Primary Author: David Goldman

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are “commercial computer software” or “commercial technical data” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>vii</td>
</tr>
</tbody>
</table>

## 1 Introduction
<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OracleAS CDC Adapter for DB2 Overview</td>
<td>1-1</td>
</tr>
<tr>
<td>OracleAS CDC Adapter for DB2 Architecture</td>
<td>1-2</td>
</tr>
<tr>
<td>The Staging Area</td>
<td>1-4</td>
</tr>
</tbody>
</table>

## 2 Configuring the OracleAS CDC Adapter for DB2
<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting Up the IBM z/OS Platform in Oracle Studio</td>
<td>2-1</td>
</tr>
<tr>
<td>Securing Access to Oracle Connect</td>
<td>2-2</td>
</tr>
<tr>
<td>Setting Password Access to Oracle Studio</td>
<td>2-2</td>
</tr>
<tr>
<td>Specifying Users with Administrative Rights</td>
<td>2-4</td>
</tr>
<tr>
<td>Setting Up Run-Time User Access to the IBM z/OS Platform</td>
<td>2-5</td>
</tr>
<tr>
<td>Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2</td>
<td>2-6</td>
</tr>
<tr>
<td>Configuring the DB2 Tables for CDC</td>
<td>2-6</td>
</tr>
<tr>
<td>Setting up Security for the OracleAS CDC Adapter for DB2</td>
<td>2-6</td>
</tr>
<tr>
<td>Configuring the ATTSRVR Started Task</td>
<td>2-7</td>
</tr>
<tr>
<td>Setting up a Change Data Capture in Oracle Studio</td>
<td>2-7</td>
</tr>
<tr>
<td>Create a CDC Project</td>
<td>2-8</td>
</tr>
<tr>
<td>Set up the CDC Server</td>
<td>2-11</td>
</tr>
<tr>
<td>Set up the Staging Area Server</td>
<td>2-15</td>
</tr>
<tr>
<td>Deploying a Change Data Capture</td>
<td>2-24</td>
</tr>
<tr>
<td>Activating and Deactivating Solution Workspaces</td>
<td>2-27</td>
</tr>
</tbody>
</table>

## 3 WebLogic Deployment and Adapter Integration
<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring the DB2 CDC Adapter In the Application Server</td>
<td>3-1</td>
</tr>
<tr>
<td>CDC Stream Positions Table Definition</td>
<td>3-1</td>
</tr>
<tr>
<td>Configuring the DB2 CDC Adapter Connection Factory</td>
<td>3-2</td>
</tr>
<tr>
<td>Configuring the JCA Configuration Files for the OracleAS CDC Adapter for DB2</td>
<td>3-3</td>
</tr>
<tr>
<td>Creating JCA Configuration Files</td>
<td>3-3</td>
</tr>
<tr>
<td>Configuring the Binding File</td>
<td>3-4</td>
</tr>
<tr>
<td>High Availability</td>
<td>3-5</td>
</tr>
<tr>
<td>Troubleshooting Rollback Loops</td>
<td>3-6</td>
</tr>
</tbody>
</table>
4 Configuring a DB2 CDC Adapter

Creating Outbound Interactions with the OracleAS Adapter for DB2 ........................................ 4-1
  Requirements .......................................................................................................................... 4-1
Adding the DB2 Hospital Data Source .................................................................................... 4-1
  Prepare the System .............................................................................................................. 4-1
  Set up Machine Access to Oracle Connect ......................................................................... 4-2
Create a Change Data Capture .............................................................................................. 4-3
  Setting up a Change Data Capture ..................................................................................... 4-3
  Deploy the Solution ............................................................................................................. 4-5
Create the JCA Configuration Files ...................................................................................... 4-5

5 Creating the Project in JDeveloper Examples

Getting Started .................................................................................................................. 5-1
  Prerequisites ..................................................................................................................... 5-1
  Beginning Tasks ................................................................................................................. 5-2
Deploy a BPEL Inbound Process .......................................................................................... 5-2
  Design-Time Configuration ............................................................................................... 5-3
    Create a BPEL Inbound Process ..................................................................................... 5-3
    Use a Third-Party Adapter for the Inbound Process ......................................................... 5-4
    Design the BPEL Inbound Process ............................................................................... 5-6
    Deploy the Project ......................................................................................................... 5-9
Run-time Configuration ...................................................................................................... 5-9
  Triggering the Event using the XML Utility ....................................................................... 5-9
Mediator Inbound Process .................................................................................................... 5-10
  Beginning Tasks ................................................................................................................ 5-11
  Create a Mediator Inbound Process .................................................................................. 5-11
  Use a Third-Party Adapter for the Inbound Process ......................................................... 5-13
  Create a File Adapter ....................................................................................................... 5-14
  Design an Inbound Mediator ............................................................................................ 5-16
  Create Mapping Rules and Map the Files ....................................................................... 5-17
  Deploy the Project ............................................................................................................. 5-17

6 Run-time Tasks and Troubleshooting

Adding and Removing Tables ............................................................................................... 6-1
Handling Metadata Changes .............................................................................................. 6-2
Staging Area Maintenance ................................................................................................. 6-3
Monitoring the Change Data Capture .................................................................................. 6-4
  Service Context Table ....................................................................................................... 6-4
  Monitoring the Status ....................................................................................................... 6-5
Daemon Life-Cycle Tasks ..................................................................................................... 6-8
  Starting the Daemon ......................................................................................................... 6-8
    Task: Starting the Daemon ............................................................................................ 6-8
  Shutting Down the Daemon ............................................................................................. 6-8
  Monitoring the Daemon During Run time ....................................................................... 6-8
    Daemon (Computer) Options ....................................................................................... 6-9
    Workspace Options ....................................................................................................... 6-9
7 Advanced Features of OracleAS CDC Adapter for DB2

Configuring the Daemon for High Availability ............................................................ 7-1
   Adding a New Daemon Workspace Configuration .................................................. 7-1
   Editing the Workspace ......................................................................................... 7-2
   Configuring the Server Mode .............................................................................. 7-2
Configuring a Binding Environment ........................................................................ 7-6
   Debug ................................................................................................................... 7-7
   General .................................................................................................................. 7-8
   Language ............................................................................................................... 7-8
   Modeling ............................................................................................................... 7-9
   ODBC .................................................................................................................... 7-9
   OLEDB .................................................................................................................. 7-9
   Optimizer ............................................................................................................ 7-9
   Parallel Processing.............................................................................................. 7-10
   Query Processor ................................................................................................. 7-10
   Transaction ......................................................................................................... 7-13
   Tuning .................................................................................................................. 7-13
   XML ...................................................................................................................... 7-14
Migration Considerations .......................................................................................... 7-15

Security Considerations ............................................................................................ 7-15
   Setting Design Time Security ........................................................................... 7-16
   Setting Run-time Security .................................................................................. 7-16
Transaction Support .................................................................................................. 7-17

A Advanced Tuning of the Metadata

Metadata for the DB2 Data Source ........................................................................... A-1
   General Tab ......................................................................................................... A-2
   Columns Tab ....................................................................................................... A-3
      Column Definition Section ............................................................................. A-4
      Column Properties ......................................................................................... A-5
   Indexes Tab ......................................................................................................... A-7
   Table Information .............................................................................................. A-7
      Properties ....................................................................................................... A-7
   Statistics Tab ..................................................................................................... A-9
      Generating Statistics ..................................................................................... A-10
   Advanced Tab .................................................................................................. A-11
Adapter Metadata .................................................................................................. A-13
   Adapter Metadata General Properties ............................................................. A-13
   Adapter Metadata Interactions .......................................................................... A-15
      Interaction Advanced Tab ............................................................................ A-16
This guide is the primary source of user and reference information on the OracleAS CDC Adapter for DB2, which enables Oracle users to use change data capture in their integration architecture, working with the Oracle Application Server. This preface includes the following topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

**Audience**

This guide is intended for Oracle integration administrators and developers who perform the following tasks:

- Installing and configuring OracleAS CDC Adapter for DB2
- Diagnosing errors
- Using OracleAS for developing integration solutions using change data capture

---

**Note:** You should understand the fundamentals of OracleAS, WebLogic, the UNIX and Microsoft Windows operating system before using this guide to install or administer OracleAS Adapters for DB2.

---

**Note:** For the purposes of this version of the Oracle Weblogic Server, Oracle Application Server refers to the Oracle WebLogic Server and OracleAS refers to Oracle WLS.

---

**Documentation Accessibility**

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be
accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

**Accessibility of Code Examples in Documentation**
Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

**Accessibility of Links to External Web Sites in Documentation**
This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

**Access to Oracle Support**
Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

**Related Documents**
For more information, see the following documents in the Oracle Other Product One Release 10.0 documentation set or in the Oracle Other Product Two Release 6.1 documentation set:

- Oracle Application Server Adapter Concepts Guide
- Oracle Application Server Adapter Installation Guide
- Oracle Application Server Adapter Concepts Guide
- Oracle Application Server Containers for J2EE User’s Guide
- Oracle Application Server Containers for J2EE Services Guide
- Oracle Application Server Containers for J2EE Security Guide

**Conventions**
The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Change data capture (CDC) is an integration technology that enables efficient near real-time integration by capturing changes made to enterprise data sources and providing them for processing in integration middleware, such as Oracle Fusion Middleware. The CDC Adapter can be used by Oracle SOA Suite (BPEL Process Manager, Oracle Enterprise Service Bus) and Oracle Data Integrator (ODI).

The Oracle CDC Adapter for DB2 captures changes that are written to DB2 log and archive files.

The DB2 CDC adapter uses the DSNJU004 module internally to find an archive or log file that contains the first LRBA corresponding to the provided time stamp, based on the provided bootstrap data set.

The DB2 CDC adapter uses the IFI interface to capture the records from the log. The monitoring process is started using the IFI command, `-STA TRA(MON) CLASS(1) IFCID(306)`.

**Notes:**
- The DB2 agent does not support DB2 Data Sharing (also known as DB2plex).
- The DB2 agent for mainframe systems does not support DB2 instances that use the coupling facility.

OracleAS CDC Adapter

This chapter provides an overview of the features and architecture of the OracleAS CDC Adapter for DB2. It includes the following topics:

- **OracleAS CDC Adapter for DB2 Overview**
- **OracleAS CDC Adapter for DB2 Architecture**

**OracleAS CDC Adapter for DB2 Overview**

The OracleAS CDC Adapter for DB2 captures and delivers changes (such as insert, update, and delete operations) made to data in DB2 data sources. By using the CDC Adapter, you can build efficient data integration solutions that use DB2 data, synchronize systems to ensure data integrity, update data warehouses and operational data stores in near real-time, and enable event-driven business activity monitoring and processing.
The OracleAS CDC Adapter for DB2 has the following capabilities:

- **Non-intrusive change capture**: The OracleAS CDC Adapter for DB2 is a log-based CDC technology. The CDC agent accesses a logstream that is polled to access the changes. The agent begins polling from a specific point called the stream position (or agent context). The last position read is saved as the stream position. This next time the agent polls the logstream for information, it starts at the saved stream position.

- **Real-time data capture**: The OracleAS CDC Adapter for DB2 captures changes to data in near-real-time for applications that demand zero latency and require the most up-to-date data. Unlike scheduled CDC solutions, near real-time data capture guarantees that a change event is can be made available in a matter of seconds at the consumer. The term near-real-time data capture is used because after a period of inactivity changes may be available after a few seconds.

- **Change storage**: The Oracle CDC Adapter for DB2 provides a staging area that stores the captured changes in a format that is easy to access and process. The staging area ensures that changes are persisted until they are processed, and provides facilities to clean up old change records.

- **Change access and delivery**: The Oracle CDC Adapter for DB2 includes a JCA Resource Adapter that can be configured in BPEL and ESB as an inbound endpoint, delivering changes in XML format.

- **Filtering**: The OracleAS CDC Adapter for DB2 enables you to filter captured data based on the type of operation (INSERT, UPDATE, DELETE), based on column selection and even based on specific column values.

- **Reliable delivery and recovery**: The Oracle CDC Adapter for DB2 stores changes in a reliable persistent storage and uses the concept of a ‘stream position’ that indicates the location in the change stream that was successfully moved and processed. By storing stream positions, the CDC Adapter can always restart after failure, recover, and continue moving and delivering changes from wherever it was stopped.

- **Ease of use**: The OracleAS CDC Adapter for DB2 provides an easy and intuitive way to configure CDC solutions using Oracle Studio.

Using the OracleAS CDC Adapter for DB2 makes ETL (extract, transform, and load) processes more efficient and less disruptive to the operation of the source database systems, eliminating the need for the traditional ETL downtime window.

**OracleAS CDC Adapter for DB2 Architecture**

The following figure illustrates the system components used for change data capture using the OracleAS CDC Adapter for DB2.
The OracleAS CDC Adapter for DB2 component architecture includes the following components:

- **Database Platform**: The database platform is the data source that contains the data to be captured.

- **Database Log**: The database log is a log stream that contains the raw change data.

- **Oracle Connect**: Oracle Connect runs on the back-end system and handles the special J2CA adapter that runs on the Oracle Application Server.

- **CDC agent**: This is an OracleAS CDC Adapter for DB2 component that runs under the control of the daemon (a listener process) on the computer where the back-end data is located. The capture agent provides access to change events in the transaction log for the change router.

  The agent is located on the same computer as the changes to be captured. It maintains the last position read in the journal (the stream position or context) and starts at this point the next time it polls the journal. The context is stored in the repository where the adapter definition is stored. The adapter definition includes a single primary interaction, which is used to access the appropriate journal and includes the list of tables to monitor for changes.

- **BPEL/ESB**: This is where the change events are consumed by a JCA provider that sends change records to an end point configured in either ESB or BPEL.

- **JC2A Resource Adapter**: This is a special version of the J2CA 1.5 DB2 adapter that can read change records from the staging area. It implements inbound endpoints for BPEL and ESB that pulls change events from the staging area and deliver them as XML change documents. Each endpoint handles change records for a single captured table at the source database.

  The J2CA adapter uses multiple connections to the staging area to process changes from multiple tables in parallel.

- **JDBC Driver**: This enables access to changes in the staging area using SQL queries.
- **WebLogic / BPEL Components**: Any OracleAS component that hosts and gets messages from the JC2A adapter.

- **Staging Area**: This is a set of change tables where change records are stored before being sent to BPEL/ESB. Old records from the Staging Area are deleted based upon the specified aging policy.

The Staging Area contains a DISAM data source for SQL access and a Database Events adapter for the XML access. For more information, see The Staging Area for more information.

The staging area includes the following components:

- **Change Router**: This is an OracleAS CDC Adapter for DB2 service that runs under the control of the daemon (a listener process) on the Oracle platform. The change router reads the change stream from the CDC agent, sorts them in transaction/commit order eliminating rolled-back changes, and distributes the changes into the various change tables.

  Only one instance of the change router is active at a time against a captured database. The change router also deletes old changes from the change tables according to a configurable aging policy. policy (48 hours is the default, which means that records that have been in the staging area for more than 48 hours are deleted).

- **Change Tables**: These are tables in the staging area that contain changes to one captured table in the source database. These change tables maintain change events in transaction order and in occurrence order within a transaction for each captured table. The change tables are DISAM files that are indexed to support fast access to the change data, that can be accessed directly using the stream position.

- **Change Access Service**: This is an instance Oracle Connect that is installed on a separate computer or the same computer as BPEL/ESB that allows access to the change tables. This instance of Oracle Connect is configure as the staging area.

- **Oracle Studio**: Provides the developer or administrator with a graphical user interface for setting up the CDC Agent and Staging Area, using intuitive wizards. Oracle Studio can be installed on Linux, UNIX, or Windows.

**The Staging Area**

The staging area is an area used by Oracle Connect to store captured data from a journal. The journal is scanned once and changes for every required table read during that scan are passed to the staging area, where they are stored. Therefore, the journal is scanned once each time it is polled. After the changes are written to the staging area, processing of these changes is carried out independently of the journal.

The staging area is a benefit when transactions are used. The changed data is not written to the change queue until the transaction is committed. Thus, if the transaction fails, there is no overhead of having to back out any processing done with the steps in the failed transaction.

The staging area can be on any computer with a Windows, Linux, or UNIX operating system that is running the Oracle Application Server and does not need to be run on the same server as the DB2 CDC adapter. Once the information is extracted from the journal and written to the staging area, processing of changes is carried out on the staging area only. Thus, the staging area should be set up to consider the network configuration and where the consumer application runs.
The staging area maintains the last position read by the consumer application (the staging area context) and starts at this point the next time a request from the consumer application is received. The context is stored in the repository where the staging area is maintained.

The staging area is indexed so that access to the staging area for a specific stream is quick.

The staging area is automatically cleared every 48 hours. All events that have been in the staging area for more than 48 hours are deleted.
This chapter describes how to configure Oracle Connect using Oracle Studio.

All modeling of Oracle Connect is performed using Oracle Studio. To use Oracle Studio, you first configure it to enable access to the z/OS platform with the DB2 data.

---

**Note:** The following tasks assume you have permission to access the IBM z/OS platform and that the Oracle Connect daemon is running on this computer.

Check with the system administrator to ensure these requirements are fulfilled.

---

This chapter includes the following topics:

- Setting Up the IBM z/OS Platform in Oracle Studio
- Securing Access to Oracle Connect
- Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2

### Setting Up the IBM z/OS Platform in Oracle Studio

Using Oracle Studio, perform the following steps to configure the IBM z/OS platform:

1. From the Start menu, select **Programs**, **Oracle**, and then select **Studio**. Oracle Studio opens, displaying the Design perspective.

2. Right-click **Machines** in the Configuration Explorer and select **Add Machine**. The Add Machine screen is displayed.
3. Enter the name of the computer you want to connect to, or click Browse to select the computer from the list of computers that is displayed and which use the default port (2551).

4. Specify the user name and password of the user who was specified as the administrator when Oracle Connect was installed.

   **Note:** Selecting Anonymous connection enables anyone having access to the computer to be an administrator, if this was defined for the computer.

5. Click Finish.

   The computer is displayed in the Configuration Explorer.

---

### Securing Access to Oracle Connect

Oracle Studio includes mechanisms to secure access to Oracle Connect both during modeling and at run time.

During modeling, the following security mechanisms can be applied:

- Setting Password Access to Oracle Studio
- Specifying Users with Administrative Rights

At run time client access to Oracle Connect is provided by the user profile:

- Setting Up Run-Time User Access to the IBM z/OS Platform

---

### Setting Password Access to Oracle Studio

Initially, any operation performed using Oracle Studio does not require a password. You can set a password so that the first operation that involves accessing the server from Oracle Studio requires a password to be entered.

Perform the following steps to set password access to Oracle Studio:
1. From the Start menu, select, Programs, Oracle, and then select Studio. Oracle Studio opens.

2. Select Window from the menu bar, and then select Preferences. The Preferences screen is displayed.

3. Click Studio in the left pane as shown in the following figure:

\textit{Figure 2–2 The Preferences screen}

4. Click Change Studio Master Password. The Change Master Password screen is displayed, as shown in the following figure:

\textit{Figure 2–3 The Change Master Password screen}

5. Leave the Enter current master password field blank and type a new master password in the Enter new master password field.

6. Enter the new passoword again in the Confirm new master password field.

7. Click OK.
Specifying Users with Administrative Rights

By default, only the user who was specified during the installation as an administrator has the authorization to modify settings on that computer from Oracle Studio. This user can then authorize other users to make changes or to view the definitions for a selected computer. Adding a computer to Oracle Studio is described in "Setting Up the IBM z/OS Platform in Oracle Studio".

---

**Note:** The default during installation is to enable all users to be administrators.

---

1. From the **Start** menu, select **Programs, Oracle**, and then select **Studio**. Oracle Studio opens.

2. In the Design perspective Configuration view, Right-click the computer and select **Administration Authorization**.

   The Administration Authorization screen is displayed as shown in the following figure:

![The Administration Authorization screen](image)

The screen has the following sections:

**Administrators:** Administrators can view and modify all the definitions in Oracle Studio for the selected computer. On initial entry to Oracle Studio, every user is defined as a system administrator.

**Designers:** Designers can view all the definitions for the computer in Oracle Studio and can modify any of the definitions under the Bindings and Users nodes for the selected computer. For example, Oracle Studio database administrator can add new data sources and adapters and can change the metadata definition for a table in a data source.
**Securing Access to Oracle Connect**

**Users**: Users can view all the definitions for the computer in Oracle Studio for the selected computer. Regular users cannot modify any of the definitions.

3. Add users or groups of users by clicking **Add User** or **Add Group** for the relevant sections.

   The user or group that is added must be recognized as a valid user or group for the computer. Once a name has been added to a section, only the user or group that logs on with that user name has the relevant authorization.

**Setting Up Run-Time User Access to the IBM z/OS Platform**

During run time, client access to Oracle Connect is provided by the user profile. A user profile contains name and password pairs that are used to access a computer, data source or application at run time, when anonymous access is not allowed.

1. From the **Start** menu, select, **Programs, Oracle**, and then select **Studio**. Oracle Studio opens.

2. From the Design perspective, Configuration view, expand the **Machines** folder, then expand the machine where you want to set the user name and password.

3. Expand **Users**.

4. Right-click the **NAV** user profile and select **Open**. The NAV user profile editor is displayed:

   **Figure 2–5  The User Editor**

5. In the User editor, click **Add**. The Add Authenticator screen is displayed:
Figure 2–6  The Add Authenticator screen

7. Enter the name of the z/OS computer defined in Oracle Studio.
8. Enter the name and password used to access the computer and confirm the password.
9. Click OK.

Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2

You must set up the Oracle Connect DB2 CDC adapter on the z/OS platform to handle capture changes to the DB2 data. To work with the DB2 CDC adapter, you must configure the DFSFLGX0 exit and set up the security parameters on the z/OS computer and then configure the change data capture using the Oracle Studio CDC Solution perspective. Oracle Studio must be installed on a Windows or UNIX computer.

Perform the following steps to setup the change data capture and configure the CDC adapter:

■ Configuring the DB2 Tables for CDC
■ Setting up Security for the OracleAS CDC Adapter for DB2
■ Configuring the ATTSRVR Started Task
■ Setting up a Change Data Capture in Oracle Studio
■ Deploying a Change Data Capture

Configuring the DB2 Tables for CDC

In this task you define the data capture changes attribute on the tables where the changes are captured using the DB2 agent. You can use the following DB2 command:

```
ALTER TABLE <TABLE_NAME> DATA CAPTURE CHANGES;
```

Setting up Security for the OracleAS CDC Adapter for DB2

To work with Oracle Connect and DB2, the following requirements must be met:
All the libraries in the ATTSRVR STEPLIB must be APF-Authorized.

Provide the following grants to the owner of the ATTSRVR started task.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRANT TRACE</td>
<td>This command grants the -start trace() privilege.</td>
</tr>
<tr>
<td>GRANT MONITOR2</td>
<td>This command grants the READA and READS IFI requests privilege.</td>
</tr>
</tbody>
</table>

The owner of the ATTSRVR started task must have privileges in DB2 to run offline.

For details about setting DB2 security to use DB2 with Oracle Connect, see Oracle Application Server CDC Adapters Installation Guide.

**Configuring the ATTSRVR Started Task**

In the ATTSRVR started task STEPLIB, check that there is a DD card that defines the used DB2 load library (usually expressed as HLQ.SDSNLOAD).

Configure the ODBCINI file defined in the DSNAOINI DD card of the ATTSRVR started task. In most cases, you can use the default configurations in the ODBCINI file however, you can make changes to the file, if needed.

The following is an example of the ODBCINI file:

```plaintext
; This is a comment line...
; Example COMMON odbcini
COMMON
MVSDEFAULTSSID=DSN1
; Example SUBSYSTEM odbcini for DSN1 subsystem
DSN1
MVSATTACHTYPE=CAF
PLANNAME=DSNACLI
```

The following table describes the configurations that are important for the DB2 CDC agent.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVSDEFAULTSSID</td>
<td>The Sub-System ID (SSID) of the default used DB2 instance.</td>
</tr>
<tr>
<td>MVSATTACHTYPE</td>
<td>Use the CAF only.</td>
</tr>
<tr>
<td></td>
<td>If MVSATTACHTYPE has the RRSAF value, the DB2 agent does not work.</td>
</tr>
<tr>
<td>PLANNAME</td>
<td>The DB2 Calling Level Interface (CLI) plan name. Usually the name of the CLI plan is DSNACLI (as defined in HLQ.SDSNSAMP (DSNTIJCL) job).</td>
</tr>
</tbody>
</table>

You can specify other parameters in this file, as described in the IBM ODBC Guide and Reference.

**Setting up a Change Data Capture in Oracle Studio**

You set up the change data capture in Oracle Studio. Oracle Studio can be installed on Windows XP or Vista operating systems, or on UNIX.

A change data capture is defined in the CDC Solution perspective, which contains a series of links to guide you through the CDC set up process. The CDC solution
perspective guides display the following symbols in front of a link to show you what tasks should be done, and what tasks were completed.

- Triangle: This indicates that there are subtasks associated with this link. When you click the link, the list expands to display the subtasks.
- Asterisk (*): This indicates that you should click that link and perform the tasks and any subtasks presented. If multiple links have an asterisk, you can perform the marked tasks in any order.
- Check mark (✓): This indicates that the tasks for this link and any sublink are complete. You can double click the link to edit the configuration at any time.
- Exclamation mark (!): This indicates a potential validation error.

Perform the following to set up a change data capture:

- Create a CDC Project
- Set up the CDC Server
- Set up the Staging Area Server

### Create a CDC Project

To create a CDC Project

1. From the Start menu, select, Programs, Oracle, and then select Studio.
2. Open the CDC Solution perspective, click the Perspective button on the perspective toolbar and select CDC Solution from the list.
   
   The CDC Solution perspective opens with the Getting Started guide in the left pane of the workbench.
3. Click Create new project.
   
   The Create new project screen opens.
4. In the Project name field, enter a name for your project.
   
   The types of projects available are listed in the left pane.
5. Select Change Data Capture.
   
   From the right pane, select DB2 (Mainframe).
6. Click Finish. The Project Overview guide is displayed in the left pane.

7. Click Design. The Design wizard opens. Use this wizard to enter the basic settings for your project.

Note: The wizard screens are divided into sections. Some sections provide information only and other sections let you to enter information about the project. If you do not see any information or fields for entering information, click the triangle next to the section name to expand the section.
8. In the Client Type you can select Oracle SOA/ODI only. The Use staging area is selected and cannot be changed, you must use a staging area with the OracleAS CDC Adapter for DB2.

9. Click Next.

The Design Wizard’s second screen is displayed. In this step you configure the machines used in your solution. Enter the following information:

- **Server Machine Details**: Information about the machine where Oracle Connect is installed. The selection here is always Server Machine and Mainframe.

- **Staging Area Details**: Information about the machine platform where the staging area is located.

For the server machine Name, select one of the following:

- **CDC Stream Service**: Select this if the Staging Area is on a staging area computer. This is the default selection.

- **Server Machine**: Select this if the staging area is on the same computer where Oracle Connect is installed.

- **Client Machine**: Select this if the Staging area is on the local computer.

In the Platform list, select the operating system for the staging area. This can be Windows, Linux or UNIX. The available options are:

- Microsoft Windows
- HP-UX
- IBM AIX
- Sun Solaris
– Linux (Red Hat)
– Suse (Linux)

**Figure 2–9  Design Wizard (Configure Solution Machines)**

10. Click Finish. The wizard closes.

**Set up the CDC Server**

Click Implement in the Getting Started guide to open the Implementation guide. In the Implementation guide, do the following to set up the CDC server:

- **Set up the Machine**
- **Set up the Data Source**
- **Set up the CDC Service**

**Set up the Machine**

You do the following to define the IP Address/host name and Port for the CDC server machine.

1. Click Machine.

   The machine definition screen is displayed:
2. In the IP address/host name field, do one of the following:
   - Enter the server machine’s numeric IP address.
   - Click the Browse button and select the host machine from the ones presented, then click Finish.

3. Enter the port number.
   The default port number is 2551.

4. If you want to connect with user authentication, enter a user name and password, with confirmation, in the Authentication Information area.

**Note:** The machine you enter must be compatible with the platform designated in the Design Wizard (Configure Solution Machines) screen.
5. Select the **Connect via NAT with a fixed IP address** check box if you are using Network Access Translation and want to always use a fixed IP address for this machine.

6. Click **OK**.

Continue setting up the CDC Server on the z/OS computer.

