## Contents

### Preface

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>ix</td>
</tr>
</tbody>
</table>

### 1 Introduction

<table>
<thead>
<tr>
<th>Section</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 Installing Oracle Connect and Oracle Studio

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preinstallation Tasks</td>
<td>2-1</td>
</tr>
<tr>
<td>z/OS Hardware and Software Requirements</td>
<td>2-1</td>
</tr>
<tr>
<td>Hardware Requirements</td>
<td>2-1</td>
</tr>
<tr>
<td>Software Requirements</td>
<td>2-2</td>
</tr>
<tr>
<td>Windows Hardware and Software Requirements</td>
<td>2-2</td>
</tr>
<tr>
<td>Hardware Requirements</td>
<td>2-2</td>
</tr>
<tr>
<td>Software Requirements</td>
<td>2-3</td>
</tr>
<tr>
<td>UNIX Requirements</td>
<td>2-3</td>
</tr>
<tr>
<td>Installing Oracle Connect on a z/OS Series Platform</td>
<td>2-3</td>
</tr>
<tr>
<td>Installation Worksheet</td>
<td>2-3</td>
</tr>
<tr>
<td>Preinstallation Tasks</td>
<td>2-5</td>
</tr>
<tr>
<td>Importing the Installation Kit</td>
<td>2-5</td>
</tr>
<tr>
<td>Transferring the Kit to the Dataset</td>
<td>2-5</td>
</tr>
<tr>
<td>Installation Instructions</td>
<td>2-6</td>
</tr>
<tr>
<td>Data Source Configuration Instructions</td>
<td>2-9</td>
</tr>
<tr>
<td>Post-Installation Instructions</td>
<td>2-9</td>
</tr>
<tr>
<td>Copy and Install Programs into CICS.</td>
<td>2-9</td>
</tr>
<tr>
<td>Post-installation Procedures</td>
<td>2-10</td>
</tr>
<tr>
<td>Starting the Daemon</td>
<td>2-11</td>
</tr>
<tr>
<td>Setting Up Oracle Connect for Reentrancy</td>
<td>2-12</td>
</tr>
<tr>
<td>Setting up Oracle Connect to Access DB2CLI</td>
<td>2-12</td>
</tr>
<tr>
<td>Updating an Existing Oracle Connect Installation with DB2</td>
<td>2-13</td>
</tr>
<tr>
<td>Installing Oracle Connect on UNIX</td>
<td>2-14</td>
</tr>
<tr>
<td>Preinstallation Tasks</td>
<td>2-14</td>
</tr>
<tr>
<td>Installing Oracle Connect</td>
<td>2-15</td>
</tr>
<tr>
<td>Installation Tasks</td>
<td>2-15</td>
</tr>
</tbody>
</table>
# Runtime Tasks and Troubleshooting

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Configuring the OracleAS CDC Adapter for DB2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Setting Up the IBM z/OS Platform in Oracle Studio</td>
<td>3-1</td>
</tr>
<tr>
<td></td>
<td>Securing Access to Oracle Connect</td>
<td>3-2</td>
</tr>
<tr>
<td></td>
<td>Setting Password Access to Oracle Studio</td>
<td>3-2</td>
</tr>
<tr>
<td></td>
<td>Specifying Users with Administrative Rights</td>
<td>3-4</td>
</tr>
<tr>
<td></td>
<td>Setting Up Run-Time User Access to the IBM z/OS Platform</td>
<td>3-5</td>
</tr>
<tr>
<td></td>
<td>Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2</td>
<td>3-6</td>
</tr>
<tr>
<td></td>
<td>Configuring the DB2 Tables for CDC</td>
<td>3-6</td>
</tr>
<tr>
<td></td>
<td>Setting up Security for the OracleAS CDC Adapter for DB2</td>
<td>3-6</td>
</tr>
<tr>
<td></td>
<td>Configuring the ATTSRVR Started Task</td>
<td>3-7</td>
</tr>
<tr>
<td></td>
<td>Setting up a Change Data Capture in Oracle Studio</td>
<td>3-7</td>
</tr>
<tr>
<td></td>
<td>Create a CDC Project</td>
<td>3-8</td>
</tr>
<tr>
<td></td>
<td>Set up the CDC Server</td>
<td>3-11</td>
</tr>
<tr>
<td></td>
<td>Set up the Staging Area Server</td>
<td>3-15</td>
</tr>
<tr>
<td></td>
<td>Deploying a Change Data Capture</td>
<td>3-24</td>
</tr>
<tr>
<td></td>
<td>Activating and Deactivating Solution Workspaces</td>
<td>3-27</td>
</tr>
<tr>
<td>4</td>
<td>Oracle BPEL Process Manager Deployment and Integration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Overview of Oracle BPEL Process Manager Integration</td>
<td>4-1</td>
</tr>
<tr>
<td></td>
<td>Configuring the DB2 CDC Adapter In the Application Server</td>
<td>4-1</td>
</tr>
<tr>
<td></td>
<td>CDC Stream Positions Table Definition</td>
<td>4-2</td>
</tr>
<tr>
<td></td>
<td>Configuring the DB2 CDC Adapter Connection Factory</td>
<td>4-3</td>
</tr>
<tr>
<td></td>
<td>Configuring the Oracle BPEL Process Manager</td>
<td>4-4</td>
</tr>
<tr>
<td></td>
<td>Setting up the Connection to the Oracle Connect Server Metadata</td>
<td>4-4</td>
</tr>
<tr>
<td></td>
<td>Using JDeveloper to Integrate a CDC Inbound Process to the Inbound Application</td>
<td>4-4</td>
</tr>
<tr>
<td></td>
<td>Configuring a CDC Adapter Partner Link</td>
<td>4-5</td>
</tr>
<tr>
<td></td>
<td>Configuring WSDL</td>
<td>4-5</td>
</tr>
<tr>
<td></td>
<td>Troubleshooting Rollback Loops</td>
<td>4-8</td>
</tr>
<tr>
<td>5</td>
<td>Runtime Tasks and Troubleshooting</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adding and Removing Tables</td>
<td>5-1</td>
</tr>
<tr>
<td></td>
<td>Handling Metadata Changes</td>
<td>5-2</td>
</tr>
<tr>
<td></td>
<td>Staging Area Maintenance</td>
<td>5-3</td>
</tr>
<tr>
<td></td>
<td>Monitoring the Change Data Capture</td>
<td>5-4</td>
</tr>
<tr>
<td></td>
<td>Service Context Table</td>
<td>5-4</td>
</tr>
<tr>
<td></td>
<td>Monitoring the Status</td>
<td>5-5</td>
</tr>
</tbody>
</table>
 Daemon Life-Cycle Tasks .................................................................................................................... 5-6
 Starting the Daemon ......................................................................................................................... 5-7
 Task: Starting the Daemon .............................................................................................................. 5-7
 Shutting Down the Daemon .............................................................................................................. 5-7
 Monitoring the Daemon During Runtime ...................................................................................... 5-7
 Daemon (Computer) Options ........................................................................................................... 5-8
 Workspace Options ......................................................................................................................... 5-8
 Server Options ............................................................................................................................... 5-9
 Daemon Logs ................................................................................................................................... 5-9
 The Log Monitor ............................................................................................................................... 5-10
 Resolving Communication Errors .............................................................................................. 5-11
 Resolving Specific Errors .............................................................................................................. 5-11

 6 Advanced Features of OracleAS CDC Adapter for DB2

 Configuring the Daemon for High Availability ............................................................................. 6-1
 Adding a New Daemon Workspace Configuration......................................................................... 6-1
 Editing the Workspace .................................................................................................................... 6-2
 Configuring the Server Mode ......................................................................................................... 6-2
 Configuring a Binding Environment ............................................................................................. 6-6
 Debug ............................................................................................................................................ 6-7
 General .......................................................................................................................................... 6-8
 Language ....................................................................................................................................... 6-8
 Modeling ......................................................................................................................................... 6-9
 ODBC ............................................................................................................................................ 6-9
 OLEDB .......................................................................................................................................... 6-9
 Optimizer ....................................................................................................................................... 6-9
 Parallel Processing ......................................................................................................................... 6-9
 Query Processor ............................................................................................................................. 6-10
 Transaction ..................................................................................................................................... 6-13
 Tuning ........................................................................................................................................... 6-13
 XML ................................................................................................................................................ 6-14
 Migration Considerations .............................................................................................................. 6-15
 Security Considerations .................................................................................................................. 6-15
 Setting Design Time Security ......................................................................................................... 6-16
 Setting Runtime Security ................................................................................................................ 6-16
 Transaction Support ....................................................................................................................... 6-17

 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-8
### Advanced Tuning of the Daemon

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>B-1</td>
</tr>
<tr>
<td>Logging</td>
<td>B-3</td>
</tr>
<tr>
<td>Security</td>
<td>B-5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workspaces</td>
<td>B-7</td>
</tr>
<tr>
<td>General</td>
<td>B-7</td>
</tr>
<tr>
<td>Server Mode</td>
<td>B-11</td>
</tr>
<tr>
<td>Security</td>
<td>B-16</td>
</tr>
</tbody>
</table>

### Editing Properties for the OracleAS CDC Adapter for DB2

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Editing Properties in Oracle Studio</td>
<td>C-1</td>
</tr>
<tr>
<td>Editing the OracleAS CDC Adapter Properties</td>
<td>C-1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration Properties</td>
<td>C-2</td>
</tr>
<tr>
<td>OracleAS CDC Adapter Configuration Properties</td>
<td>C-2</td>
</tr>
<tr>
<td>OracleAS CDC Adapter for DB2 Properties</td>
<td>C-2</td>
</tr>
<tr>
<td>Interaction Properties</td>
<td>C-3</td>
</tr>
<tr>
<td>CDC Queue Adapter Properties</td>
<td>C-3</td>
</tr>
<tr>
<td>Common CDC Adapter Properties</td>
<td>C-3</td>
</tr>
<tr>
<td>Change Router Properties</td>
<td>C-4</td>
</tr>
<tr>
<td>Referential Integrity Considerations</td>
<td>C-7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access to Change Events</td>
<td>C-9</td>
</tr>
<tr>
<td>Transaction Support</td>
<td>C-10</td>
</tr>
<tr>
<td>Security</td>
<td>C-10</td>
</tr>
</tbody>
</table>

### Back-end Adapter Data Type Support

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Type Mapping</td>
<td>D-1</td>
</tr>
</tbody>
</table>

### Globalization Settings

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defining the Language and Codepage</td>
<td>E-1</td>
</tr>
</tbody>
</table>

### Editing XML Files in Oracle Studio

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preparing to Edit XML Files in Oracle Studio</td>
<td>F-1</td>
</tr>
<tr>
<td>Making Changes to the XML File</td>
<td>F-2</td>
</tr>
<tr>
<td>Remove Objects</td>
<td>F-2</td>
</tr>
<tr>
<td>Add DTD Information</td>
<td>F-2</td>
</tr>
</tbody>
</table>
Edit Namespaces ............................................................................................................................ F-3
Add Elements and Attributes .............................................................................................................. F-5
Replace an Element .......................................................................................................................... F-5

Index
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 document describes the features of the OracleAS CDC adapter for DB2.

This preface covers the following topics:

■ Audience
■ Documentation Accessibility
■ Related Documents
■ Conventions

Audience

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

■ Installing and configuring OracleAS CDC Adapters for DB2
■ Using OracleAS for developing integration solutions using change data capture

Note: You should understand the fundamentals of OracleAS, OC4J, the UNIX and Microsoft Windows operating system before using this guide to install or administer OracleAS Adapters for DB2.

■ Diagnosing errors

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.

TTY Access to Oracle Support Services
To reach AT&T Customer Assistants, dial 711 or 1.800.855.2880. An AT&T Customer Assistant will relay information between the customer and Oracle Support Services at 1.800.223.1711. Complete instructions for using the AT&T relay services are available at http://www.consumer.att.com/relay/tty/standard2.html. After the AT&T Customer Assistant contacts Oracle Support Services, an Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process.

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, ESB) 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 timestamp, 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.

This section 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 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 backend system and handles from 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 backend 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 means of a JCA provider that sends change records to an end point configured in either ESB or BPEL.

- **JC2A Resource Adapter**: 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 be able to process changes from multiple tables in parallel.

- **JDBC Driver**: Enables access to changes in the staging area using SQL queries.
- **OC4j / BPEL Components**: Any OracleAS component that hosts and gets messages from the J2CA 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.

