple directories: • In UNIX, list the directories using a comma but without embedded spaces. • In Windows, list the directories using a comma but without embedded spaces, and enclose the directory names in double quotes (“ ”).
`-aiarcinterval number-of-seconds`	Specifies the archive interval in seconds. For timed mode, use a value between 120 (2 minutes) and 86400 (24 hours). For on-demand mode, omit this parameter.

Assume that the **sports** database is running. Here is an example of how to perform a full backup and enable after-imaging and AI file management for sports online—with full AI extents set to be archived to the ai directory at most every 5 minutes:

```
probkup online sports enableai sports.bkp enableaiarchiver –aiarcdir ..\ai
–aiarcinterval 300
```

3

Managing AI extents



After you enable after-imaging and automatic AI file management—whether online or offline—for a database, OpenEdge RDBMS continuously writes AI notes about database transactions to the AI extents. During that time, the AI archiver archives full AI extents and empties them for reuse.

Over time, you will need to perform one or more of the following tasks:

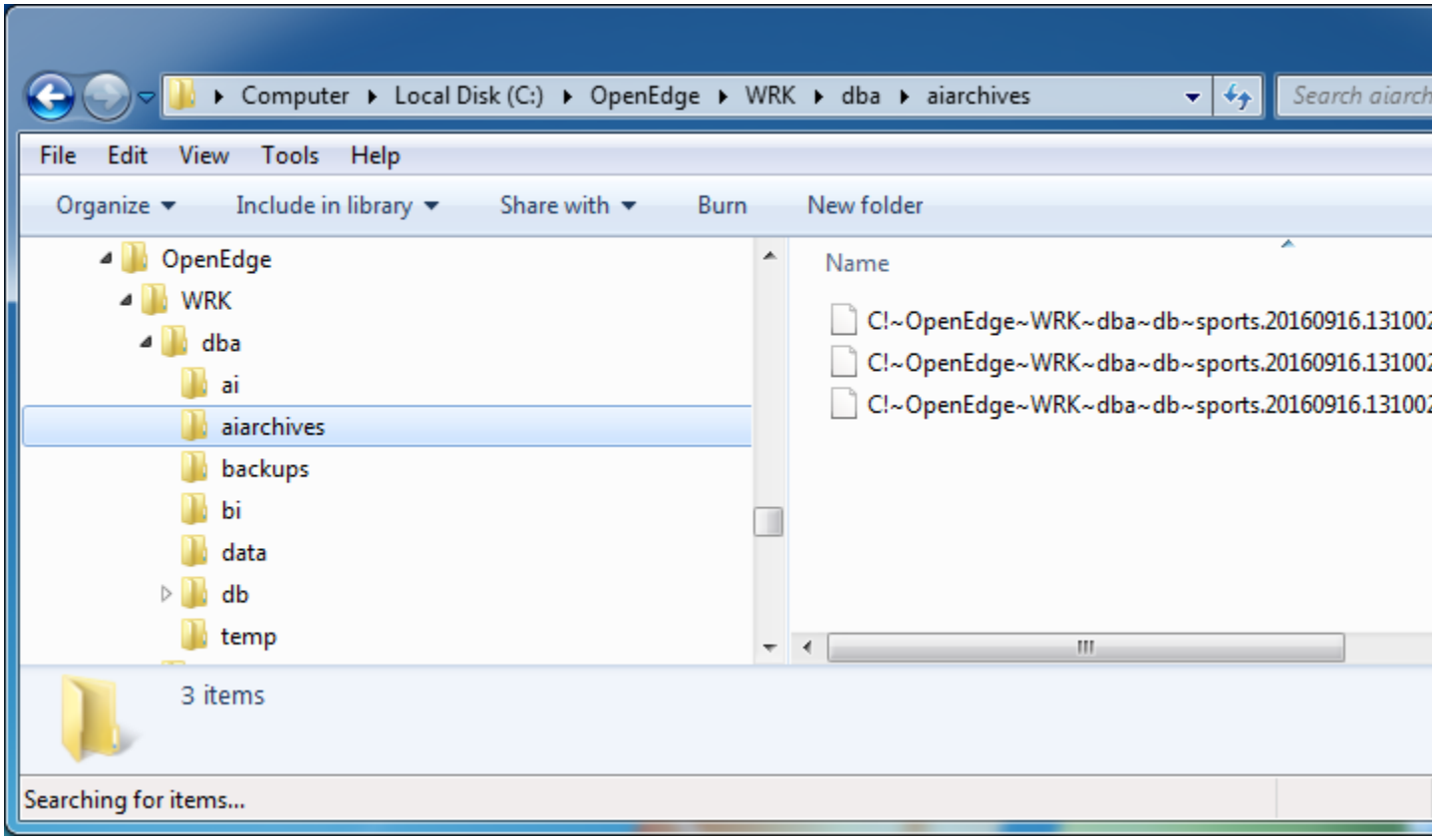
- Manage archived AI files.
- Adjust automatic AI file management parameters.