**Set up the Data Source**

To configure the DB2 data source as part of the DB2 CDC solution, perform the following procedure:

1. In the Solution perspective, click **Implement**.

2. In the Server Configuration section, click **Data Source**. The following is displayed.

![Figure 2–12 The DB2 Data Source Configuration](image)

3. Enter the following information in the Data Source Configuration window:
   - **Location**: Enter the DB2 location name for the connected DB2 instance. The parameter should be specified if the connected DB2 instance is different than the instance defined in the MVSDEFAULTSID parameter of the ODBCINI file.
   - **Database name**: Enter the existing DB2 database name, only if you are creating new tables using AIS.

4. Click **Next**. Enter the following information in the next window:
   - **Default Table Name**: Enter the name of the default table owner for the DB2 database tables.
   - **User Name** and **Password**, if you must provide security credentials to the DB2 database.

5. Click **Finish**.

For information on DB2 data source properties, see the DB2 data source reference chapter.

**Set up the CDC Service**

In this step you define the starting point or event for the change capture and then indicate the name of the change logger. Do the following to set up the CDC service.
1. In the Solution perspective, click **Implement**.

2. In the Server Configuration section, click **CDC Service**. The CDC Service wizard is displayed.

3. In the first screen select *one* of the following to determine the Change Capture starting point:
   - On first access to the CDC (immediately when a staging area is used, otherwise, when a client first requests changes
   - Changes recorded in the journal after a specific date and time.

   When you select this option, click **Set time**, and select the time and date from the dialog box that is displayed.

---

**Note:** The DB2 agent does not support the All changes recorded in the journal mode. If you choose to consume changes from a specific date and time (time stamp), the DB2 agent uses the DSNJU004 module internally to find an archive or active log file that contains the corresponding log records. Then it reads the changes sequentially from the beginning of the file until the log record corresponding to the provided time stamp is found.

4. Click **Next** to define the logger. The following is displayed.

   **Figure 2–13 CDC Logger Definition Window**

   ![CDC Logger Definition Window](image)

5. In the **Bootstrap dataset name** field, enter the name of the data set used to keep track of DB2 logs.

6. Click **Next** to go to the next step where you set the CDC Service Logging. Select the log level to use from the **Logging level** list.
Figure 2–14 Logging Level

Select one of the following from the list:

- None
- API
- Debug
- Info
- Internal Calls

7. Click Finish.

Continue setting up the CDC Server on the z/OS computer.

Set up the Staging Area Server

Click Implement in the Getting Started guide to open the Implementation guide. In the Implementation guide, do the following under the Stream Service Configuration section, to set up the staging area server:

- Set Up the Staging Area Machine
- Set up the Stream Service
- Configure the Access Service Manager

Set Up the Staging Area Machine

To set up the machine for the staging area, do the following.

1. Under the Stream Service Configuration section, click Machine.
2. Use the same configurations used to Set up the Machine for the CDC server.

Continue setting up the staging area Server.

Set up the Stream Service

In this step you set up the stream service. The Stream Service configures the following:

- Staging area
- Filtering of changed columns
■ Auditing

**Note:** Null filtering is currently unsupported. Filtering empty values is supported. Space values are truncated and are handled as empty values.

1. Click **Stream Service**. The Stream Service wizard opens.

**Figure 2–15  Staging Area**

![Stream Service](image)

**Note:** This screen appears only if you selected the inclusion of a staging area in your solution.

2. You can configure the following parameters in this screen:
   - Select **Eliminate uncommitted changes** to eliminate uncommitted changes from your CDC project.
   - Select the **Use secured connection** check box to configure the staging area to have a secured connection to the server. This is available only if you logged into the server using user name and password authentication.
   - Set the event expiration time in hours.
   - Under File Locations, click the **Browse** buttons to select the location of the changed files, and temporary staging files, if necessary.
   
3. Click **Next** to select the tables to include in the filtering process.
4. Click the required tables in the left pane and move them to the right pane using the arrow keys.

   **Note:** You can remove the tables and add new ones to be captured after you add the tables to the right pane. For more information, see Adding and Removing Tables.

5. Click **Next**. From the tables selected above, select the columns that receive changes. Select the check box next to the table to use all columns in the table.

---

**Figure 2–16  Select Tables**

![Select Tables](image1.png)

**Figure 2–17  Column Selection**

![Column Selection](image2.png)
Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2

Any data changes in the columns selected are recorded.

6. Click Next. The Filter selection screen is displayed. the types of changes you want to receive in the tables and which columns to display.

**Figure 2–18** Filter Selection

7. You can do the following in this screen:

Select the actions from which you want to receive change information:
- Update
- Insert
- Delete

**Note:** These items are all selected by default.

- Under the Changed Columns Filter column, select the columns for which you want to receive notification of changes.
8. In the **Content Filter** column of the Filter screen, double-click a table column and then click the **Browse** button to filter content from the selected column.

The Content Filter screen is displayed.

**Figure 2–19  Content Filter**

9. Select a filter type:
   - Select **In** for events to be returned where the relevant column value equals the values you specify (if a column is NULL, it is not captured).
   - Select **Not In** for events to be returned where the column value is *not* in the values you specify (if the column is NULL, it is captured).
   - Select **Between** for when the column value is between the two values you specify (if a column is NULL, it is not captured).

10. Click **Add** in the lower-left corner of the **Content Filter** screen.

    **Note:** If you select multiple conditions, you receive the change information if one condition is true.

11. Depending on your selection, do one of the following:
   - If you selected **In/Not In**, continue with step 12.
   - If you selected **Between**, continue with step 14.

12. Click **Add** in the Add items to the list screen. Enter a value for events to be returned where the column value appears (or does not appear) in that value. To
filter empty values ("") for the **Not In** filter type, leave this field blank in the dialog box that is displayed.

**Figure 2–20  Add Items (In or Not In)**

13. Repeat steps 12 as many times as necessary, and then proceed to step 16.
14. Click **Add** in the Add items to list screen.
   The Add between values screen is displayed.

**Figure 2–21  Add Items (Between)**

15. Enter values for events to be returned where the column value is between the two values you specify.
16. In the **Content Filter** screen, click **Next**.
17. Select the required auditing level when receiving changes. Your options are:
   - **None**: For no changes.
   - **Summary**: For an audit that includes the total number of records delivered, system messages, and error messages.
   - **Headers**: For an audit that includes the total number of records delivered, system and error messages, and the record headers for each captured record.
   - **Detailed**: For an audit that includes the total number of records delivered, system and error messages, the record headers for each captured record, and the content of the records.

18. Click **Finish**.

Continue setting up the staging area Server.

**Configure the Access Service Manager**

In this step you set up a daemon workspace for the CDC adapter. Do the following to configure the access service manager.

1. Click **Access Service Manager**.
   
   The Setup Workspace wizard opens.
2. Select the scenario that best meets your site requirements:
   - **Application Server using connection pooling**
   - **Stand-alone applications that connect and disconnect frequently**
   - **Applications that require long connections, such as reporting programs and bulk extractors**
3. Click Next.

The Application Server with connection pooling scenario screen is used to create a workspace server pool. The parameters available depend on the selection you made in the first screen. The following are the available parameters:

- **If you selected Application Server using connection pooling:**
  - **What is the average number of expected concurrent connections?** Enter the number of expected connections, which cannot be greater than the number of actual available connections.
  - **What is the maximum number of connections you want to open?** Enter the number of connections you want opened.

- **If you selected Stand-alone applications that connect and disconnect frequently,** in addition to the choices listed in the item above, you can also set the following:
  - **What is the minimum number of server instances you want available at any time?** Enter the number of instances, which cannot be greater than the number of actual available instances.
  - **What is the maximum number of server instances you want available at any time?** Enter the number of instances you want to be available.

- **If you selected Stand-alone applications that connect and disconnect frequently:**
  - **How many connections do you want to run concurrently?** Enter the number of concurrent connections to run.
4. Click Next. In the next screen you set time out parameters. These parameters should be changed if the system is slow or overloaded. The parameters are:

- **How long do you want to wait for a new connection to be established?** Enter the amount of time you want to wait for a new connection to be established (in seconds).
- **How long do you want to wait for a response that is usually quick?** Change this parameter if you have a fast connection. Enter the amount of time to wait for a response (in seconds).

5. Click Next. In the next screen you set security parameters. You should consult with the site security manager before changing these parameters.

   Edit the following parameters in this screen:

   - Enter the operating system account (user name) used to start server instances.
   - Select **Allow anonymous users to connect via this workspace**, to allow this option.
   - Enter the permissions for the workspace. You can allow **All users** to access the workspace, or select **Selected users only** to allow only selected users/groups to have exclusive access.
   - Select **Do you want to access server instances via specific ports**, to allow this option. If this option is cleared, the defaults are used.

   If you select this option, indicate the **From port** and **To port** and ensure that you reserve these ports in the TCP/IP system settings.

   ![Figure 2–24 Site Security](image)

6. Click Next.

   The summary screen opens.
7. Click **Save** and then click **Finish**.

When you complete all the Implementation operations, a check mark (✓) is displayed next to every link. Click **Done** to return so you can begin Deploying a Change Data Capture.

Continue setting up the staging area Server.

### Deploying a Change Data Capture

After you complete the design and implementation guides, the following procedures are available.

- **Deployment Procedure**: This section is used to deploy the project.
- **Control**: This section is used to activate or deactivate workspaces after the project is deployed and you are ready to consume changes. In this section, you can deactivate the workspace anytime you want to suspend consumption of changes from the staging area.

Do the following to deploy the CDC solution:

1. Click **Deploy**. The Deployment Procedure and Control sections are displayed in the Deployment view.

2. Click **Deploy** in the **Deployment Procedure** section.

   Oracle Studio processes the naming information. This may take a few minutes. If there are naming collisions, a message is displayed asking if you want Oracle Studio to resolve them.
3. Click **Yes** to resolve any naming collisions.
   The Deployment Guide screen is displayed.

4. If you are ready to deploy, click **Finish**.
   Otherwise, click **Cancel** and you can return to Create a CDC Project, Set up the CDC Server, or Set up the Staging Area Server to make any changes.
   If this project was deployed previously, you are notified that re-deployment overrides the previous instance.
Notes:

- When you redeploy a project where the metadata is changed, the Staging Area (SA) tables should be deleted so that no incorrect information is reported.
- When you redeploy a solution, a new binding is created for the solution. The new binding is created with the default parameters only. Any temporary features that were added are lost.

5. Where applicable, click OK to redeploy.

6. Click the Deployment Summary link.

   The Deployment Summary is displayed. It includes the ODBC connection string and JDBC connection string, as well as specific logger scripts to enable CDC capturing.

   ![Deployment Summary](image)

   **Figure 2–28 Deployment Summary**

7. Cut and paste any information required from the Deployment Summary screen to your environment as necessary.

8. If there is nothing wrong with your deployment results, click Finish.
If you found problems, click Cancel and to return Create a CDC Project, Set up the CDC Server, or Set up the Staging Area Server to modify the solution.

**Note:** If you are redeploying a solution you must follow these directions to ensure that the context and agent_context fields of the SERVICE_CONTEXT table must be saved. Follow these directions to save the fields:

1. In the staging area data source run: select context, agent_context from SERVICE_CONTEXT; and save the returned values.
2. Delete the SERVICE_CONTEXT table physical files.
3. Redeploy the solution.
4. Activate the router to create the SERVICE_CONTEXT table.
5. Disable the router.
6. In the staging area data source run: insert into SERVICE_CONTEXT (context, agent_context) values('XXX', 'YYY'). This inserts the saved values to the SERVICE_CONTEXT table.
7. Activate the solution.

---

**Activating and Deactivating Solution Workspaces**

In the Project guide for your OracleAS CDC adapter solution, click Deploy, then do one of the following to activate or deactivate the workspaces for a solution:

- To activate workspaces, under the Control section click the Activate Workspaces link.
- To deactivate workspaces, click the Deactivate Workspaces link.

During the activation/deactivation process, you may receive messages indicating that the daemon settings on one or more of the machines involved in your solution have changed. Click Yes to proceed.
WebLogic Deployment and Adapter Integration

This chapter includes the following topics:

- Configuring the DB2 CDC Adapter In the Application Server
- Configuring the JCA Configuration Files for the OracleAS CDC Adapter for DB2

Configuring the DB2 CDC Adapter In the Application Server

You can use WebLogic application server to deploy the Oracle Connect CDC adapter. This section describes how to deploy the CDC adapter, set up the necessary connections, and build the CDC Stream Positions table.

This section includes the following topics:

- CDC Stream Positions Table Definition
- Configuring the DB2 CDC Adapter Connection Factory

CDC Stream Positions Table Definition

You should create one CDC Stream Position table for each CDC adapter. The CDC Stream Positions table contains one row for each CDC process (for each captured table) to control its stream position and XA transactions. The following SQL statement is used to define the table in an Oracle database:

```sql
create table CDC_STREAM_POSITIONS(
    TABLE_NAME varchar(127) not null,
    STREAM_POSITION varchar(127),
    PREPARED_STREAM_POSITION varchar(127),
    ROLLBACK_STREAM_POSITION varchar(127),
    NEW_STREAM_POSITION varchar(127),
    XID_FORMAT int,
    XID_BRANCH varchar(128),
    XID_GLOBAL varchar(128));
create unique index CDC_STREAM_POSITIONS_INDEX on CDC_STREAM_POSITIONS (TABLE_NAME);
```

You can use any table name, but you cannot change the column names.

The following table describes the CDC Stream Positions table columns.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE_NAME</td>
<td>The name of the CDC table.</td>
</tr>
<tr>
<td></td>
<td>You should not change the data in this column.</td>
</tr>
</tbody>
</table>
You should configure a WebLogic data source that provides access to the CDC Stream Position table.

### Configuring the DB2 CDC Adapter Connection Factory

Deploy the Oracle Connect RAR into the WebLogic server.

You may create the appropriate connection factories for each CDC adapter.

This table describes the connection factory properties that are relevant for the DB2 CDC Adapter. Do not change the values of any of the other properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>eisName</td>
<td>Required. Sets the name of the adapter to use. The adapter is defined in the Oracle Connect server using Oracle Studio.</td>
</tr>
<tr>
<td>serverName</td>
<td>Required. Enter the name of the server with the TCP/IP address or host name where the Oracle Connect daemon is running. For more information, see Advanced Tuning of the Daemon for details about the daemon.</td>
</tr>
<tr>
<td>workspace</td>
<td>Required. Specifies the name of an Oracle Connect server workspace to use. For more information, see Workspaces for details about workspaces.</td>
</tr>
<tr>
<td>portNumber</td>
<td>Optional. Specifies the TCP/IP port where the Oracle Connect daemon is running on the server. The default port is 2551.</td>
</tr>
</tbody>
</table>
Configuring the JCA Configuration Files for the OracleAS CDC Adapter for DB2

This section includes the following topics:

- Creating JCA Configuration Files
- Configuring the Binding File

Creating JCA Configuration Files

Perform the following steps to create a binding file and a WSDL file for the CDC adapter using Oracle Studio:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. In the Design perspective, Configuration view, expand the Machine folder.
3. Expand the machine with the CDC staging area.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>userName</td>
<td>Optional. Specifies a user who can access the Oracle Connect server. The user is defined in the Oracle Connect daemon configuration. For more information, see daemon Security and workspace Security for details about users allowed to access an Oracle Connect server.</td>
</tr>
<tr>
<td>password</td>
<td>Optional. Specifies a valid password for the user.</td>
</tr>
<tr>
<td>firewallProtocol</td>
<td>Optional. Specifies the firewall protocol used: either none or fixedNat (the Nat protocol using a fixed address for the daemon). The default is set to none.</td>
</tr>
<tr>
<td>connectTimeout</td>
<td>Optional. Specifies the connection timeout in seconds. The default is 0, indicating that there is no connection timeout.</td>
</tr>
<tr>
<td>encryptionProtocol</td>
<td>Optional. Specifies the name of encryption protocol to use. The default is set to RC4. If the value of the property is not defined, the RC4 protocol is used.</td>
</tr>
<tr>
<td>encryptionKeyName</td>
<td>Optional. Specifies the name of the symmetric encryption key to use.</td>
</tr>
<tr>
<td>encryptionKeyValue</td>
<td>Optional. Specifies the value of the symmetric encryption key to use.</td>
</tr>
<tr>
<td>retryInterval</td>
<td>Optional. The amount of time the resource adapter waits if no events are found in the server before issuing the next request. The default value is 15 seconds.</td>
</tr>
<tr>
<td>jdbcDataSource</td>
<td>Required. The JNDI location of the JDBC DataSource that provides access to cdcStreamPositionsTable.</td>
</tr>
<tr>
<td>cdcStreamPositionsTable</td>
<td>The SQL table that is used to control the CDC inbound process. For more information, see CDC Stream Positions Table Definition for more information.</td>
</tr>
</tbody>
</table>
4. Expand the **Bindings**. The binding configurations available on this computer are listed.

5. Expand the binding for the CDC staging area that you created. The name of the binding is the Project name for the CDC solution that you created in Oracle Studio with the suffix _SA.

6. Expand the **Adapters** folder, right-click the adapter, and select **Generate JCA Configuration Files**.

   The JCA Configuration Files dialog box opens.

7. In the **Save to** field, type the path to the location where you want to save the created files, or select **Browse** to browse to a location.

8. In the **JNDI Location** change the default value for the WebLogic Connection Factory JNDI location, if necessary.

9. Click **OK** to create the files in the specified location.

   The View Generated Content dialog box opens. It provides links to the binding file (*.jca) and the WSDL file that are created. If you want to view or edit the files, click the appropriate link in the dialog box.

   Use these files when working with the BPEL Process Manager or Mediator.

### Configuring the Binding File

You should not edit the WSDL file generated by Oracle Studio. You may edit the binding file to configure the **AttuCDCActivationSpec** properties. Most of these properties can also be provided using the Connection Factory. For information on how to configure the Connection Factory, see **Configuring the DB2 CDC Adapter Connection Factory**. You may also need to configure the **retryInterval** property in the binding file.

The binding file specifies the name of the adapter’s connection factory as the value of the `adapterInstanceJndi` attribute of the `<jca:address>` element in the `<service>` section. If a connection factory exists on the application server, its properties are taken. Otherwise, the properties specified by the `AttuCDCActivationSpec` are used. If a value is specified by both the connection factory and the `AttuCDCActivationSpec`, the `AttuCDCActivationSpec` property overrides the value in the connection factory. If you want to use the value specified in the connection factory, you must delete the property from the binding file.

The following is an example of a CDC binding file:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<adapter-config adapter='Legacy Adapter' name="calc"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory UIConnectionName="pcbpel" csfKey="pcbpel"
location="/eis/legacy/calc"/>

  <endpoint-interaction operation="add" portType="calcPortType">
    <interaction-spec
      className="com.attunity.adapter.oracle.AttuInteractionSpec">
      <property name='FunctionName' value='add'/>
      <property name='ExecutionTimeout' value='120'/>
    </interaction-spec>
  </endpoint-interaction>

  <endpoint-interaction operation="display" portType="calcPortType">
    <interaction-spec
      className="com.attunity.adapter.oracle.AttuInteractionSpec">
      <property name='FunctionName' value='display'/>
    </interaction-spec>
  </endpoint-interaction>
</adapter-config>
High Availability

The Oracle legacy adapters for OracleAS support High Availability using Oracle AS clusters with active-active and active-passive topologies.

In an active-active topology Oracle Application Server provides an active-active redundant model for all its components with OracleAS Clusters. In an OracleAS Cluster, two or more Oracle Application Server instances are configured to serve the same application workload. These instances can reside on the same computer or on different computers. The active instances may be front-ended by an external load balancer, which can redirect requests to any of the active instances, or by some other application-level configuration, such as address lists, to distribute the requests.

In an active-passive topology Oracle Application Server supports Active-passive model using OracleAS Cold Failover Clusters. In this case two or more application server instances are configured to serve the same application workload, but only one is active at any time.

For High Availability support in CDC Adapters, all the instances must have access to the Oracle database specified by the jdbcDatasource parameter in the binding file.

You must configure the following property in the binding file.

retryInterval: The amount of time the resource adapter waits if no events are found in the server before issuing the next request. The default is set to 15 seconds. This is optional.
Troubleshooting Rollback Loops

When a problem occurs in the CDC resource adapter work thread, the thread waits for a defined period (retryInterval) then retries the last operation. If you want to stop the adapter, you must deactivate the endpoint. The behavior of the resource adapter is different according to the type of rollback loop. In all cases, the resource adapter writes the corresponding error messages to the resource adapter log file and sends the message to the message listener onAlert method.

- If the connection to Oracle Connect is lost, then the resource adapter always waits for a defined period (retryInterval) until the connection is reestablished.

- In all other cases, the resource adapter retries the last operation immediately and if the problem returns, the resource adapter waits for ascending periods of time, beginning with the retryInterval and doubling the wait time for each successive attempt, but not for more than one minute.

If the OracleAS resource manager invokes a rollback before a 2PC prepare operation, then the CDC resource adapter take steps to perform a special operation that assists you in troubleshooting the problem. In this case, the resource adapter separately sends each event from the last event array to the BPEL endpoint, which saves the stream position of the last problematic event in the ROLLBACK_STREAM_POSITION column.

If you notice that a rollback loop occurs in the CDC resource adapter work thread, check the value of the ROLLBACK_STREAM_POSITION is not null. If the value is not null, check whether there was a data problem. You can select the problematic event from the staging area, using the current stream position value. In this way you can analyze the data to determine what the problem is. In addition, you can skip the problematic event. In this case you stop the endpoint and update the NEW_STREAM_POSITION column with the value of the ROLLBACK_STREAM_POSITION.
This chapter provides an example for creating an OracleAS adapter for DB2. To work with an OracleAS adapter for DB2, you must also configure either a BPEL process or a Mediator process. For information on using BPEL or Mediator, see Creating the Project in JDeveloper Examples.

Creating Outbound Interactions with the OracleAS Adapter for DB2

To work with the OracleAS Adapter for DB2 and create outbound interactions, you must create a new DB2 database and use the data to create a CDC Solution.

Requirements

The following is required to perform the tasks described in this chapter:

- Computer with Windows operating system
- Mainframe computer with z/OS operating system
- Oracle Connect v 11.1.1.2 installed on the Mainframe computer
- Oracle Studio v 11.1.1.2 installed on the Windows computer.

Adding the DB2 Hospital Data Source

To create an DB2 data source you must do the following:

- Prepare the System
- Set up Machine Access to Oracle Connect

Prepare the System

In your DB2 database, you should create a database called Hospital. In this database create the following tables and columns as described in Table 4–1.

<table>
<thead>
<tr>
<th>Table 4–1  DB2 Tables and Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
</tr>
<tr>
<td>HOSPITAL</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Set up Machine Access to Oracle Connect

You begin by configuring access to the Windows computer with your DB2 data. You configure this access with Oracle Connect. Open Oracle connect and follow the directions Setting Up the IBM z/OS Platform in Oracle Studio. Enter the following information in the Add machine dialog box:

- **Host name/IP address**: Enter the name or IP address for the Mainframe computer with the Hospital database. This should be installed in the same directory as Oracle Connect.
- **Port**: Enter the port number where the daemon is running. The default port is 2551.
- **Display name**: You do not need to enter any information in this field (By default, the display name is the host name and the port number).
- **User name**: If the computer you are accessing needs an administrator password. If so enter the name of the computer’s administrator.
- **Password**: If necessary, enter the computer administrator’s password.

### Table 4–1 (Cont.) DB2 Tables and Columns

<table>
<thead>
<tr>
<th>Tables</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>PATIENT</td>
<td>HOSPNAME</td>
</tr>
<tr>
<td></td>
<td>WARDNO</td>
</tr>
<tr>
<td></td>
<td>PATNAME</td>
</tr>
<tr>
<td></td>
<td>PATADDRESS</td>
</tr>
<tr>
<td></td>
<td>PAT_PHONE</td>
</tr>
<tr>
<td></td>
<td>BEDIDENT</td>
</tr>
<tr>
<td></td>
<td>DATEADMT</td>
</tr>
<tr>
<td></td>
<td>PREV_STAY_FLAG</td>
</tr>
<tr>
<td></td>
<td>PREV_HOSP</td>
</tr>
<tr>
<td></td>
<td>PREV_DATE</td>
</tr>
<tr>
<td></td>
<td>PREV_REASON</td>
</tr>
<tr>
<td>SYMPTOM</td>
<td>HOSPNAME</td>
</tr>
<tr>
<td></td>
<td>WARDNO</td>
</tr>
<tr>
<td></td>
<td>BEDIDENT</td>
</tr>
<tr>
<td></td>
<td>DIAGNOSE</td>
</tr>
<tr>
<td></td>
<td>SYMPDATE</td>
</tr>
<tr>
<td></td>
<td>PREV_TREAT_FLAG</td>
</tr>
<tr>
<td></td>
<td>TREAT_DESC</td>
</tr>
<tr>
<td></td>
<td>SYMP_DOCTOR</td>
</tr>
<tr>
<td></td>
<td>SYMP_DOCT_PHONE</td>
</tr>
<tr>
<td>WARD</td>
<td>HOSPNAME</td>
</tr>
<tr>
<td></td>
<td>WARDNO</td>
</tr>
<tr>
<td></td>
<td>TOT_ROOMS</td>
</tr>
<tr>
<td></td>
<td>TOT_BEDS</td>
</tr>
<tr>
<td></td>
<td>BEDAVAIL</td>
</tr>
<tr>
<td></td>
<td>WARDTYPE</td>
</tr>
</tbody>
</table>

(Please note: This table continues on the next page.)
Connect via NAT with fixed IP address: Select this if the machine uses the NAT (Network Address Translation) firewall protocol, with a fixed configuration, mapping each external IP to one internal IP, regardless of the port specified.

Figure 4–1 shows the Add Machine dialog box with the correct information:

Create a Change Data Capture

You now create the change data capture for the tables you created. When you are finished you can create WSDL and binding (jca) files. See Create the JCA Configuration Files for more information.

These files contain the information needed to create the BPEL and Mediator projects in JDeveloper. For more information on using JDeveloper for the BPEL Process Manger and Mediator, see Creating the Project in JDeveloper Examples.

Before you begin to set up your change data capture ensure that the DB2 is set up to handle a change data capture. For information on how this is done see these topics.

- Configuring the DB2 Tables for CDC
- Setting up Security for the OracleAS CDC Adapter for DB2
- Configuring the ATTSRVR Started Task

The following sections describe how to create a change data capture in Oracle Studio.

- Setting up a Change Data Capture
- Deploy the Solution

Setting up a Change Data Capture

The following procedure describes how to set up your change data capture for this sample. For detailed information on setting up a change data capture for DB2, see Setting up a Change Data Capture in Oracle Studio.

1. Open the Solution perspective in Oracle Studio.
2. In the left pane, click Create new project.
3. Enter the following:
   - Project name: Hospital
   - Type: DB2

   Figure 4–2 shows the correct data in the Create new project screen.

![Figure 4–2 Create New Project](image)

4. Click Design and then click Next and Finish to use the default values in the screens. If you want to set up your staging area on a UNIX computer, then you must change that setting in the second screen. For more information, see Create a CDC Project.

5. Click Implement to go to the Implementation guide.

6. For **Server Configuration**, enter the following information:
   - **Machine**: Enter the information for the Windows computer with Oracle Connect installed.
   - **Data Source**: Enter the following:
     - **Location**: Enter the DB2 location name for the DB2 instance that you are using for this project. The parameter should be specified if the connected DB2 instance is different than the instance defined in the MVSDEFAULTSSID parameter of the ODBCINI file.
     - **Database Name**: Enter the name of the Hospital DB2 database that you created for this project.

   For more information, see **Set up the Data Source**.

   - **CDC Service**: You can use the default settings, which does not include the capture of before-image records.
   - **Logger**: Enter the name of the bootstrap dataset you are using to keep track of the DB2 logs.
   - **Logging Level**: Use the default settings.

7. For **Stream Service Configuration**, enter the following information:
■ **Machine**: Enter the information for the computer you are using as your staging area. This must be a Windows or UNIX machine.

■ **Stream Service**: Enter the following:
- In the Staging Area screen, enter a location for the change files on the computer you specified and the Staging Area computer.
- In the Changed Data Capture Table Selection, select all of the tables.
- In the remaining screens, click Next to use the default values.

■ **Access Service Manager**: Click Next in all of the screens to use the default values.

For more information, see Set up the CDC Service.

You must now Deploy the Solution.

**Deploy the Solution**

To deploy the solution, click Activate Workspaces. Click OK in any screen that opens. If you get a message about a naming collision, you must resolve this first.

For information on deploying a change data capture, see Deploying a Change Data Capture.

**Create the JCA Configuration Files**

You must create both a WSDL and JCA file to set up your connection with JDeveloper. You create the files using Oracle Studio before setting up the BPEL process or Mediator connections.