  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 will be deleted).

- **Change Tables**: The staging area is made of change tables, each containing 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. This means that 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 setup 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 cleared by default every 48 hours. All events that have been in the staging area for more than 48 hours are deleted.
Installing Oracle Connect and Oracle Studio

This section describes how to install Oracle Connect and Oracle Studio.

**Note:** In addition to the installation procedures described in this section, the J2CA 1.5 DB2 adapter must be installed with Oracle Application Server. Installing the J2CA 1.5 DB2 adapter is described in *Oracle Application Server Adapter Installation Guide*.

This section includes the following topics:

- Preinstallation Tasks
- Installing Oracle Connect on a z/OS Series Platform
- Updating an Existing Oracle Connect Installation with DB2
- Installing Oracle Connect on UNIX
- Installing Oracle Connect on Windows
- Installing Oracle Studio

**Preinstallation Tasks**

Before installing OracleAS Adapter for DB2, ensure that your computer meets the following requirements:

- z/OS Hardware and Software Requirements
- Windows Hardware and Software Requirements
- UNIX Requirements

**z/OS Hardware and Software Requirements**

This section describes the following requirements for installing Oracle Connect on a z/OS platform:

- Hardware Requirements
- Software Requirements

**Hardware Requirements**

The following table summarizes the hardware requirements for Oracle Connect.
Preinstallation Tasks

Software Requirements

The following table summarizes the software requirements for Oracle Connect:

<table>
<thead>
<tr>
<th>Software Component</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>IBM z/OS</td>
</tr>
<tr>
<td>CICS TP Monitor (if accessing DB2 data under CICS)</td>
<td>V4R1 or higher (recommended to use CICS V6R1 or higher)</td>
</tr>
<tr>
<td></td>
<td>CICS EXCI support must be installed and IRCSTRT=YES must be specified in the CICS initialization parameters, so that the IRC (Inter Region Communication) starts.</td>
</tr>
<tr>
<td></td>
<td>You can also set the IRC to open by issuing the following command: CEMT SET IRC OPEN. Also the IBM group DFH$EXCI (or an equivalent user-defined group) must be installed in the CICS region: using the CEDA RDO facility.</td>
</tr>
<tr>
<td>TCP/IP</td>
<td>The operating system must support the TCP/IP protocol for using the Internet.</td>
</tr>
<tr>
<td>C Runtime Library</td>
<td>The C runtime library has all the standard C runtime programs.</td>
</tr>
<tr>
<td>Oracle Application Server</td>
<td>Oracle Application Server 10g (10.1.3.4).</td>
</tr>
</tbody>
</table>

Windows Hardware and Software Requirements

This section describes the following requirements for installing Oracle Connect:

- **Hardware Requirements**
- **Software Requirements**

**Hardware Requirements**

The following table summarizes the hardware requirements for Oracle Connect.

<table>
<thead>
<tr>
<th>Hardware Component</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>An Intel or 100% compatible computer, based on a Pentium processor</td>
</tr>
<tr>
<td>Memory</td>
<td>256 MB of RAM</td>
</tr>
<tr>
<td>Disk Space</td>
<td>40 MB of free disk space</td>
</tr>
</tbody>
</table>
Software Requirements
The following table summarizes the software requirements for Oracle Connect.

<table>
<thead>
<tr>
<th>Software Component</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Microsoft 2000 with service pack 2 or higher, Microsoft Windows XP, or Microsoft Windows Server 2003.</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Network transport protocol software, TCP/IP, included with Microsoft Windows.</td>
</tr>
</tbody>
</table>

UNIX Requirements
Oracle Connect can be installed on the following UNIX operating systems:

- Red Hat Linux
- Linux SUSE
- HP-UX
- IBM AIX
- Sun Solaris

For all of the operating systems, the required disk space is 48MB.

Installing Oracle Connect on a z/OS Series Platform
This section explains how to install Oracle Connect. This section includes the following:

- Installation Worksheet
- Preinstallation Tasks
- Importing the Installation Kit
- Installation Instructions
- Data Source Configuration Instructions
- Post-Installation Instructions

Note: If you have an Oracle Connect back-end adapter already installed on the IBM z/OS platform, follow the instructions described in "Updating an Existing Oracle Connect Installation with DB2".

Installation Worksheet
Verify that you have all the information detailed in the following installation worksheets, so you can refer to it during the configuration process.
### Table 2–5 Preinstallation Information

<table>
<thead>
<tr>
<th>Topic</th>
<th>Required Information</th>
<th>Default</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>Operating system</td>
<td>-</td>
<td>z/OS</td>
</tr>
<tr>
<td></td>
<td>Disk space</td>
<td>-</td>
<td>150 cylinders.</td>
</tr>
<tr>
<td></td>
<td>Memory</td>
<td>-</td>
<td>The minimum requirement is 4MB for each connection. A connection is defined as a connection to a server process or daemon. The actual memory requirement depends on such things as the size of the database and the number of databases accessed.</td>
</tr>
<tr>
<td></td>
<td>Installation high-level qualifier</td>
<td>OCL10134</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Volume</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Unit</td>
<td>3390</td>
<td>SMS only: unit where SMS resides.</td>
</tr>
<tr>
<td></td>
<td>Output class</td>
<td>A</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>JCL job card</td>
<td>-</td>
<td>An optional card (up to 6 lines) to replace the prefix job (entered as it will appear in the job).</td>
</tr>
<tr>
<td>CICS</td>
<td>ISPF load library name</td>
<td>ISP:SISPLOAD</td>
<td>-</td>
</tr>
<tr>
<td>CICS</td>
<td>CICS EXCI load library name</td>
<td>CICS.CICS.SDFHEXCL</td>
<td>To access DB2 data under CICS.</td>
</tr>
</tbody>
</table>

### Table 2–6 Required Permissions

**Permission**
- Permission to define an APF-authorized library
- Permission to write to an active proclib, such as user.proclib
- Permission to read the CICS EXCI library (when accessing DB2 data under CICS)
- Permission to update the security manager, such as RACF
- Optionally, permission to specify an output class for Oracle Connect output

### Table 2–7 Installation Checklist

<table>
<thead>
<tr>
<th>Step</th>
<th>Comment/Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>tso profile prefix</td>
<td>Ensures that the user name is not used as part of the dataset name allocated in the next steps.</td>
</tr>
<tr>
<td>Allocate dataset: [HLQ].TRANSMIT.KIT</td>
<td>130 tracks (3390), format=FB, record length=80, block size=3120</td>
</tr>
<tr>
<td>Allocate dataset: [HLQ].TRANSMIT.LOAD</td>
<td>500 tracks (3390), format=FB, record length=80, block size=3120</td>
</tr>
<tr>
<td>FTP files to z/OS</td>
<td>FTP using binary mode.</td>
</tr>
<tr>
<td>RECEIVE INDSNAME('[HLQ].TRANSMIT.KIT')</td>
<td>-</td>
</tr>
<tr>
<td>da('[HLQ].TRANSMIT.LIB') UNIT(unit) VOLUME(volume)</td>
<td>-</td>
</tr>
<tr>
<td>EX [HLQ].TRANSMIT.LIB(PREPARE)</td>
<td>Successful MAXCC is 0, 4 or 8</td>
</tr>
<tr>
<td></td>
<td>BUILDKIT.SRC and BUILDKIT.LOAD created.</td>
</tr>
<tr>
<td>EX [HLQ].BUILDKIT.SRC(NAVINST)</td>
<td>Successful MAXCC is 0 or 4.</td>
</tr>
</tbody>
</table>
Preinstallation Tasks

Before starting the installation, ensure that you have the following information available:

- The output class for the installation output if you do not want to use the default value, which is A
- The unit where SMS resides if you use SMS to manage all datasets. If you use SMS to manage all datasets, you cannot provide unit and volume information.

Before starting the installation, ensure that you have the following permissions:

- Permission to define an APF-authorized library.
- Permission to write to an active proclib, such as user.proclib.
- Permission to read the CICS EXCI library, when accessing DB2 data under CICS.
- Permission to update the security manager, such as RACF.

Note: If necessary, ensure that you have permission to specify an output class for Oracle Connect output. Assigning a device which is set on HOLD prevents the loss of log information when Oracle Connect started tasks finish.

Oracle Connect for the IBM z/OS platform is contained in the following datasets:

- OCL10134.TRANSMIT.KIT
- OCL10134.TRANSMIT.LOAD

Importing the Installation Kit

Perform the following steps on the mainframe to import Oracle Connect installation kit to the mainframe:

1. Run the following command:
   ```
   tso profile noprefix
   ```
   The user name will not be used as part of the dataset name. On some systems this is the default.

2. Allocate datasets with the following space for each of these files:
   ```
   - OCL10134.TRANSMIT.KIT = 130 tracks (3380 and 3390 disks)
   - OCL10134.TRANSMIT.LOAD = 420 tracks (3380 and 3390 disks)
   ```
   For each dataset: RECFM=FB and LRECL=80. The block size is 3120.

Transferring the Kit to the Dataset

You need to move the contents of the kit to the mainframe computer in the dataset that you allocated. When you copy the contents of the kit, you will see that the contents are now utilized.

Perform the following steps to copy the Oracle Connect installation kit contents to the mainframe:

1. Extract the compressed file to a folder on a computer running Microsoft Windows.
2. Open the Command console on the Windows computer.
3. Type FTP and the name of the mainframe computer where you want to connect. This creates an FTP connection to the mainframe machine. You may have to enter a user name and password to use the machine.

4. Type bin to transfer binary data.

5. Enter the following command to transfer the data from the .KIT file to the mainframe computer:

```plaintext
put OCL10134.transmit.kit '[$HLQ].transmit.kit' [replace]
```

where:
- OCL10134.transmit.kit is the name of your installation file.
- [replace] ensures that any data on the volume where the kit is installed is overwritten.

6. Enter the following command to transfer the data from the .LOAD file to the mainframe computer:

```plaintext
put OCL10134.transmit.load '[$HLQ].transmit.load' [replace]
```

7. Close the FTP connection.

**Installation Instructions**

Perform the following steps to install Oracle Connect:

1. Run the following command at the TSO prompt:

```plaintext
RECEIVE INDSNAME('nnn.TRANSMIT.KIT')
```

Where *nnn* represents the high-level qualifier you want to assign for the Oracle Connect installation. Assign the high-level qualifier you specified in step 7 of the preinstallation tasks. The default value is OCL10134.

---

**Note:** You can use more than one high-level qualifier (such as ACME.DEV.OCL10134) with the following conditions:

- The total length must be less than or equal to twenty characters.
- The words transmit and buildkit cannot be used as high-level qualifiers.

2. Enter the following command when prompted for the restore parameters:

```plaintext
da('nnn.TRANSMIT.LIB') \[UNIT(unit) VOLUME(volume)]
```

This extracts the *nnn*.TRANSMIT.LIB library from the *nnn*.TRANSMIT.KIT kit to the specified unit and volume. If a unit and volume are not specified the library is extracted to the current unit and volume.

3. In the Data Set List Utility screen, type the name of the high-level qualifier on the Dsname Level line. This will return a list of the files. Verify that the *<nnn>*.TRANSMIT.LIB file was created.

4. In the Command column next to the *<nnn>*.TRANSMIT>LIB, type M.

5. Execute the PREPARE member of the *nn*.TRANSMIT.LIB library:

```plaintext
ex \PREPARE
```
Follow the instructions in the Response column in the following table for each entry in the Screen column.

