



Deploying Oracle Application Server 10g with the F5 BIG-IP System

- Introducing the BIG-IP and Oracle Application Server 10g configuration
- Configuring the BIG-IP system for deployment with Oracle Application Server 10g
- Appendix A: Backing up and restoring the BIG-IP system configuration

Introducing the BIG-IP and Oracle Application Server 10g configuration

Oracle and F5 have created a highly effective way to direct traffic for Oracle Application Server 10g deployments using the BIG-IP application traffic management device. When deployed with Oracle Application Server 10g, the BIG-IP product ensures fast delivery, always-on access, peak security and easy expansion for applications running on Oracle.

Oracle Application Server 10g meets customers' demand for up-to-date business information with reliable, scalable and cost-effective grid computing. With grid computing, organizations can leverage the use of many low-cost, modular servers acting as one computer, making their applications more scalable and less expensive to deploy and manage. Oracle Application Server 10g's built-in clustering can be combined with the BIG-IP traffic management solution to help ensure that user requests are always routed to the most available server on the grid.

Prerequisites and configuration notes

The following are prerequisites for this Deployment Guide:

- ◆ You must have an Oracle Application Server 10g deployment.
- ◆ The BIG-IP system must be running version 4.5 or later (the step-by-step configuration procedures in this Deployment Guide are for 4.5 or later, but do not include configuration steps for BIG-IP version 9.0 and later).
- ◆ Briefly review the basic configuration tasks and the few pieces of information, such as IP addresses, that you should gather in preparation for completing the BIG-IP system configuration.

◆ Note

All of the configuration procedures in this Deployment Guide are performed on the BIG-IP system. For specific information on how to configure Oracle Application Server 10g, consult the Oracle Documentation.

This document is written with the assumption that you are familiar with both the BIG-IP system and Oracle Application Server 10g. For more information on configuring these products, consult the appropriate documentation.

Configuration example

The BIG-IP system provides intelligent traffic management and fail-over for Oracle Application Server 10g middle tier and infrastructure servers. Through advanced health checking capabilities, the BIG-IP product recognizes when resources are unavailable or under-performing and directs traffic to another resource. The BIG-IP product can also track Oracle Application Server 10g end-user sessions, enabling the application server to maintain client session data. The following diagram shows an example deployment with Oracle Application Server 10g and the BIG-IP system.