For information on how to create JCA configuration files, see Create the JCA Configuration Files. To create the files for the Hospital_CDCQueue adapter, do the following:

1. From the Configuration pane in Oracle Studio, expand the binding for the CDC staging area that you created. The name of the binding is the Project name for the CDC solution that you created in Oracle Studio with the suffix _SA.

2. Right-click the Hospital_CDCQueue adapter and select Generate JCA Configuration Files.

3. Save the files to a place you can find when you create the BPEL process or Mediator connection.
Creating the Project in JDeveloper Examples

This chapter contains the following topics:

- Getting Started
- Deploy a BPEL Inbound Process
- Mediator Inbound Process

Getting Started

This topic describes the components necessary to work with the samples and how to prepare your system. This section has the following topics:

- Prerequisites
- Beginning Tasks

Prerequisites

The following are installation and configuration requirements for using the examples.

- Personal computer running Microsoft Windows XP, Windows Vista, Windows Server 2003, or Windows Server 2008 with one GB of RAM.
- The CDC adapter that you are using must be deployed to the Oracle Application Server.
- OracleAS CDC Adapter must be configured for inbound processing. See Configuring the DB2 CDC Adapter In the Application Server.
- Oracle JDeveloper must be installed.
- CDC adapter must be deployed.
- Oracle Studio with the necessary machines loaded. See Securing Access to Oracle Connect.

This chapter provides examples of the configuration steps that are used when using BPEL and Mediator to work with the Oracle CDC adapters. You should be familiar with the following before working with these examples:

- How to create J2CA connections. See Configuring the JCA Configuration Files for the OracleAS CDC Adapter for DB2 for more information.
Beginning Tasks

Before you design the inbound process, you must perform the following:

■ Perform the tasks in Configuring a DB2 CDC Adapter to create the CDC Solution and the adapters necessary to create the JCA configuration files.

■ Configure the connection for the J2CA CDC resource adapter. You do this by creating a connection factory.

For more information, see Configuring the DB2 CDC Adapter Connection Factory.

■ Create the WSDL and binding files for the interaction. You do this in Oracle Studio before you begin to build your endpoint in JDeveloper. Oracle Studio creates the WSDL and binding (jca) files automatically, based on the interaction you create.

See Creating JCA Configuration Files for information on creating these files.

Deploy a BPEL Inbound Process

This section describes how to create a BPEL inbound process that connects to an Adabas CDC adapter. For information on how to configure an Adabas CDC adapter for inbound, see Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2.

To configure a process with the BPEL process manager, you use JDeveloper. This section describes how to create a BPEL inbound process for DB2 adapters using JDeveloper. There are two configuration types necessary to perform this inbound sample.

■ Design-Time Configuration

■ Run-time Configuration

Before you design the inbound process, make sure to integrate and connect your OracleAS CDC adapter for Adabas with the BPEL process manager. See Configuring the DB2 CDC Adapter In the Application Server for more information.

You must also make sure to enter the value for the JdbcDataSource in the jca file. shows where you add this value to the jca file.
Deploy a BPEL Inbound Process

Creating the Project in JDeveloper Examples

Design-Time Configuration

This section describes the design-time steps necessary to deploy a BPEL inbound process:

- Create a BPEL Inbound Process
- Use a Third-Party Adapter for the Inbound Process
- Design the BPEL Inbound Process
- Deploy the Project

Create a BPEL Inbound Process

Perform the following to create a BPEL project for inbound.

1. In JDeveloper, create a new SOA project
2. Right-click the SOA project and select New. The New Gallery is displayed as shown in Figure 5–2.

![Figure 5–2 New Gallery](image)

3. In the Categories list, expand SOA Tier and select Service Components.
4. From the Items list, select **BPEL Process** then click **OK**. The Create BPEL Process screen is displayed as shown in Figure 5–3.

**Figure 5–3  BPEL Process**

5. Enter the following in the BPEL Process screen:
   - Enter a name for the BPEL process, for example **BPELInbound1**.
   - From the Template list, select **Define Service Later**.

6. Click **OK**. The BPEL process is displayed in the visual editor.

Continue with **Use a Third-Party Adapter for the Inbound Process**.

**Use a Third-Party Adapter for the Inbound Process**

You use a third-party adapter as a partner link to create the inbound process. SOA executes the SQL statement defined in the interaction and passes it to a resource adapter. Perform the following to create the third-party adapter.

1. From the **Projects** area, on the left side, double-click the **composite.xml**. At the bottom of the visual editor, click the **Design** tab, if necessary. **Figure 5–4** shows the Composite view in the visual editor.

**Figure 5–4  Composite View**
2. Drag a third-party adapter into the **Exposed Services** lane of the visual editor, as shown in **Figure 5–5**.

**Figure 5–5 Third-Party Adapter**

When you drag the third-party adapter into JDeveloper, the Create Third Party Adapter Service screen is displayed as shown in **Figure 5–6**.

**Figure 5–6 Create Third-Party Adapter Service**

3. Click the button next to the **WSDL URL** field and browse to the WSDL file you are using.

4. In the **Type** list, select **Service**.

5. Click the button next to the **JCA File** field and browse to the binding file you are using.

6. Click **OK** to close. You should now have a composite design as shown in **Figure 5–7**.
Deploy a BPEL Inbound Process

Design the BPEL Inbound Process

Before you begin to design the inbound process procedure, you may need to configure some activation spec properties in the binding (jca) file. For more information on working with the inbound binding file, see Configuring the DB2 CDC Adapter Connection Factory.

Do the following to define the BPEL inbound process.

1. In the Composite, connect the BPEL process to the third-party adapter as shown in Figure 5–8.

2. Double-click the BPEL Process.

3. From the Component Palette, BPEL Process Activities section, drag a Receive and an Assign process into the editor as shown in Figure 5–9.
4. In the Visual Editor, connect the Receive activity to the third-party adapter service (Service1). The Edit Receive dialog box is displayed as shown in Figure 5–10.

![Inbound Process](image)

**Figure 5–9  Inbound Process**

5. Do the following in the Edit Receive dialog box:
   - Create the variable. Click the plus sign to the right of the **Variable** field, then click **OK** in the Create Variable dialog box that is displayed.
   - Make sure that the **Create Instance** check box is selected.

6. Click the (x) in the visual editor as shown in Figure 5–11. The Variables screen is displayed.
Deploy a BPEL Inbound Process

7. In the Variables screen, click the Plus sign to open the Create Variable screen.

8. In the Create Variable screen, select **Message Type**, and then click the icon next to the field to open the Type Chooser.

9. In the Type Chooser, expand the files in the tree until you find the **HOSPITALStream** under the Message Types folder as shown in **Figure 5–12**.

Click **OK**, then click **OK** in the Create Variable screen.

10. Double-click the **Assign_1** process to open the Assign screen.

11. From the Assign screen, click the plus sign at the top as shown in and select **Copy Operation**. The Create Copy Operation screen is displayed as shown in **Figure 5–13**.
12. From the Create Copy Operation screen, for both the From and To sides, expand Variable_1, then HOSPITALstream, then select ns2:HOSPITALstream as shown in Figure 5–13.

13. Click OK to close the Create Copy Operation screen, then click OK to close the Assign screen and accept the information.

14. Deploy the Project.

**Deploy the Project**

Perform the following to deploy the project to the Oracle Application Server:

1. From the Application Navigator, right-click the project and find Deploy and then select the project you are deploying.
2. From the Deploy wizard, select Deploy to Application Server.
3. Continue to click Next till the end of the wizard and then click Finish to deploy.

**Run-time Configuration**

This section describes how to test the project and run it to see that the inbound process is executed successfully. By using the XML utility, you call an event in Oracle Connect that is defined in the file adapter you created. See Triggering the Event using the XML Utility. Because this event is called from Oracle Connect it is an inbound event. Events that originate with SOA services in the OracleAS, it is an outbound event.

**Triggering the Event using the XML Utility**

Use the XML utility to create an SQL statement to trigger the event you are requesting. Do the following to trigger the event.

1. From the Start menu, select All Programs, then Oracle, then Server Utilities, and then XML Utility. The XML Utility is displayed.
2. Enter the following information in the Connect section of the XML utility:
   - **Server**: The name of the machine where the OracleAS Adapter for DB2 is located.
   - **Workspace**: The name of the workspace where the OracleAS Adapter for DB2 is located. A list of available workspaces is shown in the list for this field. The workspaces are defined in Oracle Studio.
   - **Adapter**: The name of the OracleAS Adapter for DB2. A list of available adapters is shown in the list for this field. The adapters are defined in Oracle Studio.

3. Click **Connect** to connect to the selected machine.

4. In the **Interaction** field, enter the name of the interaction to call. This is defined in Oracle Studio.

5. In the **Input** window, enter a statement to call the interaction. You can see an example in the above figure.

6. Click **Execute**. If successful, you get a return similar to the following example:

   ```xml
   <?xml version="1.0"?>
   <bigOrderEvent ORDER_ID="1" ORDER_PRICE="1" ORDERED_BY="1" xmlns="noNamespace://ordersQueue"/>
   ```

---

**Mediator Inbound Process**

This section describes how to create a Mediator inbound process that connects to a OracleAS CDC Adapter for Adabas.

You use JDeveloper to configure a process with Mediator. This section describes how to create a Mediator inbound process for CDC resource adapters using JDeveloper. The following topics describe how to create a Mediator inbound process.
Beginning Tasks

Before you design the inbound process, you must perform the following:

- Integrate the OracleAS CDC Adapter for DB2. See Configuring the DB2 CDC Adapter In the Application Server for more information.

- Configure the connection for the J2CA CDC resource adapter. You do this by creating a connection factory. See Configuring the JCA Configuration Files for the OracleAS CDC Adapter for DB2 for information on how to do this.

- Create the WSDL and binding file for the interaction. You do this in Oracle Studio before you begin to build your endpoint in JDeveloper. Oracle Studio creates the WSDL and binding (jca) files automatically based on the interaction you create. See Creating JCA Configuration Files for information on creating these files.

Create a Mediator Inbound Process

Perform the following to create an inbound Mediator project.

1. In JDeveloper, create a new SOA project
2. Right-click the SOA project and select New. The New Gallery is displayed as shown in Figure 5–15.
3. In the Categories list, expand SOA Tier and select Service Components.

4. From the Items list, select Mediator then click OK. The Create Mediator screen is displayed as shown in Figure 5–16.

5. Enter the following in the BPEL Process screen:
   - Enter a name for the Mediator process, for example MedInbound1.
   - From the Template list, select Define Service Later.

6. Click OK. The Mediator process is displayed in the visual editor. Continue with Use a Third-Party Adapter for the Inbound Process.
Use a Third-Party Adapter for the Inbound Process

You use a third-party adapter as a partner link to create the inbound Mediator. SOA executes the SQL statement defined in the interaction and passes it to a resource adapter. Perform the following to create the third-party adapter.

1. From the Projects area, on the left side, double-click the composite.xml. At the bottom of the visual editor, click the Design tab, if necessary. Figure 5–17 shows the Composite view in the visual editor.

2. Drag a third-party adapter into the Exposed Services lane of the visual editor, as shown in Figure 5–18.

When you drag the third-party adapter into JDeveloper, the Create Third Party Adapter Service screen is displayed as shown in Figure 5–19.
3. Click the button next to the WSDL URL field and browse to the WSDL file you are using.

4. In the Type list, select Service.

5. Click the button next to the JCA File field and browse to the binding file you are using.

6. Click OK to close. You should now have a composite design as shown in Figure 5–20.

Continue with Create a File Adapter.

Create a File Adapter

Perform the following to create a file adapter.

1. In the Composite, connect the Mediator to the third-party adapter.
2. From the component pallet, SOA section, on the right, drag a file adapter into the External References lane of the visual editor. The Adapter Configuration wizard is displayed.

3. In the first screen of the Adapter Configuration wizard, enter a name for the adapter service, and click Next.

4. In the Adapter Interface screen, select Import an existing WSDL. Select the same WSDL file as you selected in Use a Third-Party Adapter for the Inbound Process. Figure 5–21 shows the Adapter Interface screen.

Figure 5–21 File Adapter

5. In the Operation screen, select Write File as the Operation Type.

6. In the File Configuration screen, enter a path to the directory where the files are written, and also enter the file naming convention as shown in Figure 5–22.
7. Click **Next** to use the default settings for each of the remaining screens in the wizard. When you get to the end click **Finish**.

8. Connect Mediator to the file adapter.

9. In the Choose Operations screen, select the Hospital for both operations. Click **OK**. Continue with **Design an Inbound Mediator**.

**Design an Inbound Mediator**

In this step, you design the BPEL process. Perform the following to design the inbound Mediator.

1. In the Composite, double-click the Mediator.

2. Expand the Filter Expression as shown in **Figure 5–23**.

**Figure 5–23  Filter Expression**

3. Continue with **Create Mapping Rules and Map the Files**.
Create Mapping Rules and Map the Files

You must perform the following steps to create mapping files and map the files.

1. In the Filter Expression screen, click the icon next to the **Transform using** field as shown in Figure 5–23. The Request Transformation Map screen is displayed as shown in Figure 5–24.

   ![Figure 5–24 Request Transformation Map](image)

2. In the Request Transformation Map, select **Create New Mapper** file. Use the default value.

3. Click **OK**.

4. In the visual editor, click the HOSPITALStream to HOSPITALStream tab. This displays the mapper.

5. Connect the `legacyReq:HOSPITALStream` elements. This is shown in red in Figure 5–25. Then expand the nodes to see the connections.

   Note: The red line does not appear in JDeveloper.

   ![Figure 5–25 Mapper Screen](image)

6. Save and then **Deploy the Project**.

Deploy the Project

Perform the following to deploy the project to the Oracle Application Server:

1. From the Application Navigator, right-click the project and find **Deploy** and then select the project you are deploying.

2. From the Deploy wizard, select **Deploy to Application Server**.

3. Continue to click **Next** till the end of the wizard and then click **Finish** to deploy.
Run-time tasks are executed after you install and configure the OracleAS CDC Adapter for DB2. This chapter describes how to perform basic maintenance tasks, and make changes to CDC solutions. These tasks include:

- Adding and Removing Tables
- Handling Metadata Changes
- Staging Area Maintenance
- Monitoring the Change Data Capture
- Daemon Life-Cycle Tasks
- Resolving Communication Errors
- Resolving Specific Errors

## Adding and Removing Tables

After you deploy the OracleAS CDC adapter solution and start working with it, you may want to change the tables that are being monitored. You do this in the CDC Solution perspective of Oracle Studio. The following describes the main steps that you must follow to make changes to the table. During this operation you must go back and change some configurations that you made when you set up the OracleAS CDC adapter solution during design time. Do the following to change tables:

1. From the Start menu, select Start, Programs, Oracle, and then select Studio.
2. From the Window menu, click the Open Perspective button and select CDC Solution.
3. In the Getting Started pane, under Recent projects, click the OracleAS CDC solution that you are changing.
   
   If you do not see the CDC solution you want to edit in the list under Recent projects, click Open an existing project and select your solution from the list in the dialog box that is displayed.
4. Click Implement.
5. Under the Stream Service Configuration section, click Stream Service.
6. In the first screen of the Stream Service wizard, you can change the location of the change files or click Next to make changes to the tables to be captured.
7. Select one or more tables from either pane.
Handling Metadata Changes

When you make changes to the metadata, you must ensure that the solution recognizes the changes and works with them. This section provides you with a procedure to handle the metadata when working with an OracleAS CDC adapter for DB2 solution if changes are made after deploying the solution. You should perform these steps at a time when there is little or no activity in the system. If you want to receive new events with a new structure, consume the changes for the table you are updating before carrying out any the steps in this process. Do the following to handle any changes to the metadata:

1. Deactivate the Solution workspaces using the CDC Solution perspective in Oracle Studio.
2. Update the metadata on the back-end database for the table you are working with.
3. Update the metadata in the Staging Area by doing one of the following:
   - If you made manual changes to the CDC solution after deployment, or if you do not want to redeploy the solution, then on the Router's (Staging Area) computer, do the following:
     a. Run Oracle Studio, and open the Design perspective.
     b. Edit the Metadata for the Router’s Data source.
     c. Expand the table list and edit the metadata for the table.
        If you are adding a new column, ensure to add it to the end of the COLUMN list. This operation can also be done using the Source view. Ensure that you select the correct data type.
        If you are modifying a data type, ensure that you select the corresponding data type when making the modification.
     - For cases where you can redeploy the solution:

8. Click Next, in the screen displayed, clear check boxes from the columns you no longer want to capture. Select check boxes from columns that are not currently included in the change data capture to include them.
9. Click Next, in the screen displayed, clear and select the check boxes to change the filters used in the change data capture.
10. Click Next, in the final screen you can make changes to the auditing configuration.
11. Click Finish to close the wizard and save the changes.
12. You must re-deploy the solution for the changes to be recognized at run time. For more information, see Deploying a Change Data Capture.

For more information on configuring the Stream Service, see Set up the Stream Service.

Note: When adding a new table, be sure to create a corresponding endpoint for capturing changes. For more information, see Creating JCA Configuration Files.
a. Run Oracle Studio, and open the Solution perspective.

b. Open the CDC solution project.

c. Click Implement and then click Stream Service.

d. Run the wizard.

e. Redeploy the solution, but do not activate it.

For more information, see Deploying a Change Data Capture.

4. Delete the physical files that represent the modified tables from the Staging Area. Ensure not to delete the SERVICE_CONTEXT and CONTROL_TABLE files.

5. Reactivate the solution using Oracle Studio. For more information, see Activating and Deactivating Solution Workspaces.

Staging Area Maintenance

The Staging Area files for the OracleAS CDC adapter solution are DISAM files that store changes until the client application consumes them. When you delete old changed records they are actually only marked as deleted. New changes continue to be written to the deleted records.

In busy production sites the DISAM files can get very large, which can affect the performance of the system.

To ensure that system performance is not degraded, you should defragment the Staging Area files to better maintain the Staging Area repository. Use the following procedures and suggestions when you perform the defragmentation of the staging area:

- Defragment the DISAM files at least once a week. The frequency can vary depending on the amount of changes in the staging area.
- In addition to defragmenting the staging area files, you should also check for corrupt DISAM files.
- You should perform the maintenance processes when the lowest possible activity in your system occurs.

To defragment and maintain the staging area, use the DCHECK and DPACK DISAM utilities. The DCHECK utility is used to check for corruption of DISAM files and rebuild the bad indexes and the DPACK utility defragments the DISAM files and rebuilds them without the deleted records. This reduces the size of the files and ensures that the DISAM files contain only active and relevant records.

Do the following to perform the maintenance activities:

1. Deactivate the OracleAS CDC Solution with the staging area you working with. For more information, see Activating and Deactivating Solution Workspaces.

2. Run the DCHECK utility on each file to ensure it is not corrupted. For example, at the command prompt enter the following to check the DIASM file that represents the table called employees.

   dcheck employees
   If the file is corrupt you can use the -b switch to rebuild all indexes.

3. Defragment each file using the DPACK utility. For example, at the command prompt enter the following to defragment the DIASM files that represent the tables called employees and salaries.

   dpack employees
4. Reactivate all solutions. For more information, see Activating and Deactivating Solution Workspaces.

Monitoring the Change Data Capture

After you deploy the OracleAS Change Data Capture, you can monitor its progress. Monitoring provides you with information about the OracleAS CDC adapter’s status, troubleshooting and tuning. This section contains the following topics that explain monitoring in a CDC.

- Service Context Table
- Monitoring the Status

Service Context Table

A control table is maintained by the event router that reports its current state and other important statistics. It can be accessed with any tool that supports SQL access.

The control table is called SERVICE_CONTEXT. This table has a single row with the following columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>context</td>
<td>string (32)</td>
<td>The context value of the last change record in the most recently committed transaction. Use value to synchronize retrieval of transactions among different tables.</td>
</tr>
<tr>
<td>agent_context</td>
<td>string (64)</td>
<td>This is the agent context that the staging area would return to if it were to restart for whatever reason. The agent context value is calculated as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If there are pending uncommitted transactions, the agent_context value is the agent context of the first event of the oldest uncommitted transaction.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If there are no pending uncommitted transactions the agent context of the last event of the most recently committed transaction, prefixed with 'next' and indicates that on recovery, the next event after that is to be processed.</td>
</tr>
<tr>
<td>start_time</td>
<td>timestamp</td>
<td>The time when the staging area started.</td>
</tr>
<tr>
<td>status</td>
<td>string (16)</td>
<td>Staging area status. For more information, see Monitoring the Status.</td>
</tr>
<tr>
<td>sub_status</td>
<td>string (64)</td>
<td>A second level status. For more information, see Monitoring the Status.</td>
</tr>
</tbody>
</table>
The CONTROL table is also used by the event router to persist its state for purpose of recovery. This table must not be modified by the users.

### Monitoring the Status

The following table describes the status for the CDC adapters when they are running. The status is defined as a state in the SERVICE_CONTEXT table. The table describes the different statuses available for a CDC adapter.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status_message</td>
<td>string (80)</td>
<td>Message that is returned that describes the staging area status.</td>
</tr>
<tr>
<td>status_time</td>
<td>timestamp</td>
<td>The time that the status is updated.</td>
</tr>
<tr>
<td>completed_transactions</td>
<td>uint4</td>
<td>Number of transactions processed.</td>
</tr>
<tr>
<td>active_transactions</td>
<td>uint4</td>
<td>Number of transactions in progress (in memory, not yet committed or rolled back).</td>
</tr>
<tr>
<td>timedout_transactions</td>
<td>uint4</td>
<td>Number of transactions that have timed out (were in memory for too long, declared to have timed out and written to a file).</td>
</tr>
<tr>
<td>rollback_transactions</td>
<td>uint4</td>
<td>Number of rolled back transactions.</td>
</tr>
<tr>
<td>processed_change_events</td>
<td>uint4</td>
<td>Number of change events written out.</td>
</tr>
<tr>
<td>deleted_change_events</td>
<td>uint4</td>
<td>Number of change events deleted from change table.</td>
</tr>
<tr>
<td>bytes_written</td>
<td>uint4</td>
<td>Accumulated size in bytes of change records written.</td>
</tr>
<tr>
<td>opened_files</td>
<td>uint4</td>
<td>Current number of physically opened files by the staging area.</td>
</tr>
<tr>
<td>opened_files_virtual</td>
<td>uint4</td>
<td>Current number of logically opened files by the staging area.</td>
</tr>
<tr>
<td>memory_usage</td>
<td>uint4</td>
<td>Amount of memory currently allocated for staging.</td>
</tr>
<tr>
<td>node_id</td>
<td>uint4</td>
<td>A two-digit identifier with the same value as the nodeID config property when in multi-router mode. In regular mode, this column has no value.</td>
</tr>
<tr>
<td>last_transaction_timestamp</td>
<td>string (26)</td>
<td>The time of the last transaction.</td>
</tr>
<tr>
<td>Version</td>
<td>uint4</td>
<td>The version number for the router.</td>
</tr>
<tr>
<td>errors</td>
<td>uint4</td>
<td>Total number of errors reported.</td>
</tr>
<tr>
<td>Reduced_transactions</td>
<td>uint4</td>
<td>The number of transactions reduced to disc.</td>
</tr>
<tr>
<td>compensation_records</td>
<td>uint4</td>
<td>The number of compensation records captured.</td>
</tr>
</tbody>
</table>

The CONTROL table is also used by the event router to persist its state for purpose of recovery. This table must not be modified by the users.
<table>
<thead>
<tr>
<th>State</th>
<th>Sub State</th>
<th>State Details</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Processing</td>
<td>■ Reads the change events</td>
<td>The router is connected to the CDC adapter and is processing or waiting for the change events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Writes the change events</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Reduces the timed-out transaction to disc</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Deletes any expired change events</td>
<td></td>
</tr>
<tr>
<td>Idle</td>
<td></td>
<td>Waits for new change events</td>
<td>The router’s adapter reaches the end of its journal and does not have any new change events.</td>
</tr>
<tr>
<td>error</td>
<td>router.discWriteEerror</td>
<td>Detailed error text</td>
<td>This indicates that the change router operation involving writing to disk failed. The most common reason is not enough disk space. Other reasons such as permissions, wrong path, or locking can also cause this.</td>
</tr>
</tbody>
</table>
This error type occurs in adapters and routers.

The following are the errors that are returned for this error type:

- `xmlError`
- `requestError`
- `noActiveConnection`
- `resourceLimit`
- `noSuchResource`
- `authenticationError`
- `noSuchInteraction`
- `noSuchConnection`
- `notImplemented`
- `autogenRejected`
- `resourceNotAvailable`
- `authorizationError`
- `configurationError`
- `noSuchStream`
- `temporarilyUnavailable`
- `dataError`
- `interventionRequired`

**Disconnected**

<table>
<thead>
<tr>
<th>State</th>
<th>Sub State</th>
<th>State Details</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disconnected</td>
<td></td>
<td>Detailed error text</td>
<td>This indicates that the change router operation with the CDC adapter failed and cannot be restored.</td>
</tr>
</tbody>
</table>

**Paused**

<table>
<thead>
<tr>
<th>State</th>
<th>Sub State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paused</td>
<td>N/A</td>
<td>Operator manually paused the change router using the <code>sqlrtr_pause</code> control reset.</td>
</tr>
</tbody>
</table>

**Down**

<table>
<thead>
<tr>
<th>State</th>
<th>Sub State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Down</td>
<td>N/A</td>
<td>Down message (orderly shutdown or abort message) This indicates that the change router is not running.</td>
</tr>
</tbody>
</table>
Daemon Life-Cycle Tasks

Most of the daemon run-time tasks between Oracle Application Server and OracleAS CDC Adapter for DB2 are carried out using Oracle Studio, which is used to monitor the daemon and server activity and control what happens to the daemon and server processes.

See Also: Appendix B, "Advanced Tuning of the Daemon" for details about the configuration settings.

This section contains the following topics:

- Starting the Daemon
- Shutting Down the Daemon
- Monitoring the Daemon During Run time
- Daemon Logs

Starting the Daemon

The daemon is started when OracleAS Adapter for DB2 is installed. In case you have shut down the daemon, as described in Shutting Down the Daemon, you can restart the daemon as described in the following task.

Task: Starting the Daemon

Activate `INSTROOT.USERLIB(ATTDAEMN)` as a started task to invoke the daemon. For example, in the SDSF screen, enter the following command:

```
'/s ATTDAEMN'
```

Where `INSTROOT` is the high-level qualifier where Oracle Connect is installed.

Shutting Down the Daemon

To shut down the daemon use Oracle Studio, as follows:

1. From the Start menu, select Start, Programs, Oracle, and then select Studio.
2. Expand the Machines folder.
3. Right-click the computer defined in Setting Up the IBM z/OS Platform in Oracle Studio and select Open Runtime Perspective.
4. In the Runtime Explorer, right-click the daemon you want to shut down and select Shutdown Daemon.

Monitoring the Daemon During Run time

Use the Runtime Manager perspective of Oracle Studio to monitor the daemon during run time.

Perform the following steps:

1. From the Start menu, select Start, Programs, Oracle, and then select Studio.
2. Right-click the computer defined in Setting Up the IBM z/OS Platform in Oracle Studio, and select Open Runtime Perspective.

You can manage the daemon by expanding the relevant node, daemon, workspace or server process, as described in the following sections.

**Daemon (Computer) Options**

Right-click the daemon to display the options available for it, including the ability to display the daemon log.

The following table lists the available daemon options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Daemon Configuration</td>
<td>Opens the daemon editor, which enables you to reconfigure the daemon.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Appendix B, &quot;Advanced Tuning of the Daemon&quot; for details about the configuration settings.</td>
</tr>
<tr>
<td>Status</td>
<td>Checks the status of the daemon. The information about the daemon includes the name of the daemon configuration used, the active client sessions, and logging information.</td>
</tr>
<tr>
<td>Reload Configuration</td>
<td>Reloads the configuration after any changes. Any servers currently started are not affected by the changed configuration.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Appendix B, &quot;Advanced Tuning of the Daemon&quot; for details about the configuration settings.</td>
</tr>
<tr>
<td>View Log</td>
<td>Displays the daemon log. For more information, see Daemon Logs.</td>
</tr>
<tr>
<td>View Events</td>
<td>Displays the daemon events log.</td>
</tr>
<tr>
<td>Daemon Properties</td>
<td>Displays information about the computer where the daemon is running, such as the physical address and any user name and password needed to access the computer.</td>
</tr>
<tr>
<td>Recycle servers</td>
<td>Closes all unused servers and prepares all active servers to close when the client disconnects. New connection requests are allocated with new servers.</td>
</tr>
<tr>
<td>Kill servers</td>
<td>Immediately closes all active and unused servers.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> It is recommended to use this option with caution, as it may lead to data loss.</td>
</tr>
<tr>
<td>Shutdown Daemon</td>
<td>Shuts down the daemon on the computer.</td>
</tr>
<tr>
<td>Rename</td>
<td>Enables changing the name of the daemon displayed in the Runtime Explorer.</td>
</tr>
<tr>
<td>Remove</td>
<td>Removes the daemon from the Runtime Explorer.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Refreshes the display.</td>
</tr>
</tbody>
</table>

**Workspace Options**

Right-click a workspace to display the options available for the workspace, including the ability to display the workspace log.