**Table 2–8 Installation Prepare Job Prompts and Responses**

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>DO YOU WANT TO USE SMS MANAGED STORAGE FOR THIS INSTALLATION Y/N [N] :</td>
<td>If you want to manage the storage using SMS, then answer Y, otherwise answer N.</td>
</tr>
<tr>
<td>ENTER THE STORCLASS FOR INSTALLATION TEMP DATASETS [ ] :</td>
<td>This prompt is displayed only if SMS is used to manage the installation (you answered Y to the first prompt). Enter the storage class.</td>
</tr>
<tr>
<td>ENTER THE UNIT NAME FOR INSTALLATION TEMP DATASETS [3390] :</td>
<td>If a storage class is not specified, then enter the unit name for temporary datasets used during the installation procedure.</td>
</tr>
<tr>
<td>ENTER THE VOLUME NAME FOR INSTALLATION TEMP DATASETS :</td>
<td>This prompt is displayed only if SMS is not used to manage the installation (you answered N to the first prompt). The volume name for temporary datasets used during the installation procedure.</td>
</tr>
<tr>
<td>ENTER THE OUTPUT CLASS FOR INSTALLATION OUTPUT [A] :</td>
<td>Enter the output class only if you do not want the default class used (the default is A)</td>
</tr>
<tr>
<td>DO YOU WANT TO USE THE DEFAULT JOB CARD Y/N [Y]</td>
<td>A job card is displayed. If you want to use a replacement card, then it must be entered as it will appear in the job. You can enter up to six lines. Enter a blank card to end input. If you do not enter a card, then the Oracle Connect default card is used.</td>
</tr>
<tr>
<td>DO YOU WANT TO PERFORM A MANUAL (M) OR AUTOMATIC (A) INSTALLATION [A] :</td>
<td>If you want to review the JCL used to install Oracle Connect, before it is submitted, then respond M for a manual installation.</td>
</tr>
<tr>
<td>PLEASE REVIEW AND SUBMIT FOR EXECUTION THE HLQ.TRANSMIT.LIB(INSTJO)</td>
<td>This prompt is displayed only if a manual installation is requested (you answered M to the previous prompt).</td>
</tr>
</tbody>
</table>

The following libraries are generated:

```
nnn.BUILDKIT.LOAD
nnn.BUILDKIT.SRC
nnn.BUILDKIT.GENDEMO
```

Where *nnn* is the high-level qualifiers you assigned in step 1.

**Note:** This command always finishes successfully. To make sure that no errors occurred, check the *instjo* JCL.

6. In the *nnn.BUILDKIT.SRC* library, execute the *NAVINST* member:

```
ex NAVINST
```

Follow the instructions in the Response column in the following table for each entry in the Screen column.
The installation is completed. All JCL jobs and REXX procedures are written to the \texttt{INSTROOT.USERLIB} library. \texttt{INSTROOT} is the high-level qualifier for the installation.
Data Source Configuration Instructions

In the $nnn.BUILDKIT.SRC$ library, execute the DB2 member:

```
exit DB2
```

Follow the instructions in the Response column for each entry in the Screen column.

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTER THE DB2 HLQ</td>
<td>Enter the HLQ where DB2 is installed.</td>
</tr>
<tr>
<td>ENTER THE DB2 SUBSYSTEM NAME</td>
<td>Enter the IDB2 subsystem.</td>
</tr>
<tr>
<td>THE DATABASE DB2 RESIDES ON THIS MACHINE</td>
<td>This is a summary of the DB2 questions. Check the summary to be sure all is correct.</td>
</tr>
<tr>
<td>THE DB2 SUBSYSTEM NAME IS:</td>
<td>■ Yes, to use the DB2 database as configured</td>
</tr>
<tr>
<td>PLEASE CONFIRM (YES/NO/QUIT):YES:</td>
<td>■ No, to not use the DB2 database</td>
</tr>
<tr>
<td></td>
<td>■ Quit, to leave the DB2 process</td>
</tr>
</tbody>
</table>

Post-Installation Instructions

The following post-installation tasks must be done to work with Oracle Connect:

- Copy and Install Programs into CICS
- Post-installation Procedures
- Starting the Daemon
- Setting Up Oracle Connect for Reentrancy
- Setting up Oracle Connect to Access DB2CLI

Copy and Install Programs into CICS

You must copy and install some supporting programs when you use a CICS session.

The following describes the steps necessary to copy and install these programs under CICS.

1. In the Data Set List Utility screen, type the name of the `<HLQ>` on the Dsname Level line. This will return a list of the files.
2. In the Command column next to the `<HLQ> LOAD`, type `M`.
3. Type `C` next to following programs you are using. The available programs are listed below:

   - **ATTHRDPL**: Use this program when working under CICS. This is the CICS queue.
   - **ATTCISD**: Use this program when working under CICS.
   - **ATTHRDPL**: Use this program when working under CICS if you are using connection pooling.
   - **ATTCALL**: Use this program when working under CICS if you are using connection pooling.
   - **ATTCNTRL**: Use this program when working under CICS if you are using connection pooling.
■ TRANS3GL: Use this program if you are using the 3GL interface. This is an example program.

4. Exit the session, then start a new session.

5. Enter a new CICS session.

6. At the code line, run the following:

   CEDA INS PROG(<PROG NAME>) G(ATY)

   The <PROG NAME> is on one of the programs listed in step 3. Run this for each of the programs you are using.

Post-installation Procedures
Perform the following procedures after completing the installation to configure Oracle Connect.

■ Allocate a dataset for **INSTROOT.DEF.BRANDBIN**, using 1 track and with
RECFM=VB and LRECL=256. The block size is 6233.

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

   Using FTP, copy the BRANDBIN file, in binary mode, from the
   Oracle Connect\DB2/DB Legacy Adapter directory to the mainframe, to
   **INSTROOT.DEF.BRANDBIN**.

■ Define the LOADAUT library as an APF-authorized library

---

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

```
*/setprog apf,add,dsn=INSTROOT.loadaut,volume=vol002`
```

where vol002 is the volume where you installed Oracle Connect and
**INSTROOT** is the high-level qualifier where Oracle Connect is installed.

If the site uses SMS, then when defining APF-authorization in the
SDSF screen, enter the following command:

```
*/setprog apf,add,dsn=INSTROOT.loadaut,SMS`
```

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

---

■ Move the **INSTROOT.USERLIB(ATTTDAEMN)** and
**INSTROOT.USERLIB(ATTSRVR)** members to any active proclib, such as
user.proclib, ATTTDAEMN and ATTSRVR are run as started tasks.

If you decide to change the name of the ATTSRVR member when you move it to a
general high-level qualifier, then change the name specified in the
**StartupScript** parameter in the daemon configuration to the new name:

■ Run **INSTROOT.USERLIB(NAVCMD)** and enter **EDIT DAEMON IRPCDINI** at
the prompt.

■ Change the **startupScript** parameter from ATTSRVR to the new name for
the server:

```
<Workspace name="Navigator"
    startupScript="NEW_NAME"
```
Install serverMode="reusable"
... />

- Exit and save the change.
- Change the following line in the ATTDAEMN script to include the IP address and port of the z/OS platform.
  For example, before:
  // PARM='-B START IRPCDINI'

  After:
  // PARM='-B -L ip_address:2551 START IRPCDINI'

Where ip_address specifies the IP address of the computer, 2551 is the default port for starting the daemon and IRPCDINI is the default daemon configuration.

- The ATTDAEMN and ATTSRVR started tasks need permission to use an Open Edition TCP/IP stack. The owner must be a user with OMVS segment defined and OMVS UID= 000000000.

- In the security manager, such as RACF, define ATTDAEMN and ATTSRVR with a started task class and a general profile that enables the following:
  - Permission to issue master console commands.
  - START authority for the ATTSRVR job.
  - Access to an Open z/OS segment (that defines access to TCP/IP OA sockets).
  - ALTER authority on datasets under INSTROOT (to access to read, write, allocate and delete datasets under INSTROOT).

- The installation includes a PS, INSTROOT.DEF.GBLPARMS, that contains global environment information. This PS is read at startup and the correct software version is used, based on the details provided in the startup task.
  If you change the location of this member, you must also change the relevant cards in the following jobs to the new locations:
  - ATTSRVR: located in an active proclib, such as user.proclib
  - ATTDAEMN: located in an active proclib, such as user.proclib
  - NAVSQL: located in INSTROOT.USERLIB

- The input during the installation procedure is written to nnn.BUILDKIT.SRC(PARS). You can use this file to provide the same inputs if you rerun the installation, where nnn is the high-level qualifier you assign for the installation.

- For information about specifying Oracle Connect as the service using port 2551 in the TCP/IP network services file, consult TCP/IP documentation.

Starting the Daemon

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

'/s ATTDAEMN'

Where INSTROOT is the high-level qualifier where Oracle Connect is installed.
To submit the daemon as a job, uncomment the first two lines of the ATTDAEMN JCL, change the PARM line as described earlier, and run the job using the subcommand. The ATTDAEMN JCL is similar to the following:

```hll
//*ATTDAEMN JOB 'RR','TTT',MSGLEVEL=(1,1),CLASS=A,
//* MSGCLASS=A,NOTIFY=&SYSUID,REGION=8M
//*STEP1 EXEC PGM=IRPCD,
//* PARM=-B START IRPCDINI'
//* PARM=-B -L :8883 START'
//STEP1 LIB DD DSN=INSTROOT.LOADAUT,DISP=SHR
//SYSPRINT DD SYSOUT=A
//GBLPARMS DD DSN=INSTROOT.DEF.GBLPARMS,DISP=SHR
// EXEC PGM=IRPCD,COND=((1,EQ,STEP1),(2,EQ,STEP1)),
// PARM=-KATTDAEMN START ''INSTROOT.DEF.IRPCDINI''
//STEP1 LIB DD DSN=INSTROOT.LOADAUT,DISP=SHR
//SYSPRINT DD SYSOUT=A
//GBLPARMS DD DSN=INSTROOT.DEF.GBLPARMS,DISP=SHR
//SYSDUMP DD DUMMY
```

### Setting Up Oracle Connect for Reentrancy

All Oracle Connect load modules are reentrant to enable sub-tasking. Therefore, move INSTROOT.LOAD to the Link Pack Area (LPA).

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

Using the LPA reduces real storage usage (because everyone shares the LPA copy) and fetch time.

**Note:** If you intend on using impersonation, so that you can run in a security context that is different than the context of the process that owns the server, then do the following:

- Place the INSTROOT.LOAD (ATYSVCW) member in an APF-authorized library outside the LPA.
- Change the ATTSRVR member (located in the active proclib), by adding the following to the STEPLIB list:

```hll
// DD DSN=apf_library,DISP=SHR
```

Where `apf_library` is the APF-authorized library outside the LPA where the ATYSCVW member was moved.

### Setting up Oracle Connect to Access DB2CLI

In order to access DB2 CLI, you must have DB2 CLI installed on the machine. For details of DB2 CLI, refer to the *IBM Call Level Interface Guide and Reference*.

The Oracle Connect DB2 CLI driver uses an ODBCINI file. During installation of Oracle Connect, an ODBCINI file is defined as a member in NAVROOT.USERLIB.

The ODBCINI file is similar to the following:

```hll
; This is a comment line...
; Example COMMON odbcini
[COMMON]
MVSDEFAULTSSID=DSN1
; Example SUBSYSTEM odbcini for DSN1 subsystem
[DSN1]
MVSATTACHTYPE=CAF
PLANNNAME=DSNACLI
```
The **PLANNAME** value is the default DB2 Calling Level Interface (CLI) plan name. If a different plan is used, change the name to the correct plan.

---

**Note:** In the example, CAF is used, so only one-phase commit transactions are supported. Oracle Connect also supports two-phase commit transactions, by setting the `MVSATTACHTYPE` parameter to `RRSAF`.

---

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

---

**Note:** The `AUTOCOMMIT` initialization parameter is set automatically by Oracle Connect at runtime, and thus should not be set in the `ODBCINI` file.

---

You must bind to the plan specified in the `PLANNAME` parameter. The bind to the plan, check that the job `DSNxxx.SDSNSAMP(DSNTIJCL)` includes the following line:

```
BIND PLAN(DSNACLI)
```

Where `DSNxxx` is the DB2 high-level qualifier (the default is `DSN610`) and `DSNACLI` is the plan name specified for the `PLANNAME` parameter.

**Oracle Connect and DB2 Security**

All the libraries in the STEPLIB must be APF-authorized. Grant DB2 DISPLAY GROUP authorization to the ATTSRVR started task. Issue the following commands to the owner of the ATTSRVR started task:

```
GRANT TRACE(STAT)
GRANT TRACE(MON)
GRANT MONITOR2
```

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

```
-start trace(stat)
```

---

**Updating an Existing Oracle Connect Installation with DB2**

Verify that you have all the information detailed in the following installation worksheets, so you can refer to it during the configuration process.

---

**Table 2–11 Preinstallation Information**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Required Information</th>
<th>Default</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>CICS</td>
<td>CICS EXCI load library name</td>
<td>CICS.CICS.SDFLEXCL</td>
<td>-</td>
</tr>
</tbody>
</table>

**Table 2–12 Required Permissions**

<table>
<thead>
<tr>
<th>Permission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permission to read the CICS EXCI library</td>
</tr>
</tbody>
</table>

---

*Installing Oracle Connect and Oracle Studio* 2-13
In the `nnn.BUILDKIT.SRC` library, execute the DB2 member:

```
ex DB2
```

Follow the instructions in the Response column for each entry in the Screen column.