You will learn how to perform each of these tasks next.

For details, see the following topics:

- Managing archived AI files
- Adjusting automatic AI file management parameters

Managing archived AI files



The AI archiver automatically creates and names archived AI files in the AI archive directory. It is recommended that you retain at least one week's worth of archived AI files.

Using OS tools, you should monitor the archived AI files to see what has been archived and periodically delete old archived files to free up disk space.

The archived AI file name identifies the AI extent and the state of the database at the time of the archive operation. Each archived AI file name comprises the following components:

- The fully qualified file name of the database
- The date and time of the archive
- The AI sequence number
- The name of the AI extent

Here is an example of the name of an archived AI file of the sports database:

```
C!~OpenEdge~WRK~dba~db~sports.20160914.090933.00000001.sports.a1
```

The following table annotates the preceding code example:

Code	Description
C!~OpenEdge~WRK~dba~db~sports	The fully qualified file name of the database. The “~” sign replaces the directory separator (/). In Windows, an exclamation point (!) replaces the colon (:) following the disk drive letter.
20160914.090933	The date and time of the archive.
00000001	The AI sequence number.
sports.a1	The name of the AI extent.

Adjusting automatic AI file management parameters

Once you set aside enough space for your archived AI files and enable automatic AI file management, the AI archiver runs by itself without requiring much intervention. There are times, however, when you may need to step in and help the AI archiver, including:

- When you need to direct the AI archiver to a different list of directories because the disk that contains the current list of directories is becoming full.
- When you need to modify the time interval because you have a more accurate idea of how frequently AI extents are being archived.

To specify a different AI archive directory list, you use RFUTIL with the AIARCHIVER SETDIR qualifier online or offline.

The command to specify a different AI archive directory list online or offline is:

```
rfutil db-name –C aiarchiver setdir dirlist
```

To specify a different time interval, use RFUTIL with the AIARCHIVER SETINTERVAL qualifier online or offline.

The command to specify a different time interval online or offline is:

```
rfutil db-name –C aiarchiver setinterval number-of-seconds
```

Here is an example of how to specify a different list of AI archive directories (**aiarchive1** and **aiarchive2**) for the **sports** database online or offline:

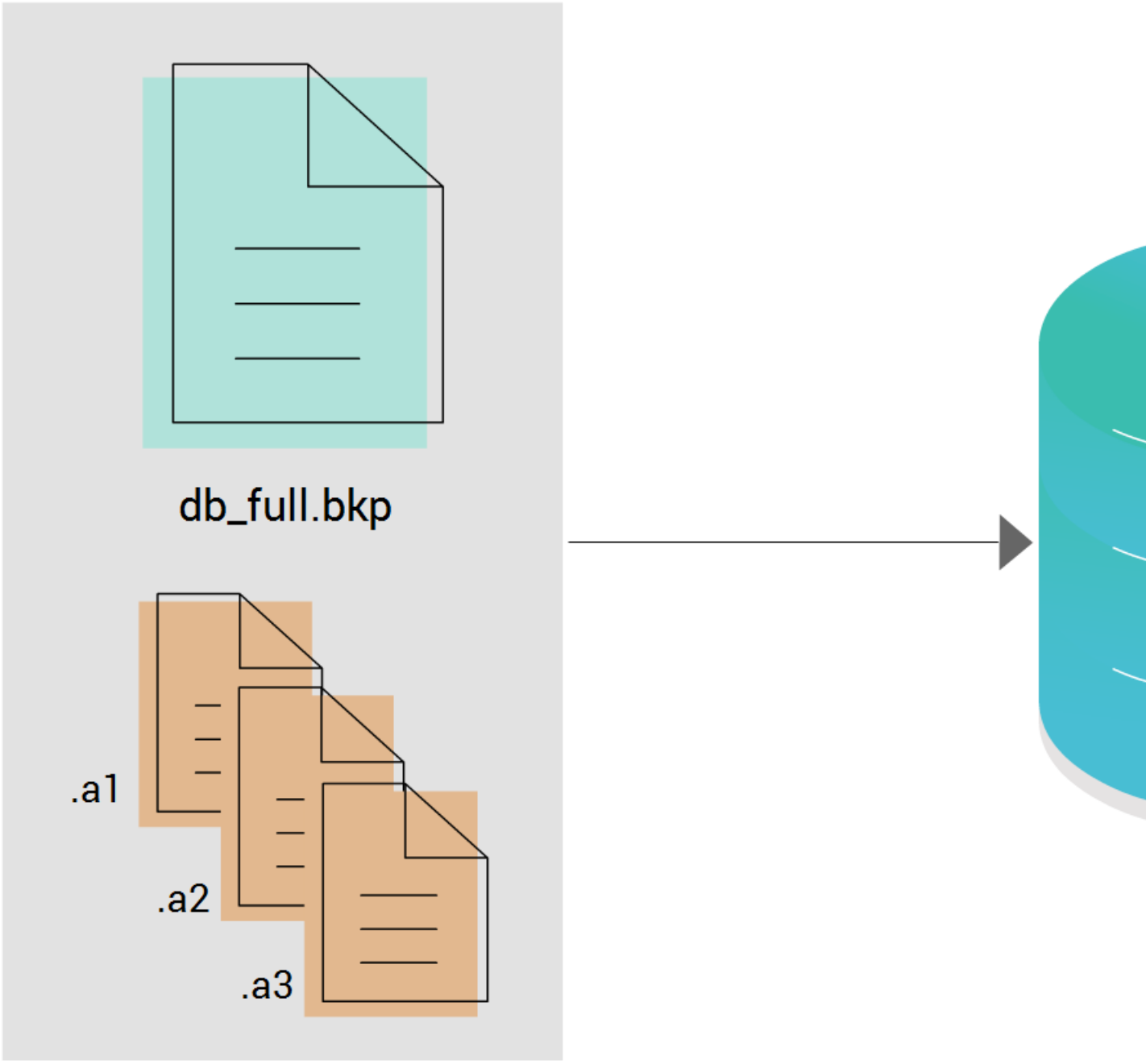
```
rfutil sports -C aiarchiver setdir ../aiarchive1,../aiarchive2
```

Here is an example of how to specify a different time interval (every 3 minutes) for the sports database online or offline:

```
rfutil sports –C aiarchiver setinterval 180
```

4

Performing roll-forward recovery



While your database is running, if a failure happens—be it power outage, disk damage, or database corruption—you would still be able to recover committed transactions up to the moment of failure. To do this, you restore the database from the most recent full and incremental backups, and then apply to the restored database all AI extents since the most recent backup up to the moment of failure. The end result is a new database that contains most, or all, of committed transactions up to the moment of failure.

In your work environment, you might never be required to perform roll-forward recovery for a production database. However, roll-forward recovery is a mission-critical task. It is very important that you practice your roll-forward recovery plan in a test environment. This enables you to accomplish the following:

- Ensure that your roll-forward recovery procedures work as planned.
- Determine how long the recovery process takes so that you will not be surprised when you are required to do the actual recovery.
- Be confident that you can perform the task because you have done it before.
- To perform roll-forward recovery, you perform the following tasks:

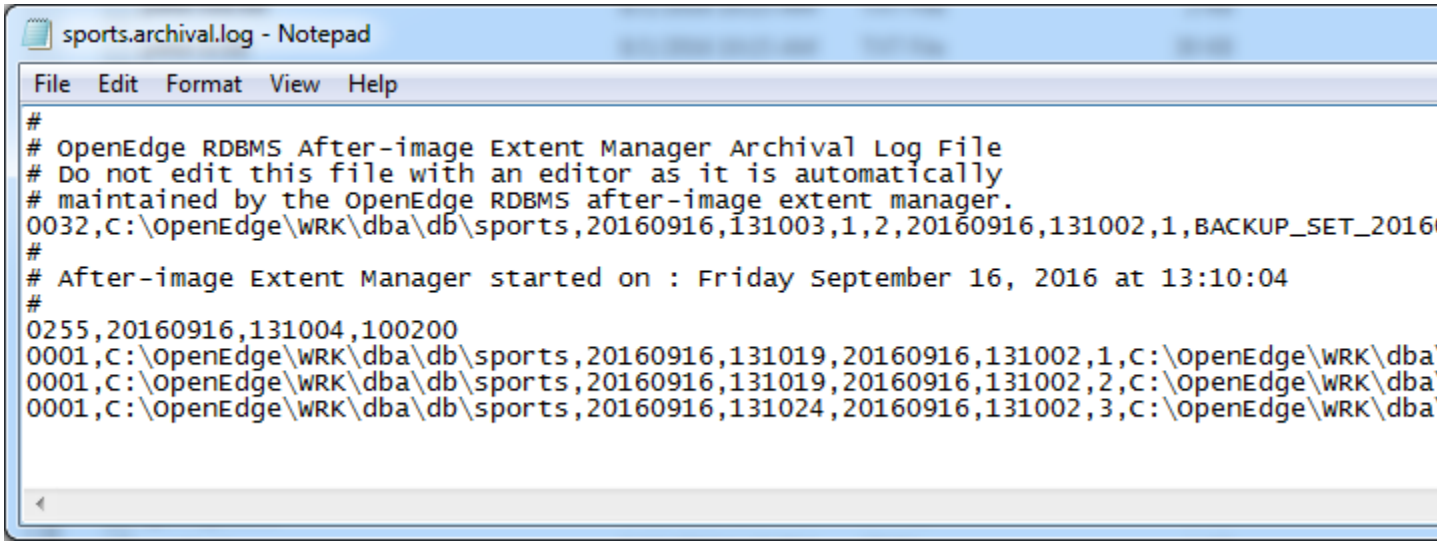
1. Identify a roll-forward recovery set in the archival log file.
2. Create a roll-forward recovery batch file.
3. Perform roll-forward recovery using the batch file.

Next, you will learn how to perform each of these tasks.

For details, see the following topics:

- Identifying a roll-forward recovery set
- Procedure: Creating a roll-forward recovery batch file
- Procedure: Performing roll-forward recovery using a batch file

Identifying a roll-forward recovery set



A roll-forward recovery set should include the following files:

- The most recent full backup of the database
- All incremental backups, if any, since the last full backup
- All archived AI files since the last backup

- If possible, the AI extents (full or busy) that have not been archived at the time of failure. If these extents are not available, committed transactions recorded in them would be lost.

The AI archiver makes it easy for you to identify a roll-forward recovery set by creating and maintaining an archival log file. The archival log records the names of database backups and archived AI files. The log resides in the same directory as your database .db file, and is named, db-name.archival.log. This file can be read, but must not be altered.

Using the information in the archival log file, make sure you can locate all the files in the roll-forward recovery set.

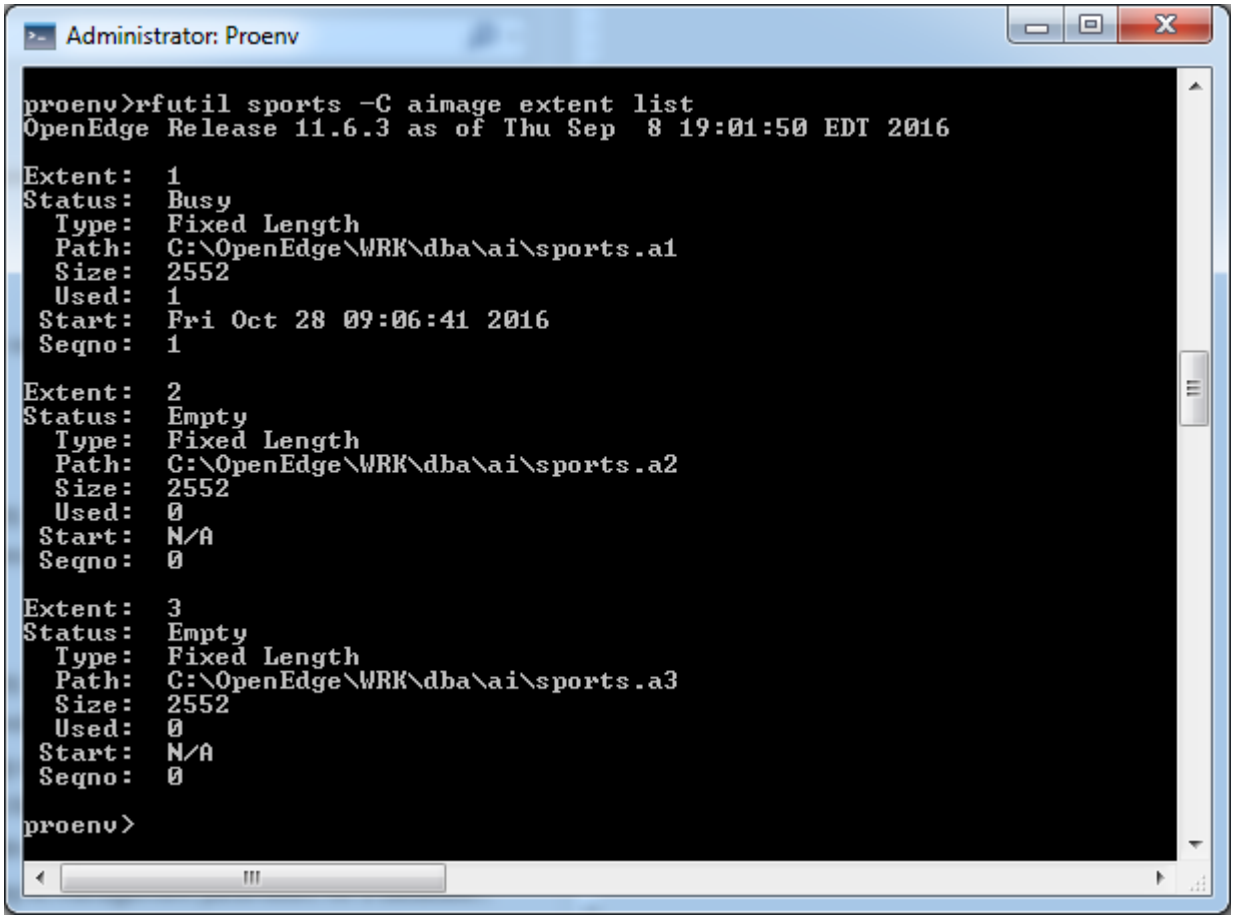
Note: As a best practice, every time you back up a database, you should also back up the archival log file.