The following table lists the available options:
Server Options

Right-click a server to display the options available for the server, including the ability to display the server log.

The options available at the server level are listed in the following table:

### Table 6–5 Server Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Checks the status of the server. The information about the server includes the server mode and the number of active client sessions for the server.</td>
</tr>
<tr>
<td>View Log</td>
<td>Displays the server log. For more information, see Daemon Logs.</td>
</tr>
<tr>
<td>View Events</td>
<td>Displays the server events log.</td>
</tr>
<tr>
<td>Kill server</td>
<td>Ends the server process, regardless of its activity status.</td>
</tr>
<tr>
<td>Note:</td>
<td>Use this option with caution, as it may lead to data loss.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Refreshes the display.</td>
</tr>
</tbody>
</table>

Daemon Logs

Oracle Connect produces many logs that you can use to troubleshoot problems. The daemon manages the following logs:

- **Server Options**
- **Workspace Options**
- **Daemon Logs**
Daemon Life-Cycle Tasks

- **Daemon**
- **Workspace**
- **Server process**

The Runtime Manager perspective of Oracle Studio provides a monitor for these logs as shown in the following figure:

To display the required log, do the following:

1. In Oracle Studio, Runtime Manager perspective, right-click, expand the Daemons folder
   - If you want to view the workspace log, then expand the daemon with the workspace you want to view.
   - If you want to view the server, right-click the workspace with the server you want to view.

2. Right click the daemon, workspace, or server and select **View Log**.

Each log is displayed in a different tab. You can browse the different logs by clicking the tab at the bottom of the screen.

**The Log Monitor**

The logs display daemon, workspace, or server events as they happen. You can view the following types of logs in the monitor:

**Daemon logs**: Display activity between clients and the daemon, including clients logging in and logging out from the daemon.

**Workspace logs**: Display information about the workspace being used by the client.

**Server logs**: Display activity between clients and the server process used by that client to handle the client request.

You can change the logging level. To change the logging level, click **Properties**. The following levels of logging are available in the dialog box:

- **none**: The log displays users that log in and out.
- **error**: The log displays users that log in and out and any errors that are generated.
debug: The log displays users that log in and out, any errors that are generated, and any tracing that was configured. For information on configuring the tracing options, see daemon Logging and workspace General.

You can start and stop the logging display.
- Click Suspend to stop collecting logging information.
- Click Resume to start collecting logging information.

You can remove the information displayed in the log.
To remove the information, click Clear.
If logging is enabled, new information continues to be displayed. The cleared information cannot be viewed again.

Resolving Communication Errors

When Oracle Studio disconnects from the IBM z/OS computer, the computer is displayed in Oracle Studio with an X in a red circle. If this situation occurs, try to access the computer later.

The following table describes the various scenarios that may exist when Oracle Application Server disconnects from the IBM z/OS computer.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Idle (Not Processing a Client Request)</th>
<th>Processing a Client Request</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit Disconnect (client explicitly closes connection or client program terminates)</td>
<td>The server is immediately notified of the disconnect and either becomes available for use by another client or terminates (if it is not reusable).</td>
<td>The server does not know that the client has disconnected and continues processing. When processing completes, the server tries to reply to the client and immediately gets an error that the connection was lost. The server either becomes available for use by another client or terminates (if it is not reusable).</td>
</tr>
<tr>
<td>Abrupt Disconnect (client closed without proper shutdown or client system hanged and communication disconnected)</td>
<td>The server does not know that the client has disconnected and remains in the idle state. After timing out based on whichever comes first of the value for the client idle timeout daemon workspace parameter or the TCP/IP KEEPALIVE parameter, the server is notified of the disconnect and either becomes available for use by another client or terminates (if it is not reusable).</td>
<td>The server does not know that the client has disconnected and continues processing. When processing completes, the server tries to reply to the client. After an interval (typically several minutes, depending on the TCP/IP configuration), during which the TCP/IP subsystem retries sending the message to the client, the server assumes that the client has terminated and notifies the server that the connection has been closed. The server either becomes available for use by another client or terminates (if it is not reusable).</td>
</tr>
</tbody>
</table>

To troubleshoot client/server communication problems, you must be familiar with the following:
- Daemon configuration settings
- Oracle Connect security
- TCP/IP subsystem. Oracle Application Server Adapter for DB2 uses TPC/IP for internal intercomputer communications.
- System details, such as the account name and password of the administrator account, the IP address of the computers involved and whether a portmapper is being used.
Resolving Specific Errors

The following error messages relate to errors received from Oracle Connect.

**C007: Server initialization failed.**
- **Cause:** The daemon failed to start its network service.
- **Action:** Check the processes being run on the system to see whether another daemon or program is using the port specified in the oc4j-ra-xml file for the adapter.
- **Action:** Check the TCP/IP subsystem on the current computer by trying to ping it or run FTP or Telnet to or from it.
- **Action:** Check whether the daemon has privileges to use the TCP/IP services on the current computer with the port specified in the oc4j-ra-xml file for the adapter.

**C008: Setting server event handler failed.**
- **Cause:** Internal error.
- **Action:** Contact Oracle Support Services.

**C009: IRPCD process has been terminated by user request.**
- **Cause:** This message is informational only. The daemon successfully shut down.
- **Action:** No action required.

**C00A: Application %s not found.**
- **Cause:** The requested workspace does not exist.
- **Action:** Check that the workspace defined in the oc4j-ra-xml file is also defined in the daemon configuration on the IBM z/OS platform. Use the Status option in the Runtime Manager perspective.

**C00B: Invalid IRPCD client context.**
- **Cause:** A non-Oracle Connect program is trying to connect to the daemon.
- **Action:** Check the processes and stop the relevant process with a system command.

**C00C: Daemon request requires a server login.**
- **Cause:** A non-Oracle Connect server or program was trying to use a daemon service reserved for Oracle Connect servers.
- **Action:** Check the processes and stop the relevant process with a system command.

**C00D: Daemon request requires a client login.**
- **Cause:** The requested daemon requires a valid client login, which was not supplied.
- **Action:** Reissue the command and specify a user name and password.
- **Action:** Edit the user profile in Oracle Studio to specify a valid user name and password for the IBM z/OS platform.

**See Also:** Setting Up Run-Time User Access to the IBM z/OS Platform.

**C00E: Daemon request requires an administrator login.**
- **Cause:** The requested daemon service requires an administrative login.
Action: Edit the daemon security in Oracle Studio to specify a valid administrator user name and password.

See Also: Daemon Security.

C00F: Anonymous client logins are not allowed.
Cause: The daemon is configured to require a valid user name and password, which were not supplied.
Action: Enable anonymous client access in daemon security in Oracle Studio.

See Also: Daemon Security.

Action: Edit the user profile in Oracle Studio to specify a valid user name and password for the IBM z/OS platform.

See Also: Setting Up Run-Time User Access to the IBM z/OS Platform.

C010: Anonymous server logins are not allowed.
Cause: Internal error.
Action: Contact Oracle Support Services.

C011: Client has already timed out.
Cause: A server process was started on behalf of a client and the client has timed out before the server completed its startup.
Action: Increase the Connect timeout value for the server workspace in the General tab of the workspace configuration.

See Also: Workspace General.

C012: Invalid username/password.
Cause: Invalid user name/password supplied when logging on to the daemon.
Action: See the daemon log file because the user name/password were not accepted.

Action: Edit the user profile in Oracle Studio to specify a valid user name and password for the z/OS platform.

See Also: Setting Up the IBM z/OS Platform in Oracle Studio.

Action: Ensure that the daemon is started from an APF-authorized account that is allowed to check for system user names and passwords.

C014: Client connection limit reached - try later.
Cause: The maximum number of server processes for the workspace has been reached, and none of the active servers could accept the client connection.
Action: Increase the value of the Number of sub-tasks in the Server Mode section of the workspace configuration.

See Also: Workspace Server Mode.

Action: Try running the command later.
C015: Failed to start server process.
   **Cause:** The Oracle Connect daemon failed to start a server process or the started server failed upon starting up.
   **Action:** See the daemon and server logs for the reason the server did not start. For example, you might receive an message with a reason specified in the log file similar to the following: [C015] Failed to start NAVIGATOR server process: No server account name defined for anonymous client; code: -1601: SQL code: 0
   **Action:** If you use impersonation, check the user profile on the client. Also see C069.

   **To set impersonation:** APF authorize all the steplib in the server script on a z/OS computer. For example:
   ```
   setprog... ada622-volume adavol
   CICS.CICS.SDFHEXCI - p390dx
   INSTROOT.load - 111111
   INSTROOT.loadaut - 111111
   ```
   `INSTROOT` is the high level qualifier where Oracle Connect is installed.
   In the **Security** tab of the Navigator workspace, under the daemon node in the Configuration view, select **Use specific workspace account** and clear the **Workspace account** field of all values.

C016: Unexpected server state.
   **Cause:** Internal error.
   **Action:** Contact Oracle Support Services.

C017: Active daemon clients exist. Shutdown canceled.
   **Cause:** One or more clients are still connected to the daemon.
   **Action:** Wait until all the clients log off the daemon and then retry the shutdown operation.

C019: Request is not granted because someone else is locking it.
   **Cause:** A request to lock a resource managed by the daemon was denied because another user has locked the resource.
   **Action:** Wait for the other user to release the resource.

C01A: Lock %s not found.
   **Cause:** A request to free a resource was denied because the caller did not lock that resource (for example, another user shut down the daemon you are working with).
   **Action:** Contact Oracle Support Services.

C01B: Unexpected error in %s.
   **Cause:** Internal error.
   **Action:** Contact Oracle Support Services.

C01C: Cannot update configuration without _APPLICATIONS lock.
   **Cause:** Internal error.
   **Action:** Contact Oracle Support Services.
C01D: Need to lock the application first.
   Cause: Internal error.
   Action: Contact Oracle Support Services.

C01F: Cannot set configuration of a deleted application.
   Cause: Internal error.
   Action: Contact Oracle Support Services.

C020: Failed in looking up host name (gethostname())
   Cause: Cannot connect to the remote computer.
   Action: Check that the name specified for the computer in the oc4j-ra-xml file is correct.
   Action: Check that a domain name server (DNS) is available to look up the host name.
   Action: Check the TCP/IP subsystem on the computer by trying to ping it or run FTP or Telnet to or from it.

C021: Required variable %s not found
   Cause: An environment variable required by the Oracle Connect server was not defined when the server started up.
   Action: Check whether the startup script makes any changes to the environment variables used by Oracle Connect.
   Action: Check whether the system-defined environment size is sufficiently large for Oracle Connect.

C022: Server failed to connect and register with the daemon.
   Cause: An Oracle Connect server started by the daemon was not able to connect or register back with the daemon.
   Action: Try to connect again.
   Action: Increase the Connect timeout value for the server workspace in the General tab of the workspace configuration.
   See Also: Workspace General.

   Action: Check that the startup script for the workspace launches the correct version of Oracle Connect.
   Action: Increase the value of the Set maximum number of server processes parameter for the Clients per server limit in the Server Mode tab of the workspace configuration.
   See Also: Workspace Server Mode.

C023: Call made to unregistered module %d.
   Cause: Internal error.
   Action: Contact Oracle Support Services.

C024: Failed to create a socket.
   Cause: An error occurred within the TCP/IP subsystem.
Action: Check whether you have sufficient system privileges.

Action: Check the TCP/IP subsystem on the computer by trying to ping it or run FTP or Telnet to or from it.

C025: Failed to set socket option %s
Cause: An error occurred within the TCP/IP subsystem.
Action: Check whether you have sufficient system privileges.

Action: Check the TCP/IP subsystem on the computer by trying to ping it or run FTP or Telnet to or from it.

C026: Failed to bind server to port %s
Cause: An Oracle Connect server or daemon was not able to bind to the specified port.
Action: Check whether another program is holding the port that was specified in the oc4j-ra-xml file for the adapter.

Action: Check whether you have sufficient system privileges.

C027: Cannot create TCP service for %s
Cause: An error occurred within the TCP/IP subsystem
Action: Check the TCP/IP subsystem on the computer by trying to ping it or run FTP or Telnet to or from it.

C028: Unable to register (%s, %d, tcp)
Cause: This error may happen when a portmapper is used (host:a) but the portmapper is not available.
Action: Enable the portmapper.

Action: Avoid using the portmapper (by not using :a when starting the daemon).

C029: Failed to create a server thread
Cause: Internal error.
Action: Contact Oracle Support Services.

C02A: Server thread failed to start
Cause: Internal error.
Action: Contact Oracle Support Services.

C02B: Stopping the %s server - no client
Cause: A server that was started by the Oracle Connect daemon to service a client did not get a client connection request within one minute. The server terminates.
Action: In most cases, the client was terminated by a user request, so no specific action is required.

Action: If no client can connect to the server, it may be that the server has multiple network cards and the Oracle Connect daemon is not aware of this. In this case, start the daemon with an IP address.

C02C: Unexpected event - a termination signal intercepted
Cause: Internal error.
Action: Contact Oracle Support Services.
C02D: Modified transport, context unknown/lost  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C02F: Corrupted arguments passed to procedure  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C030: Unable to free arguments for %s() of %s  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C031: Cannot register a non-module RPC %s  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C032: An IRPCD program is required  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C033: An IRPCD super-server is required for module events  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C034: An invalid super-server module ID was specified, %d  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C035: Out of memory  
  **Cause:** Not enough memory to service a client request.  
  **Action:** Increase process memory quota or add memory to the system.

C036: Failed to register RPC procedure module %s  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C037: Failed to register an invalid RPC procedure number %x  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C038: Cannot reregister RPC procedure number %x  
  **Cause:** Internal error.  
  **Action:** Contact Oracle Support Services.

C042: Remote call to %s failed; %s  
  **Cause:** Remote call to API failed.  
  **Action:** Check the daemon log file.

  **Action:** If necessary, change the level of detail written to the log file to help resolve the problem.
C043: Failed to connect to host %s

Cause: The remote host is not correctly defined to Oracle Connect or is not working.

Action: Check the remote computer definition in the oc4j-ra-xml file for the adapter.

Action: Check that the daemon is up on the z/OS platform. Use the Status option in the Runtime Manager perspective.

Action: Check the network connection by trying to ping the host computer or run FTP or Telnet to or from it.

C047: %s out of memory

Cause: Not enough memory was available to Oracle Connect to complete a requested operation.

Action: Terminate unnecessary processes running on the server.

Action: Add more memory to the system.

Action: Allow the process to use more memory.

Action: Limit the number of processes the daemon may start. If the demand for servers exceeds the number of available servers, clients get a message telling them the maximum number of servers has been reached and asking them to try again later.

C066: Communication error with the server %s

Cause: Connection to the Oracle Connect daemon or server failed, or an established session with a server has failed.

Action: Check the remote computer definition in the weblogic-ra-xml file.

Action: Check that the daemon is up on the IBM z/OS platform. Use the Status option in the Runtime Manager perspective.

Action: In case of a network problem, check the network connection by trying to ping the host computer or run ftp or telnet to or from it.

C067: Unexpected error occurred in server function %s

Cause: One of the server functions has exited with an exception (such as an abend, or an Invalid Instruction).

Action: Contact Oracle Support Services.

C068: Fail to login daemon

Cause: The daemon is not running on the server computer.

Action: Use the Status in Oracle Studio Runtime Manager perspective to check whether a daemon is running on the server.

Action: Have the system administrator reinstall Oracle Connect on the server.

C069: Fail to get server

See Also: Daemon Logging.
Resolving Specific Errors

**Cause:** The Oracle Connect daemon on the server computer could not start a server process to serve the client. A separate message provides more detail on why the server process could not start.

**Action:** There are many possible causes of this error. If the cause is not clear from the related message, see the Oracle Connect daemon log file on the server.

**Action:** The resolution to this error is highly dependent on the particular cause. The following are some typical causes and resolutions.

**Action:** Some process creation quota was exceeded. Either try again later or increase the quota or the other relevant system resources.

**Action:** The server startup script failed.

**Action:** The user name given is not allowed to use the requested server. Use an authorized user name.

**Action:** A limit on concurrent clients for a server has been reached. Try again later.

**Action:** If you use impersonation, check the user profile on the client. Also see C015.

**C06A: Failed to connect to server**

**Cause:** The server assigned to the client did not accept the client connection. A separate message provides more detail about why the server process did not accept the connection.

**Action:** See the daemon and server log files because the server was not available to accept its assigned client.

**C06B: Disconnecting from server**

**Cause:** A network failure, or a server computer failure or a server program failure caused the connection to end. The currently active transaction is also stopped.

**Action:** Oracle Connect automatically tries to reestablish a connection with a server upon the next SQL command issued against the server. Once the network or computer failure is corrected, the connection to the daemon is reestablished automatically.

**C070: Server failed to send reply to the client**

**Cause:** Server terminated unexpectedly.

**Action:** Unless the client was intentionally stopped (for example, using Control-C), contact Oracle Support Services.

**C071: Connection to server %s was disconnected. Cursors state was lost.**

**Cause:** Either a network failure, a server computer failure or a server program failure caused the connection to end. The currently active transaction is also stopped.

**Action:** Normally, Oracle Connect automatically tries to create a session with the server upon the next attempt to access the server. If the network and server are accessible, the next operation should succeed. Otherwise, the network or server computer should be fixed before connection can be resumed.

**Action:** In case of a server failure not related to callable user code, contact Oracle Support Services.

**C072: Reconnect to server %s**
Resolving Specific Errors

Run-time Tasks and Troubleshooting

6-21

Cause: This is an informational message only. The client has reestablished its connection with the server.
Action: No action is required.

C073: The parameters passed to the admin server are invalid: %s
Cause: Internal error.
Action: Contact Oracle Support Services.

C074: No authorization to perform the requested operation (%s)
Cause: User/account has insufficient privileges.
Action: Grant administrative privileges to the user/account using the Administrator parameter of the Security tabs in the daemon and workspace configurations.

See Also: Damon Security or Workspace Security.

C075: Failed to register daemon in the TCP/IP service table
Cause: Registration of the daemon in the TCP/IP services file has failed.
Action: Check that the account running the daemon has the permissions to update the TCP/IP services file.

E001: Failed in lock/release operation
Cause: A lock or release operation of a global resource has failed. A separate message provides more details. The separate message specifies the cause of this error.
Action: There are various causes for this error, including lack of sufficient privileges or a system resource shortage.

J0006: Operation on already closed connection was requested
Cause: A request using a connection that was closed was attempted.
Action: Reopen the connection and try again.

J0028: Internal Error: Unknown XML tag %s
Cause: Internal error.
Action: Contact Oracle Support Services.

J0030: Internal Error: Method %s needs to be overwritten
Cause: Internal error.
Action: Contact Oracle Support Services.

J0031: Internal Error: Required attribute %s not found in %s verb
Cause: Internal error.
Action: Contact Oracle Support Services.

J0032: Internal Error: %s ACP object was returned instead of %s as expected
Cause: Internal error.
Action: Contact Oracle Support Services.

J0033: Internal Error: Attempt to work with closed socket
Cause: Internal error.
Action: Contact Oracle Support Services.
J0034: Internal Error: corrupted message; %s bytes read instead of %s as expected
   Cause: XML sent from the client to the server has become corrupted.
   Action: Check compression settings for XML transferred from the client to the server. If the setting are OK, retry sending the request from the client to the server.

J0035: Internal Error: Invalid redirection address %s returned by daemon
   Cause: Internal error.
   Action: Contact Oracle Support Services.

J0036: %s: %s
   Cause: One of the following errors was received from the server: 0 - server.internalError, 1 - client.xmlError, 2 - client.requestError, 3 - client.noActiveConnection, 4 - server.resourceLimit, 5 - server.redirect, 6 - client.noSuchResource, 7 - client.authenticationError, 8 - client.noSuchInteraction, 9 - client.noSuchConnection, 10 - server.notImplemented, 11 - server.xaProtocolError, 12 - server.xaUnknownXID, 13 - server.xaDuplicateXID, 14 - server.xaTransactionTooFresh, 17 - server.resourceNotAvailable, 18 - client.authorizationError, 19 - server.configurationError
   Action: Review the server log file to determine the problem.

J0037: Internal Error: No ACP response when %s was expected
   Cause: Internal error.
   Action: Contact Oracle Support Services.

J0039: Internal Error: ACP root is not found in the XML
   Cause: Internal error.
   Action: Contact Oracle Support Services.

J0040: Internal Error: Input record is required for interaction %s execution
   Cause: Internal error.
   Action: Contact Oracle Support Services.

J0048: Invalid metadata type %s is passed to %s function
   Cause: A request for metadata was not fulfilled.
   Action: Check the validity of the request.

J0050: Key of the put method must be of type string
   Cause: In either a GET or PUT operation, the key must be a string.
   Action: Change the key used in the operation to a valid key.

J0059: Value %s is invalid for attribute %s
   Cause: A request for metadata was not fulfilled.
   Action: Check the validity of the request.

J0068: Value must be of type string
   Cause: In a PUT operation, the value must be a string.
   Action: Change the value used in the operation to a valid value.

J0069: Value must be of type MappedRecord
   Cause: In a PUT operation, the value must be a mapped record.
**Action:** Change the value used in the operation to a valid value.

**J0070:** Value must be of type MappedRecord[]
- **Cause:** In a PUT operation, the value must be mapped record array.
- **Action:** Change the value used in the operation to a valid value.

**J0071: Bad key for mapped record, #element or #element[] is required**
- **Cause:** In a PUT operation, the value must be mapped record array.
- **Action:** Change the key used in the record to a valid key.

**J0072: Value must be of type Object[]**
- **Cause:** In a PUT operation, the value must be mapped record array.
- **Action:** Change the value used in the operation to a valid value.

**J0078: In nonpersistent connection and the nonkeep alive encryption is not supported - ignored**
- **Cause:** Encryption is not supported for nonpersistent connections.
- **Action:** There is no action to take. This warning can be ignored.

**J0079: Invalid argument passed to %s - Argument: %s, Value: %s**
- **Cause:** The value pass.
- **Action:** Change the argument used to a number.
Oracle Connect includes several tuning parameters that can improve performance. Specifically, the daemon can be configured to optimize communication between the Windows operating system and a client.

In addition, the binding environment can be tuned to optimize the request handling.

This chapter contains the following topics:

- Configuring the Daemon for High Availability
- Configuring a Binding Environment
- Migration Considerations
- Security Considerations
- Transaction Support

**Configuring the Daemon for High Availability**

The daemon workspace is responsible for allocating server processes to clients. You can configure a workspace to use a pool of server processes so that a server process is always available for a client request.

Use Oracle Studio to maintain daemon and daemon workspace parameters to control the allocation of server processes and their management in a pool.

You can also have several daemon workspace configurations. This lets you create individual workspaces for use with different adapters or CDC adapters.

**Adding a New Daemon Workspace Configuration**

Use Oracle Studio to add a new daemon configuration. You can set up different daemon configurations for different situations.

Carry out the following steps to add a new daemon workspace configuration:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. In the Design Perspective Configuration view, expand the Machines folder and then expand the machine where you want to add the workspace.
3. Expand the Daemons folder. The daemon available on this computer are listed.
4. Right-click IRPCD and select New Workspace. The New Daemon Workspace screen is displayed.
5. Enter a name for the new workspace and then enter a description, if desired.

6. Select whether to use default settings or copy the properties of an existing workspace.

   To copy the properties of an existing workspace, click the Browse button and select the workspace from which you want to copy the properties.

7. Click Next. The Select Scenario screen is displayed.

8. Select Application Server using connection pooling and click Next.

9. Continue through the wizard, entering the required values for the workspace.

10. Click Finish.

The workspace is displayed under the IRPCD daemon node.

Editing the Workspace

You edit a workspace by using the tabs described in the following table:

### Table 7-1  Workspace Properties

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>Specifies general information including the server type, the command procedure used to start the workspace, the binding configuration associated with this workspace (which dictates the data sources and applications that can be accessed), the timeout parameters, and logging information (which dictates the data sources and applications that can be accessed), the timeout parameters, and logging information.</td>
</tr>
<tr>
<td>Server Mode</td>
<td>Contains the workspace server information including features that control the operation of the servers started up by the workspace and allocated to clients.</td>
</tr>
<tr>
<td>Security</td>
<td>Specifies administration privileges, user access, ports available to access the workspace and workspace account specifications.</td>
</tr>
</tbody>
</table>

Use Oracle Studio to access these tabs, as follows:

1. From the Start menu, select Programs, Oracle, and then select Studio.

2. In the Design perspective Configuration view, expand the Machines folder and then expand the machine where you want to edit the workspace.

3. Expand the Daemons folder. The daemon available on this computer are listed.

4. Expand the IRPCD daemon. The daemon workspaces are listed.

5. Right-click the workspace you are editing and select Open.

6. Click the tab that contains the information you want to edit. For full details of the tabs and the fields in these tabs, see Workspaces.

7. After editing the workspace, click Save.

Configuring the Server Mode

The server mode defines how the daemon starts new processes. The daemon supports the following server modes:
- **singleClient**: Each client receives a dedicated server process. The account in which a server process runs is determined either by the client login information or by the specific server workspace.

  This mode enables servers to run under a particular user account and isolates clients from each other (because each receives its own process). However, this server mode incurs a high overhead due to process startup times and may use a lot of server resources (because it requires as many server processes as concurrent clients).

- **multiClient**: Clients share a server process and are processed serially. This mode has low overhead because the server processes are initialized. However, because clients share the same process, they may impact one another, especially if they issue lengthy queries.

  The number of clients that share a process is determined by the Clients per server limit (the maximum number of concurrent clients a server process for the current workspace accepts).

- **reusable**: This is an extension of the single client mode. Once the client processing finishes, the server process does not die and can be used by another client, reducing startup times and application startup overhead.

  This mode does not have the high overhead of single client mode because the servers are initialized. However, this server mode may use a lot of server resources (because it requires as many server processes as concurrent clients).

  The other modes can be set so that the server processes are reusable by setting the number of times a process can be reused with the Reuse limit value (the maximum number of times a particular server process can be reused or how many clients it can serve before it is retired). Reuse of servers enhances performance because it eliminates the need to repeat initializations. However, reuse runs a risk of higher memory leakage over time. The default value for the Reuse limit field is None, indicating that no reuse limit is enforced.

Set the server mode in the **Server Mode** tab of the daemon workspace editor as shown in the following figure:
Configuring the Daemon for High Availability

Figure 7–1  The Server Mode tab

When using any of the server modes you can specify a pool of server processes. These server processes are started when the daemon starts and are maintained in a pool. The server processes are available for use by new client requests from the pool, saving initialization time. Instead of starting a new server process each time one is requested by a client, the client receives a process immediately from the pool of available processes. When the client finishes processing, this server process either dies, or if reusable servers have been specified, it is returned to the pool.

You set up a pool of server processes by specifying the following parameters in the Server Mode tab.

- **Port Range**: Select the range for specific firewall ports through which you access the workspace. Determines the range of ports available for this workspace when starting server processes. Use this option when you want to control the port number, so that Oracle Connect can be accessed through a firewall.

  Enter the port range in the following fields:

  - **From**: Enter the highest numbered port in the range
  - **To**: Enter the lowest numbered port in the range
  - **Use Default Port Range**: Select this to use the port range that is defined in the daemon.

- **Maximum number of server processes**: Enter the maximum number of server processes that can run at the same time.

- **Limit server reuse**: Select this to limit the number of servers that can be reused. If this is selected, the Reuse limit parameter is available.

  If Limit server reuse is selected, in the field next to the check box, enter the maximum number of times a server can be reused. Select the maximum of clients accepted in a server process.
A one-client server can be reused after its (single) client has disconnected. Reuse of servers enhances startup performance because it eliminates the need to repeat initializations.

This parameter is not available if the Limit server reuse parameter is not selected.

This parameter is not available if the server mode value is `singleClient`.

- **Limit Concurrent clients per server**: Select this to limit the number of clients that a server can accept for the current workspace process.
  
  If this is not selected, the number of clients is unlimited.