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTER THE DB2 HLQ</td>
<td>Enter the HLQ where DB2 is installed.</td>
</tr>
<tr>
<td>ENTER THE DB2 SUBSYSTEM NAME</td>
<td>Enter the IDB2 subsystem.</td>
</tr>
<tr>
<td>THE DATABASE DB2 RESIDES ON THIS MACHINE</td>
<td>This is a summary of the DB2 questions. Check the summary to be sure all is correct.</td>
</tr>
<tr>
<td>THE DB2 SUBSYSTEM NAME IS:</td>
<td>- Yes, to use the DB2 database as configured</td>
</tr>
<tr>
<td>PLEASE CONFIRM (YES/NO/QUIT):YES:</td>
<td>- No, to not use the DB2 database</td>
</tr>
<tr>
<td></td>
<td>- Quit, to leave the DB2 process</td>
</tr>
</tbody>
</table>

The installation is completed. All JCL jobs and REXX procedures are written to the `INSTROOT.USERLIB` library. `INSTROOT` is the high-level qualifier for the installation.

After completing the installation, perform post-installation tasks, as described in "Post-Installation Instructions" on page 2-9, as required.

**Installing Oracle Connect on UNIX**

You must install Oracle connect on any UNIX machine that you are using for a staging area for the CDC adapter. The following sections explain how to install Oracle Connect on a UNIX machine.

- Installing Oracle Connect on a UNIX Platform

**Installing Oracle Connect on a UNIX Platform**

This section explains how to install Oracle Connect on a UNIX platform. This section includes the following:

- Preinstallation Tasks
- Installing Oracle Connect
- Installation Tasks
- Post-Installation Tasks

**Preinstallation Tasks**

Before starting the installation procedure, ensure that you have the following information is available:

- The root directory where you want to install Oracle Connect.

  **Notes:** The root directory cannot be a system root directory, `/var` or `/tmp`.

- The account name where Oracle Connect will run.
- Whether the installation source media is removable media or a disk archive file.
If you are installing from the removable media that is not a default device in the system, you need to know the media device name (such as /dev/rmt/0m). See the operating system manuals or ask the system administrator to find out the device name for the site.

If you are installing from the disk archive file, you need to know the name of the Oracle Connect disk archive file (such as /tmp/nav.2.0.tar).

- The shell being used: C-shell Korn-shell, or Bourne-shell. The installation creates a startup file according to the indicated shell.

When running the installation below, use the following as the <filename>:

- For RedHat Linux: OCL10134-linuxrh.tar.Z
- For Linux Suse: OCL10134-linuxsuse.tar.Z
- For HP-UX: OCL10134-hpux.tar.Z
- For IBM AIX: OCL10134-ibmaix.tar.Z
- For Solaris Operating System (SPARC): OCL10134-sunsol2.8.tar.Z

**Installing Oracle Connect**

Perform the following steps to install Oracle Connect:

1. Transfer the tar.Z file to the system.

2. Decompress the file using the following command:
   
   `uncompress <filename>`

3. Run the tar command, as shown in the following example:
   
   `tar xvf <filename> nav_install`

   The following message is displayed:
   
   `x nav_install, nnnn bytes, mmmm tape blocks`

---

**Note:** Ensure that the directory used to run the installation files has WRITE privileges.

**Installation Tasks**

Perform the following steps to install Oracle Connect:

1. Run the following command:
   
   `./nav_install`

   This command initiates the installation procedure. The installation procedure is displayed in a series of screen prompts and responses.

2. Enter the full path of the disk archive (.tar) file, and press Enter.

3. Enter the root directory name for the installation, and press Enter. You must have a WRITE permission for this directory. The default directory is the users home directory.
4. Confirm the directory name in which Oracle Connect will be installed, and press Enter.

5. Enter the account name where you want Oracle Connect to run, and press Enter. This account name will be used for anonymous access to the server by clients. It can be changed after the installation is complete.

6. Confirm the account name, and press Enter.

7. Specify the required shell, under which Oracle Connect should run, and press Enter. The following options are displayed:
   - C-shell (/bin/csh).
   - Korn-shell (/bin/ksh)
   - Bourne-shell (/bin/sh)

8. Enter the account name for a user with administrative authorization. Optionally, press Enter to enable any user to administer Oracle Connect.

Post-Installation Tasks
After installing Oracle Connect, perform the following post-installation tasks:

- Configuring the Oracle Connect Environment
- Configuring the Oracle Connect Script
- Starting the Oracle Connect Daemon

Configuring the Oracle Connect Environment
When Oracle Connect is installed on a UNIX platform, using FTP, copy the brand.bin file to the Oracle Connect computer, to NAVROOT/bin.

Where NAVROOT is the directory where Oracle Connect is installed.

Configuring the Oracle Connect Script
The program that manages Oracle Connect server processes (nav_server) is accessed by a symbolic link to a file for the C-shell, Bourne and Korn shells.

To set up nav_server, perform the following steps:

1. In the bin directory, under the directory where Oracle Connect is installed, delete the existing link to nav_server using the following command:
   \[ rm nav_server \]

2. In the bin directory, under the directory where Oracle Connect is installed, link to the required version of nav_server as follows:
   - C-shell: \[ ln -s nav_server.csh nav_server \]
   - Bourne: \[ ln -s nav_server.sh nav_server \]
   - Korn: \[ ln -s nav_server.ksh nav_server \]

Notes:
- The root directory cannot be a system root directory or /var or /tmp directory.
- Oracle Connect is installed into a fixed directory named navroot.
The Oracle Connect `nav_login` procedure defines the default environment when Oracle Connect runs. If you want site-dependent variables to be included in the environment, create a file called `site_nav_login` and save this file in the bin directory under the Oracle Connect root directory. `nav_login` runs `site_nav_login` automatically.

`nav_login` must be invoked to run Oracle Connect. It can be invoked from the user login script.

The command line for invoking `nav_login` varies according to the shell the user is running. The following table lists the different options for invoking the command line:

<table>
<thead>
<tr>
<th>Shell</th>
<th>nav_login Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSH</td>
<td><code>source root/bin/nav_login</code></td>
</tr>
<tr>
<td>Bourne</td>
<td><code>. root/bin/nav_login.sh</code></td>
</tr>
<tr>
<td>Korn</td>
<td><code>. root/bin/nav_login.sh</code></td>
</tr>
</tbody>
</table>

In the `nav_login` command, `root` represents the root directory of the Oracle Connect installation. After running the login procedure, the environment variable `NAVROOT` points to this root directory.

Ensure that users have **READ** and **EXECUTE** permissions on the Server files. Use the `chmod` command to change the permissions.

**Starting the Oracle Connect Daemon**

The Oracle Connect daemon must run on a server for client/server access to Oracle Connect. To start the daemon with the system startup, add the following command invoking the daemon to the end of the `/etc/inittab` file:

```
nv:3:once:navroot/bin/irpcd -l ip:2551 start >/dev/console 2>&1
```

In this command, the symbol `navroot` should be replaced with the directory where Oracle Connect is installed and `ip` replaced by the ip address of the computer.

**Note:** To allow automatic client/server access to Oracle Connect, start the daemon at system startup time from a super user account.

**Installing Oracle Connect on Windows**

You must install Oracle connect on any Windows machine that you are using for a staging area for the CDC adapter. This following sections explain how to install Oracle Connect on a Windows machine.

- **Installing Oracle Connect on a Windows Platform**

**Installing Oracle Connect on a Windows Platform**

This section explains how to install Oracle Connect. Do the following to install Oracle Connect.
Installing Oracle Studio

This following sections explain how to install Oracle Studio.

- Oracle Studio Requirements
- Installing Oracle Studio on Windows
- Installing Oracle Studio on Linux

**Note:** If you have Oracle Studio version 10.1.3.4 or higher installed on your computer because you are using an OracleAS legacy adapter or OracleAS CDC adapter, you do not need to reinstall it. If you have an older version of Oracle Studio, you must install the newest version.

Oracle Studio Requirements

The following are the hardware requirements for Oracle Studio:

- Processor: Intel or 100% compatible computer, based on a Pentium processor
- Memory: 256 MB
- Disk space: 120 MB of free disk space

You can install Oracle Studio on the following operating systems:

- Windows XP with Service Pack 2 or higher
  - Windows 2003
  - Windows Vista

1. Copy the following installation files into a folder on the Windows computer where you are installing Oracle Connect.
   - OCL_10134_windows.exe
   - brand.bin

2. Open the Windows command line interface.

3. Change to the directory where you installed the installation files.

4. Type the following at the command prompt to install Oracle Connect.

   "OCL_10134_windows.exe" -a -bBRAND=".\brand.bin" -bSERVER

   The install wizard opens. Follow the directions in the installation wizard to complete the installation.

**Note:** If you are upgrading from version 10.1.3.3, you must use the upgrade option in the installation wizard. Do not uninstall Version 10.1.3.3 and then install the newer version. If you do this, you will lose all data in the DEF directory.

**Note:** If you are installing Oracle Connect on a Windows XP computer, you cannot use a logical drive as the destination folder for the installation.
Installing Oracle Studio on Windows

Oracle Studio is installed with a standard install wizard. Do the following to install Oracle Studio.

- Run the installation file, either using the Run option in the Windows Start menu or through Windows Explorer. Follow the instructions on the wizard screen.

Installing Oracle Studio on Linux

The Oracle Studio installation on Linux can be carried out by one of the following:

- Installing Oracle Studio with the Wizard (SH installation)
- Installing with a Silent Installation (RPM)

Installing Oracle Studio with the Wizard (SH installation)

You can install Oracle Studio by using the installation wizard. This allows you to easily carry out any standard or custom installation. Do the following for the Linux SH installation.

1. Install into a directory where you have permission.
2. Change the mode to execute mode. Type in:
   
   ```
   Chmod +x <file name>
   ```
3. Enter the following:
   
   ```
   ./ <file name>
   ```
   If you are not installing to the current directory, enter the full path.

Installing with a Silent Installation (RPM)

You can use the Linux RPM method to install Oracle Studio. Do the following to carry out a silent installation on Linux.

- Enter the following:
  
  ```
  rpm -i <file name>
  ```
  You can use RPM parameters to create a custom installation.
This section 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 section 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.
Figure 3–1 The Add Machine screen

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 username 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:

   **Figure 3–2**  *The Preferences screen*

   ![Preferences Screen](image)

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

   **Figure 3–3**  *The Change Master Password Screen*

   ![Change Master Password Screen](image)

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 password 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:

Figure 3–4  The Administration Authorization screen

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 metadata definitions for a table in a data source.
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 who 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 3–5 The User Editor

5. In the User editor, click Add. The Add Authenticator screen is displayed:
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:

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.
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 Connect and DB2 Security.

