# Deploying Oracle Business Intelligence Enterprise Edition with Oracle Active Data Guard

*Oracle Maximum Availability Architecture White Paper*
*March 2010*

Maximum
Availability
Architecture

Oracle Best Practices For High Availability

# Executive Overview

This paper describes the Enterprise Deployment architecture for Oracle Business Intelligence Enterprise Edition Plus (Oracle BI EE Plus). It also describes how to configure Oracle BI EE Plus with Oracle Active Data Guard.

**Oracle BI EE Plus** is a comprehensive suite of enterprise BI products that delivers a full range of analysis and reporting capabilities. Featuring a unified, highly scalable, modern architecture, Oracle BI EE Plus provides intelligence and analytics from data spanning enterprise sources and applications, empowering the largest communities with complete and relevant insight.

**Oracle Active Data Guard** is an important option for Oracle Database 11*g* Enterprise Edition that enables a physical standby database to be opened for read-only access while Redo Apply is active. This makes it possible to run queries and reports against an up-to-date physical standby database without compromising data protection or extending recovery time in the event a failover is required. Additionally, you can use the standby database to offload backups from production databases or for Quality Assurance to run other test activities that require read/write access to an exact replica of the production database. The Active Data Guard feature provides capabilities that are unique to Oracle Database, making it an attractive option for deployment in a variety of business applications.

This Maximum Availability Architecture (MAA) [1] white paper augments the Oracle documentation by profiling several use cases that are well suited to Oracle BI and Active Data Guard, and explaining the best practices for configuration and usage.

Unless stated otherwise, the enterprise topology uses Oracle BI EE with Oracle Application Server 10.1.3.4 and Oracle Database 11*g* release 11.1.0.7. Oracle BI EE supports Oracle Application Server release 10.1.3.1.0 or later. The Active Data Guard option is available with Oracle Database 11*g* Enterprise Edition and higher.

## About Oracle BI EE Plus and Oracle Active Data Guard

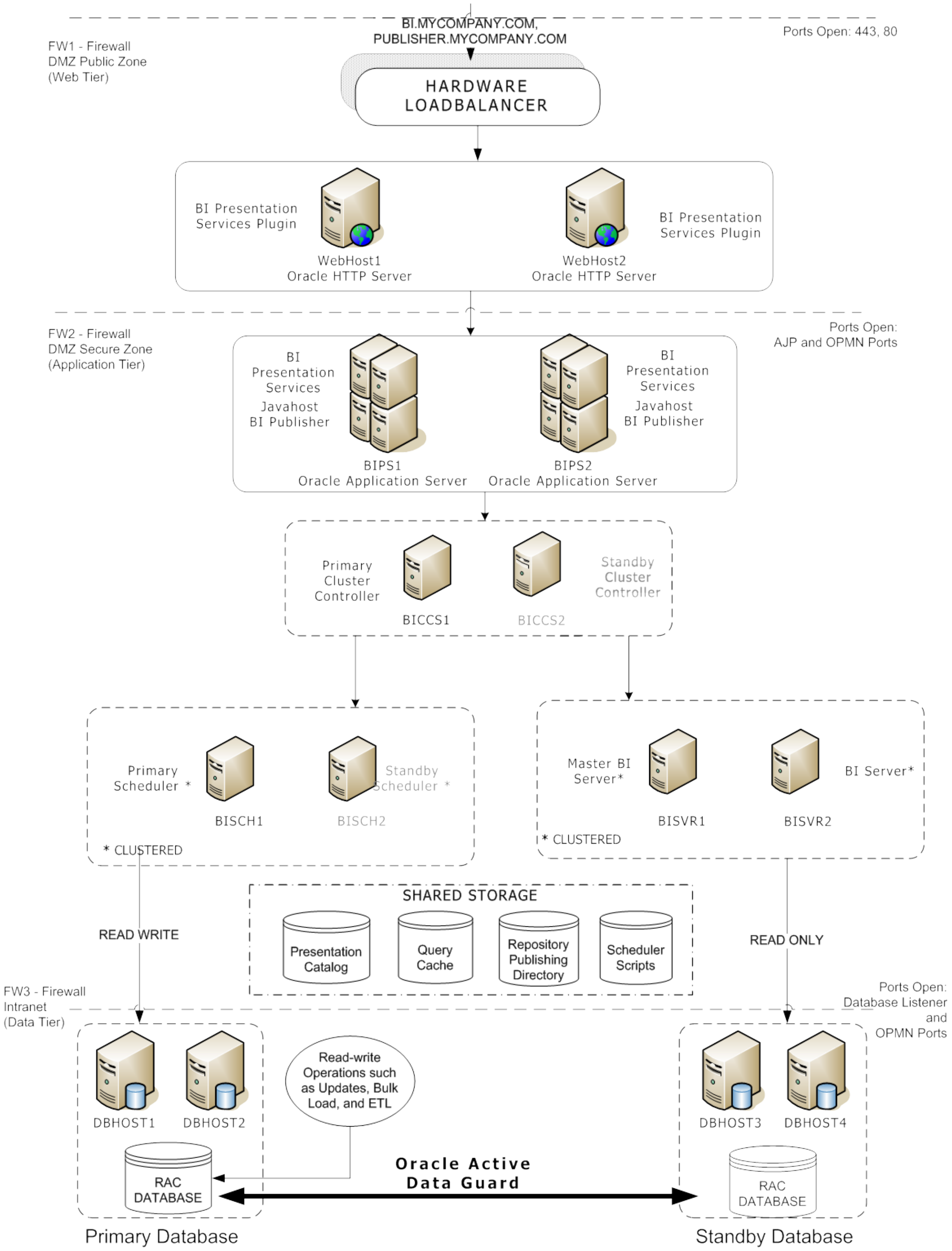
The instructions and diagrams in this guide reference the topology shown in Figure 1, to which variations may be applied. Oracle BI components are supported in a many-to-many architecture, and the component interaction is illustrated in the configuration:

- End-user Web requests are directed to one of many **BI Presentation Services** servers. In turn, each BI Presentation Services server can take advantage of the availability of multiple **BI Servers**.
- The **BI Cluster Server** allows multiple BI servers to be deployed for scalability and availability needs.
- A **Cluster Controller** serves as the entry point to the cluster servers. The server metadata is contained in the **repository publishing directory file** (`.rpd`).
- **BI Schedulers** participate in the cluster in an active-passive configuration.
- One BI Server is designated as the Master. Online changes to the rpd file are made on the **Master BI Server** and these changes are replicated to other members of the cluster. A cluster-aware cache capability offers support for a common query cache that is visible to all BI Servers in the cluster.

Cluster capability for the BI Presentation Services layer allows for the deployment of a multi-server environment to better manage large volumes of users and provide high availability. Multiple BI Presentation Services instances in the cluster can either share a common **Presentation Catalog** on a network storage device, or the catalog may be replicated across each BI Presentation Services instance. Native load balancing and failover capabilities are offered for the components of the Presentation Layer.

**Figure 1** shows the reference enterprise topology that uses Oracle BI EE with Oracle Application Server Release 10.1.3.4 and Oracle Database 11*g* Release 11.1.0.7. The testing, discussions, and examples in this paper are based on the Linux platform, but the methodologies and best practices should apply to all UNIX platforms. The sections after **Figure 1** discuss the network, Web tier, application tier, and data tier in more detail.

## Figure 1: MyBICompany Topology



## Configuring the Network for Communication between Tiers

Firewalls separate the various tiers. The following describes how to configure the firewalls to open communications between tiers.

- Configure the firewall FW1 for communication into the Web Tier from the Application Tier:
  - http://webhost1.mycompany.com:7777/
  - http://webhost2.mycompany.com:7777/
  - Oracle Notification Services (ONS) remote port. For example: 6202, on `WEBHOST1` and `WEBHOST2`.
- Configure the firewall FW2 for communication into, and out of, the Application Tier:
  - AJP ports on BIPS1 and BIPS2.

    **Note:** The AJP ports used by applications fall within the range 12501 to 12600. Ensure that all of the AJP ports used are open on the firewall between the Web server and the application. If a port is not open, an error occurs when access to the application from the Web tier is attempted (that is, when the following URL is requested `http://webhost1.mycompany.com:7777/analytics`). For example:

    ```
    mod_oc4j: request to OC4J bips1.mycompany.com:12501 failed:
    Connect failed (errno=110)
    ```

    This error creates an entry in a log file in the `ohs/logs` directory.
  - ONS remote port, for example 6201, on BIPS1 and BIPS2.
- Configure the firewall FW3 for communication from the DMZ Secure Zone to the intranet:
  - Configure access to the database using a listener that is listening on port 1521.

Table 1summarizes the ports that must be open on each firewall.

TABLE 1. PORTS THAT MUST BE OPEN ON EACH FIREWALL

| FIREWALL ZONES | PORTS | PURPOSE |
|---|---|---|
| Internet to Web Tier | HTTP and HTTPS Ports<br>(for example, 80 and 443) | To receive client requests |
| Web Tier to Application Tier | AJP Ports | WEBHOST1 and WEBHOST2, to access BIPS1 and BIPS2 AJP ports |
| | OPMN Port<br>(for example, 6201) | OPMN cluster gateway |
| Application Tier to Web Tier | OPMN Port<br>(for example, 6202) | OPMN cluster gateway |
| | HTTP and HTTPS Ports<br>(for example, 7777, 443) | BIPS1 and BIPS2 loopback access to `bi.mycompany.com:7777` |
| Application Tier to Data Tier | OPMN Port<br>(for example, 6201) | OPMN cluster gateway |
| | Database Listener Port<br>(for example, 1521) | Database Access |

## Web Tier

**Figure 1** shows that the nodes WEBHOST1 and WEBHOST2 in this tier run Oracle HTTP Server. The Oracle HTTP Servers on WEBHOST1 and WEBHOST2 are installed using the same multicast address and port on both hosts.