  If `Limit concurrent clients per server` is selected, in the field next to the check box, enter the maximum number of clients that a server process for the current workspace accepts. The default for this field is `None`, indicating that the number of clients for each server is unlimited. This field is available if the server mode value is `multiClient` or `multiThreaded`.

- **Specify Server Priority**: Set the priority for servers. For example, a workspace for applications with online transaction processing can be assigned a higher priority than a workspace that requires only query processing. The lower the number, the higher the priority. For example, workspaces with a priority of 1 are given a higher priority than workspaces with a priority of 2.

  **Note**: This is unavailable if `Use default server priority` is selected.

- **Use default server priority**: Sets the priority to 0. There is no specific priority for this workspace. Clear this check box to set a priority in the `Specify Server Priority` parameter.

- **Keep when daemon ends**: Select this to stop all servers started by that daemon when a daemon is shutdown, even if they are active. Select this if you want the servers for the workspace to remain active, even after the daemon has been shut down. If selected, it is the responsibility of the system operator or manager to ensure that the servers are eventually stopped. This must be done at the system level.

- **Number of prestarted servers in pool**: Enter the number of server processes that are prestarted for this workspace when the daemon starts. These are available for use by new client processes with minimal initialization time. Instead of starting a new server process each time one is requested by a client, the daemon immediately allocates (to the client) a server from a pool of available servers. When the number of available server processes drops lower than the value specified in the Minimum number of available servers field, the daemon again starts server processes until the specified number of available servers is reached. The default for this parameter is 0, meaning that no servers are prestarted for this workspace.

- **Number of spare servers**: Enter the minimum number of server processes in the prestarted server’s pool before the Oracle Connect daemon resumes creating new server processes (up to the number specified in the Initial number of servers field value, described earlier). If this parameter is set to a value greater than the Initial number of servers field value, the daemon considers the value to be the same as the value specified in the Initial number of servers field. In this case, a new server process is started and added to the pool each time a server process is removed from the pool and allocated to a client). The default for this parameter is 0, which means that new servers are created only when there are no other available servers.

- **Number of sub-tasks**: Enter the number of sub-tasks for a server that are prestarted for this workspace when the daemon starts. In addition to setting up a
pool of server processes as described earlier, you can set additional server processes as sub-tasks by specifying this parameter. If you set 10 servers and 10 prestarted sub-tasks then 100 tasks are started (10 sub-tasks for each process).

### Configuring a Binding Environment

Each binding configuration includes the following information:

- Environment settings, which are used to configure the environment used by any of the adapters or CDC adapters defined in the binding.
- Application adapters on the current computer.

To configure environment settings in Oracle Studio, perform the following steps:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. In the Design perspective Configuration view, expand the Machine folder, then expand the machine where you want to configure the binding.
3. Expand the Bindings folder. The binding available on this computer are listed.
4. Right-click NAV and select Open.
5. In the Environment tab, edit the environment settings as needed. To edit an environment setting, expand the property category and click the value to edit.

The binding Environment tab is shown in the following figure:

**Figure 7–2 The Binding Environment tab**

The binding environment is divided into the following categories:

- Debug
- General
The following table lists the parameters that define debugging and logging operations:

**Table 7-2 Debug Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACX trace</td>
<td>Select this for the input xml sent to the back-end adapter and the output xml returned by the back-end adapter to be written to the log.</td>
</tr>
<tr>
<td>GDB Trace</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
<tr>
<td>General trace</td>
<td>Select this to log general trace information. The default writes only error messages to the log.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: Changing the default setting can degrade performance.</td>
</tr>
<tr>
<td>Query warnings</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
<tr>
<td>Add timestamp to traced</td>
<td>Select this to add a time stamp on each event row in the log.</td>
</tr>
<tr>
<td>events</td>
<td></td>
</tr>
<tr>
<td>Query Processor trace</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
<tr>
<td>Binary XML Log Level</td>
<td>Select the binary XML log level from the list. The following logging levels are available:</td>
</tr>
<tr>
<td></td>
<td>■ None</td>
</tr>
<tr>
<td></td>
<td>■ API</td>
</tr>
<tr>
<td></td>
<td>■ Info</td>
</tr>
<tr>
<td></td>
<td>■ Debug</td>
</tr>
<tr>
<td>Log file</td>
<td>The high-level qualifier of the log file for messages. The following type of message are written to the log:</td>
</tr>
<tr>
<td></td>
<td>■ Error messages</td>
</tr>
<tr>
<td></td>
<td>■ Trace information and information about the query optimization strategy, if General Trace is selected.</td>
</tr>
<tr>
<td>Trace Directory</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
<tr>
<td>Optimizer trace</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
</tbody>
</table>
General

The following table lists the parameters that define general operations where temporary files are written.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NA V_UTIL editor</td>
<td>This parameter is not applicable for use with OracleAS CDC Adapter for DB2.</td>
</tr>
<tr>
<td>Temporary Dir</td>
<td>The directory where temporary files are written, including the temporary files created for use by hash joins and for sorting files. The default is the current high-level qualifier.</td>
</tr>
</tbody>
</table>
| Year 2000 policy   | Determines the way two-digit years are converted into four-digit years. When the year2000Policy parameter is not set, or when it is set to a value outside the range of values defined for the policy, as described in the following paragraphs, a default value of 5 and the Sliding Base Year policy is used. Two policies are provided:

Fixed Base Year: year2000Policy is set to a value greater than, or equal to 1900. In this case, the value of year2000Policy is the first four-digit year after 1900 that can be represented by a two-digit year. For example, if year2000Policy is set to 1905, the years 2000->2004 is represented by 00->04. All other two digits are mapped to 19xx.

This solution is most required if there is live data at the low end (close to the year 1900), which the user wants to keep with the current two-digit format.

The user probably changes the base date only after ensuring that these old dates have been deleted from the data source.

Sliding Base Year: year2000Policy is set to a positive value less than 100. In this case, the value of year2000Policy represents the number of years ahead of the current year that can be represented by a two-digit number. With each passing year the earliest year that can be represented by a two-digit number changes to a year later. |
| Cache buffer size  | Enter the number of bytes to be used for a memory buffer on a client computer, which is used by the Oracle Connect client/server to store read-ahead data. The default is 200000. |

Language

The following table lists the parameters that define globalization support:
Modeling

The **Modeling** parameters are not applicable with OracleAS CDC Adapter for DB2.

**ODBC**

The **ODBC** parameters are not applicable for use with OracleAS CDC Adapter for DB2.

**OLEDB**

The **OLEDB** parameters are not applicable for use with OracleAS CDC Adapter for DB2.

Optimizer

The following parameters enable you to customize the performance:

**Optimizer goal:** The optimization policy. Select one of the following from the list to the right:

- **none** (default): All row optimization is used.
- **first**: First row optimization is performed based on the assumption that the results produced by the query are used as the rows are retrieved. The query optimizer uses a strategy that retrieves the first rows as fast as possible, which might result in a slower overall time to retrieve all the rows.
- **all**: Optimization is performed based on the assumption that the results produced by the query are used after all the rows have been retrieved. The query optimizer uses a strategy that retrieves all the rows as fast as possible, which might result in a slower time to retrieve the first few rows.

Aggregate queries automatically use all row optimization, regardless of the value of this parameter.

All other optimizer parameters are not applicable for use with the OracleAS CDC Adapter for DB2.

---

### Table 7-4 Language Settings Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>Identifies the application language. A default code page is selected based on the value specified for this parameter. For more information, see Appendix E, &quot;Globalization Settings&quot;.</td>
</tr>
<tr>
<td>Code Page</td>
<td>For use with globalization support to identify the code page for the workspace. For more information, see Appendix E, &quot;Globalization Settings&quot;.</td>
</tr>
</tbody>
</table>
| NLS String | Specifies the code page used by a field whose data type is defined as **nlsString**. Use this for a field whose code page is other than that of the computer code page. This parameter includes the following values:
  - The name of the code page.
  - Whether the character set reads from right to left (as in middle eastern character sets).
  The default is **false**. |
Parallel Processing

The following list shows the parallel processing properties. The parallel processing properties control how parallel processes are handled in the binding.

Table 7–5  Parallel Processing Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable threads</td>
<td>Select this to disable multi-threading. If this is selected, the following properties are disabled:</td>
</tr>
<tr>
<td></td>
<td>■ Disable threaded read ahead (QP): Select this to disable read-ahead functionality.</td>
</tr>
<tr>
<td></td>
<td>■ Disable query read ahead (QP): Select this to disable read-ahead functionality for components using Query Processor services.</td>
</tr>
<tr>
<td></td>
<td>■ ODBC async execution: Select this property to enable ODBC asynchronous execution.</td>
</tr>
<tr>
<td></td>
<td>■ Disable QP parallel execution: Select this to disable parallel processing for query execution. This option is available only if both Disable threaded read ahead (QP) and Disable query ready ahead (QP) are not selected.</td>
</tr>
<tr>
<td></td>
<td>■ Hash parallelism: Select this to read both sides of hash joins at the same time. By default, this property is selected. If you do not want this behavior, clear the check box.</td>
</tr>
</tbody>
</table>

Query Processor

The following table lists the parameters that enable you to fine tune how queries are processed:

Table 7–6  Query Processor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable command reuse</td>
<td>Select this to disable Query Processor caching the executed state of a query for reuse.</td>
</tr>
<tr>
<td>Disable DS property cache</td>
<td>Select this to disable caching data source properties.</td>
</tr>
<tr>
<td>Disable insert parameterization</td>
<td>Select this to disable parameterization constants in INSERT statements.</td>
</tr>
<tr>
<td>Disable metadata caching</td>
<td>Select this to disable caching object metadata. If this is selected, the object metadata is taken from the from the original data source instead of the cache.</td>
</tr>
<tr>
<td>Disable query parametarization</td>
<td>Select this to not convert constants into parameters when accessing data sources.</td>
</tr>
<tr>
<td>Disable row mark field fetch</td>
<td>Select this for OLE DB getRows errors to be marked and reshow on every getRows, if the rowset is active.</td>
</tr>
<tr>
<td>Compile after load</td>
<td>Select this to compile an Oracle Connect procedure or view after it is read.</td>
</tr>
</tbody>
</table>
Ignore segments bind failure
This property determines how Oracle Connect responds when the execution of one segment of a segmented data source fails:

- Select this to Log a message and continue execution. This is the default setting.
- Clear the check box to Log a message and stop execution. By default, this property is selected. If you want to stop execution after sending a message, clear this check box.

Prompt database-user password
Select this to configure Oracle Connect to prompt the user for security information when accessing a data source.

Use alternate qualifier
Select this to use the @ symbol instead of a colon (:) when connecting to multiple data sources.

Use table filter expression
Select this to enable the use of tables that have filter expressions specified in their metadata.

Write empty string as null
Select this to replace empty strings in a SET clause of an UPDATE statement or in a VALUES list of an INSERT statement with null values.

Optimistic for update
Select this to use optimistic locking as the default locking behavior on queries with a FOR UPDATE clause.

Disable compilation cache
Select this to disable saving successfully compiled statements in the cache.

Maximum SQL cache
Enter the maximum number of SQL queries that can be stored in cache memory. This property’s value is ignored if Disable compilation cache is selected. The default is 3.

First tree extensions
Enter the maximum size allowed for an SQL query after compilation. The default is 150.

Maximum columns in parsing
Enter the maximum number of columns that a query can reference. The default is 500.

Maximum segmented database threads
Enter the maximum number of open threads allowed, when working with segmented databases.

Minimum number of parameters allocated
Enter the minimum number of parameters that can be used in a query.

Continuous query retry interval
Enter the number of seconds that the query processor waits before executing a query again, when no records are returned. The default is 2.

Continuous query timeout
Enter the number of seconds that the query processor continues to issue queries, when no records are returned. The default is 3600 (one hour), which indicates that after an hour without new messages the continuous query ends. Enter 0 to indicate that there is no timeout and the continuous query does not end automatically.

Continuous query prefix
Enter a prefix to replace the $$ prefix that is used to identify the continuous query special columns. For example, if you enter ##, then the continuous query alias is ‘##StreamPosition’ and the control command alias is ‘##ControlCommand’.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ignore segments bind failure</td>
<td>This property determines how Oracle Connect responds when the execution of one segment of a segmented data source fails:</td>
</tr>
<tr>
<td>Prompt database-user password</td>
<td>Select this to configure Oracle Connect to prompt the user for security information when accessing a data source.</td>
</tr>
<tr>
<td>Use alternate qualifier</td>
<td>Select this to use the @ symbol instead of a colon (:) when connecting to multiple data sources.</td>
</tr>
<tr>
<td>Use table filter expression</td>
<td>Select this to enable the use of tables that have filter expressions specified in their metadata.</td>
</tr>
<tr>
<td>Write empty string as null</td>
<td>Select this to replace empty strings in a SET clause of an UPDATE statement or in a VALUES list of an INSERT statement with null values.</td>
</tr>
<tr>
<td>Optimistic for update</td>
<td>Select this to use optimistic locking as the default locking behavior on queries with a FOR UPDATE clause.</td>
</tr>
<tr>
<td>Disable compilation cache</td>
<td>Select this to disable saving successfully compiled statements in the cache.</td>
</tr>
<tr>
<td>Maximum SQL cache</td>
<td>Enter the maximum number of SQL queries that can be stored in cache memory. This property’s value is ignored if Disable compilation cache is selected. The default is 3.</td>
</tr>
<tr>
<td>First tree extensions</td>
<td>Enter the maximum size allowed for an SQL query after compilation. The default is 150.</td>
</tr>
<tr>
<td>Maximum columns in parsing</td>
<td>Enter the maximum number of columns that a query can reference. The default is 500.</td>
</tr>
<tr>
<td>Maximum segmented database threads</td>
<td>Enter the maximum number of open threads allowed, when working with segmented databases.</td>
</tr>
<tr>
<td>Minimum number of parameters allocated</td>
<td>Enter the minimum number of parameters that can be used in a query.</td>
</tr>
<tr>
<td>Continuous query retry interval</td>
<td>Enter the number of seconds that the query processor waits before executing a query again, when no records are returned. The default is 2.</td>
</tr>
<tr>
<td>Continuous query timeout</td>
<td>Enter the number of seconds that the query processor continues to issue queries, when no records are returned. The default is 3600 (one hour), which indicates that after an hour without new messages the continuous query ends. Enter 0 to indicate that there is no timeout and the continuous query does not end automatically.</td>
</tr>
<tr>
<td>Continuous query prefix</td>
<td>Enter a prefix to replace the $$ prefix that is used to identify the continuous query special columns. For example, if you enter ##, then the continuous query alias is ‘##StreamPosition’ and the control command alias is ‘##ControlCommand’.</td>
</tr>
</tbody>
</table>
Table 7–6 (Cont.) Query Processor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arithmetic fixed precision</td>
<td>Enter an integer determine the precision scale factor for floating decimal position. The default is 0, which indicates that the exact arithmetic function is not used. When the value is set to a small positive integer, the special precise floating point arithmetic is used in the query processor. The value determines the precision scale factor (for example, a value of 2 indicates two digits decimal precision). Setting this parameter can be done at a workspace level and it affects all queries running in that workspace with no change to the query or to the underlying data source. The query processor ADD(), SUBTRACT() and SUM() functions that currently use double arithmetic for both floating and decimal types use this logic. When the value is set to the default, 0, the exact arithmetic function is not used. This property is used to set the Exact Arithmetic function. The \texttt{qpArithmeticFixedPrecision} property is an integer value that determines the fixed precision the Oracle Connect query processor uses for precise floating point arithmetic. It is used to create an accurate result when using the SUM function. Because floating point data types are not accurate their results over time does not correspond to the expected arithmetic sum. In other words, in the floating point representation, values such as 0.7 cannot be represented precisely. If there are eight precision digits, there is usually imprecision in the least significant digit so the number is actually approximately 0.699999995. The \texttt{qpArithmeticFixedPrecision} property corrects this imprecision by using an exact floating point.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parser depth</th>
<th>The maximum depth of the expression tree. The default is 500.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Token size</td>
<td>Enter the maximum length of a string in an SQL query. The minimum value is 64. The default value is 350.</td>
</tr>
<tr>
<td>Insert from select commit rate</td>
<td>Enter the commit rate to use when executing an INSERT-FROM-SELECT operation. If a value more than 0 is entered, a commit is performed automatically after inserting the indicated number of rows. For example, if the value is 5, a commit is performed every time 5 rows are inserted.</td>
</tr>
<tr>
<td>Disable SQS cache</td>
<td>Select this to always read compiled Oracle Connect procedures and views from a disk. In this case, they are not saved in the cache.</td>
</tr>
<tr>
<td>Procedures cache size</td>
<td>Enter the number of stored queries created with a CREATE PROCEDURE statement that can be kept in cache memory. This property’s value is ignored if \texttt{Disable SQS cache} size is selected.</td>
</tr>
<tr>
<td>Expose XML fields</td>
<td>Expose XML fields: Select this to display data returned for a query as XML, representing the true structure of the result. This is useful when querying a data source table that contains arrays or variants.</td>
</tr>
<tr>
<td></td>
<td>■ XML field name: Enter the name used in a query to indicate that the data is returned as XML, instead of the keyword XML. This is available only if \texttt{Expose XML fields} is selected.</td>
</tr>
</tbody>
</table>
Transaction

The following table lists the parameters that define transaction support:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction extended logging</td>
<td>Select this to write extended information about transactions to the transaction manager log files.</td>
</tr>
<tr>
<td>Commit on destroy</td>
<td>Select this to commit all single-phase commit transactions opened for a data source, if a connection closes while the transaction is still open.</td>
</tr>
<tr>
<td>Disable 2PC</td>
<td>Select this to disable two-phase commit capabilities, even in drivers that support two phase commit.</td>
</tr>
<tr>
<td>User commit confirm table</td>
<td>This parameter is not applicable for use with OracleAS CDCAdapter for DB2.</td>
</tr>
<tr>
<td>Transaction log file</td>
<td>The high-level qualifier and name of the log file that logs activity when using transactions.</td>
</tr>
<tr>
<td>Recovery Delay</td>
<td>The number of minutes from the start of a transaction before any recovery operation on that transaction can be attempted. The default is 15 minutes.</td>
</tr>
<tr>
<td>Time limit</td>
<td>Enter the time to wait for a transaction to complete before an error is returned. This parameter is also used when performing a RECOVERY, and it then indicates the number of minutes to wait before a forced activity can be performed, since the last transaction activity.</td>
</tr>
<tr>
<td>Conversions</td>
<td>Select one:</td>
</tr>
<tr>
<td></td>
<td>No conversion: Select this if you want all transactions to remain as sent. This is selected by default</td>
</tr>
<tr>
<td></td>
<td>Convert all to distributed: Select this to convert all simple transactions into distributed transactions.</td>
</tr>
<tr>
<td></td>
<td>Convert all to simple: Select this to convert all distributed transactions into simple transactions.</td>
</tr>
</tbody>
</table>

Tuning

The following table lists the parameters that define the tuning:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dsm maximum buffer size</td>
<td>Enter the maximum size of a cache memory. This cache is used when memory is required on a temporary basis (as when Oracle Connect sorts data for a query output, for a subquery, or for aggregate queries). This cache size is not used for hash joins and lookup joins (for more information, see the hashBufferSize parameter). The default value is 1000000 bytes.</td>
</tr>
<tr>
<td>Dsm maximum Sort buffer size</td>
<td>Enter the maximum size of the sort buffers. Use this parameter instead of dsmMaxBufferSize for sorts only. The default value is 1000000 bytes.</td>
</tr>
</tbody>
</table>
### Table 7–8 (Cont.) Tuning Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dsm middle buffer size</td>
<td>Enter the maximum size of the index cache. This cache is not used for hash joins and lookup joins. The default value is 100000 bytes.</td>
</tr>
<tr>
<td>File pool size</td>
<td>Enter the maximum number of files that can be opened in the file pool. The default is 10.</td>
</tr>
<tr>
<td>File pool size per file</td>
<td>Enter the size of the file in the pool. The default is 3.</td>
</tr>
<tr>
<td>File close on transaction</td>
<td>Select this if you want the File Pool to close when a transaction is committed.</td>
</tr>
<tr>
<td>Use global file pool</td>
<td>Select this to use a global file pool. When the workspace server mode parameter is set to multiClient or reusable, this parameter also indicates whether the file pool closes upon the client disconnection.</td>
</tr>
<tr>
<td>Hash buffer size</td>
<td>Enter the amount of cache memory that is available for each hash join or lookup join. The default is 100000 bytes.</td>
</tr>
<tr>
<td>Hash max open files</td>
<td>Enter the maximum number of files that a query can open simultaneously for use when performing hash joins. The number assigned to this parameter must not exceed the system maximum. The default is set to 90 files. Note: The hash join optimization strategy results in several files being opened to perform the join. The larger the table size, the more files are opened. By adjusting this parameter you can disable hash joins on very large tables, while allowing hash joins for small tables.</td>
</tr>
<tr>
<td>Hash primary extent size</td>
<td>Enter the primary extent size.</td>
</tr>
<tr>
<td>Hash secondary extent size</td>
<td>Enter the secondary extent size.</td>
</tr>
<tr>
<td>Hash enable RO</td>
<td>Select this for the QP to store the first hash bucket in memory instead of a sequential file.</td>
</tr>
</tbody>
</table>

### XML

The following table lists the parameters that define XML support:

### Table 7–9 XML Category Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COM maximum XML in memory</td>
<td>Specifies the maximum size of an XML document held in memory. The default is 65535 bytes.</td>
</tr>
<tr>
<td>COM maximum XML size</td>
<td>Specifies the maximum size of an XML document passed to another computer. The default is 65535 bytes. Note: When you increase this value for this property, you may need to increase the value for the Maximum XML in memory property in the daemon. For more information on daemons, see Control.</td>
</tr>
<tr>
<td>COM XML transport buffer size</td>
<td>Enter the maximum size of the internal communications buffer. The default value (-1) indicates there is no size limit.</td>
</tr>
</tbody>
</table>
Migration Considerations

You can migrate an adapter configuration from one platform to another. The configuration information is stored in the Oracle Connect repository on the source platform and is exported to an XML file which can then be imported to the target platform.

Note that when migrating a configuration, file names and paths that are specific to the source platform must be changed to valid files on the target platform.

To migrate an adapter configuration using Oracle Studio, perform the following steps:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. In the Design perspective, Configuration view, expand the Machines folder.
3. Right-click the required computer, and select Export XML definitions.
4. In the File name field, type the path and name of the XML file where the XML representation of the computer and its complete configuration is stored.
5. Edit any paths in the XML definition to the paths required on the target platform. For example, the setting for the serverLogFile might need changing, depending on the platform.
6. Set up the target platform in Oracle Studio in the same way you set up the source platform, as described in Setting Up the IBM z/OS Platform in Oracle Studio.
7. In the Configuration Explorer, right-click the target computer and select Import XML definitions.
8. Import the XML file to the target platform.

Security Considerations

Oracle Connect works within the confines of the platform security system.

In addition, Oracle Connect provides the following security:

- A binary XML encryption mechanism, which is activated as follows:
  1. The client’s first message to the server includes a pre-defined shared key, including the key name and value in the connection string. The server gets the key value for the key name passed from the client from the native object store (NOS).
2. The server generates a random 128-bit RC4 session key which is returned encrypted to the client, using the shared key. If no predefined shared key is provided, then a predefined, hardcoded key is used (this key is hardcoded on the client and on the server).

3. Passwords are always encrypted when passed over the wire, using an RC4, 128-bit session key, regardless of whether the entire session is encrypted or not.

4. If a predefined shared key was provided, then the entire session is encrypted. Otherwise, only the password exchange is encrypted (using the hardcoded key).

- Credentials: Passwords and user names exchanged over the network are encrypted using a pre-defined, hardcoded, 128-bit RC4 session key.
- Design Time: Security within Oracle Studio to grant access to Oracle Studio itself and to grant access to computers, user profiles and workspaces.
- Run time: Security used to access DB2, including controlling the daemon for the access.

Setting Design Time Security

Set the design time security as described in the following sections:

- Securing access to Oracle Studio is described in Setting Password Access to Oracle Studio.
- Securing rights to configure a computer in Oracle Studio is described in Specifying Users with Administrative Rights.
- Securing access to user profiles is accomplished by right-clicking the relevant user profile in Oracle Studio and selecting Change Master Password. In the dialog box that is displayed, specify a password that must be provided in the future to edit the specific user profile.
- Securing access to workspaces is accomplished by right-clicking the relevant workspace in Oracle Studio and selecting Set Authorization. In the dialog box that is displayed, specify a valid user and password that must be provided in the future to edit the specific workspace.

Setting Run-time Security

During run time, security considerations are implemented as follows:

- When the client request accesses the legacy platform through the daemon, either anonymous access is allowed or a valid user name and password must be provided for the computer in the user profile. The userName and password properties in the J2CA 1.5 DB2 adapter are used at this stage to access the daemon.

Note: The user name used to access the daemon must also be the name of a user profile used.

- Access by the client must be through a valid port, according to the port range entered in the Server section of the workspace Server Mode tab in Oracle Studio. For more information, see the explanation of the workspace Server Mode.
To be allocated a server process, the client must be granted anonymous access to
the workspace or be listed in the **Authorized Workspace Users** section of the
workspace **Security** tab in Oracle Studio. For more information, see the
explanation of the workspace **Security** tab.

The ability to run commands on the daemon, such as starting or stopping a
daemon or ending server processes is available only to administrators who have
been registered in Oracle Connect as a daemon administrator. A client is registered
as a valid daemon administrator in the **Administrator privileges** section of the
daemon **Security** tab in Oracle Studio. For more information, see explanation of
the daemon **Security** tab.

**Note:** You can also specify administrators who can run commands
only at the level of the workspace. Select these administrators in the
workspace **Security** tab, as described in the explanation of the
workspace **Security**.

---

**Transaction Support**

OracleAS CDC Adapter for DB2 supports global transactions and can fully participate
in a distributed transaction.

To work with global transactions, select **Convert all to distributed** in the Transaction
section of the binding environmental properties. For more information, see **Transaction**
in the **Configuring a Binding Environment** section.

To use the global-transaction capability to access data on the **z/OS** computer, define
every library in the ATTTSRVR JCL as an APF-authorized library.

To define a DSN as APF-authorized, in the SDSF screen, enter the following command:

```
/setprog apf,add,dsn=instroot.library,volume=ac002
```

Where **ac002** is the volume where you installed Oracle Connect and **INSTROOT** is the
high-level qualifier where Oracle Connect is installed.

If the Oracle Connect installation volume is managed by SMS, when defining
APF-authorization enter the following command in the SDSF screen:

```
/setprog apf,add,dsn=instroot.library,SMS
```

Ensure that the library is APF-authorized, even after an IPL (restart) of the computer.

If RRS is not running, OracleAS CDC Adapter for DB2 can participate in a distributed
transaction, as the only one-phase commit resource, if the **Transaction log file**
environment property includes the keyword **NORRS**. For more information, see **Transaction**
in the **Configuring a Binding Environment** section.

**Note:** If a log file is not specified, then the format for the **logFile**
parameter when RRS is not running is **,NORRS**.
Advanced Tuning of the Metadata

Oracle Studio enables you to define outbound adapter interactions. In addition, Oracle Studio defines input and output structures used for these interactions. The interactions and input and output structures are maintained as metadata by Oracle Studio in the Metadata tab of the Design perspective.

This appendix contains the following sections:

- Metadata for the DB2 Data Source
- Adapter Metadata

Metadata for the DB2 Data Source

Using Oracle Studio, perform the following steps to maintain the metadata for the DB2 data source:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. Expand the Machines folder, then expand the CDC Server machine that you defined when you Set up the CDC Server.
3. Expand the Bindings folder.
4. Expand the binding with the name of your CDC solution and the suffix _ag.
5. Expand the Data sources folder to display the data sources.
6. Right-click the DB2 data source and select Show Metadata View to display the Metadata view.
7. Right-click the table you want to view in the Metadata Explorer and select Edit.

The Metadata tab is shown in the following figure:

The metadata editor opens, displaying the General tab, with general table details. The following tabs are used to view and edit the metadata:

- General Tab
- Columns Tab
- Indexes Tab
- Statistics Tab
- Advanced Tab
General Tab

Use the General tab to maintain information about the whole table, such as the table name and the way the table is organized.