Example of a roll-forward recovery set

The following is an example of a roll-forward recovery set in the sports.archival.log file associated with the sports database. As you can see, the set consists of a full backup, sports.bkp, and three associated archived AI files (note the fully qualified file names are highlighted in bold).

```
#
# OpenEdge RDBMS After-image Extent Manager Archival Log File
# Do not edit this file with an editor as it is automatically
# maintained by the OpenEdge RDBMS after-image extent manager.
0032,C:\OpenEdge\WRK\dba\db\sports,20160916,131003,1,2,20160916,131002,1,BACKUP_SET_20160916_131002,1,sports.bkp
#
# After-image Extent Manager started on : Friday September 16, 2016 at 13:10:04
#
0255,20160916,131004,100200
001,C:\OpenEdge\WRK\dba\sports,20160916,130921,20160916,130921,C:\OpenEdge\WRK\dba\sports.a1,C:\OpenEdge\WRK\dba\C-OpenEdge\WRK\dba\sports.20160916130200001.sports.a1
001,C:\OpenEdge\WRK\dba\sports,20160916,130922,20160916,130922,C:\OpenEdge\WRK\dba\sports.a2,C:\OpenEdge\WRK\dba\C-OpenEdge\WRK\dba\sports.20160916130200002.sports.a2
001,C:\OpenEdge\WRK\dba\sports,20160916,130923,20160916,130923,C:\OpenEdge\WRK\dba\sports.a3,C:\OpenEdge\WRK\dba\C-OpenEdge\WRK\dba\sports.20160916130200003.sports.a3
```

Identifying busy or full AI extents



In addition, you should try to identify any busy or full AI extents that have not been archived at the time of failure. If you can locate these extents and they are intact, you will be able to recover the committed transactions they contain that were not yet archived.

You use the RFUTIL utility with the AIMAGE EXTENT LIST qualifier to generate a list of AI extents.

The command to list AI extents is:

```
rfutil db-name –C aimage extent list
```

The following table lists each AI extent property with its description:

Property	Description
Extent	The number of the AI extent as defined in the .st file.

Property	Description
Status	Key values are: Empty—The AI extent is available for use. Busy—The AI extent is currently being used. Full—The AI extent is full. Note: There are other values if you use OpenEdge Replication. See the Progress OpenEdge Database Administration guide.
Type	Fixed extent or variable extent.
Path	The fully qualified name of the AI extent.
Size	The size of the AI extent in 1K blocks.
Used	The number of 1K blocks used in each AI extent.
Start	The date and time the AI extent was started to be written.
Seqno	The sequence number of the AI extent. This is useful for performing roll-forward recovery because in that process you must apply the AI extents in the sequence they were written by OpenEdge RDBMS.