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:

```
; 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 will 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. For more information, see Installing Oracle Studio.
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 carry out the tasks and any subtasks presented. If more than one link has an asterisk, you can carry out 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.

Carry out 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**

Do the following 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 just below.
5. Select **Change Data Capture**.
   
   From the 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 machine 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
Setting Up a Change Data Capture with the OracleAS CDC Adapter for DB2

Configuring the OracleAS CDC Adapter for DB2

- Linux (Red Hat)
- Suse (Linux)

Figure 3–9  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** checkbox 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, carry out the following procedure:

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

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

### Figure 3–12 The DB2 Data Source Configuration

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 MVSDEFAULTSSID 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 need to provide security credentials to the DB2 database.

5. Click **Finish**.

For information on DB2 data source properties, see in 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 (timestamp), 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 timestamp is found.

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

   **Figure 3–13  CDC Logger Definition Window**

5. In the **Bootstrap dataset name** field, enter the name of the dataset 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 3–14 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 3–15  Staging Area**

Note: This screen will only appear 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 have already added 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 3–16  Select Tables**

![Select Tables](image)

**Figure 3–17  Column Selection**

![Column Selection](image)
Any data changes in the columns selected will be 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 3–18 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.

![Content Filter Screen](image)

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 more than one condition, you will receive the change information as long as one of the conditions 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 3–20 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 3–21 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 recorded delivered, as well as system 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 next screen that 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 that you want 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**, if you want 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 the users/groups that you want to have exclusive access.
   - Select **Do you want to access server instances via specific ports**, if you want to allow this options. If this option is cleared, the defaults are used.

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

6. Click Next.

   The summary screen opens.
Figure 3–25  Workspace Setup Summary

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.

   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 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 will be notified that re-deployment will override 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, the 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 shown. It includes the ODBC connection string and JDBC connection string, as well as specific logger scripts to enable CDC capturing.

Figure 3-28 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 make sure that the `context` and `agent_context` fields of the `SERVICE_CONTEXT` table should 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 datasource run: `insert into SERVICE_CONTEXT (context, agent_context) values('XXX', 'YYY');`. This will Insert 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.
To deploy and integrate OracleAS CDC Adapters for DB2 with Oracle BPEL Process Manager, you need to configure the BPEL Process Manager.

This section includes the following topics:

- Overview of Oracle BPEL Process Manager Integration
- Configuring the DB2 CDC Adapter In the Application Server
- Configuring the Oracle BPEL Process Manager

Overview of Oracle BPEL Process Manager Integration

Oracle BPEL Process Manager provides a comprehensive solution for creating, deploying, and managing BPEL business processes. Oracle BPEL Process Manager is based on the Service Oriented Architecture (SOA) to provide enterprises with flexibility, interoperability, reusability, extensibility, and rapid implementation of Web services and business processes. It reduces the overall costs of management, modification, extension, and redeployment of existing business processes. Each business activity is a self-contained, self-describing, and modular application whose interface is defined by the WSDL, and the business process is modeled as a Web Service.

A Web Service is first published and then composed or orchestrated into business flows. Publishing a service is implemented by taking a function within an existing application or system and making it available in a standard way, while orchestration is implemented by composing multiple services into an end-to-end business process. The CDC solutions that are defined as part of the configuration of the OracleAS CDC Adapter for DB2 are integrated into the orchestration as PartnerLinks. Every PartnerLink is linked to a WSDL that describes the CDC table.

See Also: Oracle Application Server Adapter Concepts Guide.

Configuring the DB2 CDC Adapter In the Application Server

You can use the OC4J or WebLogic application servers to deploy the Oracle Connect Legacy adapter. This section describes how to deploy the Legacy 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

Each CDC inbound process is defined for one source table only. You create one CDC Stream Positions table for each CDC adapter. The CDC Stream Positions table contains one row for each process (for each table) to control its stream position and XA transaction. The following SQL statement is used to define the table:

```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 change the name of the table, but not the columns.

The following table describes the CDC Stream Positions table columns.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE_NAME</td>
<td>The name of the CDC table for the inbound process. You should not change the data in this column.</td>
</tr>
<tr>
<td>STREAM_POSITION</td>
<td>The last committed stream position of the CDC process for the current table. You should not change the data in this column.</td>
</tr>
<tr>
<td>PREPARED_STREAM_POSITION</td>
<td>The last prepared stream position of the CDC process for the current table. You should not change the data in this column.</td>
</tr>
<tr>
<td>ROLLBACK_STREAM_POSITION</td>
<td>In case of a Rollback Loop, this column contains the stream position of the event that occurred at the time of the rollback.</td>
</tr>
<tr>
<td>NEW_STREAM_POSITION</td>
<td>You can enter a valid stream position value in this column to manually change the current stream position. Before you update this column, deactivate the corresponding CDC process. Note: Make sure that the XID_FORMAT column has a null value. If the XID_FORMAT column is not null, do not enter a value for this column.</td>
</tr>
<tr>
<td>XID_FORMAT</td>
<td>The last prepared XID fields. If the XID_FORMAT column is null, the stream position is committed. In this case the XID_BRANCH and XID_GLOBAL columns will contain the last committed XID fields.</td>
</tr>
<tr>
<td>XID_BRANCH</td>
<td></td>
</tr>
<tr>
<td>XID_GLOBAL</td>
<td></td>
</tr>
</tbody>
</table>

You should configure the native data source that contains the CDC Stream Positions tables in your application server. The following example shows an example of a data source definition in OC4J (in the data-sources.xml file):

```xml
<native-data-source
    name="CdcContext:OracleDS"
    jndi-name="jdbc/CdcContext:OracleDS"
    description="Native CDC Context DataSource"
    data-source-class="oracle.jdbc.pool.OracleDataSource"
    user="system"
```
Configuring the DB2 CDC Adapter Connection Factory

Deploy the Oracle Connect Legacy Adapter RAR into the OC4J or WebLogic server with the name jca-legacy-adapter.

Create the appropriate connection factories in the OC4J or WebLogic application server.

This table provides a description of the connection factory 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. See Also: <em>Advanced Tuning of the Daemon</em> for details about the daemon.</td>
</tr>
<tr>
<td>workspace</td>
<td>Required. Specifies the name of an Oracle Connect server workspace to use. See Also: <em>Workspaces</em> 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>
<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. See also: 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>persistentConnection</td>
<td>By default this is set to true. You should not change this value.</td>
</tr>
<tr>
<td>keepAlive</td>
<td>By default this is set to true. You should not change this value.</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>
</tbody>
</table>
Configuring the Oracle BPEL Process Manager

This section includes the following topics:

- Setting up the Connection to the Oracle Connect Server Metadata
- Using JDeveloper to Integrate a CDC Inbound Process to the Inbound Application

Setting up the Connection to the Oracle Connect Server Metadata

Perform the following steps to set up the connection to the Oracle Connect server to provide an automatic BPEL application built by JDeveloper:

1. Open the Oracle BPEL Admin window.
2. On the Server tab, on the Configuration tab, specify the following:
   - **LegacyServer**: The IP address of the server where Oracle Connect is installed.
   - **LegacyPort**: The port number of the server where Oracle Connect is installed.
3. Repeat the previous step for each Oracle Connect server to be used by Oracle BPEL Process Manager. Use a comma as a separator between the different servers and ports.
4. Click **Apply**.
5. Restart the Oracle SOA Suite.

Using JDeveloper to Integrate a CDC Inbound Process to the Inbound Application

The following sections describe how to integrate the CDC inbound process to the inbound application:

- Configuring a CDC Adapter Partner Link
- Configuring WSDL
Configuring a CDC Adapter Partner Link

Carry out the following steps in Oracle JDeveloper to configure a CDC Adapter Partner Link.

1. Drag a Partner Link into one of the Services lanes of the visual editor.
2. In the Create Partner Link dialog box, click the Service Explorer button (the second button under WSDL Settings).
3. In the Service Explorer, expand Adapter Services.
4. Expand the connection server on which you deployed the DB2 resource adapter (this is the server where you created your CDC solution).
5. Under Adapters, expand Legacy to view a list of the Oracle Connect servers that you defined by using the Oracle BPEL Admin window.
6. Expand the Oracle Connect server with the metadata you want to check, to view a list of workspaces.
   Expand the CDC staging area workspace that you created when . This workspace will have the name with the suffix _sa.
7. Expand the CDCQueue adapter to view a list of captured tables.
8. Select the table you are working with and click OK.
9. Define the Partner Role and My Role then click OK.

Configuring WSDL

When you build an inbound Partner Link, Oracle BPEL Process Manager automatically creates the WSDL that corresponds to the captured table, including the properties of the AttuCDCActivationSpec class. These properties are the only relevant properties for a CDC inbound process. These properties, except for the TableName, 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.

The WSDL 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 ActivationSpec are used. If a value is specified by both the connection factory and the ActivationSpec, the ActivationSpec property overrides the value in the connection factory. If you want to use the value specified in the connection factory, you have to delete the property from the WSDL.

The following is an example of a CDC inbound WSDL:

```xml
<?xml version='1.0' encoding='UTF-8'?>
<definitions name="nation" targetNamespace="http://xmlns.oracle.com/pcbpel/dlg1_CDCQueue/nation" xmlns="http://schemas.xmlsoap.org/wsdl/
xmlns:legacyReq="noNamespace://dlg1_CDCQueue"
xmlns:tns="http://xmlns.oracle.com/pcbpel/dlg1_CDCQueue/nation"
xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
xmlns:jca="http://xmlns.oracle.com/pcbpel/wsdl/jca/"
  <types>
</types>
```

<xsd:element name="nationStream" type="nationStreamDescription"/>
<xsd:complexType name="nationStreamDescription">
  <xsd:sequence>
    <xsd:element name="nation" type="nationOut" minOccurs="1" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="nationOut" type="nationOut"/>
<xsd:complexType name="nationOut">
  <xsd:sequence>
    <xsd:element minOccurs="0" maxOccurs="1" name="header" type="nation__header"/>
    <xsd:element minOccurs="0" maxOccurs="1" name="data" type="nation__data"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:element name="nation__data" type="nation__data"/>
<xsd:complexType name="nation__data">
  <xsd:attribute name="n_nationkey" type="xsd:int" use="required"/>
  <xsd:attribute name="n_name" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="25"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
  <xsd:attribute name="n_regionkey" type="xsd:int" use="required"/>
  <xsd:attribute name="n_comment" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="152"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
<xsd:element name="nation__header" type="nation__header"/>
<xsd:complexType name="nation__header">
  <xsd:attribute name="timestamp" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="26"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
  <xsd:attribute name="tableName" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="64"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
  <xsd:attribute name="operation" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="12"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
  <xsd:attribute name="transactionID" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:maxLength value="64"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
<xsd:attribute name="context" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="32"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>

<xsd:attribute name="agent_context" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="64"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>

<xsd:attribute name="terminalID" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="4"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>

<xsd:attribute name="taskID" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="8"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>

<xsd:complexType>
    <xsd:schema>
        <message name="event">
            <part name="event_nation" element="legacyReq:nationStream"/>
        </message>

        <portType name="nationPortType">
            <operation name="nation">
                <input name="Event_nation" message="tns:event"/>
            </operation>
        </portType>

        <binding name="nationJCABinding" type="tns:nationPortType">
            <jca:operation
                ActivationSpec="com.oracle.adapter.oracle.AttuCDCActivationSpec"
                TableName="nation" JdbcDataSource="jdbc/CdcContextOracleDS"
                CdcStreamPositionsTable="" EisName="dlgl_CDCQueue" ServerName="server1-xp"
                PortNumber='2551' UserName="" Password="" Workspace="dlgl_sa" RetryInterval="15"
                ConnectTimeout="0"/>
            <input/>
        </jca:operation>
    </binding>

    <service name="nationService">
        <port name="nationPort" binding="tns:nationJCABinding"/>
    </service>
</xsd:complexType>
</xsd:schema>
Troubleshooting Rollback Loops

When a problem occurs in the CDC resource adapter work thread, the thread will wait for a defined period of time (retryInterval) then it will retry 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, the resource adapter always waits for a defined period of time (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, the CDC resource adapter take steps to carry out 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.

Note: In this example, the element <nation> is defined as an array. The maximum number of elements in this array is defined by the maxRecords property in the corresponding CDC Queue adapter.
Runtime Tasks and Troubleshooting

Runtime tasks are executed after you install and configure the OracleAS CDC Adapter for DB2. This chapter describes how to carry out 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 need to follow to make changes to the table. During this operation you will need to go back and change some of the configurations 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 drop-down 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.
Select a table from the right pane then click the left arrow to move a selected table into the left pane and remove it from the list of tables to be captured.

Select a table from the left pane then click the right arrow to move a selected table into the right pane and add it to the list of tables to be captured.

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 runtime. 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. See **Using JDeveloper to Integrate a CDC Inbound Process to the Inbound Application**.

---

**Handling Metadata Changes**

When you make changes to the metadata, you need to be sure 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 carry out 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 of 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 backend 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) machine, do the following:
     - Run Oracle Studio, and open the Design perspective.
     - Edit the Metadata for the Router’s Data source.
     - Expand the table list and edit the metadata for the table.
     - If you are adding a new column, make sure to add it to the end of the **COLUMN** list. This operation can also be done using the Source view. Make sure you select the correct datatype.
     - If you are modifying a datatype, make sure to select the corresponding datatype when making the modification.
     - For cases where you can redeploy the solution:
Run Oracle Studio, and open the Solution perspective.
Open the CDC solution project.
Click **Implement** and then click **Stream Service**.
Run the wizard.
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. Make sure *not* to delete the SERVICE_CONTEXT and CONTROL_TABLE files.

5. Reactivate the solution using Oracle Studio. 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 carry out 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 carry out 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 carry out the maintenance activities:

1. Deactivate the OracleAS CDC Solution with the staging area you working with. See Activating and Deactivating Solution Workspaces.