The General tab is shown in the following figure:

**Figure A–1  The General Tab**

![The General Tab](image)

The General tab comprises fields, as listed in the following table:

**Table A–1  General Tab Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>An optional description of the table.</td>
</tr>
<tr>
<td>Table Properties</td>
<td>The name of the file that contains the table. You must enter the full path and include the file extension for the file. For example, D:\COBOL\orders.cob. You can click Browse and browse to find and enter the location of the table file. Note: Do not enter the file extension for DIASM or CIASM files.</td>
</tr>
<tr>
<td>Data file location</td>
<td></td>
</tr>
</tbody>
</table>
Columns Tab

Use the Columns tab to specify metadata that describe the table columns. The tab is divided into the following:

- Column Definition Section
- Column Properties

Table A–1 (Cont.) General Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Select how the record represented by the table is organized. The options that are displayed depend on the record. The following options are available:</td>
</tr>
<tr>
<td></td>
<td>■ Index</td>
</tr>
<tr>
<td></td>
<td>■ Sequential</td>
</tr>
<tr>
<td></td>
<td>■ Relative: Used with RRDS files. Access to a specific record number of a relative file is performed by using a pseudo column to specify the record position. The hash symbol (#) is used to specify a pseudo column. For example:</td>
</tr>
<tr>
<td></td>
<td>SELECT * FROM colleges WHERE # = 6</td>
</tr>
<tr>
<td>Record format</td>
<td>Specifies how the record, represented by the table, is formatted. The options that are displayed depend on the record. The DB2 records have a fixed format.</td>
</tr>
<tr>
<td>Maximum record</td>
<td>The maximum allowable size of a record (in bytes).</td>
</tr>
<tr>
<td>length</td>
<td>DB Command DB2 specific commands are displayed in the DB Command section of the General tab. The information displayed is generated automatically when the metadata is generated and should not be modified.</td>
</tr>
<tr>
<td>Filter Expression</td>
<td>A WHERE clause. This clause is added to every query accessed using this metadata. The filter is specified without the WHERE keyword. Specify a filter when multiple logical tables are stored in the same physical file.</td>
</tr>
</tbody>
</table>
The **Columns** tab is shown in the following figure:

**Figure A–2  The Columns Tab**

![Columns Tab](image)

### Column Definition Section

The top section of this tab lets you define the columns in the source data. You can click in any row (which represents a column in the database table) to edit the information. The following table describes this section.

### Table A–2  Metadata Column Tab Definitions

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the column</td>
</tr>
<tr>
<td>Data type</td>
<td>The data type of the column. Selecting this field displays a list of the possible data types.</td>
</tr>
<tr>
<td>Size</td>
<td>The size of the column</td>
</tr>
</tbody>
</table>
| Scale      | The information entered in this field depends on the data type:  
For decimal data types, this is the number of digits to the right of the decimal place. This number must not be greater than the number of digits. The default value is 0.  
For scaled data types, this is the total number of digits. The number must be negative. |
The buttons on the right side of the tab are used to manipulate the data in this section of the tab. The following table describes how you can move around in this section.

### Table A–3 Definition Section Buttons

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert</td>
<td>Inserts a column to the table. You can insert a new column. If the table has arrays, you can add a new child column.</td>
</tr>
<tr>
<td>Up</td>
<td>Moves your selection to the column directly above the currently selected column.</td>
</tr>
<tr>
<td>Down</td>
<td>Moves your selection to the column directly below the currently selected column.</td>
</tr>
<tr>
<td>Rename</td>
<td>Lets you rename the selected column.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the selected column.</td>
</tr>
<tr>
<td>Find</td>
<td>Click this button to open a list of all columns in the database. Select a column and click <strong>OK</strong> to select it in the table.</td>
</tr>
</tbody>
</table>

### Column Properties

You can change the property value by clicking in the **Value** column. To display column properties, select a column from the Column Definition (top) section.

The properties for the column are displayed at the bottom of the tab.

The following table shows properties available for selected columns.
### Table A–4 Metadata Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias</td>
<td>A name used to replace the default virtual table name for an array. Virtual table names are created by adding the array name to the record name. When an array includes another array the name of the nested array is the name of the record and the parent array and the nested array. When the default generated virtual table name is too long, use an Alias to replace the long name.</td>
</tr>
<tr>
<td>Autoincrement</td>
<td>The current field is updated automatically by the data source during an INSERT statement and is not explicitly defined in the INSERT statement. The INSERT statement should include an explicit list of values. This attribute is used for fields such as an order number field whose value is incremental each time a new order is entered to the data source.</td>
</tr>
<tr>
<td>Comment</td>
<td>A short note or description about the column.</td>
</tr>
<tr>
<td>DB command</td>
<td>DB2 specific commands for the column. The information displayed is generated automatically when the metadata is generated and should not be modified.</td>
</tr>
<tr>
<td>Empty value</td>
<td>The value for the field in an insert operation, when a value is not specified.</td>
</tr>
<tr>
<td>Explicit Select</td>
<td>When true, the current field is not returned when you execute a SELECT * FROM... statement. To return this field, you must explicitly ask for it in a query, for example, SELECT NATION_ID, SYSKEY FROM NATION where SYSKEY is a field defined with Explicit Select. You cannot use an asterisk (*) in a query where you want to retrieve a field defined with the Explicit Select value.</td>
</tr>
<tr>
<td>Hidden</td>
<td>The current field is hidden from users. The field is not displayed when a DESCRIBE statement is executed on the table.</td>
</tr>
<tr>
<td>Non Selectable</td>
<td>When true, the current field is never returned when you execute an SQL statement. The field is displayed when a DESCRIBE statement is executed on the table.</td>
</tr>
<tr>
<td>Non Updateable</td>
<td>If true, the current field cannot be updated.</td>
</tr>
<tr>
<td>Nullable</td>
<td>This value allows the current field to contain NULL values.</td>
</tr>
<tr>
<td>Null value</td>
<td>The null value for the field during an insert operation, when a value is not specified.</td>
</tr>
<tr>
<td>Chapter of</td>
<td>This property shows that the set member field is a chapter of an owner field. A value for this property must be used when accessing a set member as a chapter in an ADO application. This property is used for DBMS metadata.</td>
</tr>
<tr>
<td>OnBit</td>
<td>The position of the bit in a BIT field and the starting bit in a BITS field.</td>
</tr>
<tr>
<td>Subfield of</td>
<td>The value is generated automatically when you generate metadata from DB2 data that includes a superdescriptor based on a subfield. A field is created to base this index on, set to the offset specified as the value of the Subfield start field. If no value is entered in the Subfield start field, the subfield is set by default to an offset of 1.</td>
</tr>
<tr>
<td>Subfield start</td>
<td>The offset within the parent field where a subfield starts.</td>
</tr>
</tbody>
</table>
Indexes Tab

Use the **Indexes** tab to specify metadata describing the indexes of a table.

**Note:** The Indexes tab contains information only if the **Organization** field in the Table tab is set to **Index**.

The **Indexes** tab is shown in the following figure:

**Figure A–3  The Indexes Tab**

This tab has two sections. The first section lets you define the index keys for the columns in the table. The bottom of the tab lists the properties for each of the columns at the top.

The following sections describe the **Indexes** tab:

- **Table Information**
- **Properties**

**Table Information**

The following table describes the fields for the top part of the tab, which defines the indexes used for the table.
The buttons on the right side of the tab are used to manipulate the data in this section of the tab. The following table describes how you can move around in this section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The names of existing indexes for the current table.</td>
</tr>
<tr>
<td>Order</td>
<td>The ordering of the rows retrieved by the index.</td>
</tr>
<tr>
<td>DB Command</td>
<td>DB2-specific commands for the index. The information displayed is generated automatically when the metadata is generated and should not be modified.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert</td>
<td>Inserts an index to the table.</td>
</tr>
<tr>
<td>Rename Index</td>
<td>Lets you rename the selected index.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the selected index.</td>
</tr>
</tbody>
</table>

**Properties**

You can index properties for each index column. To display the index properties, select a column from the Index Definitions (top) section.

The properties for the column are displayed at the bottom of the tab.

The properties displayed at the bottom of the tab describe the index or segment. The properties available depend on the data source.
Use the **Statistics** tab to update metadata statistics for a table. The **Statistics** tab is shown in the following figure:

![The Statistics tab](image)

The **Statistics** tab has three sections. The following tables describe each section.

### Table A–7  Table Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rows</td>
<td>Enter or use arrows to select the approximate number of rows in the table. If the value is -1, then the number of rows in the table is unknown (a value was not supplied and the update statistics utility was not run to update the value). A value of 0 indicates that this table is empty.</td>
</tr>
<tr>
<td>Blocks</td>
<td>Enter or use arrows to select the approximate number of blocks in the table.</td>
</tr>
</tbody>
</table>

**Note:** If neither the number of rows nor the number of blocks is specified for a table, queries over the table might be executed in a nonoptimal manner.

Use the Columns group box to specify cardinality for each of the columns in the table:
Use the Indexes group box to specify cardinality for the columns in each of the indexes in the table:

### Table A–9 Indexes group Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indexes and segments</td>
<td>The indexes and segments in the table.</td>
</tr>
<tr>
<td>Cardinality</td>
<td>The number of distinct key values in the index. If the value is −1, then the number of distinct key values in the index is unknown (a value was not supplied and the update statistics utility was not run to update the value). A value of 0 indicates that there are no distinct key values in the index.</td>
</tr>
</tbody>
</table>

### Generating Statistics

Click **Update** in the **Statistics** tab to generate updated statistics for the table. The Update Statistics screen is displayed, as shown in the following figure:

![Update Statistics screen](image)

The following tables list the fields that are used to update statistics:

The **Type** section lets you edit the following:
The Resolution section lets you specify the statistical information returned:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated</td>
<td>An estimation of the amount of statistical information returned.</td>
</tr>
<tr>
<td>Estimated with rows</td>
<td>An estimation of the amount of statistical information returned. The estimate includes an estimation of the number of rows in the table. Specify the number in the text box. This number is used to shorten the time to produce the statistics if the value specified here is the correct value, or close to the correct value.</td>
</tr>
<tr>
<td>Exact</td>
<td>The exact statistical information returned. Note that this can be a lengthy task and can lead to disk space problems with large tables.</td>
</tr>
</tbody>
</table>

The Advanced tab lets you enter information about the virtual view policy for arrays. These parameters are valid only if you are using virtual array views. You configure virtual array views in the Modeling section when Configuring a Binding Environment.
The configurations made in this editor are for the selected table, only. The same parameters are configured on the data source level in the data source editor.

**Figure A–6 Data Source Metadata Advanced Tab**

Enter the following information in this tab:

- **Generate sequential view**: Select this to map non-relational files to a single table.
- **Generate virtual views**: Select this to have individual tables created for each array in the non-relational file.
- **Include row number column**: Select one of the following:
  - **true**: Select true, to include a column that specifies the row number in the virtual or sequential view. This is true for this table only, even in the data source is not configured to include the row number column.
  - **false**: Select false, to not include a column that specifies the row number in the virtual or sequential view for this table even if the data source is configured to include the row number column.
  - **default**: Select default to use the default data source behavior for this parameter.
- **Inherit all parent columns**: Select one of the following:
  - **true**: Select true, for virtual views to include all the columns in the parent record. This is true for this table only, even in the data source is not configured to include all of the parent record columns.
Adapter Metadata

Use Oracle Studio to maintain the metadata for the CDC Queue adapter in your staging area.

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. Expand the Machines folder, then expand the machine with your staging area.
3. Expand the Bindings folder.
4. Expand the binding for your CDC staging area (with the _sa suffix).
5. Expand the Adapters folder to display the adapters list.
6. Right-click the CDC Queue adapter for your solution, and select Show Metadata View to open the Metadata view.

You can create and edit the following adapter metadata properties:

- **Adapter Metadata General Properties**: Enter and edit information about the adapter, such as the adapter name and the way in which you connect to the adapter. You make these changes in the Design perspective, Metadata view.
- **Adapter Metadata Interactions**: Enter details of an interaction. The interaction Advanced tab is displayed for some adapters only, such as the Database adapter and includes more details about the interaction.
- **Adapter Metadata Schema Records**: The input and output record structure for a record in the adapter definition.

### Adapter Metadata General Properties

You can enter and edit information about the adapter, such as the adapter name and the way in which you connect to the adapter. You make these changes in the Design perspective, Metadata view. The following describes how to open the Adapter General Properties editor.

1. In Oracle Studio Design perspective, Metadata view, expand the Adapters folder.
2. Right-click the adapter to edit, and select Open.

   The General properties editor is displayed.

The Adapter General Properties editor is shown in the following figure:

- **false**: Select false, so virtual views do not include the columns in the parent record for this table even if the data source is configured to include all of the parent record columns.
- **default**: Select default to use the default data source behavior for this parameter.
The **General** properties are listed in the following table:

**Table A–12  General tab Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Specifies an identifying description of the adapter.</td>
</tr>
<tr>
<td>Authentication mechanism</td>
<td>Specifies the authentication to access the adapter. The available mechanisms are:</td>
</tr>
<tr>
<td></td>
<td>■ kerby5</td>
</tr>
<tr>
<td></td>
<td>■ none</td>
</tr>
<tr>
<td>Max request size</td>
<td>Specifies the maximum size in bytes for an XML request or reply. Larger messages are rejected with an error.</td>
</tr>
<tr>
<td>Max active connections</td>
<td>Specifies the maximum number of simultaneous connections for an adapter (per process).</td>
</tr>
<tr>
<td>Max idle timeout</td>
<td>Specifies the maximum time, in seconds, that an active connection can stay idle. After that time, the connection is closed.</td>
</tr>
<tr>
<td>Adapter Specifications</td>
<td>Specifies the adapter-specific properties for an interaction. The CDC Queue adapter does not have any adapter-specific properties.</td>
</tr>
</tbody>
</table>
Adapter Metadata Interactions

The Adapter Metadata Interactions editor defines an interaction and its input and output definitions. The following describes how to open the Adapter Metadata editor.

1. In Oracle Studio Design perspective, Metadata view, expand the **Adapters** folder.
2. Expand the adapter with the interaction to edit.
3. Expand the **Adapter** folder.
4. Right-click the adapter you want to edit and select **Open**.

The Adapter Metadata Interactions editor is displayed.

The Adapter Metadata Interactions editor is shown in the following figure:

**Figure A–8 Adapter Metadata Interactions**

![Image of Adapter Metadata Interactions]

The Adapter Metadata Interaction properties are listed in the following table:

**Table A–13 Interaction General tab Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Provides a descriptive identifier for the interaction.</td>
</tr>
</tbody>
</table>
In the Adapter Metadata Interactions editor, click Advanced at the bottom of the editor to open the Interaction Advanced tab. Use this to enter advanced details for the interaction or to create interaction manually.

The following figure shows the Interactions Advanced tab.

Table A–13  Interaction General tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>Determines the interaction mode. The following interaction modes are available:</td>
</tr>
<tr>
<td></td>
<td>- sync-send-receive: The interaction sends a request and expects to receive a response.</td>
</tr>
<tr>
<td></td>
<td>- sync-send: The interaction sends a request and does not expect to receive a response.</td>
</tr>
<tr>
<td></td>
<td>- sync-receive: The interaction expects to receive a response.</td>
</tr>
<tr>
<td>Input record</td>
<td>Identifies an input record.</td>
</tr>
<tr>
<td>Output record</td>
<td>Identifies an output record for the results of an interaction.</td>
</tr>
<tr>
<td>Interaction Specific</td>
<td>Specific properties for the interaction. When an Interaction Advanced tab is used, this section is not displayed. The parameters shown are for a Legacy Plug Adapter.</td>
</tr>
</tbody>
</table>
This tab has three sections.

The **SQL Statement** section lets you build any valid SQL statement. Use the tabs to select the tables and statement types and build a statement. When you select **Enable manual query editing**, you can manually enter a valid SQL statement at the bottom of the screen.

The **Interaction Properties** section lets you select any of the following:

- **Pass Through**: Select this to pass a query directly to the DB2 data.
- **Reuse compiled query**: Select this to save query objects created in the previous execution to the cache. This allows the objects to be reused.
- **Fail on no rows returned**: Select this if you want the system to return an error if no rows are selected.
- **Encoding**: Select one of the following from the list:
  - **base64**: Select this for base 64 encoding
- **hex**: Select this for hexadecimal encoding

The **Parameters** section lets you create parameters to use for the interaction. To create a parameter, click **Add**. Enter a name in the dialog box and click **OK**. The parameter is entered in the list. You can edit the following properties for each parameter you create. Click in the corresponding cell to edit the properties.

- **Name**: The name of the parameter. This is automatically entered when you create a new parameter. You can click in the cell to change this parameter.

- **Type**: The type of parameter. Select one of the following types from the list:
  - **string**
  - **number**
  - **timestamp**
  - **binary**
  - **xml**

- **Nullable**: Select True or False to determine whether the parameter can be nullable.

- **Default**: Enter a default value for the parameter, which is used if the parameter attribute is missing in the input record.

---

**Notes:**

- If a field is not nullable and a default value is not supplied in the schema part of the Adapter Definition, an error occurs if the parameter attribute is missing in the input record.

- The parameters must be entered in the same order as they are used in the SQL statement.

---

**Adapter Metadata Schema Records**

The Adapter Metadata Schema Records editor defines the general details of the input and output record structures for the interaction. The following describes how to open the Adapter Metadata Schema Records editor:

1. In Oracle Studio Design perspective, Metadata view, expand the **Adapters** folder.
2. Expand the adapter with the schema records to edit.
3. Expand the **Schemas**.
4. Right-click the schema you want to edit and select **Open**.
   
   The Adapter Metadata Schema Records editor is displayed.
The Adapter Metadata Schema Records editor is shown in the following figure:

**Figure A–10  Adapter Metadata Schema Records**

The Adapter Metadata Schema Records properties are listed in the following tables:

**Table A–14  Schema Record Tab**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields list</td>
<td>Defines the single data item within a record. This section has a table with the following three columns:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Name: The name of the field</td>
</tr>
<tr>
<td></td>
<td>■ Type: The data type of the field. For more information, see the Valid Data Types table for a list of the valid data types.</td>
</tr>
<tr>
<td></td>
<td>■ Length: The size of the field including a null terminator, when the data type supports null termination (such as the string data type).</td>
</tr>
<tr>
<td>Specifications</td>
<td>Defines specific field properties. To display the properties, select the specific field in the Fields list.</td>
</tr>
</tbody>
</table>

The following table describes the valid data types that can be used when defining these specifications in the Schema Record editor.

**Table A–15  Valid Data Types**

<table>
<thead>
<tr>
<th>Binary</th>
<th>Boolean</th>
<th>Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>Double</td>
<td>Enum</td>
</tr>
</tbody>
</table>
You can also edit the adapter metadata by viewing its XML schema. The following describes how to open the adapter metadata XML editor:

1. In Oracle Studio Design perspective, Metadata view, expand the **Adapters** folder.
2. Right-click the adapter to edit, and select **Open as XML**.

The adapter XML editor is displayed in the Design view.

The adapter XML editor is shown in the following figure:

**Figure A–11  Adapter Metadata XML Editor**

<table>
<thead>
<tr>
<th>Binary</th>
<th>Boolean</th>
<th>Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td>Float</td>
<td>Int</td>
<td>Long</td>
</tr>
<tr>
<td>Numeric[p,s]</td>
<td>Short</td>
<td>String</td>
</tr>
<tr>
<td>Time</td>
<td>Timestamp</td>
<td></td>
</tr>
</tbody>
</table>

For information on how to edit the properties in the XML editor Design view, see **Editing XML Files in Oracle Studio**.

For an explanation of the attributes for the CDC Queue adapter that you should edit, see **CDC Queue Adapter Properties**.
The daemon configuration is managed using Oracle Studio. Daemon configuration is divided into the following groups:

- Control
- Logging
- Security
- Workspaces

**Control**

Using the Control tab for the daemon, you define various daemon control options. The Daemon Control tab is accessed as follows:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. From the Design perspective Configuration view expand the Machines folder.
3. Right-click the computer and select Open Runtime Perspective.
4. Right-click the required daemon in the Runtime Explorer and select Edit Daemon Configuration. The Control tab for the daemon is displayed in the editor.
5. After making changes to the daemon, right-click the daemon and select Reload Configuration.

---

**Note:** You can also change daemon settings using the Configuration Explorer, by selecting a computer and scrolling the list to the required daemon. Right-click the daemon and select Edit Daemon.

Changes made to the daemon configuration are only implemented after the configuration is reloaded using the Reload Configuration option in the Runtime Manager perspective.

---

The Control tab is shown in the following figure:
Figure B–1  The Control tab

The following table shows the parameters that can be set in the Control tab:

Table B–1  Daemon Control tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daemon IP Address</td>
<td>Enter the IP address of the computer(s) where the daemon is listening. If no IP address is entered, the daemon listens on all available IP addresses.</td>
</tr>
<tr>
<td>Daemon port</td>
<td>Enter the port where the daemon is listening. If no port is entered, the daemon listens on all available ports.</td>
</tr>
<tr>
<td>Automatically recover from failure</td>
<td>The daemon restarts automatically if it fails for any reason (any error that causes the daemon process to terminate, such as network process lost or the CPU running the daemon crashes and the backup daemon is defined on another CPU). All available and unconnected servers are terminated and any connected servers are marked and terminated on release. Also, the backup starts a backup for itself. The backup appends a new log file to the log of the original daemon, adding a line indicating that a backup daemon was started.</td>
</tr>
<tr>
<td>Maximum XML request size</td>
<td>The maximum number of bytes that the daemon handles for an XML document.</td>
</tr>
<tr>
<td>Default language</td>
<td>The language that the daemon supports. This setting is used when working with a client with a code page different from the server code page.</td>
</tr>
<tr>
<td>Maximum XML in memory</td>
<td>The maximum amount of space reserved for the XML in memory.</td>
</tr>
</tbody>
</table>
Using the **Logging** tab, you define the daemon log file settings, the log file structure and the location where the log is saved. In addition, use it to define the data that is logged and traced in the file.

The following describes how to open the **Logging** tab.

1. From the **Start** menu, select **Programs, Oracle**, and then select **Studio**.
2. From the Design perspective Configuration view expand the **Machines** folder.
3. Right-click a computer and select **Open Runtime Perspective**.
4. Right-click the daemon in the Runtime Explorer and select **Edit Daemon Configuration**.
5. Click the **Logging** tab.
6. After making changes to the daemon, right-click the daemon and select **Reload Configuration**.

---

**Table B–1 (Cont.) Daemon Control tab Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call timeout</td>
<td>The timeout period for <em>short</em> calls for all daemons. The definition of a short call is a call that should be completed in a few seconds. For example, most calls to a database such as DESCRIBE should be completed in a few seconds as opposed to call like a GETROWS call, which can take a long time. In heavily loaded or otherwise slow systems, even short calls such as calls to open a file, may take a significant amount of time. If a short call takes more than the specified time to complete, then the connection is stopped. The default value for this parameter is 60 seconds. Values of less than 60 seconds are considered to be 60 seconds. Specifying the timeout in a workspace overrides the value set in this field for that workspace.</td>
</tr>
</tbody>
</table>
| Connect timeout  | The time the client waits for a daemon server to start. If the daemon server does not start within this period, then the client is notified that the server did not respond. The value specified for this parameter serves as the default timeout for all the workspaces listed in the daemon configuration. The default value for this parameter is 60 seconds. **Notes:**
- Entering the timeout in a workspace overrides the value set in this field for that workspace.
- Even if the XML source does not list this parameter in the workspace section, the workspace gets it using the default value. To prevent a workspace from using the default value, you must enter a value of zero for this parameter in the workspace section. |
| Client idle timeout | The maximum amount of time any daemon client may be idle before the connection with the server is closed. Specifying the timeout in a Workspace overrides this setting for that workspace. |
7. Right-click the daemon and select **Recycle Servers**. Any servers in the connection pool are closed and new servers start with the new configuration.

The **Logging** tab for the daemon is shown in the following figure:

**Figure B–2 The Logging tab**

The **Daemon Logging** tab comprises fields, as listed in the following table:

**Table B–2 Daemon Logging tab Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Logging options</strong></td>
<td></td>
</tr>
<tr>
<td>Daemon log file location</td>
<td>Specifies the daemon produces its log data. The full path must be specified.</td>
</tr>
<tr>
<td>Server log filename format</td>
<td>Defines the name and location of the server log file. The field must specify the full path name. If no directory information is provided for the log file, then it is located in the login directory of the account running Oracle Connect workstation.</td>
</tr>
<tr>
<td>Daemon operations</td>
<td>Select this to log all of the daemon operations.</td>
</tr>
</tbody>
</table>
Table B–2 (Cont.) Daemon Logging tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracing and debug options</td>
<td></td>
</tr>
<tr>
<td>Daemon RPC function calls</td>
<td>Select this to log all daemon RPC function calls.</td>
</tr>
<tr>
<td>Log ACX</td>
<td>Select this if you want to log requests and processes.</td>
</tr>
<tr>
<td>Extended RPC trace</td>
<td>Generates a verbose message in the server log file for each low-level RPC function called. This is useful for troubleshooting the server.</td>
</tr>
<tr>
<td>System trace</td>
<td>Generates system-specific tracing of various operations.</td>
</tr>
<tr>
<td>Timing</td>
<td>Generates a time stamp for every entry to the server log file.</td>
</tr>
<tr>
<td>Sockets</td>
<td>Generates a message in the server log file for each socket operation.</td>
</tr>
<tr>
<td>Trace information</td>
<td>Select this to log low-level RPC operations.</td>
</tr>
<tr>
<td>No timeout</td>
<td>Disables the standard RPC timeouts, setting them to a long duration (approximately an hour) to facilitate debugging.</td>
</tr>
<tr>
<td>Call trace</td>
<td>Generates a message in the server log file for each RPC function called. This is useful for troubleshooting the server.</td>
</tr>
<tr>
<td>RPC trace</td>
<td>Enables debugging messages on the server.</td>
</tr>
<tr>
<td>Binary XML log level</td>
<td>Sets the binary XML log level. Your options are:</td>
</tr>
<tr>
<td></td>
<td>▪ debug</td>
</tr>
<tr>
<td></td>
<td>▪ none (the default)</td>
</tr>
<tr>
<td></td>
<td>▪ api</td>
</tr>
<tr>
<td></td>
<td>▪ info</td>
</tr>
</tbody>
</table>

The following tokens can appear in the log file template and are replaced accordingly:

- %A: workspace name
- %D: date (yymmd)
- %I: instance number of the given workspace server
- %L: server account login directory
- %P: server process ID
- %T: time (hhmmss)
- %U: server account name (user name)

For example, %L/server_%A%I.log may produce a log file such as:
/usr/smith/server_sales15.log.

The default log file template is %L/server_%A%I.log.

Security

The following Security tab for the daemon is used to:

- Grant administration rights for the daemon.
- Determine access to the computer.
The following shows how to open the Daemon Security tab:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. From the Design perspective Configuration view expand the Machines folder.
3. Right-click the computer and select Open Runtime Perspective.
4. Right-click the daemon in the Runtime Explorer and select Edit Daemon Configuration.
5. Click the Security tab.
6. After making changes to the daemon, right-click the daemon and select Reload Configuration.

---

**Note:** You can also change daemon settings using the Configuration Explorer, by selecting a computer and scrolling the list to the required daemon. Right-click the daemon and select Edit Daemon.

Changes made to the daemon configuration are not implemented. They are only implemented after the configuration is reloaded using the Reload Configuration option in the Runtime Manager.

---

7. Right-click the daemon and select Recycle servers. Any servers in the connection pool are closed and new servers start with the new configuration.

The Daemon Security tab is shown in the following figure:

![The Daemon Security tab](image)

The Daemon Security tab comprises fields, as listed in the following table:
A daemon can include several workspaces. A workspace defines the server processes and environment that are used for the communication between the client and the server for the duration of the client request. Each workspace has its own definition. The workspace definition is divided into the following groups:

- **General**
- **Server Mode**
- **Security**

### General

Using the **General** tab, you enter general information about the features that control the operation of the workspace, such as the server type, the command procedure used to start the workspace and the binding configuration associated with this workspace.

Do the following to open the **General** tab:

1. From the **Start** menu, select **Programs**, **Oracle**, and then select **Studio**.
2. From the Design perspective Configuration view, expand the **Machines** folder.
3. Right-click a computer and select Open Runtime Perspective.

4. Expand the Daemons node to display the workspaces in the Runtime Explorer.

5. Right-click the workspace and select Edit Workspace Configuration. The General tab opens.

6. After making changes to the workspace, right-click the daemon and select Reload Configuration.

7. Right-click the daemon and select Recycle Servers. Any servers in the connection pool are closed and new servers start with the new configuration.