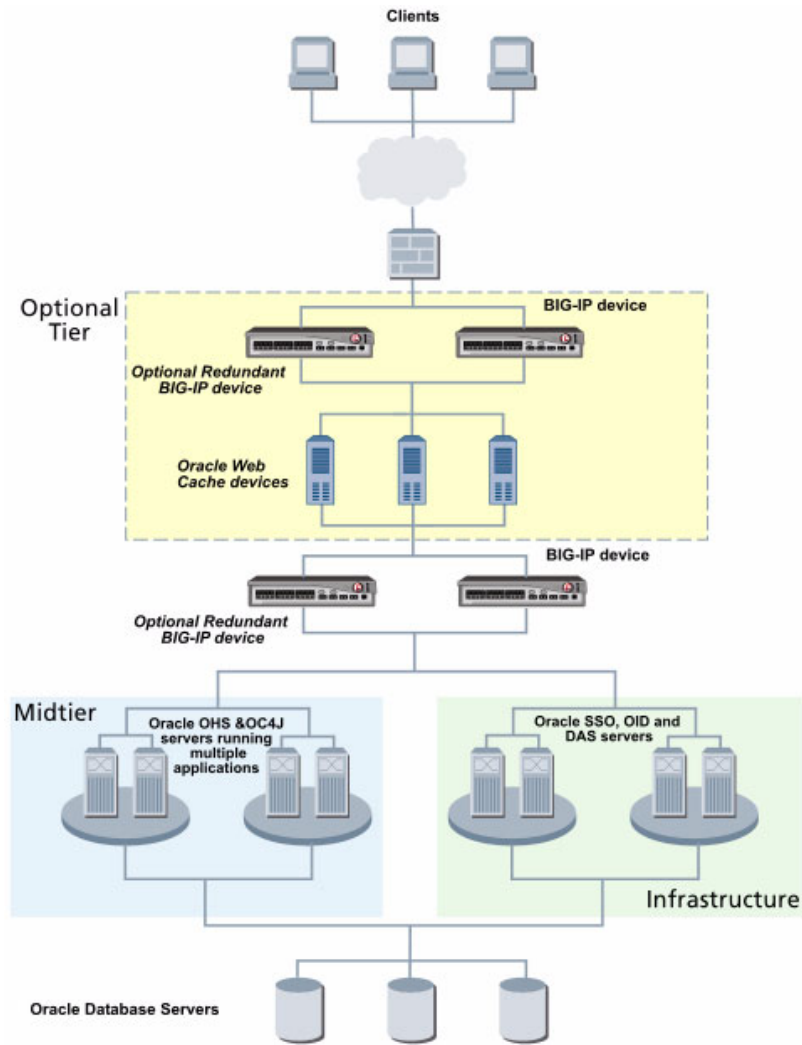


Figure 1.1 BIG-IP Oracle Application Server 10g configuration example

Configuring the BIG-IP system for deployment with Oracle Application Server 10g

To configure the BIG-IP for directing traffic to the Oracle devices, you need to complete the following procedures:

- *Connecting to the BIG-IP device*
- *Configuring the BIG-IP system for Oracle Web Cache servers*
- *Configuring the BIG-IP system for the Oracle Midtier*
- *Configuring a health monitor*
- *Synchronizing the BIG-IP configuration if using a redundant system*

◆ Tip

*We recommend you save your existing BIG-IP configuration before you begin the procedures in this Deployment Guide. To save your BIG-IP configuration, see **Appendix A: Backing up and restoring the BIG-IP system configuration**, on page 1-23.*

The BIG-IP system offers both web-based and command line configuration tools, so that users can work in the environment that they are most comfortable with. This Deployment Guide contains procedures to configure the BIG-IP system using both the BIG-IP web-based Configuration utility using a browser and the BIG-IP **bigpipe** command line interface. Unless you are familiar with using the **bigpipe** command line interface, we recommend using the Configuration utility.

Connecting to the BIG-IP device

The first step in this configuration is to connect to the BIG-IP system. You can connect to the BIG-IP system using the Configuration utility or the command line.

Connecting to the BIG-IP device using the Configuration utility

Use the following procedure to access the BIG-IP web-based Configuration utility using a web browser.

To connect to the BIG-IP system using the Configuration utility

1. In a browser, type the following URL:
https://<administrative IP address of the BIG-IP device>
A Security Alert dialog box appears, click **Yes**.
The authorization dialog box appears.
2. Type your user name and password, and click **OK**.
The Configuration Status screen opens.

Once you are logged onto the BIG-IP system, the initial screen, called the Configuration Status page, displays. From the Configuration status page, you can access the Configuration utility, documentation such as manuals and release notes, and software downloads.

3. From the Configuration Status screen, click **Configure your BIG-IP (R) using the Configuration utility**.

The Configuration utility opens to the Network Map screen.

Connecting to the BIG-IP device using the bigpipe command line interface

You can access the **bigpipe** command line utility on a BIG-IP system with connections for a monitor and keyboard. For a system without a monitor and keyboard attached, like the IP Application Switch, you can access **bigpipe** through an SSH shell from a remote administrative host. For specific information on how to access **bigpipe** through an SSH shell, consult the *BIG-IP Reference Guide*.

Configuring the BIG-IP system for Oracle Web Cache servers

If you are using Oracle Web Cache devices in your configuration, the first step in this deployment is configuring the BIG-IP system to load balance traffic to these servers.

◆ Important

The Web Cache servers are optional in an Oracle deployment. Only use the following procedures if you are using Web Cache servers.