The Web tier also includes a load balancer router to handle external requests. External requests are sent to the virtual host names that are configured on the load balancer. The load balancer then forwards the requests to Oracle HTTP Server, which manages load balancing between the available servers.

## Application Tier

In **Figure 1**:

- Oracle Application Servers, each with an OC4J instance, are installed on machines BIPS1 and BIPS2 with the cluster configuration option.

- `BICCS1` and `BICCS2` run the Oracle BI Cluster Controller in an active-passive configuration. The BI Cluster Controller running on `BICCS1` serves as the Primary Controller Server.
- Oracle BI Servers (formerly known as Analytic Server or SAS) run on `BISVR01` and `BISVR02` in an active-active configuration. The BI Server running on `BISVR01` serves as the Master BI Server. The BI Servers access data from a back-end two-node Oracle RAC database running on `DBHOST1` and `DBHOST2`.
- BI Presentation Services (formerly known as Analytics Web Server or SAW) and BI Publisher are installed on `BIPS1` and `BIPS2` in an active-active configuration. The BI Presentation Services Plug-in and BI Publisher are deployed in the Oracle Application Server homes on machines `BIPS1` and `BIPS2`.
- `BISCH1` and `BISCH2` run the BI Scheduler in an active-passive configuration.

### Data Tier

In this tier, an Oracle RAC database runs on the nodes `DBHOST1` and `DBHOST2`. The database contains the application schemas needed by the Oracle BI components. The BI components running in the application tier access this database.

If an Oracle Active Data Guard standby database is configured in this environment, you can increase the read capacity of the system and offload the read-only workload from the primary database, running on `DBHOST1` and `DBHOST2,` to the standby database, running on `DBHOST3` and `DBHOST4`.

You can manage the read-only and read/write client connections by having a separate service names for the read/write application and reporting read-only application and starting the corresponding service based on the database role. In the configuration shown in **Figure 1**, the read/write application is connecting with a service name of `bi_rw` and the reporting application is connecting with a service name of `bi_ro`. To automate starting and stopping the service, we create an “on startup” trigger that checks the database role and starts the appropriate service.

## Preconfiguring the Database and Environment

For the BI enterprise topology, the database contains the collections of schemas used by various Oracle BI components, such as BI Server and BI Scheduler. This section describes the database and network environment preconfiguration requirements for the BI enterprise topology.

## Setting Up the Database

> **Note:** You must install the database before you can configure the Oracle BI components. For the enterprise topology, an Oracle Real Application Clusters (Oracle RAC) database is highly recommended.

### Database Host Requirements

On the hosts `DBHOST1` and `DBHOST2` in the data tier, note the following requirements:

- Install Oracle Clusterware

  For Oracle Database 11*g* Release 1 (11.1) for Linux, see the *Oracle Clusterware Installation Guide for Linux*.

- Install Oracle RAC

  For Oracle Database 11*g* Release 1 (11.1) for Linux, see the *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

- Optionally, install Oracle Automatic Storage Management (Oracle ASM).

  If you install Oracle ASM, install it for the node as a whole. Oracle recommends that you install Oracle ASM in a separate Oracle Home from the database Oracle home. The Oracle Universal Installer executable, `runInstaller`, offers this option in the Select Configuration page. Select the **Configure Automatic Storage Management** option to create a separate Oracle ASM home.

### Supported Database Version Requirements

Oracle BI EE Plus supports Oracle Database 10*g* (release 10.1.3.4 or higher). If you plan to configure Oracle BI EE Plus with Oracle Active Data Guard, you must use Oracle Database 11*g* Enterprise Edition (release 11.1.0.7 or higher).

To check the release of your database, query the `PRODUCT_COMPONENT_VERSION` view as follows:

```
SQL> SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE PRODUCT LIKE
‘Oracle%’;
```

### Database Services Recommendations

- Oracle recommends using the Oracle Enterprise Manager Cluster Managed Services Page to create database services that client applications will use to connect to the database. For complete instruction on creating database services, see the chapter about workload management in the *Oracle Real Application Clusters Administration and Deployment Guide*.

Optionally, you can use SQL*Plus to configure database services by using the following instructions:

1. Use the `CREATE_SERVICE` subprogram to create the BI read/write (`bi_rw.mycompany.com`) and read-only (`bi_ro.mycompany.com`) database services. Log on to SQL*Plus as the `SYSDBA` user and run the following statements:

   ```
   SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
        (SERVICE_NAME => ‘bi_rw.mycompany.com’,
        NETWORK_NAME => ‘bi_rw.mycompany.com’);

   SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
        (SERVICE_NAME => ‘bi_ro.mycompany.com’,
        NETWORK_NAME => ‘bi_ro.mycompany.com’);
   ```

2. Add the service to the database and assign it to the instance by using the `srvctl` command:

   ```
   prompt> srvctl add service –d bidb –s bi_rw –r bidb1,bidb2
   prompt> srvctl add service –d bidb –s bi_ro –r bidb1,bidb2
   ```

3. Start the service by using the `srvctl` command:

   ```
   prompt> srvctl start service –d bidb –s bi_rw
   prompt> srvctl start service –d bidb –s bi_ro
   ```

   **Note:** For more information about the `srvctl` command, see the *Oracle Real Application Clusters Administration and Deployment Guide*.

- Oracle recommends using a specific database service for a product suite, even when the client applications share the same database. Also, the database service used should be different than the default database service.

  **Note:** If there are multiple repositories set up in the BI environment, you must create the read/write and read-only services on each repository.

## Setting Up Shared Files and Directories

The BI components deployed in a clustered environment must share certain files and directories. A shared storage device such as NAS or SAN may be used.

**Note:** If SAN is used, a clustered file system is required.

- Presentation Catalog
  - Create a shared directory for the Presentation Catalog. All instances of BI Presentation Services in the cluster must have read/write access to this share.
  - Reference the shared Presentation Catalog as `/Shared Directory Path/OracleBIData/web/catalog/customCatalog`

    In the example, `customCatalog` is an example of the catalog name.

- Repository Publishing Directory
  - Create a shared directory for the Repository Publishing Directory. The Master BI Server must have read/write access to this directory. All other BI Servers must have read access.
  - Reference the shared directory for the Repository Publishing Directory as
    `/Shared Directory Path/OracleBIData/ClusterRpd`
- Cluster-Aware Cache
  - Create a shared directory for the global cache. All BI Servers must have read/write access to this directory.
  - Reference the shared global cache as
    `/Shared Directory Path/OracleBIData/ClusterCache`
- Scheduler Scripts
  - Create a shared directory for the Scheduler scripts. The Scheduler servers must have read/write access to this shared directory.
  - Reference the shared Scheduler scripts as
    `/Shared Directory Path/OracleBI/server/Scripts/Common`
    `/Shared Directory Path/OracleBI/server/Scripts/Scheduler`
- BI Publisher Repository
  - Create a shared directory that will serve as the root directory for the Publisher repository.
  - Reference the shared Publisher repository as
    `/Shared Directory Path/Publisher/repository`

## Installing and Configuring BI EE and Active Data Guard

This section describes how to configure Oracle BI EE with Oracle Active Data Guard.

### Installing Oracle HTTP Servers on WEBHOST1 and WEBHOST2

Use the **Advanced** option of Oracle Universal Installer to install the Oracle HTTP Server instances. Follow these steps to install the Oracle HTTP Servers on `WEBHOST1` and `WEBHOST2`.

1. Ensure that you meet the system, patch, kernel and other requirements as specified in the Oracle Application Server Installation Guide for the platform you are using.  You can find this guide in the Oracle Application Server platform documentation library for the platform and version you are using.

2. Copy the `staticports.ini` file from `Disk1/stage/Response` directory to a local directory, such as `TMP`. Provide the path to this file during the installation procedure.

3. Edit the `staticport.ini` file to assign the following custom port:

   ```
   Oracle HTTP Server port = 7777
   ```

   > **Note:** Ensure that this port is not already in use by any other service on the computer. Using the Static Ports feature to install the Application Server Tier ensures that the port assignments will be consistent, if the ports are correctly specified in the file and the port is not already in use. If a port is incorrectly specified, Oracle Universal Installer assigns the default port. If a port is already in use, Oracle Universal Installer selects the next available port.

4. Start Oracle Universal Installer. For example, on UNIX, issue the `runInstaller` command:

   The Oracle Application Server 10.1.3.1.0 Installation screen displays.

5. Specify an installation directory for the instance.

6. Select **Advanced Installation Mode**.

7. Click **Install**.

   The Select Installation Type screen displays.

8. Select Web Server and click **Next**.

   The Specify Port Configuration Options screen displays.

9. Select **Manual**, specify the location at the `staticports.ini` file, and click **Next**.

   The Specify Instance Name screen displays.