Here is an example of how to generate a list of the AI extents in the sports database:

```
rfutil sports –C aimage extent list
```

Procedure: Creating a roll-forward recovery batch file

After you identify the roll-forward recovery set and locate all the files, you are ready to create a batch file to perform roll-forward recovery.

Follow these steps to create a roll-forward recovery batch file:

Step	Action
1	Copy and paste the roll-forward recovery set into a text editor.
2	Edit the text so that you have only the backup file names and the fully qualified AI file names.
3	Add the fully qualified file names of any full or busy AI extents.

Steps	Action
4	Add a CALL command to invoke PROREST to restore the most recent full backup. Use this syntax: `call prorest db-name backup-filename`
5	Add CALL commands to invoke PROREST to restore each incremental backup.

6	Add CALL commands to invoke RFUTIL with the ROLL FORWARD qualifier to apply each archived AI file.
	Here is the syntax for RFUTIL with the ROLL FORWARD qualifier:
	`rfutil db-name -C roll forward –a ai-filename`
	Here is the syntax for the CALL command to invoke RFUTIL ROLL FORWARD:
	`call rfutil db-name -C roll forward –a ai-filename`

7	Add CALL commands to invoke RFUTIL with the ROLL FORWARD qualifier to apply the AI extents (full or busy) that were not archived.
8	Save the file as a .bat file.

Example of a roll-forward recovery batch file

Here is an example of a roll-forward recovery batch file that creates a new database, sports2, based on the most recent full backup of the sports database and applies the three associated archived AI extents—including the one busy extent that was not archived (note the last line in the file) at the time of failure. Notice that the CALL command is used to invoke PROREST and RFUTIL.

```
call prorest sports2 C:\OpenEdge\WRK\dba\db\sports.bkp
call rfutil sports2 -C roll forward -a
C:\OpenEdge\WRK\dba\ai\C!~OpenEdge~WRK~dba~db~sports.20160916.131002.00000001.sports.a1
call rfutil sports2 -C roll forward -a
C:\OpenEdge\WRK\dba\ai\C!~OpenEdge~WRK~dba~db~sports.20160916.131002.00000002.sports.a2
call rfutil sports2 -C roll forward -a
C:\OpenEdge\WRK\dba\ai\C!~OpenEdge~WRK~dba~db~sports.20160916.131002.00000003.sports.a3
call rfutil sports2 -C roll forward -a C:\OpenEdge\WRK\dba\ai\sports.a1
```

Procedure: Performing roll-forward recovery using a batch file

After you create a roll-forward recovery batch file, you are ready to perform roll-forward recovery.

Follow these steps to perform roll-forward recovery using a batch file:

Steps	Action
1	Ensure that you have sufficient disk space for the new database.
2	Use the .st file of the original database to create a new .st file that reflects the location of the new database.
3	Run the roll-forward recovery batch file to perform roll-forward recovery. When the batch file runs successfully, at the end you should see information like the following: `At the end of the .ai file, 0 transactions were still active. (1636)Roll forward of the after-image busy extent is complete. (3786)All roll forward for database db-name is complete. (3787)` **Note:** "0 transactions" indicates that all committed transactions have been recovered up to the moment of failure.
4	Back up the new database.
5	Enable after-imaging for the new database.
6	Enable automatic AI file management for the new database.

5

Disabling after-imaging

There are times when disabling after-imaging is required such as when you need to:

- Remove an AI extent.
- Rebuild indexes.
- Convert an OpenEdge Release 10 to an OpenEdge Release 11 database.

To improve performance, you should also disable after-imaging if you want to run a batch file that involves many database updates.

To disable after-imaging, you perform the following steps:

1. Shut down the database.
2. Back up any full AI extents that have not been backed up.Important: When you disable after-imaging, OpenEdge RDBMS marks all the AI extents as empty. As a result, you will not be able to access any data in the AI extents that have not been backed up.
3. Back up the AI extent that is busy.
4. Disable after-imaging.
5. Back up the database.

Note: When you disable after-imaging for a database, automatic AI file management is also disabled.

You use RFUTIL with the AIMAGE END qualifier to disable after-imaging.

The command to disable after-imaging is:

```
rfutil db-name -C aimage end
```

Here is an example of how to disable after-imaging for the sports database:

```
rfutil sports -C aimage end
```

Automatic AI file management is also disabled.