To configure the BIG-IP system for load balancing traffic to Web Cache devices, you must complete the following procedures:

- *Creating pools for the Web Cache devices*
- *Creating virtual servers*
- *Configuring a health monitor*

Creating pools for the Web Cache devices

The first procedure in this configuration is to configure pools for the Web Cache devices. A BIG-IP pool is a set of devices grouped together to receive traffic according to a load balancing method. You must create a separate pool for each service on which there will be traffic. For this configuration, you configure two pools for the Web Cache devices, one pool for the Web Caches themselves, and another pool for Web Cache Invalidation traffic.

Creating the Web Cache pool

The first pool we create is for the Web Cache servers. You can create this pool from the Configuration utility or the command line. For this pool, we recommend using cookie persistence, Insert mode.

To create the Web Cache pool from the Configuration utility

1. In the navigation pane, click **Pools**.
The Pools screen opens.
2. Click the **Add** button.
The Add Pool screen opens (see Figure 1.2).
3. In the **Pool Name** box, enter a name for your pool.
In our example, we use **oracle_cache_pool**.
4. In the **Load Balancing Method** box, enter your preferred load balancing method (different load balancing methods may yield optimal results for a particular network).
In our example, we select **Predictive**.
5. In the **Resources** section, you add the Oracle Web Cache servers to the pool.
 - a) In the **Member Address** box, type the IP address of the Oracle server.
In our example, we type **192.168.201.21**.
 - b) In the **Service** box, type the service number you want to use for this device, or specify a service by choosing a service name from the list (for example **7777**).
In our example, we type **7777**.
 - c) The **Member Ratio** and **Member Priority** boxes are optional.
 - d) Click the Add button (>>) to add the member to the **Current Members** list.
 - e) Repeat steps a-d for each Web Cache server you want to add to the pool. In our example, we repeat these steps once for the other Web Cache server (**192.168.201.22**). See Figure 1.2.
6. The other fields in the Add Pool screen are optional. Configure these fields as applicable for your network. (For additional information about configuring a pool, click the **Help** button.)

7. Click the **Done** button.

Member Address:	Service:	Member Ratio:	Member Priority:	Current Members:
192.168.201.22	: 7777			192.168.201.21:7777 r 1 p 1 192.168.201.22:7777 r 1 p 1
192.168.201.1	or choose...			

Figure 1.2 Adding a pool for Oracle Web Cache devices in the BIG-IP Configuration utility

8. In the Pool screen, from the Pool Name list, click the name of the pool you just created.
In our example, we click **oracle_cache_pool**.
9. Click the Persistence tab.
The Persistence screen for the pool opens.
10. In the Persistence Type section, click the option button for **Active HTTP Cookie**.
11. From the Method list, select **Insert**.
12. Setting a Expiration for the cookie is optional.

13. Click the **Apply** button.

The screenshot shows the configuration interface for a pool named 'oracle_cache_pool'. The 'Persistence Type' section is expanded, and 'Active HTTP Cookie' is selected. The 'Method' dropdown is set to 'Insert', and the 'Expiration' fields are set to 0 days, 0 hours, 0 minutes, and 0 seconds. The 'Cookie Name' field is empty.

Figure 1.3 Configuring active cookie persistence, Insert mode

To create the Web Cache pool from the command line

To create the pool from the command line, use the following syntax

```
b pool <pool name> { [lb_method <load balancing method>]member <IP address>:<port> persist
  cookie cookie_mode insert }
```

In our example, we type:

```
b pool oracle_cache_pool { lb_method predictive member 192.168.201.21:7777 member
  192.168.201.22:7777 persist cookie cookie_mode insert }
```

Creating the Web Cache Invalidation pool

The next step is to create a pool for the Web Cache Invalidation traffic. Invalidation keeps content on the cache consistent with content on the origin web servers and databases. This is particularly important for dynamically generated web pages that change frequently, such as stock quotations or news updates.

Again, you can create the pool from the Configuration utility or from the command line.