2. Run the DCHECK utility on each file to make sure it's 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
   ```
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. This value can be used 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>
<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>
</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>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>
</tbody>
</table>
Daemon Life-Cycle Tasks

Most of the daemon runtime 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 Runtime
- 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.

---

**Note:** The daemon is started on the IBM z/OS platform. It cannot be started remotely using Oracle Studio.

---

**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.

**See Also:** Starting the Daemon for details about the `ATTDAEMN` JCL

---

**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 Runtime**

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

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 5–3 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. See Also: 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. See Also: Appendix B, &quot;Advanced Tuning of the Daemon&quot; for details about the configuration settings.</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. See Also: 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 details 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 username 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. Note: 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:

Table 5–4 Workspace Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Workspace Configuration</td>
<td>Opens the daemon editor to enable you to reconfigure the workspace. See Also: Appendix B, &quot;Advanced Tuning of the Daemon&quot; for details about the configuration settings.</td>
</tr>
</tbody>
</table>
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>
<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 details see Daemon Logs.</td>
</tr>
<tr>
<td>View Events</td>
<td>Displays the server events log.</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. Note: It is recommended to use this option with caution, as it may lead to data loss.</td>
</tr>
<tr>
<td>Remove</td>
<td>Removes the selected workspace from the Runtime Explorer.</td>
</tr>
<tr>
<td>Disable</td>
<td>Disables the selected workspace.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Refreshes the display.</td>
</tr>
</tbody>
</table>

Daemon Logs
Oracle Connect produces a number of logs that you can use to troubleshoot problems. The daemon manages the following logs:

- 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 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 have been generated.
- **debug**: The log displays users that log in and out, any errors that have been 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 will continue 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 need to 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 kill 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 kill 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 username and password.
Action: Edit the user profile in Oracle Studio to specify a valid username 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 username and password.

See Also: Daemon Security.
C00F: Anonymous client logins are not allowed.
  **Cause:** The daemon is configured to require a valid username 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 username 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 username/password supplied when logging on to the daemon.
  **Action:** See the daemon log file for the reason that the username/password were not accepted.

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

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

  **Action:** Make sure the daemon is started from an APF-authorized account that is allowed to check for system usernames 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 steplibs 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.

  See Also: Daemon Logging.

C043: Failed to connect to host %s; %s
Resolving Specific Errors

**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 oc4j-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**

**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 username given is not allowed to use the requested server. Use an authorized username.

**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 for the reason that 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 abort. The currently active transaction is aborted as well.

**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 abort. The currently active transaction is aborted as well.

**Action:** Normally, Oracle Connect automatically tries to create a new 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 crash not related to callable user code, contact Oracle Support Services.

**C072: Reconnect to server %s**

**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**
Resolving Specific Errors

Runtime Tasks and Troubleshooting

5-21

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.xaInvalidArgument, 15 - client.autogenRejected, 16 - 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 a number of 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 section 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 a number of 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>
<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 up 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 already 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 already 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:
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 if you want 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 avoids the need 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 kill 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 killed. This must be done at the system level.

- **Number of prestarted servers in pool**: The number of server processes that are prestarted for this workspace when the daemon starts up. 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**: 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**: The number of sub-tasks for a server that are prestarted for this workspace when the daemon starts up. 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:

![The binding Properties tab](image)

The binding environment is divided into the following categories:

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

<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>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 events</td>
<td>Select this to add a timestamp on each event row in the log.</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>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>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>
<tr>
<td>Transaction extended logging</td>
<td>Select this for the transaction manager to write additional information about transactions to the log.</td>
</tr>
</tbody>
</table>
General

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

<table>
<thead>
<tr>
<th>Table 6–3 General Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>NAV_UTIL editor</td>
</tr>
<tr>
<td>Temporary Dir</td>
</tr>
<tr>
<td>Year 2000 policy</td>
</tr>
<tr>
<td>Cache buffer size</td>
</tr>
</tbody>
</table>

Language

The following table lists the parameters that define globalization support:

<table>
<thead>
<tr>
<th>Table 6–4 Language Settings Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>Language</td>
</tr>
<tr>
<td>Code Page</td>
</tr>
</tbody>
</table>
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.

Parallel Processing

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

---

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLS String</td>
<td>Specifies the codepage used by a field whose data type is defined as nlsString. Use this for a field whose codepage is other than that of the computer codepage. This parameter includes the following values:</td>
</tr>
<tr>
<td></td>
<td>- The name of the codepage.</td>
</tr>
<tr>
<td></td>
<td>- Whether the character set reads from right to left (as in middle eastern character sets).</td>
</tr>
<tr>
<td></td>
<td>The default is false.</td>
</tr>
</tbody>
</table>
### Query Processor

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

**Table 6–5 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>Disable threaded read ahead (QP)</td>
<td>Select this to disable read-ahead functionality.</td>
</tr>
<tr>
<td>Disable query read ahead (QP)</td>
<td>Select this to disable read-ahead functionality for components using Query Processor services.</td>
</tr>
<tr>
<td>ODBC async execution</td>
<td>Select this to enable ODBC asynchronous execution.</td>
</tr>
<tr>
<td>Disable QP parallel execution</td>
<td>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>Hash parallelism</td>
<td>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>

**Table 6–6 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 reshowed on every getRows, if the rowset is active.</td>
</tr>
<tr>
<td>Compile after load</td>
<td>Select this to always compiles an Oracle Connect procedure or view after it is read.</td>
</tr>
<tr>
<td>Ignore segments bind failure</td>
<td>This property determines how Oracle Connect responds when the execution of one of the segments of a segmented data source fails:</td>
</tr>
<tr>
<td></td>
<td>Select this to Log a message and continue execution. This is the default setting.</td>
</tr>
<tr>
<td></td>
<td>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.</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>
</tbody>
</table>
### Table 6–6 (Cont.) Query Processor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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 will continue 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 will end. Enter 0 to indicate that there is no timeout and the continuous query will 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>
Arithmetic fixed precision

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 will 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 datatypes 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.

Parser depth

The maximum depth of the expression tree. The default is 500.

Token size

Enter the maximum length of a string in an SQL query. The minimum value is 64. The default value is 350.

Insert from select commit rate

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.

Disable SQS cache

Select this to always read compiled Oracle Connect procedures and views from a disk. In this case, they are not saved in the cache.

Procedures cache size

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 Disable SQS cache size is selected.

Expose XML fields

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.

- **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 Expose XML fields is selected.
**Transaction**

The following table lists the parameters that define transaction support:

<table>
<thead>
<tr>
<th><strong>Table 6–7 Transaction Parameters</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Parameter</strong></td>
</tr>
<tr>
<td>Transaction extended logging</td>
</tr>
<tr>
<td>Commit on destroy</td>
</tr>
<tr>
<td>Disable 2PC</td>
</tr>
<tr>
<td>User commit confirm table</td>
</tr>
<tr>
<td>Transaction log file</td>
</tr>
<tr>
<td>Recovery Delay</td>
</tr>
<tr>
<td>Time limit</td>
</tr>
</tbody>
</table>

**Conversions**

Select one:

- **No conversion**: Select this if you want all transactions to remain as sent. This is selected by default
- **Convert all to distributed**: Select this to convert all simple transactions into distributed transactions.
- **Convert all to simple**: Select this to convert all distributed transactions into simple transactions.

**Tuning**

The following table lists the parameters that define the tuning:

<table>
<thead>
<tr>
<th><strong>Table 6–8 Tuning Parameters</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Parameter</strong></td>
</tr>
<tr>
<td>Dsm maximum buffer size</td>
</tr>
<tr>
<td>Dsm maximum Sort buffer size</td>
</tr>
</tbody>
</table>
### Table 6-8 (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 at one time 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.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The hash join optimization strategy results in a number of 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 6-9 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.</td>
</tr>
<tr>
<td><strong>Note:</strong> 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>
<td></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, any 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 usernames 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**

Setting design time security is 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 Runtime 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 privleges 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. 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 ATTSRVR 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,Vol=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
```

Make sure 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**. 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.
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.

   The Metadata tab is shown in the following figure:

7. Right-click the table you want to view in the Metadata Explorer and select Edit.

   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*

<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></td>
</tr>
<tr>
<td>Data file location</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>
</tbody>
</table>
Use the **Columns** tab to specify metadata that describe the table columns. The tab is divided into the following:

- **Column Definition Section**
- **Column Properties**

The **Columns** tab is shown in the following figure:
**Figure A–2 The Columns Tab**

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>
<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 drop-down box listing the possible data types.</td>
</tr>
<tr>
<td>Size</td>
<td>The size of the column</td>
</tr>
<tr>
<td>Scale</td>
<td>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.</td>
</tr>
<tr>
<td>Dimension</td>
<td>The maximum number of occurrences of a group of columns that make up an array. The (+) to the left of a column indicates a group field. This type of field will have a Dimension value. Click (+) to display the group members.</td>
</tr>
<tr>
<td>Offset</td>
<td>An absolute offset for the field in a record</td>
</tr>
</tbody>
</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 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 where the currently selected column</td>
</tr>
<tr>
<td>Down</td>
<td>Moves your selection to the column directly below where 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. Follow these steps for displaying the column properties.

**To display the 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 some of the properties available for selected columns.
<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>

Properties
You can index properties for each index column. Follow these steps for displaying the properties for each index.

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.

This properties displayed at the bottom of the tab describe the index or segment. The properties available depend on the data source.

Statistics Tab
Use the Statistics tab to update metadata statistics for a table. The Statistics tab is shown in the following figure:
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:

### Table A–8  Columns group Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column name</td>
<td>The columns in the table.</td>
</tr>
</tbody>
</table>
Use the Indexes group box to specify cardinality for the columns in each of the indexes in the table:

<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:

**Figure A–5 The Update Statistics screen**

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 A–10  Type Components**

<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, assuming that 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>

**Table A–11  Resolution Components**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>Only information about the table and indexes is collected. Information for partial indexes and columns is not collected.</td>
</tr>
<tr>
<td>All columns and indexes</td>
<td>Information about the table, indexes, partial indexes and columns is collected.</td>
</tr>
<tr>
<td>Select columns and indexes</td>
<td>Enables you to select the columns and indexes for which you want to collect statistics. In the enabled list of columns or indexes, left click those columns you want included (you can use shift-click and control-click to select a number of columns or indexes).</td>
</tr>
</tbody>
</table>

**Advanced Tab**

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.
  - **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.
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 that you want to edit, and select Open.

The General properties editor is displayed.

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

- **default**: Select default to use the default data source behavior for this parameter.

---

**Adapter Metadata**

- **default**: Select default to use the default data source behavior for this parameter.
The **General** properties are listed in the following table:

<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>■ kerbv5</td>
</tr>
<tr>
<td></td>
<td>■ none</td>
</tr>
<tr>
<td></td>
<td>■ basic password</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 that you want 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**

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>
Table A–13  (Cont.) 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 Parameters</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>

Interaction Advanced Tab

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.
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 that you want 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:
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>■ Name: The name of the field</td>
</tr>
<tr>
<td></td>
<td>■ Type: The data type of the field. 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>
<tr>
<td>Float</td>
<td>Int</td>
<td>Long</td>
</tr>
</tbody>
</table>
Table A–15  (Cont.)  Valid Data Types

<table>
<thead>
<tr>
<th>Binary</th>
<th>Boolean</th>
<th>Byte</th>
</tr>
</thead>
<tbody>
<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>

**Adapter Metadata XML**

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 that you want 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*

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

See *CDC Queue Adapter Properties* for an explanation of the attributes for the CDC Queue adapter that you should edit.
Advanced Tuning of the Daemon

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

### Table 6–10 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 machine(s) where the daemon is listening. If no IP address is entered, the daemon will listen 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 6–10 (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 <strong>DESCRIBE</strong> should be completed in a few seconds as opposed to call like a <strong>GETROWS</strong> 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. If you want 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 6–4  The Logging tab**

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

**Table 6–11  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 will be 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>
The following tokens can appear in the log file template and will be replaced accordingly:

- `%A`: workspace name
- `%D`: date (ymmd)
- `%I`: instance number of the given workspace server
- `%L`: server account login directory
- `%P`: server process ID
- `%T`: time (hh:mm:ss)
- `%U`: server account name (username)

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:

**Figure 6–5  The Daemon Security tab**

The Daemon Security tab comprises fields, as listed in the following table:
Workspaces

A daemon can include a number of 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.