The General tab is shown in the following figure:

Figure B–4 The General tab

The General tab comprises fields, as listed in the following table:

Table B–4 General Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info</td>
<td></td>
</tr>
</tbody>
</table>
### Table B–4 (Cont.) General Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Workspace name         | The name used to identify the workspace.  
  **Note:** The default configuration includes the default Navigator workspace. This workspace is automatically used if a workspace is not specified as part of the connection settings. |
| Description            | A description of the workspace. |
| Startup script         | The full path name of the script that starts the workspace server processes. The script specified here must always activate the `nav_login` procedure and then run the server program (svc). If you do not specify the directory, the startup procedure is taken from the directory where the daemon resides. Oracle Connect includes a default startup script, which it is recommended to use. |
| Server type            | This field is not applicable for use with OracleAS Adapters for DB2. |
| Workspace binding name | This field is not applicable for use with OracleAS Adapters for DB2. |
| Timeout parameters     | The time the client waits for the workspace server to start. If the workspace server does not start within this period, then the client is notified that the server did not respond. Specifying the timeout here overrides the default setting, specified in the Control section.  
  For more information, see Control for details about the Daemon Control section. |
| Client idle timeout    | The maximum amount of time a workspace client can be idle before the connection with the server is closed. |
| Connect timeout        | The time the client waits for a workspace server to start. If the workspace server does not start within this period, then the client is notified that the server did not respond. |
| Call timeout           | The timeout period for short calls for all daemons. The definition of a short call is a call that should be completed in a few seconds. For example, most calls to a database such as `DESCRIBE` should be completed in a few seconds as opposed to call like a `GETROWS` call, which can take a long time. In heavily loaded or otherwise slow systems, even short calls such as calls to open a file, may take a significant amount of time. If a short call takes more than the specified time to complete, then the connection is stopped. The default value for this parameter is 60 seconds. Values of less than 60 seconds are considered to be 60 seconds.  
  Specifying the timeout in a workspace overrides the value set in this field for that workspace. |
| Logging and Trace Options | |

---

Advanced Tuning of the Daemon  B-9
### Specific log file format

Defines the name and location of the server log file if you want the data written to a file instead of SYSOUT for the server process. The parameter must specify the name and the high level qualifier.

The following tokens can appear in the log file template and are replaced accordingly:

- `%A`: workspace name
- `%D`: date (yyymmdd)
- `%I`: instance number of the given workspace server
- `%L`: server account’s login directory
- `%P`: server’s process ID
- `%T`: time (hhmmss)
- `%U`: server’s account name (user name)

### Trace options

Specifies the type of tracing to be performed. The following tracing options are available:

- **No timeout**: Select this to disable the standard RPC timeouts, setting them to a long duration (approximately an hour) to facilitate debugging.
- **Call trace**: Select this to generate a message in the server log file for each RPC function called. This is useful for troubleshooting the server.
- **RPC trace**: Select this to enable debugging messages on the server.
- **Sockets**: Select this to generate a message in the server log file for each socket operation. This is useful for troubleshooting client/server communication.
- **Extended RPC trace**: Select this to generate a verbose message in the server log file for each low-level RPC function called. This is useful for troubleshooting the server.
- **System trace**: Select this to generate operating system-specific tracing.
- **Timing**: Select this to generate a time stamp for every entry to the server log file.

### Query governing restrictions

#### Max number of rows in a table that can be read

Select the maximum number of table rows that are read in a query. When the number of rows read from a table exceeds the number stated the query returns an error.

#### Max number of rows allowed in a table before scan is rejected

Select the maximum number of table rows that can be scanned. This parameter has different behavior for query optimization and execution.

- For query optimization, the value set is compared to the table cardinality. If the cardinality is greater than the value, the scan strategy is ignored as a possible strategy (unless it is the only available strategy).
- For query execution, a scan is limited to the value set. When the number of rows scanned exceeds the number entered, the query returns an error.

---

### Table B–4 (Cont.) General Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific log file format</td>
<td>Defines the name and location of the server log file if you want the data written to a file instead of SYSOUT for the server process. The parameter must specify the name and the high level qualifier. The following tokens can appear in the log file template and are replaced accordingly:</td>
</tr>
<tr>
<td>Trace options</td>
<td>Specifies the type of tracing to be performed. The following tracing options are available:</td>
</tr>
<tr>
<td>Query governing restrictions</td>
<td>Select the maximum number of table rows that are read in a query. When the number of rows read from a table exceeds the number stated the query returns an error. Select the maximum number of table rows that can be scanned. This parameter has different behavior for query optimization and execution. For query optimization, the value set is compared to the table cardinality. If the cardinality is greater than the value, the scan strategy is ignored as a possible strategy (unless it is the only available strategy). For query execution, a scan is limited to the value set. When the number of rows scanned exceeds the number entered, the query returns an error.</td>
</tr>
</tbody>
</table>
**Server Mode**

The Server Mode tab lets you configure the features that control the operation of the servers started up by the workspace and allocated to clients.

For example, you can configure the workspace to start several servers for future use before any client request, instead of starting each server when a request is received from a client.

Do the following to open the **Server Mode** tab:

1. From the **Start** menu, select **Programs, Oracle**, and then select **Studio**.
2. From the Design Perspective Configuration view, expand the **Machines** folder.
3. Right-click the computer and select **Open Runtime Perspective**.
4. Expand the Daemons node to display the workspaces in the Runtime Explorer.
5. Right-click the workspace and select **Edit Workspace Configuration**.
6. Click the **Server Mode** tab.
7. After making changes to the workspace, right-click the daemon and select **Reload Configuration**.

---

**Notes:**

- You can also change daemon settings using the Configuration Explorer, by selecting a computer and scrolling the list to the required daemon. Right-click the daemon and select **Edit Daemon**.

- Changes made to the daemon configuration are not implemented. They are only implemented after the configuration is reloaded using the **Reload Configuration** option in the Runtime Manager.

---

8. Right-click the daemon and select **Recycle Servers**. Any servers in the connection pool are closed and new servers start with the new configuration.

The **Server Mode** tab is shown in the following figure:
The **Server Mode** tab has the following fields:
Table B–5  Server Mode Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workspace server mode</td>
<td>Specifies the type of new server processes that the daemon starts. The daemon supports the following server modes:</td>
</tr>
<tr>
<td></td>
<td>■ <strong>singleClient</strong>: Each client receives a dedicated server process. The account in which a server process runs is determined either by the client login information or by the specific server workspace. This mode enables servers to run under a particular user account and isolates clients from each other, as each receives its own process. However, this server mode incurs a high overhead due to process startup times and can use a lot of server resources as it requires as many server processes as concurrent clients.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>multiClient</strong>: Clients share a server process and are processed serially. This mode has low overhead because the server processes are initialized. However, because clients share the same process, they can impact one another, especially if they issue lengthy queries. The number of clients that share a process is determined by the Clients per server limit field.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>multiThreaded</strong>: This mode is not applicable for use with OracleAS Adapter for DB2.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>reusable</strong>: An extension of single-client mode. Once the client processing finishes, the server process does not die and can be used by another client, reducing startup times and application startup overhead. This mode does not have the high overhead of single-client mode because the servers are initialized. However, this server mode can use a lot of server resources as it requires as many server processes as concurrent clients.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: The other modes can be set so that the server processes are reusable. The Reuse limit field value controls the number of times a process can be reused.</td>
</tr>
<tr>
<td>Reuse limit</td>
<td>Sets the maximum number of times a particular server can be reused. A one-client server can be reused after its (single) client has disconnected. Reuse of servers enhances startup performance because it is not necessary to repeat initialization. The default for this field is none (0), indicating that server reuse is unlimited. This parameter is disabled only if the server mode value is <strong>singleClient</strong>.</td>
</tr>
<tr>
<td>Clients per server limit</td>
<td>Sets the maximum number of clients a server process for the current workspace accepts. The default for this field is none (0), indicating that the number of clients for each server is unlimited. This field is enabled only if the server mode value is <strong>multiClient</strong> or <strong>multiThreaded</strong>.</td>
</tr>
</tbody>
</table>
Server availability

Specifies the number of servers in a pool of servers, available to be assigned to a client.

The following options are available:

- **Initial number of servers**: The number of server processes that are prestarted for this workspace when the daemon starts. When the number of available server processes drops lower than the value specified in the Minimum number field, the daemon again starts server processes until this number of available server processes is reached. The default for this field is 0.

- **Minimum number**: The minimum number of server processes in the prestarted pool before the daemon resumes creating new server processes (to the value specified in the Initial number of servers field). If this field is set to a value higher than the Initial number of servers field, the daemon uses the value specified in the Initial number of servers field. The default for this field is 0.

- **Keep when daemon ends**: When a daemon is shutdown, all the servers started by that daemon are also stopped, even if they are active. Set this field to true if you want the servers for the workspace to remain active, even after the daemon has been shut down. If this field is set to true, it is the responsibility of the system operator or manager to ensure that the servers are eventually stopped. This must be done at the system level.

- **Set maximum number of servers**: The maximum number of available server processes. Once this number is reached, no new nonactive server processes are created for the particular workspace. For example, if several server processes are released at the same time, so that there are more available server processes than specified by this field, the additional server processes higher than this value are terminated. The default for this field is zero, meaning that there is no maximum.

Port range

Determines the range of ports available for this workspace when starting server processes. Use this option when you want to control the port number, so that Oracle Connect can be accessed through a firewall.

Enter the port range in the following fields:

- **From**: enter the lowest numbered port in the range.
- **To**: Enter the highest numbered port in the range.

Use default port range

Select this to use the port range that is defined in the daemon. This is defined in the Port range for servers field in the daemon Control tab.

Maximum number of server processes

Enter the maximum number of server processes that can run at the same time.

Limit server reuse

Select this to limit the number of servers that can be reused. If this is selected, the Reuse limit parameter is available.
If Limit server reuse is selected, in the field next to the check box, enter the maximum number of times a server can be reused. Select the maximum of clients accepted in a server process.

A one-client server can be reused after its (single) client has disconnected. Reuse of servers enhances startup performance because it is not necessary to repeat initialization.

This parameter is not available if the Limit server reuse parameter is not selected.

This parameter is not available if the server mode value is singleClient.

Limit concurrent clients per server

Select this to limit the number of clients that a server can accept for the current workspace process.

If this is not selected, the number of clients is unlimited.

If Limit concurrent clients per server is selected, in the field next to the check box, enter the maximum number of clients that a server process for the current workspace accepts. The default for this field is None, indicating that the number of clients for each server is unlimited. This field is available if the server mode value is multiClient or multiThreaded.

Specify Server Priority

Set the priority for servers. For example, a workspace for applications with online transaction processing can be assigned a higher priority than a workspace that requires only query processing. The lower the number, the higher the priority. For example, workspaces with a priority of 1 are given a higher priority than workspaces with a priority of 2.

Note: This is unavailable if Use default server priority is selected.

Use default server priority

Sets the priority to 0. There is no specific priority for this workspace. Clear this check box to set a priority in the Specify server priority parameter.

Keep when daemon ends

Select this to stop all servers started by that daemon when a daemon is shutdown, even if they are active. Select this if you want the servers for the workspace to remain active, even after the daemon has been shut down. If selected, it is the responsibility of the system operator or manager to ensure that the servers are eventually stopped. This must be done at the system level.

Server Provisioning

Number of prestarted servers in pool

Initial number of servers: The number of server processes that are prestarted for this workspace when the daemon starts. When the number of available server processes drops lower than the value specified in the Minimum number field, the daemon again starts server processes until this number of available server processes is reached. The default for this field is 0.

### Table B–5 (Cont.) Server Mode Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit concurrent clients per server</td>
<td>Select this to limit the number of clients that a server can accept for the current workspace process. If this is not selected, the number of clients is unlimited. If Limit concurrent clients per server is selected, in the field next to the check box, enter the maximum number of clients that a server process for the current workspace accepts. The default for this field is None, indicating that the number of clients for each server is unlimited. This field is available if the server mode value is multiClient or multiThreaded.</td>
</tr>
<tr>
<td>Specify Server Priority</td>
<td>Set the priority for servers. For example, a workspace for applications with online transaction processing can be assigned a higher priority than a workspace that requires only query processing. The lower the number, the higher the priority. For example, workspaces with a priority of 1 are given a higher priority than workspaces with a priority of 2. Note: This is unavailable if Use default server priority is selected.</td>
</tr>
<tr>
<td>Use default server priority</td>
<td>Sets the priority to 0. There is no specific priority for this workspace. Clear this check box to set a priority in the Specify server priority parameter.</td>
</tr>
<tr>
<td>Keep when daemon ends</td>
<td>Select this to stop all servers started by that daemon when a daemon is shutdown, even if they are active. Select this if you want the servers for the workspace to remain active, even after the daemon has been shut down. If selected, it is the responsibility of the system operator or manager to ensure that the servers are eventually stopped. This must be done at the system level.</td>
</tr>
<tr>
<td>Number of prestarted servers in pool</td>
<td>Initial number of servers: The number of server processes that are prestarted for this workspace when the daemon starts. When the number of available server processes drops lower than the value specified in the Minimum number field, the daemon again starts server processes until this number of available server processes is reached. The default for this field is 0.</td>
</tr>
</tbody>
</table>
The Security tab lets you configure the security level for a workspace. This lets you set the security options for the workspace only. The Security tab is used to:

- Grant administration rights for the workspace
- Determine access to the workspace by a client

Do the following to open the Security tab:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. From the Design Perspective Configuration view, expand the Machines folder.
3. Right-click the computer and select Open Runtime Perspective.
4. Expand the Daemons node to display the workspaces in the Runtime Explorer.
5. Right-click the workspace and select Edit Workspace Configuration.
6. Click the Security tab.
7. After making changes to the workspace, right-click the daemon and select Reload Configuration.

### Table B–5 (Cont.) Server Mode Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of spare servers</td>
<td>The minimum number of server processes in the prestarted pool before the daemon resumes creating new server processes (to the value specified in the Initial number of servers field). If this field is set to a value higher than the Initial number of servers field, the daemon uses the value specified in the Initial number of servers field. The default for this field is 0.</td>
</tr>
<tr>
<td>Prestarted server pool limit</td>
<td>The maximum number of available server processes. Once this number is reached, no new nonactive server processes are created for the particular workspace. For example, if several server processes are released at the same time, so that there are more available server processes than specified by this field, the additional server processes higher than this value are terminated. The default for this field is zero, meaning that there is no maximum.</td>
</tr>
<tr>
<td>Resource limitations</td>
<td></td>
</tr>
<tr>
<td>Number of sub-tasks</td>
<td>The number of sub-tasks for a server that are prestarted for this workspace when the daemon starts. In addition to setting up a pool of server processes as described earlier, you can set additional server processes as sub-tasks by specifying this parameter. If you set 10 servers and 10 prestarted sub-tasks then 100 tasks are started (10 sub-tasks for each process).</td>
</tr>
</tbody>
</table>
Workspaces

Notes:

- You can also change daemon settings using the Configuration Explorer, by selecting a computer and scrolling the list to the required daemon. Right-click the daemon and select **Edit Daemon**.

- Changes made to the daemon configuration are not implemented. They are only implemented after the configuration is reloaded using the **Reload Configuration** option in the Runtime Manager.

8. Right-click the daemon and select **Recycle Servers**. Any servers in the connection pool are closed and new servers start with the new configuration.

The **Security** tab is shown in the following figure:

*Figure B–6  The Security tab*

![Workspace Security](image)

The **Security** tab has the following fields:
### Table B–6  Security Tab

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server Account</strong></td>
<td>This section defines the users (accounts) allowed to access the workspace, firewall access ports, workspace account, and anonymous login permissions.</td>
</tr>
</tbody>
</table>
| **Use specific workspace account** | Select this to define the operating system account used for the workspace.  
If selected, enter the name of the workspace account in the **workspace account** field.  
If not selected, the account name that was provided by the client is used. |
| **Allow anonymous clients to use this workspace** | Select this to allow this workspace to be invoked without authentication.  
If selected, enter the name of the workspace account in the **Server account to use with anonymous clients** field. |
| **Authorized Workspace users** | Indicate which users have permission to use the workspace. Select one of the following:  

  - **All users**: Any user who has logged on to the daemon may use the workspace  
  - **Selected users only**: Select this to allow only users (or accounts) with specific permission to use the workspace.  
    When this is selected, add the names of users (or accounts) and groups that can be use the workspace in the field below.  
    **Note**: If no user is specified, any user who has logged on to the daemon may use the workspace. |
| **Authorized Administrators** | Identifies the users (accounts) with administrator privileges. Select one of the following:  

  - **All users**: Indicates that anyone can access the workspace and change the settings.  
  - **Selected users only**: Select this to allow only users (or accounts) with specific permission to be administrators.  
    When this is selected, add the names of users (or accounts) and groups that can be workspace administrators.  
    If no user is specified, any user who has logged on to the daemon may administrator this workspace. |
The OracleAS CDC Adapter for DB2 has several configuration properties. You can edit the properties in Oracle Studio after Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2. This chapter describes the configuration properties and how to edit them. It contains the following sections:

- Editing Properties in Oracle Studio
- Configuration Properties
- Access to Change Events
- Transaction Support
- Security

**Editing Properties in Oracle Studio**

After you create the OracleAS CDC solution, you can also edit the properties. The solution create two adapters, the adapter and the CDC Queue adapter. The adapter is created on the server computer and the CDC Queue adapter is created on the staging area computer. For more information, see Setting up a Change Data Capture in Oracle Studio.

**Editing the OracleAS CDC Adapter Properties**

To edit the CDC adapter properties, open the Oracle Studio Design perspective and find the binding for the CDC solution on the server computer. Then open the adapter, which contains the name of the CDC solution with the suffix `_ag` added to it. Changes to adapter properties are reset when the CDC solution is redeployed, therefore these changes must be reapplied following solution deployment. For information on deploying a solution, see Deploying a Change Data Capture.

Do the following to edit the CDC adapter properties.

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. Expand the Machines folder.
3. Expand the server machine that you created when Setting up a Change Data Capture in Oracle Studio.
4. Expand the Bindings folder, and then expand the binding the name of the CDC solution with the suffix `_ag` added to it.
5. Expand the Adapter folder.
6. Right-click the adapter the adapter with the name of the solution and the suffix _ag and select Open.
   The adapter configuration editor opens in the editor, which displays the properties for the adapter.

7. Edit the adapter parameters as required.
   For a description of the properties, see OracleAS CDC Adapter Configuration Properties.

Configuration Properties

This section describes the configuration properties for the DB2 CDC adapter. You can edit the properties using Oracle Studio. This section has the following topics:

- OracleAS CDC Adapter Configuration Properties
- Change Router Properties
- Referential Integrity Considerations

OracleAS CDC Adapter Configuration Properties

This section describes the common configuration properties for Oracle CDC adapters and the change router configuration properties, which is located on the staging area computer.

To edit the CDC adapter properties, open the Oracle Studio Design perspective and find the binding for the CDC solution you created. The binding contains the name of the CDC solution with the suffix _ag added to it. Open the adapter with the name of the solution and the suffix _ag to edit the properties. For information on how to edit adapter properties in Oracle Studio, see Editing Properties in Oracle Studio. Changes to adapter properties are reset when the CDC solution is redeployed, therefore these changes must be reapplied following solution deployment.

For a description of the configuration properties, see the following topics:

- OracleAS CDC Adapter for DB2 Properties
- Interaction Properties
- CDC Queue Adapter Properties
- Common CDC Adapter Properties

OracleAS CDC Adapter for DB2 Properties

To edit the properties for the OracleAS CDC Adapter for DB2, open the Oracle Studio Design perspective and find the binding for the CDC solution you created. The binding contains the name of the CDC solution with the suffix _ag added to it. Open the adapter with the name of the solution and the suffix _ag to edit the properties. For more information on how to edit adapter properties, see Editing the OracleAS CDC Adapter Properties.

You can edit the following properties for the OracleAS CDC Adapter for DB2. There are also additional properties that are common to all OracleAS CDC adapters. For a description of the common properties, see Common CDC Adapter Properties.

- commandRecognitionPrefix: The JES (job entry subsystem) command prefix character. The default value is '|'.

C-2 Oracle Application Server CDC Adapters for DB2 User's Guide
Interaction Properties
To edit the interaction properties for the OracleAS CDC Adapter for DB2, open the Oracle Studio Design perspective and find the binding for the CDC solution you created. The binding contains the name of the CDC solution with the suffix _ag added to it. Right-click the adapter with the name of the solution and the suffix _ag and select Show in metadata. In the Metadata view, expand the Interactions folder and then open the interaction you want to edit. For more information on how to edit an interaction in Oracle Studio, see Adapter Metadata Interactions.

You can edit the following properties for DB2 CDC agent interactions:
- **bootStrapDataset name**: This data set is used to keep track of the DB2 logs.

CDC Queue Adapter Properties
The CDC Queue adapter is a database adapter that is found in the staging area. It is created automatically when Setting up a Change Data Capture in Oracle Studio. It has three properties that can be viewed in the adapter’s editor in Oracle Studio:
- **connectString**
- **defaultDatasource**
- **multipleResults**

To ensure that the queue adapter works properly with your CDC solution, do not change the values for these properties.

The CDC Queue adapter may have additional properties that can be viewed in the adapter’s XML schema. For information on how to view the XML, see Adapter Metadata XML.

The following property, which is viewed in the XML only can be edited:
- **maxRecords**: The maximum number of records that can be returned.

For information on how to edit XML records in Oracle Studio, see Editing XML Files in Oracle Studio.

Common CDC Adapter Properties
The following table describes the common configuration properties for Oracle CDC adapters. For information on specific CDC adapter properties for the OracleAS CDC Adapter for DB2, see OracleAS CDC Adapter for DB2 Properties.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>datasource</strong></td>
<td>string</td>
<td>The name of the data source for the OracleAS CDC adapter.</td>
<td></td>
</tr>
<tr>
<td><strong>routers</strong></td>
<td></td>
<td>A list of users who can connect to the adapter and get change events from it for processing. If no routers are specified, any valid user for the workspace can get change events from the Oracle Connect adapter. To add the list of users in Oracle Studio, expand the router property then right-click users. A new entry called Item(#) is added to the Property column. In the Value column, enter the User Name for this router.</td>
<td></td>
</tr>
</tbody>
</table>
The following table describes the SQL-based change event router configuration parameters.

To edit the router properties, open the Oracle Studio Design perspective and find the binding for the CDC solution you created. The binding contains the name of the CDC solution with the suffix \_router added to it. Open the adapter with the name of the solution and the suffix \_router to edit the properties. For information on how to edit adapter properties in Oracle Studio, see Editing Properties in Oracle Studio.

### Table C-2 Change Router Configuration Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cdcDatasource</td>
<td>string</td>
<td></td>
<td>The OracleAS Change Data Source.</td>
</tr>
</tbody>
</table>
### Table C–2 (Cont.) Change Router Configuration Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>eliminateUncommittedChanges</td>
<td>Boolean</td>
<td>false</td>
<td>When set to true, only committed change records are moved to the Change Table. If false, all change records are moved to the change tables (in which case, memory usage is minimal) hence the change table may contain rolled back data. For most adapters, following the RI considerations results in rolled-back changes eliminated naturally by compensating change records generated by the adapter in a rollback. Consult the respective CDC adapter documentation for details. For more information, see Referential Integrity Considerations.</td>
</tr>
<tr>
<td>eventExpirationHours</td>
<td>int</td>
<td>48</td>
<td>Indicates how long change records are kept in change tables within the staging area. After the indicated time, change records are deleted. You can set a value between 0 and 50000. A value of 0 means that the records are never deleted. A value of 1 indicates that the records are kept for one hour.</td>
</tr>
<tr>
<td>logLevel</td>
<td>enum</td>
<td></td>
<td>The logging level for the router. The following are the available log levels:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- none</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- api</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- internalCalls</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- debug.</td>
</tr>
<tr>
<td>maxDeletedEventsInBatch</td>
<td>int</td>
<td>500</td>
<td>Controls how many expired change records are deleted in a single pass. This number may need to be lowered in some rare cases to reduce latency when a large number of change events is continuously being received.</td>
</tr>
<tr>
<td>maxOpenfiles</td>
<td>int</td>
<td>200</td>
<td>Controls the number of physical files opened by the router.</td>
</tr>
<tr>
<td>maxTransactionMemory</td>
<td>int</td>
<td>1000</td>
<td>Specifies how much memory can be stored in memory per transaction before it is off-loaded to disk. This number should be higher than the average transaction size so that the slower-than-memory disk is not used too often.</td>
</tr>
<tr>
<td></td>
<td>(in Kb)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>maxStagingMemory</td>
<td>int</td>
<td>10000</td>
<td>Specifies how much memory in total can be used for storing active transactions (ones that have not yet committed or rolled back).</td>
</tr>
<tr>
<td></td>
<td>(in Kb)</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
### Table C-2 (Cont.) Change Router Configuration Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sourceEventQueue</td>
<td>Structure</td>
<td></td>
<td>Connection information to the OracleAS CDC adapter.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ server</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ workspace</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ adapter</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ eventWait</td>
<td>int</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>■ maxEventsAsBlocks</td>
<td>int</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>■ reconnectWait</td>
<td>int</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>■ fixedNat</td>
<td>boolean</td>
<td>false</td>
<td></td>
</tr>
<tr>
<td>stagingDirectory</td>
<td>string</td>
<td></td>
<td>Specifies the directory where the staging area change files are be stored.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This directory also stores off-loaded transactions, timed-out transactions,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and error files.</td>
</tr>
<tr>
<td>transactionTimeout</td>
<td>int (in</td>
<td>3600</td>
<td>Specifies how long can a transaction be active without getting new events.</td>
</tr>
<tr>
<td></td>
<td>seconds)</td>
<td></td>
<td>This parameter should be set according to the corresponding setting of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>captured database. In particular, this setting must not be lower than the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>database’s transaction time-out setting as this may lead to the loss of</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>transactions.</td>
</tr>
<tr>
<td>useType</td>
<td>enum</td>
<td>sqlBbas ed Cdc</td>
<td>This parameter must be set to this value.</td>
</tr>
<tr>
<td>routers</td>
<td></td>
<td></td>
<td>A list of users who can connect to the change event router and get change</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>events from it for processing. If no routers are specified, any valid</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Oracle Connect user for the workspace may get change events from the adapter.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To add the list of users in Oracle Studio, expand the router property then</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>right-click users. A new entry called Item(#) is added to the Property column.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In the Value column, enter the User Name for this router.</td>
</tr>
<tr>
<td>senders</td>
<td></td>
<td></td>
<td>A list of users who can connect to the change event router and send change</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>events to it for processing. If no routers are specified, any valid Oracle</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Connect user for the workspace may get change events from the adapter.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To add the list of users in Oracle Studio, expand the sender property then</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>right-click users. A new entry called Item(#) is added to the Property column.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In the Value column, enter the User Name for this router.</td>
</tr>
</tbody>
</table>
Referential Integrity Considerations

Some related tables have referential integrity (RI) constraints enforced on them. For example, with `OrderHeader` and `OrderLines` one cannot have `OrderLines` without an associated `OrderHeader`.

When processing change events by the table (which is how an OracleAS CDC works) as opposed to by transaction, referential integrity cannot be maintained properly. For example, when first handling all `OrderHeader` records and then all `OrderLines` records then a deleted `OrderHeader` may be applied long before the required delete of the associated `OrderLines` records.

In order to reduce the potential referential integrity to a known time frame after which referential integrity is restored, a somewhat different process is needed (compared with Reading the Change Tables’).

A special `SYNC_POINTS` table should be added to maintain a common sync-point for use with multiple related tables. The table is defined as follows:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>application_name</code></td>
<td>string (64)</td>
<td>The application for which the processing is done.</td>
</tr>
<tr>
<td><code>table_name</code></td>
<td>string (64)</td>
<td>The name of the synchronization point</td>
</tr>
<tr>
<td><code>context</code></td>
<td>String (32)</td>
<td>A stream position that can be safely used as an upper bound for event retrieval of all related tables</td>
</tr>
</tbody>
</table>

This table’s primary unique key is the concatenation of `application_name` + `sync_name`. The use of this table is not mandatory but it is part of the recommended use pattern of SQL-based CDC.
The SYNC_POINTS table is created with the following definition (where filename is changed into an actual path):

```xml
<?xml version='1.0' encoding='UTF-8'?>
<navobj>
  <table name='SYNC_POINTS'
    fileName='<staging-directory-path>SYNC_POINTS' organization='index'>
    <fields>
      <field name='application_name' datatype='string' size='64'/>
      <field name='sync_name' datatype='string' size='64'/>
      <field name='context' datatype='string' size='32'/>
    </fields>
    <keys>
      <key name='Key0' size='128' unique='true'>
        <segments>
          <segment name='application_name'/>
          <segment name='sync_name'/>
        </segments>
      </key>
    </keys>
  </table>
</navobj>
```

The following procedure describes how to ensure RI is regained after a group of ETL rounds. It is an extension of the procedure described earlier for consuming change records. Here we assume that tables T1, T2 and T3 are related with RI constraints and that A is the application we are working under.