To create the Web Cache Invalidation pool from the Configuration utility

1. In the navigation pane, click **Pools**.
The Pools screen opens.

2. Click the **Add** button.
The Add Pool screen opens.
3. In the **Pool Name** box, enter a name for your pool.
In our example, we use **oracle_invalidation_pool**.
4. In the **Load Balancing Method** box, enter your preferred load balancing method (different load balancing methods may yield optimal results for a particular network).
In our example, we select **Round Robin**.
5. In the **Resources** section, you add the cache servers to the pool.
 - a) In the **Member Address** box, type the IP address of the Oracle server.
In our example, we type **192.168.201.21**.
 - b) In the **Service** box, type the service number you want to use for this device, or specify a service by choosing a service name from the list (for example **4001**).
In our example, we type **4001**.
 - c) The **Member Ratio** and **Member Priority** boxes are optional.
 - d) Click the Add button (>>) to add the member to the **Current Members** list.
 - e) Repeat steps a-d for each Oracle server you want to add to the pool. In our example, we repeat these steps once for the other Oracle server (**192.168.201.22**).
6. The other fields in the Add Pool screen are optional. Configure these fields as applicable for your network. (For additional information about configuring a pool, click the **Help** button.)
7. Click the **Done** button.

To create the Web Cache Invalidation pool from the command line

To create the pool from the command line, use the following syntax

```
b pool <pool name> { [lb_method <load balancing method>]member <IP address>:<port> }
```

In our example, we type:

```
b pool oracle_invalidation_pool { lb_method rr member 192.168.201.21:4001 member  
192.168.201.22:4001 }
```

Creating virtual servers

The next step in this configuration is to define virtual servers that reference the Web Cache pools you just created. As with a pool, you must create a virtual server for each service. Again, you can define virtual servers from the Configuration utility or the command line.

Creating the Web Cache virtual server

To create the Web Cache virtual server using the Configuration utility

1. In the navigation pane, click **Virtual Servers**.
The Virtual Servers screen opens.
2. Click the **Add** button.
The Add Virtual Server screen opens.
3. Enter the IP address and service for the virtual server, then click the **NEXT** button.
In our example, we use **192.168.200.10** with service of **7777**.
The Configure Basic Properties screen displays. Click the **Next** button again.
4. Click the **Pool** option button, and from the list, select the pool you created in the *Creating the Web Cache pool* section above.
In our example, we select **oracle_cache_pool** (see Figure 1.4).

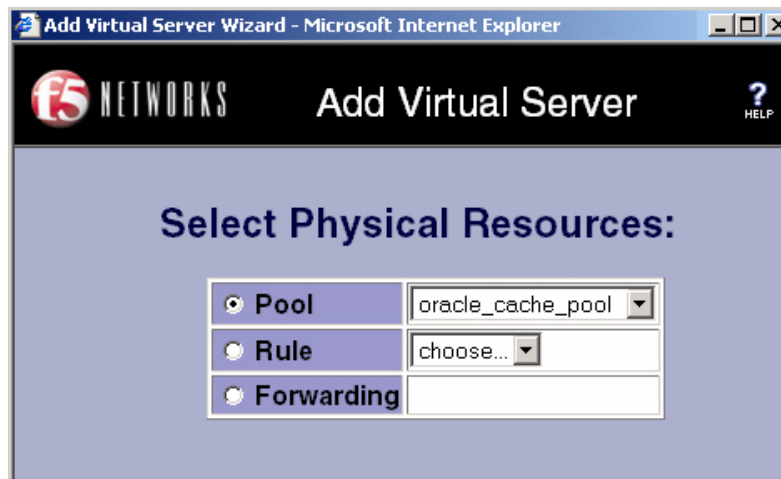


Figure 1.4 Selecting the `oracle_cache_pool` while creating the virtual server

5. Click the **Done** button. For additional information about configuring a virtual server, click the **Help** button.

To view the virtual server, click the virtual server in the list. In our example, the virtual server properties are shown in Figure 1.5.