Table 6–12 Daemon Security tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrators privileges</td>
<td>Identifies the users (accounts) allowed to perform administrative tasks (tasks that require administrative login).</td>
</tr>
<tr>
<td>All users</td>
<td>Enables all users to access the daemon and change the settings.</td>
</tr>
<tr>
<td>Selected users only</td>
<td>Identifies the names of users (accounts) and groups that can be administrators. If a user is not specified, the account from which the daemon was started is considered the administrator. Note that the daemon does not require the user to log in to the account on the system, but to log in to the daemon using the account name and password.</td>
</tr>
<tr>
<td>Machine access</td>
<td>Manages access to the computer.</td>
</tr>
<tr>
<td>Allow anonymous login</td>
<td>Whether workspaces allow anonymous logins (without user name/password entries). For the optimal level of security, keep this option unchecked and define a username for the Daemon Administrators parameter. If unchecked, then no workspace can have an anonymous client. If checked, then a particular workspace allows anonymous clients.</td>
</tr>
<tr>
<td>Cached password</td>
<td>Enables login passwords to be cached. This enhances performance by reducing login times for future connections from the same client in a session.</td>
</tr>
<tr>
<td>Encryption methods</td>
<td>Specifies the encryption method used to send information across the network. The default is an asterisk (*), meaning that all methods are acceptable. If an encryption method is specified, it must be used. The RC4 and DES3 protocols are currently supported.</td>
</tr>
</tbody>
</table>

1 The name is prefixed with '@', to utilize the operating system GROUP feature.
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**.

   **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 **General** tab is shown in the following figure:

**Figure 6–6  The General tab**

---

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info</td>
<td></td>
</tr>
</tbody>
</table>
### Table 6–13  (Cont.)  General Tab Components

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workspace name</td>
<td>The name used to identify the workspace. <strong>Note:</strong> 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.</td>
</tr>
<tr>
<td>Description</td>
<td>A description of the workspace.</td>
</tr>
<tr>
<td>Startup script</td>
<td>The full path name of the script that starts the workspace server processes. The script specified here must always activate the <code>nav_login</code> procedure and then run the server program (<code>svc</code>). 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.</td>
</tr>
<tr>
<td>Server type</td>
<td>This field is not applicable for use with Oracle AS Adapters for DB2.</td>
</tr>
<tr>
<td>Workspace binding name</td>
<td>This field is not applicable for use with Oracle AS Adapters for DB2.</td>
</tr>
<tr>
<td>Timeout parameters</td>
<td>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. <strong>See Also:</strong> Control for details about the Daemon Control section.</td>
</tr>
<tr>
<td>Client idle timeout</td>
<td>The maximum amount of time a workspace client can be idle before the connection with the server is closed.</td>
</tr>
<tr>
<td>Connect timeout</td>
<td>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.</td>
</tr>
<tr>
<td>Call timeout</td>
<td>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 <code>DESCRIBE</code> should be completed in a few seconds as opposed to call like a <code>GETROWS</code> 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>

### 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 will be replaced accordingly:

- %A: workspace name
- %D: date (yymmdd)
- %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 (username)

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 - providing a detailed trace of every 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 timestamp for every entry to the server log file.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max number of rows in a table that can be read</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.</td>
</tr>
<tr>
<td>Max number of rows allowed in a table before scan is rejected</td>
<td>Select the maximum number of table rows that can be scanned. This parameter has different behavior for query optimization and execution.</td>
</tr>
</tbody>
</table>

- 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.
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 up a number of servers for future use, prior to 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:
Figure 6–7  The Server Mode tab

The Server Mode tab has the following fields:
Table 6–14 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 up. The daemon supports the following server modes:</td>
</tr>
<tr>
<td></td>
<td>■ 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.</td>
</tr>
<tr>
<td></td>
<td>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>■ multiClient: Clients share a server process and are processed serially. This mode has low overhead because the server processes are already 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>■ multiThreaded: This mode is not applicable for use with OracleAS Adapter for DB2.</td>
</tr>
<tr>
<td></td>
<td>■ reusable: 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.</td>
</tr>
<tr>
<td></td>
<td>This mode does not have the high overhead of single-client mode because the servers are already 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>Note: The other modes can be set so that the server processes are reusable. The number of times a process can be reused is controlled by the Reuse limit field value.</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 avoids the need 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 singleClient.</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 multiClient or multiThreaded.</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 up. 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 killed, 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 killed. 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 a number of 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 if you want to limit the number of servers that can be reused. If this is selected, the Reuse limit parameter is available.
### Table 6–14 (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>
</tbody>
</table>
| 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 kill 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 killed. This must be done at the system level. |
| Server Provisioning           | **Initial number of servers:** The number of server processes that are prestarted for this workspace when the daemon starts up. 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. |

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 avoids the need 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.
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 6–14 (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 a number of 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>
</tbody>
</table>

Resource limitations

Number of sub-tasks | The number of sub-tasks for a server that are prestarted for this workspace when the daemon starts up. 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). |
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 6–8 The Security tab*

The **Security** tab has the following fields:
### Table 6–15  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 if you want 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 if you want 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 a number of 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 machine and the CDC Queue adapter is created on the staging area machine. 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 machine. 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.

See OracleAS CDC Adapter Configuration Properties for a description of the 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 machine.

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.

See the following topics for a description of the configuration properties:

- 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 ‘-‘.
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 dataset is used to keep track of the DB2 logs.

CDC Queue Adapter Properties
The CDC Queue adapter is a data base 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>datasource</td>
<td>string</td>
<td></td>
<td>The name of the data source for the OracleAS CDC adapter. 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>
</tr>
</tbody>
</table>
Change Router Properties

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>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>retryInterval</td>
<td>int</td>
<td>2</td>
<td>The polling interval for the database journal. When no events are received in the database journal, the adapter waits for the amount of time (in seconds) that is indicated for this property.</td>
</tr>
<tr>
<td>getTransactionInfo</td>
<td>boolean</td>
<td>true</td>
<td>When set to true, transaction information (begin, commit, rollback) is returned.</td>
</tr>
<tr>
<td>getBeforeImage</td>
<td>boolean</td>
<td>false</td>
<td>When true, before image information is returned.</td>
</tr>
<tr>
<td>realTime</td>
<td>boolean</td>
<td>true</td>
<td>When true, this reduces latency in getting change events, however it also increases the polling of the database journal. The change router asks for N events from the adapter. If the adapter finds fewer than N events in the database journal and realTime is true, these events are immediately returned to the change router. If realTime is false, the adapter polls the journal again after waiting for the number of seconds indicated in the retryInterval.</td>
</tr>
<tr>
<td>logLevel</td>
<td></td>
<td></td>
<td>The logging level. 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>checkTimeoutEveryNEvents</td>
<td>int</td>
<td>100</td>
<td>The number of events that occur before the CDC adapter checks the timeout value. For example, if this property is set to 100, the adapter will check the timeout value after 100 events have taken place. If the amount of time set in the timeout property is past, the adapter will timeout.</td>
</tr>
</tbody>
</table>
Table 6–17 (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 (see Referential Integrity Considerations) results in rolled-back changes eliminated naturally by means of compensating change records generated by the adapter in case of a rollback. Consult the respective CDC adapter documentation for details.</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>maxDeletedEventsInBatch</td>
<td>int</td>
<td>500</td>
<td>Controls how many expired change records are delete in a single pass. This number may need to be lowered in some rare cases in order 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>(in Kb) 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>maxStagingMemory</td>
<td>int</td>
<td>10000</td>
<td>(in Kb) 0 Specifies how much memory in total can be used for storing active transactions (ones that have not yet committed or rolled back).</td>
</tr>
</tbody>
</table>
### Configuration Properties

**Table 6–17 (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>• 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 will be stored. This directory also stores off-loaded transactions as well as timed-out transactions and error files.</td>
</tr>
<tr>
<td>transactionTimeout</td>
<td>int (in seconds)</td>
<td>3600</td>
<td>Specifies how long can a transaction be active without getting new events. This parameter should be set according to the corresponding setting of the captured database. In particular, this setting must not be lower than the database's transaction time-out setting as this may lead to the loss of transactions.</td>
</tr>
<tr>
<td>useType</td>
<td>enum</td>
<td>sqlBbased 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 events from it for processing. If no routers are specified, any valid Oracle Connect user for the workspace may get change events from the 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>
</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 events to it for processing. If no routers are specified, any valid Oracle Connect user for the workspace may get change events from the adapter. To add the list of users in Oracle Studio, expand the sender 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>
</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 at the end of 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 STREAM_POSITIONS 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 at the end of each ETL round once 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 needs to be stored back into the STREAM_POSITION table for the next ETL round. This is done with:

\[
\text{update STREAM\_POSITIONS set context='C' where application\_name='A' and table\_name='Ti';}
\]

This value can be stored more frequently during the ETL process as needed. The general guideline is that once change record data has been 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:
|             | \(<yyyyymmdd>T<hhmmss>.<nnnn><cccccc>\) |
|             | Where: |
|             | • \(<yyyyymmdd>T<hhmmss>\) 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). |
| operation   | This column lists the operations available for the CDC adapter. The available operations are: |
|             | • INSERT |
|             | • DELETE |
|             | • BEGIN |
|             | • UPDATE |
|             | • BEFOREIMAGE |
|             | • COMMIT |
|             | • ROLLBACK |
Transaction Support

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 Connect and DB2 Security.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>transactionID</td>
<td>The operation’s transaction ID.</td>
</tr>
<tr>
<td>tableName</td>
<td>The name of the table where the change was made. For INSERT, UPDATE, and BEFOREIMAGE operations, the owner name and then the table name are displayed. For COMMIT and ROLLBACK operations, this value is the same as the OPERATION value.</td>
</tr>
<tr>
<td>RBA</td>
<td>The event LRBA</td>
</tr>
<tr>
<td>timestamp</td>
<td>The date and time of the occurrence.</td>
</tr>
</tbody>
</table>
The OracleAS CDC Adapter for DB2 supports a number of 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 6–20  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</td>
<td>string</td>
</tr>
<tr>
<td>[CHARACTER]</td>
<td>int</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 [CHARACTER]</td>
<td>string</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 Codepage**

The language and codepage parameters are accessed from the computer where Oracle Studio is installed.

Perform the following steps to define the required language and codepage:

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:
From the Language list, select the NLS supported language to use in this binding. Valid values are listed in the Language Name column of NLS Language Codes table.

From the Codepage list, select the codepage that you want 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 will also change. Check to be sure that you want to use the selected code page with the language you selected.

If no codepage is selected, the default codepage 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 codepage is used by a field with a data type defined as nlsString. This parameter is used for a field with a codepage that is different than the machine’s codepage. This property includes values for the name of the codepage 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 codepage:

```xml
<misc nlsString="JA16EUC,false"/>
```

7. Save the change. New servers will use the language selected.

The following table lists the codepages:

<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 Codepages (EBCDIC based unless noted otherwise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>English UK</td>
<td>ENUK</td>
<td>Windows-1252</td>
<td>ISO-8859-15</td>
<td>IBM1146</td>
<td>IBM285, IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<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 Codepages (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>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>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>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>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>2</td>
<td>ISO-8859-15</td>
<td>IBM1140</td>
<td>IBM037, IBM500, IBM1140, IBM1148, IBM1047, ISO-8859-1 (ASCII based)</td>
</tr>
<tr>
<td>Italian</td>
<td>ITL</td>
<td>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>3</td>
<td>ISO-8859-7</td>
<td>IBM875</td>
<td>-</td>
</tr>
<tr>
<td>Russian¹</td>
<td>RUS</td>
<td>1</td>
<td>ISO-8859-5</td>
<td>IBM1154</td>
<td>IBM1025</td>
</tr>
</tbody>
</table>
### Table 6–21 (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 Codepages (EBCDIC based unless noted otherwise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turkish²</td>
<td>TUR</td>
<td>Windows-1254</td>
<td>ISO-8859-9</td>
<td>IBM1155</td>
<td>IBM1026</td>
</tr>
<tr>
<td>Hebrew</td>
<td>HEB</td>
<td>Windows-1255</td>
<td>ISO-8859-8</td>
<td>IBM424</td>
<td>IBM 862</td>
</tr>
<tr>
<td>Arabic</td>
<td>ARA</td>
<td>Windows-1256</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 codepage 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\Studio1\studio.exe -nl en"
Editing XML Files in Oracle Studio