**To create a stream position**

1. This is a one-time setup step aimed to create a stream position record for T [1/2/3] + A in the STREAM_POSITIONS table. The following SQL statement creates that record:

   ```sql
   insert into STREAM_POSITIONS values ('A', 'T1', '');
   insert into STREAM_POSITIONS values ('A', 'T2', '');
   insert into STREAMPOSITIONS values ('A', 'T3', '');
   ```

2. This step is performed at the beginning of a group of ETL rounds processing (that is before starting to process change events for T1, T2 and T3). The goal here is to get a shared sync point for retrieval of T1, T2 and T3. This is done by sampling the 'context' column of the SERVICE_CONTEXT table. This value is the stream position of the last change record in the most recently committed transaction. This is done as follows:

   ```sql
   insert into SYNC_POINTS
     select 'A' application_name, 'T123' sync_name, context from SERVICE_CONTEXT;
   ```

   Here, T123 is the name chosen for the synchronization points of tables T1, T2, and T3.

3. This step is where change data is actually read. It occurs on each ETL round.

   ```sql
   select n.* from T t, STREAM_POSITIONS sp, SYNC_POINTS sy where
   sp.application_name = 'A' and
   sp.table_name = 'T' and
   sy.application_name = sp.application_name and
   sy.sync_name = 'T123' and
   n.context > sp.context and n.context <= sy.context order by n.context;
   ```
Note that “n.context <= sy.context” is used because the context represents a change record to be processed and processing should include the change record associated with sy.context, too.

This query retrieves change records starting from just after the last handled change record but stopping at a common sync point. “n.*” can be replaced with an explicit list of columns, however it is important that the ‘context’ column must be selected as this is the change record stream position which is required for the next step.

This step occurs after each ETL round when all change records were retrieved and processed for a table Ti. Let’s assume that the value of the ‘context’ column of the last change record was ‘C’. This value must be stored back into the STREAM_POSITION table for the next ETL round. This is done with:

update STREAM_POSITIONS set context=’C’ where application_name = ‘A’ and table_name = ‘Ti’;

This value can be stored more frequently during the ETL process as needed. The general guideline is that when change record data is committed to the target database, the stream position should be updated as well.

### Access to Change Events

Changes are captured and maintained in a change table. The table contains the original table columns and CDC header columns. The header columns are described in the following table:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| context     | The change record stream position in the staging area. The column is defined as primary unique index. It is a 32-bytes string with the following structure:
|             | <yyyymmdd>T<hhmmss>.<nnnn><cccccc> |

Where:
- <yyyymmdd>T<hhmmss> is the commit processing timestamp as generated in the staging area when starting to process the Commit event.
- <nnnn> is a unique number to differentiate between transactions committed during the same second (up to 99,999 are assumed).
- <cccccc> is a counter for the change events in the transaction making every stream position unique (up to 9,999,999 are assumed).

<table>
<thead>
<tr>
<th>operation</th>
<th>This column lists the operations available for the CDC adapter. The available operations are:</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSERT</td>
<td>INSERT</td>
</tr>
<tr>
<td>DELETE</td>
<td>DELETE</td>
</tr>
<tr>
<td>BEGIN</td>
<td>BEGIN</td>
</tr>
<tr>
<td>UPDATE</td>
<td>UPDATE</td>
</tr>
<tr>
<td>BEFOREIMAGE</td>
<td>BEFOREIMAGE</td>
</tr>
<tr>
<td>COMMIT</td>
<td>COMMIT</td>
</tr>
<tr>
<td>ROLLBACK</td>
<td>ROLLBACK</td>
</tr>
</tbody>
</table>
The OracleAS CDC Adapter for DB2 supports transactions.

Security

To work with Oracle Connect and DB2, the following requirements must be met:

- All the libraries in the ATTSRVR STEPLIB must be APF-Authorized.
- Provide the following grants to the owner of the ATTSRVR started task.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRANT TRACE</td>
<td>This command grants the -start trace() privilege.</td>
</tr>
<tr>
<td>GRANT MONITOR2</td>
<td>This command grants the READA and READS IFI requests privilege.</td>
</tr>
</tbody>
</table>

- The owner of the ATTSRVR started task must have privileges in DB2 to run offline.
  
  For details about setting DB2 security to use DB2 with Oracle Connect, see Oracle Application Server CDC Adapters Installation Guide.
Back-end Adapter Data Type Support

The OracleAS CDC Adapter for DB2 supports several data types that are used when defining metadata in Oracle Studio. The data types are mapped from the COBOL data types during the import procedure.

**Note:** The mapping of data types between the OracleAS CDC Adapter for DB2 and Oracle Application Server is performed internally by Oracle Connect.

## Data Type Mapping

The COBOL data type `COMP`, in the table is an abbreviation for, and synonymous with, COMPUTATIONAL. Square brackets ([]) denote optional qualifiers for some COBOL compilers, which may not be allowed for other COBOL compilers.

### Table D–1 Data Type Mapping: COBOL and Oracle Connect Back-end Adapter

<table>
<thead>
<tr>
<th>COBOL Data Type</th>
<th>Oracle Connect Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>BINARY (with fractional data)</td>
<td>string</td>
</tr>
<tr>
<td>BINARY (without fractional data)</td>
<td>int</td>
</tr>
<tr>
<td>COMP (with fractional data)</td>
<td>string</td>
</tr>
<tr>
<td>COMP (without fractional data)</td>
<td>int</td>
</tr>
<tr>
<td>COMP-2</td>
<td>double</td>
</tr>
<tr>
<td>COMP-3</td>
<td>string</td>
</tr>
<tr>
<td>COMP-4 (with fractional data)</td>
<td>string</td>
</tr>
<tr>
<td>COMP-4 (without fractional data)</td>
<td>int</td>
</tr>
<tr>
<td>COMP-5 (with fractional data)</td>
<td>string</td>
</tr>
<tr>
<td>COMP-5 (without fractional data)</td>
<td>int</td>
</tr>
<tr>
<td>COMP-X (with fractional data)</td>
<td>string</td>
</tr>
<tr>
<td>COMP-X (without fractional data)</td>
<td>int</td>
</tr>
<tr>
<td>INDEX</td>
<td>int</td>
</tr>
<tr>
<td>[SIGN [IS]] LEADING</td>
<td>string</td>
</tr>
<tr>
<td>[SIGN [IS]] LEADING SEPARATE [CHARACTER]</td>
<td>string</td>
</tr>
<tr>
<td>NATIVE-2</td>
<td>int</td>
</tr>
<tr>
<td>COBOL Data Type</td>
<td>Oracle Connect Data Type</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>NATIVE-4</td>
<td>int</td>
</tr>
<tr>
<td>NATIVE-8</td>
<td>string</td>
</tr>
<tr>
<td>PACKED-DECIMAL</td>
<td>string</td>
</tr>
<tr>
<td>POINTER</td>
<td>int</td>
</tr>
<tr>
<td>[SIGN [IS]] TRAILING</td>
<td>string</td>
</tr>
<tr>
<td>[SIGN [IS]] TRAILING SEPARATE</td>
<td>string</td>
</tr>
<tr>
<td>[CHARACTER]</td>
<td></td>
</tr>
</tbody>
</table>
The OracleAS CDC Adapter for DB2 provides the globalization support for the following languages:

- Arabic
- English (the default)
- French
- German
- Greek
- Hebrew
- Italian
- Japanese
- Korean
- Portuguese
- Simple Chinese
- Spanish
- Traditional Chinese
- Turkish

This appendix describes how to define the language support.

**Defining the Language and Code Page**

The language and code page parameters are accessed from the computer where Oracle Studio is installed.

Perform the following steps to define the required language and code page:

1. From the Start menu, select Programs, Oracle, and then select Studio.
2. In the Design perspective Configuration view, expand the Machines folder.
3. Expand the machine for which you want to set the language.
4. Expand the Bindings folder and right-click the NAV binding.
5. Select Open.
6. Expand the Language Settings and do the following:
Defining the Language and Code Page

- From the **Language** list, select the NLS supported language to use in this binding. Valid values are listed in the Language Name column of the NLS Language Codes table.

- From the **Codepage** list, select the code page to use with this language. The code pages available are determined by the Language that is selected. If you have additional code pages available, you can manually enter them in this field.

  **Note**: If you change the language, the code page also changes. Ensure that you want to use the selected code page with the language you selected.

  If no code page is selected, the default code page for the selected language is used.

- From the **NLS string** list, select the NLS string for this language and code page. The NLS strings available are determined by the code page that is selected. If you have additional NLS strings available, you can manually enter them in this field.

  The code page is used by a field with a data type defined as `nlsString`. This parameter is used for a field with a code page that is different than the computer’s code page. This property includes values for the name of the code page and whether the character set reads from right to left (as in middle-eastern character sets).

  For example, the following specifies a Japanese EUC 16-bit code page:

  `<misc nlsString="JA16EUC,false"/>

7. Save the change. New servers are used for the language selected.

The following table lists the code pages:

**Table E–1 NLS Language Codes**

<table>
<thead>
<tr>
<th>Language Name</th>
<th>Language Code</th>
<th>Windows Default</th>
<th>ASCII Platforms (Default)</th>
<th>EBCDIC Platforms (Default)</th>
<th>Alternative Code pages (EBCDIC based unless noted otherwise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language Name</td>
<td>Language Code</td>
<td>Windows Default</td>
<td>ASCII Platforms (Default)</td>
<td>EBCDIC Platforms (Default)</td>
<td>Alternative Code pages (EBCDIC based unless noted otherwise)</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------</td>
<td>-----------------</td>
<td>---------------------------</td>
<td>----------------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>French</td>
<td>FRE</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1147</td>
<td>IBM297, IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Latin International</td>
<td>LAT</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1148</td>
<td>IBM500, IBM037, IBM1140, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Spanish</td>
<td>SPA</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1145</td>
<td>IBM284, IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>German</td>
<td>GER</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1141</td>
<td>IBM273, IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Portuguese</td>
<td>POR</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1140</td>
<td>IBM037, IBM500, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Italian</td>
<td>ITL</td>
<td>Windows-125 2</td>
<td>ISO-8859-15</td>
<td>IBM1144</td>
<td>IBM280, IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Greek</td>
<td>GRK</td>
<td>Windows-125 3</td>
<td>ISO-8859-7</td>
<td>IBM875</td>
<td>-</td>
</tr>
<tr>
<td>Russian(^1)</td>
<td>RUS</td>
<td>Windows-125 1</td>
<td>ISO-8859-5</td>
<td>IBM1154</td>
<td>IBM1025</td>
</tr>
</tbody>
</table>
Table E–1  (Cont.) NLS Language Codes

<table>
<thead>
<tr>
<th>Language Name</th>
<th>Language Code</th>
<th>Windows Default</th>
<th>ASCII Platforms (Default)</th>
<th>EBCDIC Platforms (Default)</th>
<th>Alternative Code pages (EBCDIC based unless noted otherwise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turkish(^2)</td>
<td>TUR</td>
<td>Windows-125 4</td>
<td>ISO-8859-9</td>
<td>IBM1155</td>
<td>IBM1026</td>
</tr>
<tr>
<td>Hebrew</td>
<td>HEB</td>
<td>Windows-125 5</td>
<td>ISO-8859-8</td>
<td>IBM424</td>
<td>IBM 862</td>
</tr>
<tr>
<td>Arabic</td>
<td>ARA</td>
<td>Windows-125 6</td>
<td>ISO-8859-6</td>
<td>IBM420</td>
<td></td>
</tr>
<tr>
<td>Chinese - Simplified</td>
<td>SCHI</td>
<td>GBK</td>
<td>GBK</td>
<td>IBM935</td>
<td>-</td>
</tr>
<tr>
<td>Chinese - Traditional</td>
<td>TCHI</td>
<td>BIG5</td>
<td>BIG5</td>
<td>IBM937</td>
<td>-</td>
</tr>
<tr>
<td>Korean</td>
<td>KOR</td>
<td>MS949</td>
<td>EUC-KR</td>
<td>IBM933</td>
<td>MS949</td>
</tr>
</tbody>
</table>

1 Russian users who use ANSI 1251 Cyrillic as their Windows code page must edit the RUS.TXT file and compile it to RUS.CP using the NAV_UTIL CODEPAGE.

2 To work with solutions in Oracle Studio, when using Turkish, add the -nl en switch to the Target path in the Oracle Studio shortcut properties. For example: "C:\Program Files\Oracle\Studiol\studio.exe -nl en"
In many cases you must manually edit the metadata to configure parts of a solution or composition. Metadata is created in XML format. You define aspects of a solution by changing the values of the elements and attributes of the XML files that belong to the solution. Oracle Studio provides a graphical interface where you can define the various aspects of a solution. This interface lets you make changes easily without having to manually edit the XML file.

Preparing to Edit XML Files in Oracle Studio

You can edit XML files for the following items in Oracle Studio:

- Machines
- Bindings.
- Daemons
- Users

When you open an XML file, a graphical representation of the file is opened in the editor. The editor displays the elements and attributes in the file in the first column and their corresponding values in the second column. Each entry has an icon that indicates whether the entry is an element or an attribute. Click the Source tab to view the file in its native format. The following figure is an example of the editor’s view of an XML file.

Figure F–1  XML Graphical Display
Do the following to edit an XML file in Oracle Studio

1. In the Design perspective, open the Navigator view.
2. In the Navigator view, find the item with the XML file to edit. This can be a machine, binding, daemon, or user.
3. Right-click the item and select Open as XML. A graphical list of the file’s elements and attributes opens in the editor.
4. Find the element or attribute (property) to change.
5. Click in the right column next to the property you are changing and edit or add the value.
6. Save the file, then select it again in the Project Explorer and press F5 to refresh. The XML file is updated automatically.

**Making Changes to the XML File**

You can also make the following changes to XML files in Oracle Studio:

- Remove Objects
- Add DTD Information
- Edit Namespaces
- Add Elements and Attributes
- Replace an Element

**Remove Objects**

You can delete an element, attribute, or other object from the XML file.

Do the following to remove an object

1. Right-click an object from the list in the editor.
2. Select Remove.

**Add DTD Information**

You can add DTD information to an element or attribute.

Do the following to add DTD Information

1. Right-click an element or attribute and select Add DTD Information. The Add DTD Information dialog box opens.
2. Enter the information requested in the dialog box. The following table describes the Add DTD Information dialog box.

### Table F–1 Add DTD Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root element name</td>
<td>The name of the XML root element.</td>
</tr>
<tr>
<td>Public ID</td>
<td>The value in this field is the Public Identifier. It is used to associate the XML file (using an XML catalog entry) with a DTD file by providing a hint to the XML processor. Click <strong>Browse</strong> to select an XML catalog entry from a list. An XML Catalog entry contains two parts, a Key (which represents a DTD or XML schema) and a URI (which contains information about a DTD or XML schema's location). Select the catalog entry you want to associate with your XML file.</td>
</tr>
</tbody>
</table>
| System ID    | The value in this field is the DTD the XML file is associated with. You can change the DTD the file is associated with by editing this field. The XML processor tries to use the Public ID to locate the DTD, and if this fails, it uses the System ID to find it. Click **Browse** to select a system ID. You can this in two ways:  
  - Select the file from the workbench. In this case, update the file with the import dialog box.  
  - Select an XML catalog entry. |

3. Save the file, then select it again in the Project Explorer and press F5 to refresh. The XML file is updated automatically.

**Edit Namespaces**

You can make changes to the namespaces associated with an element or attribute.

Do the following to edit namespaces:

1. Right-click an element or attribute and select **Edit namespaces**. The Edit Schema Information dialog box opens.
2. Click a button to make any changes to this information.

Do the following to add a new namespace:

1. From the Schema Information dialog box, click **Add**.
2. The Add Namespace Definitions dialog box opens. Select one of the following:
   - **Select from registered namespaces**. This selection is available when the dialog box opens. Select from the list of registered namespaces and then click OK. If no registered namespaces are available, the list is empty.
   - **Specify new namespace**. Enter the information described in the following table:

### Table F–2 New Namespace

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefix</td>
<td>The prefix is added to all qualified elements and attributes in the XML file.</td>
</tr>
<tr>
<td>Namespace Name</td>
<td>The namespace of the XML file.</td>
</tr>
<tr>
<td>Location Hint</td>
<td>The location of the XML schema of the XML file. An XML Catalog ID or a URI can be entered in this field. Click <strong>Browse</strong> to search for the schema you want. You can this in two ways: Select the schema from the workbench. In this case, update the with the import dialog box. Select an XML catalog entry. The <strong>Namespace Name</strong> and <strong>Prefix</strong> fields are be filled with the appropriate values from the schema (you must leave the fields blank for this to occur).</td>
</tr>
</tbody>
</table>

**Note**: If you are creating an XML file from an XML schema, you cannot change the Namespace Name or Location Hint values.

To edit a namespace

1. From the Schema Information dialog box, click **Edit**.
2. Enter the information in the fields.
Add Elements and Attributes

You can add additional elements and attributes to the XML file.

Do the following to add Elements and Attributes

1. Right-click an element.

2. Select one of the following:
   - **Add Attribute** to add an attribute under the selected element.
   - **Add Child** to add another element under the selected element
   - **Add Before** to add another element above the selected element
   - **Add After** to add another element below the selected element

3. Provide a name for the element or attribute if required. You may also select the element from a submenu. The element or attribute is added to the file.

4. Save the file, then select it again in the Project Explorer and press F5 to refresh. The XML file is updated automatically.

Replace an Element

You can replace an element with another legal element.

Do the following to replace an element

1. Right-click an element from the list in the editor.

2. Select **Replace with**.

3. Select an element from the submenu. Only legal elements are available.

4. The original element is replaced with the selected element.
ACX trace parameter, 7-7
adapter inbound configuration, 3-1
add timestamp to traced events parameter, 7-7
Arithmetic fixed precision parameter, 7-12

binary XML log level parameter, 7-7
binding file
  configuration, 3-4

cache
  maximum queries, 7-11
cache buffer size parameter, 7-8
CDC
  stream positions table, 3-1
CDC adapter
  connection factory, 3-2
CDC stream positions table, 3-1
code page parameter, 7-9
COM maximum XML size in memory parameter, 7-14
COM maximum XML size parameter, 7-14
COM XML transport buffer parameter, 7-14
commit on destroy, 7-13
configuring the binding file, 3-4
connection factory, 3-2
continuous query
  prefix, 7-11
  retry interval, 7-11
  timeout, 7-11
continuous query prefix parameter, 7-11
continuous query retry interval parameter, 7-11
continuous query timeout parameter, 7-11
creating JCA configuration files
  JCA configuration files
    creating, 3-3

daemon
  logging, B-3
  security, B-5
  server modes, 7-2
  shutting down, 6-8
starting, 6-8
timeout, 6-12
data sources, disabling caching, 7-10
data types
atomic metadata, D-1
NLS string parameter, 7-9
ddebug parameters
ACX trace, 7-7
add timestamp to traced events, 7-7
environment, 7-7
GDB trace, 7-7
general trace, 7-7
lbinary XML log level, 7-7
log file, 7-7
optimizer trace, 7-7
query processor trace, 7-7
query warnings, 7-7
trace directory, 7-7
Transaction extended logging, 7-8
disable 2PC, 7-13
disable command reuse parameter, 7-10
disable compilation cache parameter, 7-11
disable DS property cache parameter, 7-10
disable insert parameterization parameter, 7-10
disable metadata caching parameter, 7-10
disable query parameterization parameter, 7-10
disable row mark fail fetch parameter, 7-10
disable SQS cache parameter, 7-12
disable threads parameter, 7-10
disabling
parameterization of constants, 7-10
parameterization of queries, 7-10
dsm maximum buffer size parameter, 7-13
dsm middle buffer size parameter, 7-14
dsmMaxSortBufferSize parameter, 7-13

E
environment parameters
ACX trace, 7-7
add timestamp to traced events, 7-7
Artithmetic fixed precision, 7-12
cache buffer size, 7-8
code page, 7-9
COM maximum XML size, 7-14
COM maximum XML size in memory, 7-14
COM XML transport buffer, 7-14
commit on destroy, 7-13
continuous query prefix, 7-11
continuous query retry interval, 7-11
continuous query timeout, 7-11
debug, 7-7
disable 2PC, 7-13
disable command reuse, 7-10
disable compilation cache, 7-11
disable DS property cache, 7-10
disable insert parameterization, 7-10
disable metadata caching, 7-10
disable query parameterization, 7-10
disable row mark fail fetch, 7-10
disable SQS cache, 7-12
disable threads, 7-10
dsm maximum buffer size, 7-13
dsm maximum sort buffer size, 7-13
dsm middle buffer size, 7-14
expose XML fields, 7-12
file close on transaction, 7-14
file pool size, 7-14
file pool size per file, 7-14
first tree extensions, 7-11
GDB trace, 7-7
general trace, 7-7
hash buffer size, 7-14
hash enable RO, 7-14
hash max open files, 7-14
hash primary event size, 7-14
hash secondary event size, 7-14
ignore segment bind failure, 7-11
insert from select commit rate, 7-12
language, 7-9
log file, 7-7
maximum columns in parsing, 7-11
Maximum segmented database thread, 7-11
maximum SQL cache, 7-11
minimum number of parameters allocated, 7-11
miscellaneous, 7-8
nav utility editor, 7-8
NLS string, 7-9
odbc, 7-9
oledb, 7-9
optimistic for updates, 7-11
optimizer, 7-9
optimizer goal, 7-9
optimizer trace, 7-7
parser depth, 7-12
procedures cache size, 7-12
prompt database user password, 7-11
query processor trace, 7-7
query warnings, 7-7
queryProcessor, 7-10
recovery delay, 7-13
Replace invalid XML characters, 7-15
time limit, 7-13
token size, 7-12
trace directory, 7-7
transaction conversions parameters, 7-13
Transaction extended logging, 7-8
transaction extended logging, 7-13
transaction log file, 7-13
transactions, 7-13
temporary directory, 7-8
tuning, 7-13
use alternate qualifier, 7-11
use global file pool, 7-14
use table filter expression, 7-11
user commit confirm table, 7-13
write empty string as null, 7-11
XML, 7-14
XML date format parameter, 7-15
XML trim char column, 7-15
year 2000 policy, 7-8
error log, binary XML log level parameter, 7-7
error log, log file parameter, 7-7
exact arithmetic, 7-12
expose XML fields parameter, 7-12

F
file close on transaction, 7-14
file pool size, 7-14
file pool size per file, 7-14
first row optimization, optimizer goal parameter, 7-9
first tree extensions parameter, 7-11
floating point precision, 7-12

G
GDB trace parameter, 7-7
general parameters
cache buffer size, 7-8
nav utility editor, 7-8
temporary directory, 7-8
year 2000 policy, 7-8
general trace parameter, 7-7

H
hash buffer size, 7-14
hash buffer size parameter, 7-14
hash enable RO parameter, 7-14
hash joins, 7-14
hash max open files parameter, 7-14
hash primary event size parameter, 7-14
hash secondary event size parameter, 7-14

I
ignore segment bind failure parameter, 7-11
index cache
dsm middle buffer size parameter, 7-14
insert from select commit rate parameter, 7-12

J
JCA configuration files, 3-3
configuration, 3-3
joins
hash buffer size parameter, 7-14
hash max open files parameter, 7-14

L
language parameter, 7-9
language parameters
code page, 7-9
environment, 7-9
language, 7-9
NLS string, 7-9
locking, optimistic, 7-11
log file parameter, 7-11
log files
binary XML log level parameter, 7-7
daemon options, B-3
log file parameter, 7-7
logging
daemon configurations, B-3
optimizer strategy, 7-7, 7-8

M
maximum columns in parsing parameter, 7-11
Maximum segmented database thread, 7-11
maximum SQL cache parameter, 7-11
memory
dsm maximum buffer size parameter, 7-13
dsm maximum sort buffer size parameter, 7-13
dsm middle buffer size parameter, 7-14
metadata
atomic data types, D-1
disabling retrieval from cache, 7-10
minimum number of parameters allocated parameter, 7-11
miscellaneous parameters
environment, 7-8

N
nav utility editor parameter, 7-8
nav utility editor, text editor, 7-8
NLS string parameter, 7-9

O
odbc environment parameters, 7-9
oledb environment parameters, 7-9
optimistic for updates parameter, 7-11
optimizer
environment parameters, 7-9
optimizer goal parameter, 7-9
trace directory parameter, 7-7
optimizer goal parameter, 7-9
optimizer trace parameter, 7-7

P
parallel processing parameters
disable threads, 7-10
parameterization of constants, disabling, 7-10
parameterization of queries, disabling, 7-10
parser depth parameter, 7-12
password, setting automatic prompt, 7-11
procedures cache size parameter, 7-12
prompt database user password parameter, 7-11

Q
queries
disabling reuse, 7-10
disabling saved compilation, 7-11
enabling retrieval after failure, 7-10
maximum cached in memory, 7-11
maximum length of string, 7-12
maximum size after compilation, 7-11
query optimizer
logging strategy, 7-7, 7-8
optimizer goal parameter, 7-9
trace directory parameter, 7-7
query processor parameters
Arithmetic fixed precision, 7-12
continuous query prefix, 7-11
continuous query retry interval, 7-11
continuous query timeout, 7-11
disable command reuse, 7-10
disable compilation cache, 7-11
disable DS property cache, 7-10
disable insert parameterization, 7-10
disable metadata caching, 7-10
disable query parameterization, 7-10
disable row mark fail fetch, 7-10
disable SQS cache, 7-12
emptyStringIsNull parameter, 7-11
expose XML fields, 7-12
first tree extensions, 7-11
ignore segment bind failure, 7-11
insert from select commit rate, 7-12
maximum columns in parsing, 7-11
Maximum segmented database thread parameter, 7-11
maximum SQL cache, 7-11
minimum number of parameters allocated, 7-11
optimistic for updates, 7-11
parser depth, 7-12
procedures cache size, 7-12
prompt database user password, 7-11
token size, 7-12
use alternate qualifier, 7-11
use table filter expression, 7-11
query processor trace parameter, 7-7
query warnings parameter, 7-7
queryProcessor environment parameters, 7-10

R
recovery delay, 7-13
Replace invalid XML characters parameter, 7-15
rollback loops
  troubleshooting, 3-6
row optimization, optimizer goal parameter, 7-9

S
security
  daemon configurations, B-5
  setting automatic prompt, 7-11
segmented data sources, response upon failure, 7-11
servers
  configuring modes, 7-2
  reusable, 7-3
  Reuse limit daemon parameter, 7-3, B-13
sort buffer, dsm maximum sort buffer size parameter, 7-13
staging area, 1-4
stored queries, maximum in cache memory, 7-12
stream positions table, 3-1

T
temporary directory parameter, 7-8
temporary files, 7-8
threading model, 7-11
time limit parameter, 7-13
timeout
  client idle, 6-12
daemon, 6-12
token size parameter, 7-12
trace directory parameter, 7-7
trace information, logging, 7-7
transaction conversions parameters, 7-13
transaction extended logging, 7-13
Transaction extended logging parameter, 7-8
transaction log file, 7-13
transactions environment parameters, 7-13
transactions parameters
  commit on destroy, 7-13
  disable 2PC, 7-13
  recovery delay, 7-13
  time limit, 7-13
  transaction conversions parameters, 7-13
  transaction extended logging, 7-13
  transaction log file, 7-13
  user commit confirm table, 7-13
tuning environment parameters, 7-13
tuning parameters
  dsm maximum buffer size, 7-13
  dsm maximum sort buffer size, 7-13
  dsm middle buffer size, 7-14
  file close on transaction, 7-14
  file pool size, 7-14
  file pool size per file, 7-14
  hash buffer size, 7-14
  hash enable RO, 7-14
  hash max open files, 7-14
  hash primary event size, 7-14
  hash secondary event size, 7-14
  use global file pool, 7-14

U
use alternate qualifier parameter, 7-11
use global file pool, 7-14
use table filter expression parameter, 7-11
user commit confirm table, 7-13

W
Weblogic
deployment, 3-1
WebLogic deployment, 3-1
Workspace server mode, B-13
write empty string as null parameter, 7-11

X
XML
  environment parameters, 7-14
  XML date format parameter, 7-15
  XML parameters
    COM maximum XML size, 7-14
    COM maximum XML size in memory, 7-14
    COM XML transport buffer, 7-14
    Replace invalid XML characters, 7-15
    XML date format parameter, 7-15
    XML trim char column, 7-15
    XML trim char column parameter, 7-15
Y
Y2K

See year 2000 policy parameter
eyear 2000 policy parameter, 7-8