Figure 1.5 The HTTP virtual server in the BIG-IP Configuration utility.

To create the Web Cache virtual server from the command line

Use the bigpipe **virtual** command as shown below. You can use standard service names in place of port numbers. If you have DNS configured, you can also use host names in place of IP addresses.

```
b virtual <virtual IP address>:<port> use pool <pool_name>
```

In our example, we use:

```
b virtual 192.168.200.10:7777 use pool oracle_cache_pool
```

Creating the Web Cache Invalidation virtual server

To create the Web Cache Invalidation virtual server using the Configuration utility

1. In the navigation pane, click **Virtual Servers**.
The Virtual Servers screen opens.
2. Click the **Add** button.
The Add Virtual Server screen opens.

-
3. Enter the IP address and service for the virtual server, then click the **Next** button.

In our example, we use **192.168.202.10** with service of **4001**.

***Important:** This address must be on the same subnet as the servers that will be sending the cache content for invalidation. You may need multiple virtual servers for each Midtier subnet. In our example, we use the same subnet as our Midtier, the **202** subnet.*

The Configure Basic Properties screen displays. Click the **Next** button again.

4. Click the **Pool** option button, and from the list, select the pool you created in the *Creating the Web Cache Invalidation pool* section above.

In our example, we select **oracle_invalidations_pool**.

5. Click the **Done** button.

For additional information about configuring a virtual server, click the **Help** button.

To create the Web Cache Invalidation virtual server from the command line

Use the bigpipe **virtual** command as shown below. You can use standard service names in place of port numbers. If you have DNS configured, you can also use host names in place of IP addresses.

```
b virtual <virtual IP address>:<port> use pool <pool_name>
```

In our example, we use:

```
b virtual 192.168.202.10:4001 use pool oracle_invalidations_pool
```

Configuring the BIG-IP system for the Oracle Midtier

The next step is to configure the BIG-IP system to direct traffic to the devices in the Oracle midtier. There are a number of different Oracle applications that can be running on the midtier, and for each application you need a pool on the BIG-IP system. Repeat the following procedure for creating a pool for each Oracle Midtier application.

If you are running multiple Midtier applications, you only need one Midtier virtual server, and you can use a BIG-IP iRule which inspects the client URI and directs the traffic to the appropriate pool based on the URI content. If you do not want to use the iRule, you need to create a virtual server for each pool.

Creating the Oracle Midtier pool(s)

For each Oracle Midtier application, you must create a separate pool, even if it is running on the same physical device. It is important that the port (service) number is correct for each Oracle midtier application. For a list of Oracle applications and their respective ports, consult the Oracle documentation.

To create the Midtier pool from the Configuration utility

1. In the navigation pane, click **Pools**.
The Pools screen opens.
2. Click the **Add** button.
The Add Pool screen opens.
3. In the **Pool Name** box, enter a name for your pool.
In our example, we use **oracle_forms_pool**.
4. In the **Load Balancing Method** box, enter your preferred load balancing method (different load balancing methods may yield optimal results for a particular network).
In our example, we select **Least Connections**.
5. In the **Resources** section, you add the servers to the pool.
 - a) In the **Member Address** box, type the IP address of the Oracle server.
In our example, we type **192.168.202.51**.
 - b) In the **Service** box, type the service (port) number applicable for this application, or specify a service by choosing a service name from the list (for example **7777**).
In our example, we type **7777**.
 - c) The **Member Ratio** and **Member Priority** boxes are optional.
 - d) Click the Add button (>>) to add the member to the **Current Members** list.
 - e) Repeat steps a-d for each Oracle server you want to add to the pool. In our example, we repeat these steps once for the other Oracle server (**192.168.202.52**).
6. The other fields in the Add Pool screen are optional. Configure these fields as applicable for your network. (For additional information about configuring a pool, click the **Help** button.)
7. Click the **Done** button.
8. Repeat this procedure for each Oracle Midtier application, adding the relevant members, and naming the pool appropriately.