10. Specify the instance name and click **Next**.

    The Cluster Topology Configuration screen displays.

11. Check the box to configure the instance to be part of an Oracle Application Server cluster.

    Specify the multicast address and port.

    **Note:**

    - An example of a multicast address is 225.0.0.20, with port 8001. The address and port should be the same for each computer in a farm. This multicast address used should not be in use by any application or OracleAS Instance on the network that is not part of the farm.

- You must specify the same multicast address and port for the OHS installations on both WEBHOST1 and WEBHOST2.

12. Click **Next**.

    The Summary screen displays.

13. Click **Install**.

    The Configuration Assistant screen displays. After the configuration process has completed, the End of Installation screen displays.

14. Click **Exit**, and then confirm your choice to exit.

15. Verify that the installation was successful by viewing the Oracle HTTP Server instances. Start a browser and access the following:

    ```
    http://webhost1.mycompany.com:7777 and
    http://webhost2.mycompany.com:7777
    ```

**Note:** The `ORACLE_HOME/install/readme.txt` file contains the URLs for the installation and a command to verify the status of processes.

## Installing Application Server Instances on BIPS1 and BIPS2

Install Oracle Application Server instances consisting of only one OC4J instance, using the Advanced installation option of Oracle Universal Installer. Follow these steps to install application servers to create `ORACLE_HOME` on `BIPS1` and `BIPS2`. The Oracle BI Presentations Services, Presentation Services Plug-in, and BI Publisher will be deployed into the Oracle Application Server instances.

1. Ensure that you have met the system, patch, kernel, and other requirements that are specified in the Oracle Application Server Installation Guide. You can find this guide in the Oracle Application Server platform documentation library for the platform and version you are using.

2. Start Oracle Universal Installer. For example, on UNIX systems, enter the `runInstaller` command

   The Oracle Application Server 10.1.3.1.0 Installation screen displays with the **Basic Installation Mode** and the **Integrated Web Server, J2EE Web Server and Process Management** installation type selected.

3. Specify an installation directory for the instance, or leave the default.

4. Select **Advanced Install** and click **Next**.

   A confirmation dialog displays.

5. Click **Yes**.

   A process dialog displays, then the **Select Installation Type** screen displays.

6. Select **J2EE Server** and click **Next**.

   The Specify Port Configuration Options screen displays.

7. Select **Automatic** and click **Next**.

   The Administration Settings screen displays.

8. Specify an instance name for the application server instance.

   **Note:** The instance name you specify will be prepended to the host name. For example, if you specify `J2EE` as the instance name and the host name is `server1.mycompany.com`, the instance name will be `J2EE.server1.mycompany.com`.

9. Specify and confirm the administrator password for the default OC4J instance.

10. Use the OC4J instance name of home as the name for the default OC4J instance created by the installer (the default is `home`).

    **Note**:

    - The Oracle Business Intelligence Installer requires that the default instance be named home.
    - This default OC4J instance will not be clustered.

11. Check the box to designate the instance to be installed as an administration OC4J instance (the instance on which the Application Server Control Console will operate). Perform this step only on one of the machines, for example, `BIPS1`.

12. Click **Next**.

    The Cluster Topology Configuration screen displays.

13. Select the checkbox for the options Configure this OC4J instance to be part of an Oracle Application Server cluster topology and Access this OC4J instance from a separate Oracle HTTP Server.

14. Click **Next**.

15. Specify the multicast address and port.

    This step configures the multiple instances in the same Oracle Notification Service (ONS) topology. Availability notifications are shared among the participants in the topology, enabling dynamic routing from Oracle HTTP Server to OC4J and dynamic discovery of instances, as well as creation of OC4J groups

16. Click **Next**.

    The Summary screen displays.

17. Click **Install**.

    The **Preparing to Install** dialog displays, and then the **Install** screen displays.

18. The Configuration Assistant screen displays. When the configuration process completes, the End of Installation screen displays.

19. Click **Exit**, and then confirm your choice to exit.

20. Verify that the installation was successful by viewing the instance in Oracle Enterprise Manager 10*g*. Start a browser, log in to the Application Server Control Console, and view the application server instance at:

    `http://webhost1.mycompany.com:7777/em`

**Note:** The `ORACLE_HOME/install/readme.txt` file contains the URLs for the installation and a command to verify the status of processes.

## Configuring Oracle HTTP Servers with the Load Balancing Router

Load balancing refers to the process of distributing incoming service requests over server instances within a cluster. You must configure the virtual servers (`bi.mycompany.com` and `publisher.mycompany.com`) to receive client requests and balance them to the two Oracle HTTP Server instances on the Web tier. See the *Oracle Containers for J2EE Configuration and Administration Guide* for instructions about configuring the load balancer, and follow the instructions in this section to configure the Oracle HTTP server with the Load Balancing Router.

The following steps describe how to configure the Load Balancing Routers to:

- `bi.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the runtime BI components. Traffic to SSL is configured. Clients access this service using the address `bi.mycompany.com:443`. This virtual server is defined on the load balancer.
- `publisher.mycompany.com` is a virtual server name that acts as the access point for all HTTP traffic to the runtime BI Publisher components. Traffic to SSL is configured. Clients access this service using the address `publisher.mycompany.com:443`. This virtual server is defined on the load balancer.
- Balance requests with SSL acceleration to `WEBHOST1` and `WEBHOST2` on port `7777`.

To configure the Oracle HTTP Server with the load balancing router, perform the following steps on `WEBHOST1` and `WEBHOST2`:

1. Open the following Oracle HTTP Server configuration file:

   `ORACLE_HOME/Apache/Apache/conf/httpd.conf`

2. Perform the following steps:

   a) Add the `LoadModule certheaders_module` directive for the appropriate platform:

   - UNIX Apache 1.3:

     ```
     LoadModule certheaders_module
     libexec/mod_certheaders.so
     ```

   - UNIX Apache 2.0:

     ```
     LoadModule certheaders_module modules/mod_certheaders.so
     ```

   b) Add the following lines to the Oracle HTTP Server configuration file for the Apache version (Apache 1.3 or Apache 2.0) you are using to create a `NameVirtualHost` directive and a `VirtualHost` container for `bi.mycompany.com` and port `443`.

   **Note**: The best practice recommendation is to create the `VirtualHost` directives at the end of the `httpd.conf` file.

   **Apache 1.3:**

   ```
   NameVirtualHost *:7777
   <VirtualHost *:7777>
     ServerName bi.mycompany.com
     Port 443
     ServerAdmin you@your.address
     RewriteEngine On
     RewriteOptions inherit
     SimulateHttps On
   </VirtualHost>

   NameVirtualHost *:7777
   <VirtualHost *:7777>
     ServerName bi.mycompany.com
     Port 7777
     ServerAdmin you@your.address
     RewriteEngine On
     RewriteOptions inherit
   </VirtualHost>
   ```

```
NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName publisher.mycompany.com
  Port 443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
  SimulateHttps On
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName publisher.mycompany.com
  Port 7777
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

### Apache 2.0 (UNIX):

```
NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName bi.mycompany.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
  SimulateHttps On
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName bi.mycompany.com:7777
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

```
NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName publisher.mycompany.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
  SimulateHttps On
</VirtualHost>

NameVirtualHost *:7777
<VirtualHost *:7777>
  ServerName publisher.mycompany.com:7777
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

**Note:** The `LoadModule` directives (in particular, the `LoadModule rewrite_module` directive) must be added to the `httpd.conf` file at a location preceding the `VirtualHost` directives. The Oracle HTTP server must load all modules before it can execute the directives in the `VirtualHost` container.

3. Add the `LoadModule rewrite_module` directive before the `LoadModule certheaders_module` directive.

4. Save the `httpd.conf` file.

5. Restart the components by issuing these commands in `ORACLE_HOME/opmn/bin`:

   ```
   opmnctl stopall
   opmnctl startall
   ```

6. Verify that you can access these URLs:

   https://bi.mycompany.com:443/j2ee/
   https://publisher.mycompany.com:443/j2ee/

## Configuring the Cluster Gateway on WEBHOST1,2 and BIPS1,2

This step is optional.

Firewalls separate the various tiers. To enable communications between clusters on the tiers separated by a firewall, a cross-topology gateway needs to be configured. In a cross-topology gateway, one or more nodes on each side of the firewall are designated as entry points into the cluster.

The following instructions designate BIPS1 and WEBHOST1 as the gateway servers, but you can designate any server as the gateway server. The remote port is used for communication with the gateway server; you designate the remote port in the <*gateway*> subelement in opmn.xml, as described in the following steps.

Follow these steps to specify gateway servers on the Application Tier and the Web Tier:

1. Open the ORACLE_HOME/opmn/conf/opmn.xml file on BIPS1.

2. Create the <*gateway*> subelement, as shown in the following example:

```
<notification-server>
<port local=”6101” remote=”6200” request=”6004”/>
<ssl enabled=”true” wallet-
file=”ORACLE_HOME/opmn/conf/ssl.wlt/default”/>
<topology>
<discover list=”*225.0.0.20:8001”/>
<gateway
list=”bips1.mycompany.com:6201&ampl;bips2.mycompany.com:6201&amp;w
ebhost1.mycompany.com:6202&amp;webhost2.mycompany.com:6202/”/>
</topology>
</notification-server>
```

In the preceding example:

- Port 6201 is the OPMN remote port on BIPS1, and Port 6202 is the OPMN remote port on WEBHOST1. You must view the opmn.xml file on each server to determine the port values needed for the configuration.
- When BIPS1, BIPS2, and WEBHOST1, WEBHOST2 are configured in different subnets, a gateway entry is also required. The gateway list can consist of only one host and port from each subnet. However, for failover, you must include multiple hosts from each subnet, as shown in the example.

3. Issue this command in ORACLE_HOME/opmn/bin on BIPS1:

```
opmnctl reload
```

4. Copy the <*gateway*> sub element to the ORACLE_HOME/opmn/conf/opmn.xml location on WEBHOST1.

5. Issue this command in ORACLE_HOME/opmn/bin on WEBHOST1:

```
opmnctl reload
```

**Note:** For more information, see the discussion “Configuring Cross-Topology Gateways” in the *Oracle Containers for J2EE Configuration and Administration Guide*.

## Installing Oracle BI EE Components

This section describes installing the following Oracle BI components.

- Prerequisites
- Installing Oracle BI Cluster Controller
- Installing BI Server
- Installing BI Scheduler
- Installing BI Presentation Services , BI Javahost, and BI Publisher
- Installing BI Presentation Services Plug-in

### Prerequisites

1. On machines `BISVR1` and `BISVR2` on which the BI Server components are installed:
   - Install Java Development Kit (JDK) release 1.5 or higher.
   - Install Oracle 10g Database Client or higher and ensure that database connectivity software that BI Server uses to connect to the database is installed.
   - Source the environment file (OracleBI/setup/user.sh) so that BI Server can locate the Oracle Database clients.
2. Install Oracle Database 11*g* release 11.1.0.7 or later, including Oracle RAC.
3. Set up shared files and directories.

### Installing Oracle BI Cluster Controller

Install two instances of BI Cluster Controller on different machines. One instance will serve as the Primary Cluster Controller; and the other will serve as the Secondary Cluster Controller.

To install Oracle BI components, use the Oracle Business Intelligence Installer, as follows.

1. Run the Oracle Business Intelligence Installer.

   For example, on UNIX systems, issue the `setup.sh` command.

   The Oracle Business Intelligence Installation screen displays.
2. Specify the location for the Installation and Data for the Oracle BI Cluster Controller, select **Advanced** installation type and click **Next**.

3. Select the setup type **Custom** and click **Next**.

   a) Select the feature **Oracle Business Intelligence Cluster Controller** for installation and click **Next**.

   > **Note:** If the Advanced installation type has been chosen for installing the Oracle Business Intelligence Cluster Controller, then the Oracle Business Intelligence Systems Management (required) feature will be deployed in the Oracle Application Server installed on the machine.

   b) If you are co-locating other BI components on this machine (for example, Oracle Business Intelligence Server, Oracle Business Intelligence Scheduler, or both), then select the other desired components for installation.

### Installing BI Server

Install BI Server instances on each of the machines that you have identified to host this component. You can install a maximum of 16 BI Servers to participate in the BI Cluster.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. Run the Oracle Business Intelligence Installer.

   For example, on UNIX systems, issue the `setup.sh` command.

   The Oracle Business Intelligence Installation screen displays.

2. Specify the location for the Installation and Data for the BI Server, select **Advanced** installation type and click **Next**.

3. Select the setup type **Custom** and click **Next**.

   a) Select the feature Oracle Business Intelligence Server for installation.

   **Note:** If the Advanced installation type has been chosen for installing the Oracle Business Intelligence Server, then the Oracle Business Intelligence Systems Management (required) feature will be deployed in the Oracle Application Server installed on the machine.

   b) If you are co-locating other BI components on this machine (for example, Oracle Business Intelligence Scheduler), then select the other desired components for installation.

The installed BI Servers will be configured to participate in the BI Cluster by setting parameters in the `NQSConfig.INI` and `NQClusterConfig.INI` files on each machine that hosts the BI Server.

> **Note:** See the Configuring Oracle BI Components for Clustering, Load Balancing, and Failover section.

### Installing BI Scheduler on BISCH1 and BISCH2

Install BI Scheduler instances on each of the machines that you have identified to host this component. The BI Scheduler component participates in the BI Cluster in active-passive mode. Install BI Scheduler on two machines; one will be identified as the active node and the other as the passive node.

**Installation steps:**

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. Run the Oracle Business Intelligence Installer.

   For example, on UNIX systems, issue the `setup.sh` command.

   The Oracle Business Intelligence Installation screen displays.

2. Specify the location for the Installation and Data for the BI Scheduler, and select the **Advanced** installation type.

3. Select the setup type **Custom** and click **Next**.

   a) Select the feature **Oracle Business Intelligence Scheduler** for installation.

   b) If you are co-locating other BI components on this machine (for example, Oracle Business Intelligence Server), then select the other desired components for installation.

4. Complete the additional configuration steps.

   See the chapter about "Configuring Oracle Business Intelligence Scheduler" in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for details.

The installed BI Scheduler instances will be configured to participate in the BI Cluster.

**Post-Installation Steps:**

1. Set `ORACLE_HOME,` `TNS_ADMIN`, and `LD_LIBRARY_PATH` to reference the Oracle Client `ORACLE_HOME` and source the `user.sh` file. For example:

   ```
   Example of user.sh
   ######################################################
   # Linux: Oracle BI 32 bit mode
   ######################################################
   #set +u

   # Oracle Parameters
   #---------------------------
   # Make sure that Oracle DB 32 bit Client is installed
   ORACLE_HOME=/u01/app/oracle/product/10.2.0/dbclient
   export ORACLE_HOME
   TNS_ADMIN=$ORACLE_HOME/network/admin
   export TNS_ADMIN
   PATH=$ORACLE_HOME/bin:/opt/bin:$PATH
   ```