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 6–9  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 that you want 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) that you want 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 6–22  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 Browse 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 file is associated with by editing this field. The XML processor will try to use the Public ID to locate the DTD, and if this fails, it will use 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 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 on one of the buttons 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>
<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>
</tbody>
</table>
| Location Hint          | The location of the XML schema of the XML file. An XML Catalog ID or a URI can be entered in this field. Click **Browse** to search for the schema you want. You can do 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 **Namespace Name** and **Prefix** fields are be filled with the appropriate values from the schema (you must leave the fields blank for this to occur).  
|                        | **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 be able to select the element from a submenu. The element or attribute will be 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, 6-7
adapter inbound configuration, 4-1
add timestamp to traced events parameter, 6-7
Artithmetic fixed precision parameter, 6-12

binary XML log level parameter, 6-7
BPEL
configuring inbound WDSL, 4-5
BPEL Process Manager, 4-1

cache
maximum queries, 6-11
cache buffer size parameter, 6-8
code page parameter, 6-8
COM maximum XML size in memory parameter, 6-14
COM maximum XML size parameter, 6-14
COM XML transport buffer parameter, 6-14
commit on destroy, 6-13
continuous query
prefix, 6-11
retry interval, 6-11
timeout, 6-11
continuous query prefix parameter, 6-11
continuous query retry interval parameter, 6-11
continuous query timeout parameter, 6-11

disk space requirements
IBM mainframe, 2-2
Windows, 2-2
dsm maximum buffer size parameter, 6-13
dsm middle buffer size parameter, 6-14
dsmMaxSortBufferSize parameter, 6-13

disabling
parameterization of constants, 6-10
parameterization of queries, 6-10

E
environment parameters
ACX trace, 6-7
add timestamp to traced events, 6-7
Artithmetic fixed precision, 6-12
cache buffer size, 6-8
code page, 6-8
COM maximum XML size, 6-14
COM maximum XML size in memory, 6-14
COM XML transport buffer, 6-14
commit on destroy, 6-13
continuous query prefix, 6-11
continuous query retry interval, 6-11

disable 2PC, 6-13
disable command reuse parameter, 6-10
disable compilation cache parameter, 6-11
disable DS property cache parameter, 6-10
disable insert parameterization parameter, 6-10
disable metadata caching parameter, 6-10
disable query parameterization parameter, 6-10
disable row mark fail fetch parameter, 6-10
disable SQS cache parameter, 6-12
disable threads parameter, 6-12
dsql
maximum queries, 6-11
cache buffer size parameter, 6-8
code page parameter, 6-8
COM maximum XML size in memory parameter, 6-14
COM maximum XML size parameter, 6-14
COM XML transport buffer parameter, 6-14
commit on destroy, 6-13
continuous query
prefix, 6-11
retry interval, 6-11
timeout, 6-11
continuous query prefix parameter, 6-11
continuous query retry interval parameter, 6-11
continuous query timeout parameter, 6-11

debug parameters
continuous query timeout, 6-11
debbug, 6-7
disable 2PC, 6-13
disable command reuse, 6-10
disable compilation cache, 6-11
disable DS property cache, 6-10
disable insert parameterization, 6-10
disable metadata caching, 6-10
disable query parameterization, 6-10
disable row mark fail fetch, 6-10
disable SQS cache, 6-12
disable threads, 6-10
dsm maximum buffer size, 6-13
dsm maximum sort buffer size, 6-13
dsm middle buffer size, 6-14
expose XML fields, 6-12
file close on transaction, 6-14
file pool size, 6-14
file pool size per file, 6-14
first tree extensions, 6-11
GDB trace, 6-7
general trace, 6-7
hash buffer size, 6-14
hash enable RO, 6-14
hash max open files, 6-14
hash primary event size, 6-14
hash secondary event size, 6-14
ignore segment bind failure, 6-10
insert from select commit rate, 6-12
language, 6-8
log file, 6-7
maximum columns in parsing, 6-11
Maximum segmented database thread, 6-11
maximum SQL cache, 6-11
minimum number of parameters allocated, 6-11
miscellaneous, 6-8
nav utility editor, 6-8
NLS string, 6-9
odbc, 6-9
oledb, 6-9
optimistic for updates, 6-11
optimizer, 6-9
optimizer goal, 6-9
optimizer trace, 6-7
parser depth, 6-12
procedures cache size, 6-12
prompt database user password, 6-10
query processor trace, 6-7
query warnings, 6-7
queryProcessor, 6-10
recovery delay, 6-13
Replace invalid XML characters, 6-15
time limit, 6-13
token size, 6-12
trace directory, 6-7
temporary directory, 6-8
tuning, 6-13
use alternate qualifier, 6-11
use global file pool, 6-14
use table filter expression, 6-11
user commit confirm table, 6-13
write empty string as null, 6-11
XML, 6-14
XML date format parameter, 6-15
XML trim char column, 6-15
year 2000 policy, 6-8
error log, binary XML log level parameter, 6-7
error log, log file parameter, 6-7
exact arithmetic, 6-12
expose XML fields parameter, 6-12
file close on transaction, 6-14
file pool size, 6-14
file pool size per file, 6-14
first row optimization, optimizer goal parameter, 6-9
first tree extensions parameter, 6-11
floating point precision, 6-12
GDB trace parameter, 6-7
general parameters
    cache buffer size, 6-8
    nav utility editor, 6-8
temporary directory, 6-8
general trace parameter, 6-7
hardware requirements
    mainframe disk space, 2-2
    mainframe memory, 2-2
    UNIX, 2-2
    Windows disk space, 2-2
    Windows memory, 2-2
    Windows processor, 2-2
hash buffer size, 6-14
hash buffer size parameter, 6-14
hash enable RO parameter, 6-14
hash joins, 6-14
hash max open files parameter, 6-14
hash primary event size parameter, 6-14
hash secondary event size parameter, 6-14
IBM mainframe
    installing, 2-6
    post-installation, 2-9
    preinstallation, 2-5
ignore segment bind failure parameter, 6-10
importing installation kit, 2-5
index cache
  dsm middle buffer size parameter, 6-14
insert from select commit rate parameter, 6-12
installation kit, importing, 2-5
installing
  IBM mainframe, 2-6
  Solaris Operating System (SPARC), 2-15

J
joins
  hash buffer size parameter, 6-14
  hash max open files parameter, 6-14

L
language parameter, 6-8
language parameters
  code page, 6-8
  environment, 6-8
  language, 6-9
  NLS string, 6-9
locking, optimistic, 6-11
log file parameter, 6-7
log files
  binary XML log level parameter, 6-7
  daemon options, B-3
  log file parameter, 6-7
logging
  daemon configurations, B-3
  optimizer strategy, 6-7

M
maximum columns in parsing parameter, 6-11
Maximum segmented database thread, 6-11
maximum SQL cache parameter, 6-11
memory
  dsm maximum buffer size parameter, 6-13
  dsm maximum sort buffer size parameter, 6-13
  dsm middle buffer size parameter, 6-14
memory requirements
  mainframe, 2-2
  Windows, 2-2
metadata
  atomic data types, D-1
  disabling retrieval from cache, 6-10
Microsoft software requirements, 2-3
minimum number of parameters allocated
  parameter, 6-11
miscellaneous parameters
  environment, 6-8

N
nav utility editor parameter, 6-8
nav utility editor, text editor, 6-8
NLS string parameter, 6-9

O
odbc environment parameters, 6-9
oledb environment parameters, 6-9
operating system requirements
  UNIX, 2-2
  Windows, 2-3
optimistic for updates parameter, 6-11
optimizer
  environment parameters, 6-9
  optimizer goal parameter, 6-9
  trace directory parameter, 6-7
  optimizer goal parameter, 6-9
  optimizer trace parameter, 6-7
Oracle Application Server requirements, 2-2

P
parallel processing parameters
  disable threads, 6-10
parameterization of constants, disabling, 6-10
parameterization of queries, disabling, 6-10
parser depth parameter, 6-12
password, setting automatic prompt, 6-10
post-installation, IBM mainframe, 2-9
post-installation, Solaris Operating System (SPARC), 2-16
preinstallation, IBM mainframe, 2-5
preinstallation, Solaris Operating System (SPARC), 2-14
procedures cache size parameter, 6-12
processor requirements
  UNIX, 2-2
  Windows, 2-2
prompt database user password parameter, 6-10

Q
queries
  disabling reuse, 6-10
  disabling saved compilation, 6-11
  enabling retrieval after failure, 6-10
  maximum cached in memory, 6-11
  maximum length of string, 6-12
  maximum size after compilation, 6-11
query optimizer
  logging strategy, 6-7
  optimizer goal parameter, 6-9
  trace directory parameter, 6-7
query processor parameters
  Arithmetic fixed precision, 6-12
  continuous query prefix, 6-11
  continuous query retry interval, 6-11
  continuous query timeout, 6-11
  disable command reuse, 6-10
  disable compilation cache, 6-11
  disable DS property cache, 6-10
  disable insert parameterization, 6-10
  disable metadata caching, 6-10
  disable query parameterization, 6-10
  disable row mark fail fetch, 6-10
disable SQS cache, 6-12
emptyStringIsNull parameter, 6-11
expose XML fields, 6-12
first tree extensions, 6-11
ignore segment bind failure, 6-10
insert from select commit rate, 6-12
maximum columns in parsing, 6-11
Maximum segmented database thread parameter, 6-11
maximum SQL cache, 6-11
minimum number of parameters allocated, 6-11
optimistic for updates, 6-11
parser depth, 6-12
procedures cache size, 6-12
prompt database user pasword, 6-10
token size, 6-12
use alternate qualifier, 6-11
use table filter expression, 6-11
query processor trace parameter, 6-7
query warnings parameter, 6-7
queryProcessor environment parameters, 6-10

R

recovery delay, 6-13
Replace invalid XML characters parameter, 6-15
requirements
UNIX hardware requirements, 2-1
UNIX software requirements, 2-2
Windows hardware requirements, 2-2
Windows software requirements, 2-3
row optimization, optimizer goal parameter, 6-9

S

security
daemon configurations, B-5
setting automatic prompt, 6-10
segmented data sources, response upon failure, 6-10
servers
configuring modes, 6-2
reusable, 6-3
Reuse limit daemon parameter, 6-3, B-13
SMS
See system managed storage
software requirements
Microsoft, 2-3
Oracle Application Server, 2-2
UNIX operating system, 2-2
Windows operating system, 2-3
Solaris Operating System (SPARC)
installing, 2-15
post-installation, 2-16
preinstallation, 2-14
sort buffer, dsm maximum sort buffer size parameter, 6-13
staging area, 1-4
stored queries, maximum in cache memory, 6-12
system managed storage, 2-5

T
temporary directory parameter, 6-8
temporary files, 6-8
threading model, 6-11
time limit parameter, 6-13
timeout
client idle, 5-11
daemon, 5-11
token size parameter, 6-12
trace directory parameter, 6-7
trace information, logging, 6-7
transaction conversions parameters, 6-13
transaction extended logging, 6-13
Transaction extended logging parameter, 6-7
transaction log file, 6-13
transactions environment parameters, 6-13
transactions parameters
commit on destroy, 6-13
disable 2PC, 6-13
recovery delay, 6-13
time limit, 6-13
transaction conversions parameters, 6-13
transaction extended logging, 6-13
transaction log file, 6-13
user commit confirm table, 6-13
tuning environment parameters, 6-13
tuning parameters
dsm maximum buffer size, 6-13
dsm maximum sort buffer Size, 6-13
dsm middle buffer size, 6-14
file close on transaction, 6-14
file pool size, 6-14
file pool size per file, 6-14
hash buffer size, 6-14
hash enable RO, 6-14
hash max open files, 6-14
hash primary event size, 6-14
hash secondary event size, 6-14
use global file pool, 6-14

U

use alternate qualifier parameter, 6-11
use global file pool, 6-14
use table filter expression parameter, 6-11
user commit confirm table, 6-13

W

Workspace server mode, B-13
write empty string as null parameter, 6-11

X

XML
environment parameters, 6-14
XML date format parameter, 6-15
XML parameters
COM maximum XML size, 6-14
COM maximum XML size in memory, 6-14
COM XML transport buffer, 6-14
Replace invalid XML characters, 6-15
XML date format parameter, 6-15
XML trim char column, 6-15
XML trim char column parameter, 6-15

Y

Y2K

See year 2000 policy parameter
year 2000 policy parameter, 6-8