To create the Midtier pool from the command line

To create the pool from the command line, use the following syntax

```
b pool <pool name> { [lb_method <load balancing method>]member <IP address>:<port> }
```

In our example, we type:

```
b pool oracle_forms_pool { lb_method least_conn member 192.168.202.51:7777 member
192.168.202.52:7777 }
```

Creating the Midtier virtual servers

In this step, you must decide whether you want one virtual server with an iRule to direct traffic to the appropriate pool, or if you want to create a virtual server for each pool you configured in the preceding procedure.

Creating a virtual server for each pool

Use this procedure if you do not want to use an iRule to direct traffic to the pools.

To create the Midtier virtual server using the Configuration utility

1. In the navigation pane, click **Virtual Servers**.
The Virtual Servers screen opens.
2. Click the **Add** button.
The Add Virtual Server screen opens.
3. Enter the IP address and service for the virtual server, then click the **NEXT** button.
In our example, we use **192.168.201.20** with service of **7777**.
The Configure Basic Properties screen displays. Click the **Next** button again.
4. Click the **Pool** option button, and from the list, select the pool you created in the *Creating the Web Cache pool* section above.
In our example, we select **oracle_forms_pool**.
5. Click the **Done** button. For additional information about configuring a virtual server, click the **Help** button.

To create the Midtier virtual server from the command line

Use the bigpipe **virtual** command as shown below. You can use standard service names in place of port numbers. If you have DNS configured, you can also use host names in place of IP addresses.

```
b virtual <virtual IP address>:<port> use pool <pool_name>
```

In our example, we use:

```
b virtual 192.168.201.20:7777 use pool oracle_forms_pool
```

Creating a virtual server that references an iRule

Use this procedure if you want to create a rule on the BIG-IP system to direct traffic to the appropriate Oracle Midtier pool, based on the client URI. This procedure is not necessary if you want to create a virtual server for each pool.

First we create the iRule, then create the virtual server that references it. For this configuration, we recommend using the Configuration utility.

To create a rule to direct traffic to the Midtier pools

1. In the navigation pane, click **Rules**.
2. Click the **Add** button.
The Configure Rule Basics screen of the Add Rule dialog box opens.
3. In the **Name** box, type a name for the rule. In our example, we type **midtier_rule**.
4. In the **Type** section, click the **Text Input** option button, and then click the **Next** button.

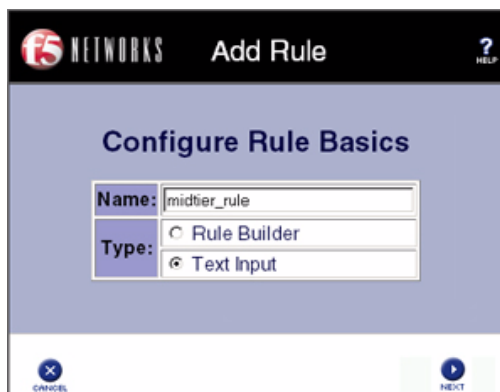


Figure 1.6 Adding the iRule in the configuration utility

-
5. In the Text Input section, enter a rule similar to the following example, substituting the names of your Midtier pools. If you have more (or less) pools, add (or subtract) **else if** statements as needed.

```
if (http_uri starts_with "/forms90") {  
    use pool forms_pool  
}  
else if (http_uri starts_with "/portal") {  
    use pool ias_portal  
}  
else {  
    use pool ias_pool  
}
```