```
export PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH
```

2. Create a soft link from `OracleBI/server/Bin/libclntsh.10.so.1` to Oracle Client `ORACLE_HOME/lib/libclntsh.10.so.1`.

   For example:

   ```
   % ln –s ORACLE_HOME/lib/libclntsh.10.so.1
   OracleBI/server/Bin/libclntsh.10.so.1
   ```

### Installing BI Presentation Services, BI Javahost, and BI Publisher

Install BI Presentation Services, BI Javahost, and BI Publisher on each machine that you have identified to host these components following their installation.

> **Note:** BI Javahost is installed along with BI Presentation Services by the BI Installer. There is no separate component option for BI Javahost.

To install Oracle BI components, use the Oracle Business Intelligence Installer, as follows:

1. Run the Oracle Business Intelligence Installer.

   For example, on UNIX systems, issue the `setup.sh` command.

   The Oracle Business Intelligence Installation screen displays.

2. Specify the location for the Installation and Data for the BI Presentation Services, and then select the **Advanced** installation type.

3. Select the setup type **Custom** and click **Next**.

4. Select the feature **Oracle Business Intelligence Presentation Services and BI Publisher** for installation.

   **Note:**

   - If you chose the Advanced installation type for installing the Oracle Business Intelligence Presentation Services, then the Oracle Business Intelligence Systems Management (required) and Oracle Business Intelligence Client (required) features will be deployed in the Oracle Application Server installed on the machine.
   - If you are co-locating other BI components on this machine (for example, Oracle Business Intelligence Presentation Services Plug-in), then select the other desired components for installation.

The installed BI Presentation Services instances are configured to participate in the BI Cluster by setting parameters in the `instanceconfig.xml` file on each machine that hosts BI Presentation Services.

**Note:** See also the Configuring BI Presentation Services on BIPS1 and BIPS2 section in this white paper.

### Installing BI Presentation Services Plug-In on BIPS1 and BIPS2

When the BI Presentation Services component is installed, the `analytics.ear` and `analytics.war` files are installed in `OracleBI/web` on Linux or UNIX systems. Deploy the files to the J2EE container on the Application Server. Perform this deployment on all instances of the J2EE container in your Web cluster.

The BI Presentation Services Plug-in (Java Servlet), which is the entry point for Web client requests to BI Presentation Services, is deployed by the installer into an Oracle Application Server OC4J instance.

To install Oracle BI components, use the Oracle Business Intelligence Installer, as follows.

1. Run the Oracle Business Intelligence Installer.

   For example, on UNIX systems, issue the `setup.sh` command.

   The Oracle Business Intelligence Installation screen displays.

2. Specify the location for the installation and for the BI Presentation Services Plug-in and BI Office, select **Advanced** installation type.

3. Select the setup type **Custom** and click **Next**.

4. Select the feature **Oracle Business Intelligence Presentation Services Plug-in and BI Office** for installation.

   **Note:**

   - If you chose the Advanced installation type for installing the Oracle Business Intelligence Presentation Services Plug-in and BI Office, then the Oracle Business Intelligence Systems Management (required) feature will be deployed in the Oracle Application Server installed on the machine.
   - If you are co-locating other BI components on this machine (for example, Oracle Business Intelligence Presentation Services Plug-in), then select the other desired components for installation.

Now that the Oracle Business Intelligence is installed, complete the process for configuring Oracle Business Intelligence as described in the next section “Configuring Oracle BI EE Components.”

## Configuring Oracle BI EE Components

After the Oracle Business Intelligence installer is finished, there are a number of additional tasks you must complete to property configure Oracle BI components. This section describes configuring Oracle BI components immediately after installation of the Oracle Business Intelligence components.

- Configuring BI Scheduler
- Configuring Oracle BI Components for Clustering, Load Balancing, and Failover

### Configuring BI Scheduler

See the chapter about configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide.*

### Configuring Oracle BI Components for Clustering, Load Balancing, and Failover

This section describes the following configuration tasks:

- Preconfiguration Tasks
- Updating the NQSConfig.ini File for BI Server ON BISVR1 and BISVR2
- Updating the NQClusterConfig.ini File for on BICCS1 and BICCS2
- Configuring BI Scheduler on BISCH1 and BISCH2
- Configuring BI Presentation Services on BIPS1 and BIPS2
- Adding Scheduler Administrator Credentials to BI Presentation Services Credential Store
- Configuring BI Presentation Services Plug-In on BIPS1 and BIPS2
- Modifying BI ODBC Data Sources for Communication with BI Cluster

#### Preconfiguration Tasks

Before configuring the Oracle BI components for clustering and load balancing, perform the following tasks.

- Identify the Cluster Controller instance to serve as the Primary Cluster Controller.

  For example, `BICCS1`

- Identify the Cluster Controller instance to serve as the Secondary Cluster Controller.

  For example, `BICCS2`

  Identify the BI Server instance to serve as the Master BI Server. For example: `BISVR1.`

- Copy the repository file (RPD) to the machines hosting the BI Servers.

  On Linux or UNIX systems, the file must be copied to `OracleBI/server/Repository`.

- Before performing the configurations, shut down all BI services or processes. Restart the BI services or processes after configuration is complete.

### Update the NQSConfig.INI File for BI Server on BISVR1 and BISVR2

You must configure the installed BI Servers to participate in the BI Cluster by setting parameters in the `NQSConfig.INI` and `NQClusterConfig.INI` files.

The BI Server software uses an initialization file named `NQSConfig.INI` to set parameters during startup. The `NQSConfig.INI` file is located in the `OracleBI/server/Config` Directory.

1. Open the `NQSConfig.INI` file for editing.
2. In the `[ REPOSITORY ]` section of the `NQSConfig.INI` file, define your repository by setting the logical repository name and file name pair. For example:

   `Star = <`*`Custom rpd filename`*`>, DEFAULT;`
3. Set cache parameters. Caching is enabled by default.

   To use the cluster-aware caching capability, set the cluster-aware cache parameters for the BI Server. In the Query Result Cache section of the `NQSConfig.INI` file, uncomment and set the following parameters:

   - `GLOBAL_CACHE_STORAGE_PATH`

     Set "<*`path to shared storage for cache`*>" and <*`size`*>
   - `MAX_GLOBAL_CACHE_ENTRIES`

     Set <*`Max number of entries`*>
   - `CACHE_POLL_SECONDS`

     Set <*`Polling interval in seconds`*>
4. In the `[ SERVER ]` section, uncomment the `CLUSTER_PARTICIPANT` parameter and set it to `YES`.
5. When the BI Server is a cluster participant, comment out the parameter `SERVER_HOSTNAME_OR_IP_ADDRESSES="ALLNICS";`
6. Set the `RPC_SERVICE_OR_PORT` parameter to the desired port on which the BI Server will listen. The default port is `9703`.

7. For online modifications to be made to the repository, uncomment and set the following parameters:
   - REPOSITORY_PUBLISHING_DIRECTORY="<*path to shared network location*>"
   - REQUIRE_PUBLISHING_DIRECTORY=YES

Example of the NQSConfig.INI file:

```
[ REPOSITORY ]
Star = custom.rpd, DEFAULT;

[ CACHE ]
ENABLE = YES;

// Cluster-aware cache
GLOBAL_CACHE_STORAGE_PATH = "/FS-HOST/OracleBIData/ClusterCache" 700 MB;
MAX_GLOBAL_CACHE_ENTRIES = 10000;
CACHE_POLL_SECONDS = 300;
CLUSTER_AWARE_CACHE_LOGGING = NO;

[ SERVER ]
#SERVER_HOSTNAME_OR_IP_ADDRESS = "ALLNICS"
CLUSTER_PARTICIPANT = YES;
REPOSITORY_PUBLISHING_DIRECTORY = "/FS-HOST/OracleBIData/ClusterRpd";
REQUIRE_PUBLISHING_DIRECTORY = YES;
```

### Updating the NQClusterConfig.INI File for Oracle BI Cluster Controller on BICCS1 and BICCS2

You must configure the Oracle BI Cluster Controllers to communicate in a clustered environment by modifying parameters in the NQClusterConfig.INI file.

Perform the following steps to update the NQSClusterConfig.INI file that is located in the OracleBI/server/Config directory.

1. Open the NQClusterConfig.INI file for editing.
2. Set the parameter ENABLE_CONTROLLER to YES to enable clustering.
3. Identify the Primary and Secondary Cluster Controllers:
   - Set the PRIMARY_CONTROLLER parameter to the machine hosting the Primary Cluster Controller.
   - Set the SECONDARY_CONTROLLER parameter to the machine hosting the Secondary Cluster Controller.
4. Set the SERVERS parameter by entering a comma-separated list of the BI Server hostnames.

   **Note**: Do not specify Fully Qualified Domain Names for the BI Servers.

5. Set the `MASTER_SERVER` parameter by entering the hostname of the Master BI Server machine.

6. Set the parameter `SCHEDULERS` for the Scheduler servers participating in the cluster.

   ```
   SCHEDULERS = "scheduler1:<rpc port>:<monitor port>",
   "scheduler2:<rpc port>:<monitor port>";
   ```

   When setting the `SCHEDULERS` parameter:

   - <*rpc port*> is the Server Port Number that is set during initial configuration of the BI Scheduler. The default port is `9705`.
   - <*monitor port*> is the Cluster Monitor Port that is set during configuration of the BI Scheduler for participation in the BI Cluster. The default is `9708`.

   Example `NQClusterConfig.INI` file:

   ```
   [ Cluster ]
   ENABLE_CONTROLLER = YES;
   PRIMARY_CONTROLLER = "BICCS1";
   SECONDARY_CONTROLLER = "BICCS2";
   SERVERS = "BISVR1", "BISVR2";
   MASTER_SERVER = "BISVR1";
   SERVER_POLL_SECONDS = 5;
   CONTROLLER_POLL_SECONDS = 5;
   CLIENT_SERVER_PORT = 9703;
   CLIENT_CONTROLLER_PORT = 9706;
   MONITOR_CONTROLLER_PORT = 9700;
   MONITOR_SERVER_PORT = 9701;
   SCHEDULERS = "BISCH1:9705:9708","BISCH2:9705:9708";
   ```

### Configuring BI Scheduler on BISCH1 and BISCH2

1. Configure BI Scheduler.

   Before configuring the BI Scheduler for participation in the BI Cluster, you must configure BI Scheduler following the steps described in the chapter about configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide.*

2. Add BI Scheduler Administrator credentials.

   The integration of BI Scheduler with BI Presentation Services requires that you add the BI Scheduler Administrator credentials to the BI Presentation Services Credential Store.

   - Update the credential store for every instance of BI Presentation Services in your deployment to store the BI Scheduler credentials, or copy the credential store with updated credentials to each BI Presentation Services machine.
   - Ensure the `instanceconfig.xml` file for each BI Presentation Services specifies the location of the credential store.

The Scheduler instances participate in the BI cluster in active-passive mode.

3. Specify a username and password for the Scheduler Administrator.

   As part of the process of setting BI Scheduler configuration options, you must specify a username and password for the Scheduler Administrator. The Scheduler Administrator must be a user in the Oracle BI repository (`.rpd` file) and have the Administrator group membership assigned.

   If you do not want to create a new administrator for Scheduler, you can set the Scheduler Administrator credentials to those of the Oracle BI user, Administrator that exists in the repository.

4. Set BI Scheduler configuration options.

   The Scheduler configuration options are set using `schconfig`, a console-based application. On machines where Scheduler instances are installed, run the `schconfig` command in the `OracleBI/setup` directory:

```
. sa-init.sh
schconfig
```

   a) From the Delivers Configuration Menu choices that display, select **1 – Configure Scheduler**.

   b) For each Scheduler Configuration Menu choice shown in the following table, select the listed parameter and configure it, as described in the table.

TABLE: BI SCHEDULER CONFIGURATION OPTIONS

| CONFIGURATION MENU | CHOICE | PARAMETER |
|---|---|---|
| 1-Database | 1-Database Type | • From the drop-down list, select the appropriate `Database Type` for the Scheduler database. The Call Interface is automatically updated according to the Database Type chosen. |
| | 3-Data Source Name | DSN created for the Scheduler database. |
| | 4-User Name | `S_NQ_SCHED` |
| | 5-Password | Password for the `S_NQ_SCHED` user. |
| 2-General | 1-Scheduler Script Path | Shared network location for default scripts. For example: `/Shared Directory Path/OracleBI/server/Scripts/Scheduler` |
| | 2-Default Script Path | Shared network location for default scripts. For example: `/Shared Directory Path/OracleBI/server/Scripts/Common` |

TABLE: BI SCHEDULER CONFIGURATION OPTIONS

| | | |
|---|---|---|
| | 10-Administrator Name | User name for the Scheduler Administrator. The default value is `Administrator`. |
| | 11-Administrator Password | User password for the Scheduler Administrator. |
| 3-Advanced | 3-Participant in Cluster | Y |
| | 4-ClusterMonitor Port | `9708`<br>**Note**: The Cluster Monitor Port must match the `<monitor port>` port number specified in the `SCHEDULERS` parameter in the `NQClusterConfig.INI` file. |

c) From the **Delivers Configuration Menu** choices that display, select **3 – Configure iBots**.

d) For each General Configuration Menu choice shown in the following table, select the listed parameter and configure it, as shown.

| CONFIGURATION MENU | PARAMETER |
|---|---|
| 1-SAW Machine Name | `BIPS1:9710, BIPS2:9710`<br>Use a comma-separated list of BI Presentation Services hosts and ports. Port `9710` is the default port on which BI Presentation Services listens to RPC calls. |

e) From the **Delivers Configuration Menu** choices that display, select **5 – Configure Java Extension**.

f) For each General Configuration Menu choice shown in the following table, select the listed parameter and configure it, as shown in the following table.

| CONFIGURATION MENU | PARAMETER |
|---|---|
| 1-Java Host Server | `BIPS1:9810, BIPS2:9810`<br>Use a comma-separated list of BI Javahost instances. Port `9810` is the default Javahost port. |

g) Select **0** to quit and save changes when prompted.

h) Select **0** to quit the utility.

### Configuring BI Presentation Services on BIPS1 and BIPS2

You must configure BI Presentation Services to:

- Communicate with BI Scheduler instances through the Primary and Secondary Cluster Controllers
- Point to the Javahost cluster
- Use the Presentation Catalog on the shared directory

To configure the BI Presentation Services, you set parameters in the configuration file `instanceconfig.xml`. The `instanceconfig.xml` file is located in the `OracleBIData/web/config` directory.

Use the following procedure to configure BI Presentation Services on each machine that hosts BI Presentation Services.

1. Open the configuration file `instanceconfig.xml` for editing.
2. Locate the `<Alerts>` element.

   Configure for communication with the clustered Scheduler instances, as follows:

   ```
   <Alerts>
         <ScheduleServer> ccsPrimary=”BICCS1” ccsPrimaryPort=”9706”
   ccsSecondary=”BICCS2”
   ccsSecondaryPort=”9706” </ScheduleServer>
   </Alerts>
   ```

   In the preceding example:

   - `ccsPrimary` is set to the `PRIMARY_CONTROLLER` parameter in the `NQClusterConfig.INI` file.
   - `ccsSecondary` is set to the value of the `SECONDARY_CONTROLLER` parameter in the `NQClusterConfig.INI` file.
   - `ccsPrimaryPort` and `ccsSecondaryPort` is set to value of the `CLIENT_CONNTROLLER_PORT` parameter in the `NQClusterConfig.INI` file.
3. Under the `ServerInstance` tag, create the `JavaHostProxy` element.

4. Set the `JavaHostProxy` element attributes and values to point to the Javahost cluster. For example:

   ```
   <ServerInstance>
   .
   .
   <JavaHostProxy>
          <Hosts>
                 <Host address=”BIPS1” port=”9810” />
                 <Host address=”BIPS2” port=”9810” />
          </Hosts>
   </JavaHostProxy>
   .
   .
   </ServerInstance>
   ```

   In the preceding example:

   - `BIPS1` and `BIPS2` are the machines that host the BI Javahost component. BI Javahost was installed along with the installation of BI Presentation Services.
   - The `Hosts` element contains Host sub-elements that identify the Javahost and port pairs. The default Javahost port is `9810.`

     Use the values in the Port element in the `OracleBI/web/javahost/config/config.xml` file on the machine where Javahost is installed.

5. Modify the <*`CatalogPath`*> element to point to the shared Presentation Catalog:

   ```
   <CatalogPath>/Shared Directory
   Path/OracleBIData/web/catalog/customCatalog</CatalogPath>
   ```

6. Under the <*`ServerInstance`*> element, create the Catalog sub element with the following attributes and values:

   ```
   <Catalog>
          <AccountIndexRefreshSecs>120</AccountIndexRefreshSecs>
          <AccountCacheTimeoutSecs>180</AccountCacheTimeoutSecs>
          <CacheTimeoutSecs>120</CacheTimeoutSecs>
          <CacheCleanupSecs>600</CacheCleanupSecs>
          <PrivilegeCacheTimeoutSecs>180</PrivilegeCacheTimeoutSecs>
   </Catalog>
   ```

   Each BI Presentation Service instance will have its own cache and the above entries are necessary to manage when the Presentation Catalog cache is to be updated from disk.

7. Save the file.

### Adding Scheduler Administrator Credentials to BI Presentation Services Credential Store

BI Presentation Services must be able to identify the Scheduler administrator and obtain the credentials to establish a connection with the Scheduler. Presentation Services stores the credentials that it uses in a Presentation Services Credential Store. You must add the Scheduler administrator credentials to the credential store, under the alias `admin`. To obtain the Scheduler Administrator credentials, BI Presentation Services searches the credential store for a username-password credential with the alias `admin`.

Use the following procedure to add the Scheduler administrator credentials to the Presentation Services Credential Store with the `admin` alias. This procedure adds the Scheduler administrator credentials to the credential store called credentialstore.xml. The default location of this file is in the `OracleBIData/web/config` directory.

To add Scheduler Administrator credentials to the credential store

1. On the machine where BI Presentation Services has been installed, run the following commands:

   ```
   $ . /OracleBI/setup/sa-init.sh
   $ . /OracleBI/setup/sysenvinit.sh
   ```

2. Navigate to the `OracleBI/web/bin` directory.

3. Execute the `CryptoTools` utility to add the Scheduler Administrator credentials to the Presentation Services Credential Store. For example:

   ```
   cryptotools credstore –add –infile
   OracleBIData/web/config/credentialstore.xml
   ```

4. Supply values for the prompted parameters, as shown in the following table.

TABLE : CRYPTOTOOLS UTILITY PARAMETERS AND RECOMMENDED VALUES

| CRYPTO TOOLS PROMPT | VALUE OR INPUT | DESCRIPTION |
|---|---|---|
| Credential Tools | `Admin` | Specify the value `admin`. Presentation Services uses this alias to identify the username-password credential for the Scheduler administrator. |
| Username | (*Username of Scheduler Administrator*) | Username of the Scheduler Administrator. For example: `Administrator.` |
| Password | (*Password for the Scheduler Administrator*) | Password for the Scheduler Administrator. |

| | | |
|---|---|---|
| Do you want to encrypt the password? | Y or N | Choosing Y encrypts the password. |
| Passphrase for encryption | (*passphrase*) | Provide a passphrase that is used to encrypt the password. |
| Do you want to write the passphrase to the xml? | Y or N | ● Choose Y to write the passphrase that is needed to decode the password in the xml file.<br>● Choose N for enhanced security. The passphrase is not written to the xml file. Instead, you must specify the passphrase in the instanceconfig.xml file. |

### Configuring BI Presentation Services Plug-in on BIPS1 and BIPS2

You must configure the BI Presentation Services Plug-in to communicate with multiple BI Presentation Services instances. For J2EE based application servers, the BI Presentation Services Plug-in is a Java Servlet deployed in a Web container.

For the Java servlet, configure the Plug-in in the `web.xml` file in the following location:

```
ORACLE_HOME/j2ee/<container name>/applications/analytics/analytics/WEB-INF
```

Perform the following steps to configure BI Presentation Services Plug-in to communicate to the multiple BI Presentation Services instances.

1. Open the `web.xml` file for editing.
2. Locate the following `param-name` and `param` value pairs:

```
<init-param>
<param-name>oracle.bi.presentation.sawserver.Host</param-name>
<param-value><BI Presentation Services></param-value>
</init-param>
<init-param>
<param-name>oracle.bi.presentation.sawserver.Port</param-name>
<param-value>9710</param-value>
</init-param>
```

3. Replace the current value with the following values:

```
<init-param>
<param-name>oracle.bi.presentation.Sawservers</param-name>
<param-value>BIPS1:9710;BIPS2:9710</param-value>
</init-param>
```

In the preceding example:

- `BIPS1 and BIPS2 are machines that host the BI Presentation Services instances.`

- `BI Presentation Services listen on port 9710 to RPC calls from the BI Presentation Services Plug-in.`

4. Save the file.
5. Copy the `web.xml` file to the `OracleBI/web/app/WEB-INF` directory.
6. Restart your Java Servlet container.

### Modifying BI ODBC Data Sources for Communication with BI Cluster

You must modify the BI ODBC Data Source Names (DSNs) for communication to occur with the BI cluster.

The following components use the BI ODBC to connect to the BI environment:

- BI Presentation Services
- BI Administration Tool

Modify the BI ODBC DSN on the following machines:

- All machines that host BI Presentation Services.
- The machine that hosts the BI Administration Tool used to connect to the clustered environment.

After the BI ODBC client is configured to communicate with the BI cluster, the Administration Tool connects to the repository on the Master BI Server.

On Linux and UNIX machines, the `odbc.ini` file is located in the `OracleBI/setup` directory.

1. Open the `odbc.ini` file for editing.
2. Make the following modifications for the DSN:

   ```
   IsClusteredDSN=Yes
   PrimaryCCS=BICCS1
   PrimaryCCSPort=9706
   SecondaryCCS=BICCS2
   SecondaryCCSPort=9706
   Regional=No
   ```

   In the preceding example:

   - Set the `PrimaryCCS` parameter to the Primary Cluster Controller host.
   - Set the `SecondaryCCS` parameter to the Secondary Cluster Controller host.
   - Set the ports to the port number specified in the `CLIENT_CONTROLLER_PORT` parameter in the `NQClusterConfig.INI` file. The default port is `9706`.

3. Save the file.

## Configuring BI Publisher

In a clustered deployment of BI Publisher, the BI Publisher instances on `BIPS1` and `BIPS2` must share a common report repository. A shared file system may be used.

Copy the repository files and folders to the shared directory. The BI Publisher repository files and folders are located in the following directory: `OracleBI/xmlp/XMLP`.

### Configuring BI Publisher to Access the Repository on the Shared File System

1. Log in to the BI Publisher Enterprise application using the URL `http://publisher.mycompany.com:7777/xmlpserver`. The default username and password is `Administrator/Administrator`.
2. Navigate to the Admin tab. Under System Maintenance, click **Server Configuration**.
3. In the Report Repository section on the Server Configuration tab of the System Maintenance page, select **File System** as the Repository Type.
4. In the Path field, enter the path to the shared file system.
5. Click **Apply**.
6. Restart the BI Publisher application for the change to take effect.

### Integrating BI Publisher in BI Presentation Services User Interface

For users to access a clustered BI Publisher application from the **More Products** **BI Publisher** link in the **BI Presentation Services** user interface, BI Presentation Services must be aware of the BI Publisher application URLs for the clustered environment.

To establish a successful connection with BI Publisher, BI Presentation Services must store the BI Publisher Administrator credentials in its credential store. The BI Publisher URLs are specified in the BI Presentation Services configuration file `instanceconfig.xml`, under the `<AdvancedReporting>` tag.

1. Open the `instanceconfig.xml` file for editing. This file is located in `OracleBI/web/config`.
2. Integrate BI Publisher in BI Presentation Services User Interface:

   a) Modify the `AdvancedReporting` element as shown in the following example, with the protocol and Virtual IP for the BI Publisher application:

   ```
   <AdvancedReporting>
   .
   .
   <ServerURL>https://publisher.mycompany.com:443/xmlpserver/servi
   ces/XMLPService</ServerURL>
   <WebURL>https://publisher.mycompany.com:443/xmlpserver</WebURL>
   ```

```
<AdminURL>https://publisher.mycompany.com:443/xmlpserver/servle
t/admin</AdminURL>
.
.
</AdvancedReporting>
```

b) Save the file.

c) Add the BI Publisher Administrator credentials to the BI Presentation Services Credential Store.

d) Specify the Credential Store in the `instanceconfig.xml` file for BI Presentation Services.

See the chapter about configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for more information on these steps.

**Note:** You must update the credential store for every instance of BI Presentation Services in your deployment to store the BI Publisher credentials, or copy the credential store with updated credentials to each BI Presentation Services machine. The `instanceconfig.xml` file for each BI Presentation Services must specify the location of the credential store.

## Configuring BI Presentation Services to Identify Credential Stores

You must direct BI Presentation Services to the credential store that contains the Scheduler administrator credentials. To do this, set parameters in the BI Presentation Services configuration file, `instanceconfig.xml`. In addition, if you have not stored the `passphrase` in the credential store, then you must also specify the passphrase to decrypt the password credential. In the above example, the `passphrase` was not stored in the credential store and needs to be specified in the `instanceconfig.xml` file.

Perform the following steps to identify the credential store to be used by BI Presentation Services:

1. Open the `OracleBIdata/web/config/instanceconfig.xml` file for editing.
2. Locate the `<CredentialStore>` node within this file.
3. Specify attribute values as shown in the following example. If the `<CredentialStore>` node does not exist, create this element with sub elements and attributes.

```
<WebConfig>
  <ServerInstance>
       .
       .
    <CredentialStore>
      <CredentialStorage type="file"
path="OracleBIData/web/config/credentialstore.xml"
passphrase="<passphrase>"/>
```

```
                    </CredentialStore>
                    .
                    .
                       </ServerInstance>
                </WebConfig>
```

4. Restart Presentation Services to reflect the configuration changes.

To test the BI Presentation Services:

1. Start the BI Presentation Services
2. Start the HTTP server being used for BI Presentation Services
3. Open a Web browser and navigate to `http://<ServerName>:<port number>/analytics/saw.dll?Dashboard`

If a login page displays, then you have successfully installed BI Presentation Services.

### Configuring Active Data Guard on DBHOST3 and DBHOST4

Oracle Active Data Guard requires a separate license, and it extends basic Data Guard functionality included with Oracle Database 11*g* Enterprise Edition. If you do not have an existing Data Guard physical standby configuration, use the following links for assistance in creating one. To create a configuration with:

- Oracle RAC primary database and a single-instance standby database: `http://www.oracle.com/technology/deploy/availability/pdf/MAA_WP_10g_RACPrimarySingleInstancePhysicalStandby.pdf`
- Oracle RAC primary database and an Oracle RAC standby database: `http://www.oracle.com/technology/deploy/availability/pdf/MAA_WP_10gRACPrimaryRACPhysicalStandby.pdf`

**Note**: For general best practices for deploying Active Data Guard, see the *Oracle Active Data Guard, Oracle Data Guard 11g Release 1* white paper at `http://www.oracle.com/technology/deploy/availability/pdf/maa_wp_11gr1_activedataguard.pdf`

## Application Connection Management

The following sections describe managing application connections:

- Managing initial client connections
- Setting up connections to the primary database for BI Scheduler on BISCH1 and BISCH2
- Setting up connections to an Active Standby Database for BI Server

### Managing Initial Client Connections

In steady state, it is important that:

- Reporting applications connect to the standby database, and primary applications that connect to the primary database, connect to the database with the correct database role.
- Following a role transition, the application should be able to automatically reconnect to the correct database.

You can accomplish these requirements by having separate service names for the primary application and reporting application and by starting the corresponding service based on the database role.

In the following example, the primary application is connecting with a service name of `bi_rw` and the reporting application is connecting with a service name of `bi_ro`. To automate starting and stopping the service, use an "on startup" trigger that checks the database role and starts the appropriate service.

```
CREATE OR REPLACE TRIGGER manage_service
after startup on database
       DECLARE
              role VARCHAR(30);
       BEGIN
              SELECT DATABASE_ROLE INTO role FROM V$DATABASE;
       IF role = ‘PRIMARY’ THEN
              DBMS_SERVICE.START_SERVICE(‘bi_rw’);
       ELSE
              DBMS_SERVICE.START_SERVICE(‘bi_ro’);
       END IF;
END;
```

Both the primary and reporting applications should connect using an Oracle Net alias that contains all hosts, both primary and standby, in the `ADDRESS_LIST`.

For example:

```
bi_rw =
(ADDRESS_LIST =
              (ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST1)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST2)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST3)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST4)(PORT = 1521))
(CONNECT_DATA = (SERVER = DEDICATED)(SERVICE_NAME = bi_rw)))

bi_ro =
(ADDRESS_LIST =
              (ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST1)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST2)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST3)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = DBHOST4)(PORT = 1521))
(CONNECT_DATA = (SERVER = DEDICATED)(SERVICE_NAME = bi_ro)))
```

When the primary application is started, the initial connection attempt will cycle through the ADDRESS_LIST until it finds the bi_rw service. The reporting application will do the same until it finds the bi_ro service.

See the "Managing Initial Client Connections" section of the Oracle Active Data Guard, Oracle Data Guard 11*g* Release 1 white paper at http://www.oracle.com/technology/deploy/availability/pdf/maa_wp_11gr1_activedataguard.pdf

### Setting Up Connections to the Primary Database for Scheduler on BISCH1 and BISCH2

You must configure the Scheduler to always write into the Scheduler schema using the Primary read/write connection.

Set the Scheduler configuration options using schconfig, a console-based application. On machines where Scheduler instances are installed, in the directory OracleBI/setup, run the command schconfig:

```
. sa-init.sh
schconfig
```

1. From the **Delivers Configuration Menu** choices that display, select **1 – Configure Scheduler**.
2. For each **Scheduler Configuration Menu** choice, select the listed parameter and configure the parameters, as described in the following table.

TABLE : SCHEDULER CONFIGURATION MENU

| CONFIGURATION MENU | CHOICE | PARAMETER |
|---|---|---|
| 1-Database | 3-DataSource Name | DSN created for the Primary Scheduler database. For example: bi_rw. |

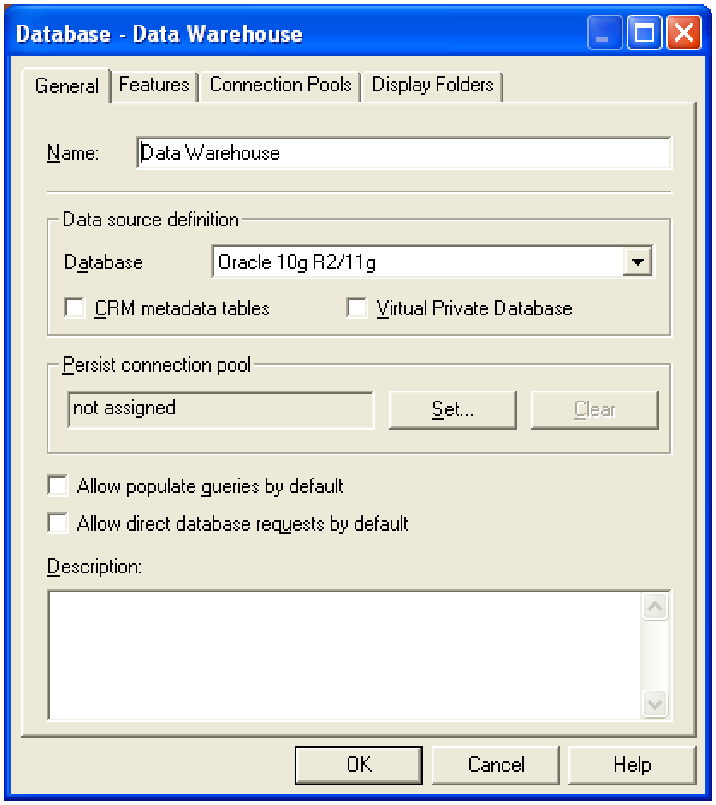
3. Select **0** to quit and save changes, when prompted.
4. Select **0** to quit the utility.

## Setting Up Connections to an Active Standby Database for BI Server

1. Create database connection and disable temporary table creation.

   The Oracle BI EE Administrator must create a database connection in the repository using the Oracle BI Administration Tool. When creating the database, in the **Database Property** dialog box, make sure the Persist connection pool is not assigned. This will prevent the Oracle BI EE server from creating temporary tables in this database.

**Figure 2: Disable Temporary Table Creation**



Name: Data Warehouse

Data source definition

Database: Oracle 10g R2/11g

□ CRM metadata tables □ Virtual Private Database

Persist connection pool: not assigned

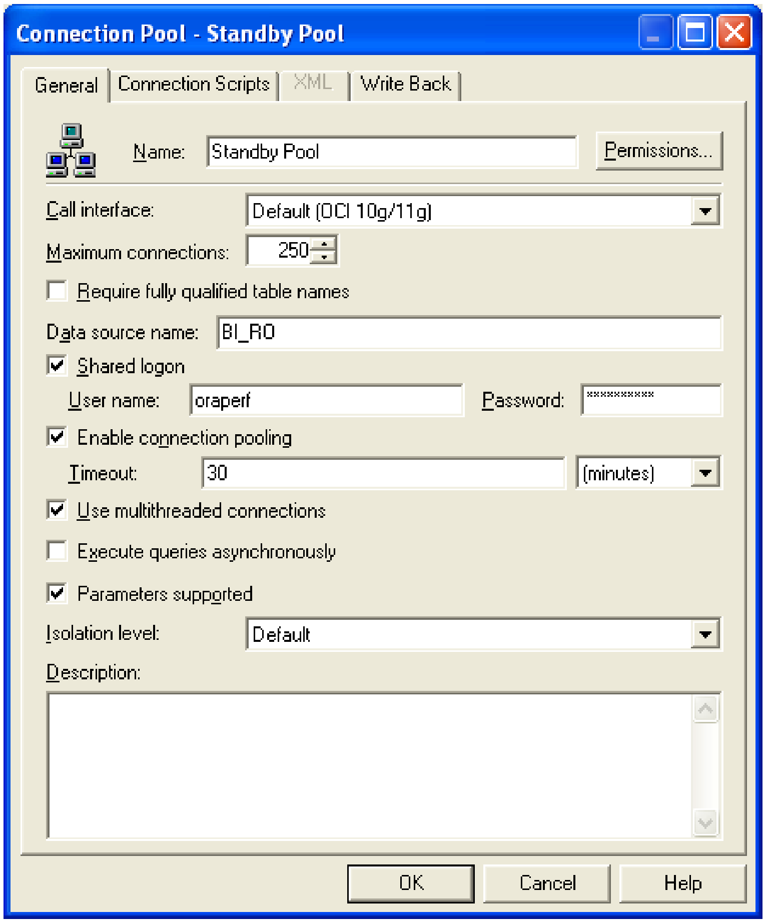
□ Allow populate queries by default

□ Allow direct database requests by default

Description:

2. After the database is created, create the first connection pool to connect to the standby database. Make sure that this Standby Pool is the first connection pool for this database, because most queries will go through the first connection.

**Figure 3: Creat Connection Pool to Standby Database**



Name: Standby Pool

Call interface: Default (OCI 10g/11g)

Maximum connections: 250

☐ Require fully qualified table names

Data source name: BI_RO

☑ Shared logon

User name: oraperf Password: ***********

☑ Enable connection pooling

Timeout: 30 (minutes)

☑ Use multithreaded connections

☐ Execute queries asynchronously

☑ Parameters supported

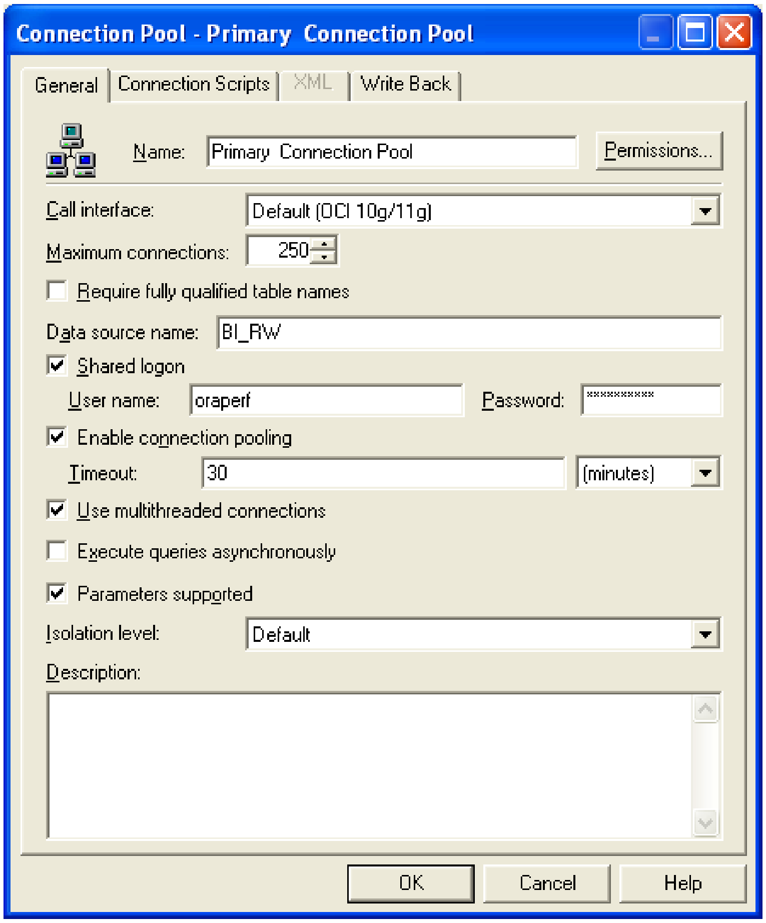
Isolation level: Default

Description:

3. Create a “Write-back” connection pool.

   With the Oracle BI Administration Tool, the Oracle BI EE Administrator should create a Write-back connection pool that points to the primary database.

Figure 4: Create Connection Pool to Primary Database



Connection Pool - Primary Connection Pool

Name: Primary Connection Pool

Call interface: Default (OCI 10g/11g)

Maximum connections: 250

☐ Require fully qualified table names

Data source name: BI_RW

☑ Shared logon

User name: oraperf    Password: **********

☑ Enable connection pooling

Timeout: 30 (minutes)

☑ Use multithreaded connections

☐ Execute queries asynchronously

☑ Parameters supported

Isolation level: Default

Description:

**Primary Connection Pool** is created in case Oracle BI EE Administrator must write information back to the database using interactive or batch processing client (such as `nqCmd`).

After these two steps, the Oracle BI EE Administrator can continue the regular configuration steps as usual.

**Note:** See “Configuring Oracle BI EE Server with Oracle Active Data Guard” for more details: http://www.oracle.com/technology/deploy/availability/pdf/maa_wp_11gbiee_activedataguard.pdf

# References

1. Oracle Maximum Availability Architecture Web site
   http://www.otn.oracle.com/goto/maa
2. Oracle Active Data Guard
   http://www.oracle.com/technology/deploy/availability/htdocs/activedataguard.html
3. Oracle Business Intelligence Suite Enterprise Edition Plus
   http://www.oracle.com/appserver/business-intelligence/enterprise-edition.html
4. Configuring Oracle BI EE Server with Oracle Active Data Guard
   http://www.oracle.com/technology/deploy/availability/pdf/maa_wp_11g_biee_activedataguard.pdf
5. Oracle Active Data Guard, Oracle Data Guard 11*g* Release 1
   http://www.oracle.com/technology/deploy/availability/pdf/maa_wp_11gr1_activedataguard.pdf

Deploying Oracle Business Intelligence
Enterprise Edition with Oracle Active Data
Guard
March 2010
Author: Susan Kornberg
Contributing Authors: Rahul Menezes

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com oracle.com

Oracle is committed to developing practices and products that help protect the environment