Figure 1.7 Example iRule for the Midtier pools

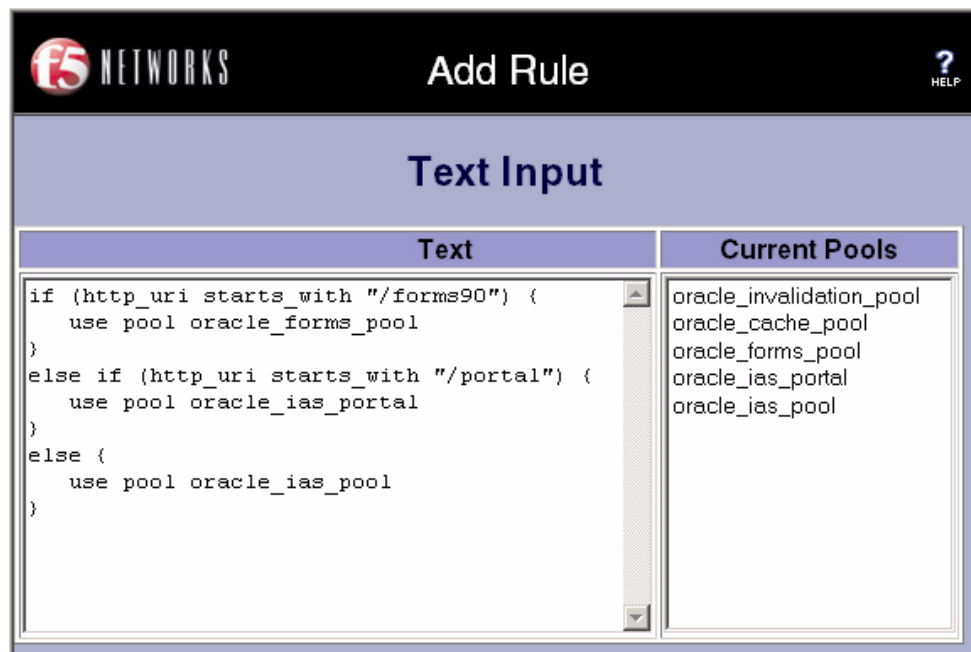


Figure 1.8 Inserting the rule in the Configuration utility

6. Click the **Done** button.

The next step is to create a virtual server that references the rule you just created.

To create a virtual server for the Midtier pools that references a rule

1. In the navigation pane, click **Virtual Servers**.
The Virtual Servers screen opens.
2. Click the **Add** button.
The Add Virtual Server screen opens.
3. Enter the IP address and service for the virtual server, then click the **NEXT** button.
In our example, we use **192.168.201.20** with service of **80**.
The Configure Basic Properties screen displays. Click the **Next** button again.
4. Click the **Rule** option button, and from the list, select the rule you created in the *To create a rule to direct traffic to the Midtier pools* section. In our example, we select **midtier_rule** (see Figure 1.9).



Figure 1.9 Configuring the virtual server to use the rule

5. Click the **Done** button. For additional information about configuring a virtual server, click the **Help** button.

Configuring the BIG-IP system for Oracle Infrastructure

The next step is to configure the BIG-IP system to direct traffic to the devices in the Oracle Infrastructure devices. Oracle Infrastructure contains devices like SSO, DAS, OID.

Configuring the Infrastructure devices on the BIG-IP system is nearly identical to configuring the Midtier devices, although the ports (services) vary by application.

For this Deployment Guide, we use Oracle SSO as an example.

Single Sign On configuration

If your configuration includes Single Sign On (SSO), you need to create a pool and virtual server on the BIG-IP system.

Creating the SSO pool

The first step is to create a pool for SSO traffic. You can create the pool from the Configuration utility or from the command line.

To create the SSO pool from the Configuration utility

1. In the navigation pane, click **Pools**.
The Pools screen opens.
2. Click the **Add** button.
The Add Pool screen opens.
3. In the **Pool Name** box, enter a name for your pool.
In our example, we use **oracle_sso_pool**.
4. In the **Load Balancing Method** box, enter your preferred load balancing method (different load balancing methods may yield optimal results for a particular network).
In our example, we select **Least Connections**. In Least Connections mode, the BIG-IP system passes a new connection to the node that has the least number of current connections. Least Connections mode works best in environments where the servers or other equipment you are load balancing have similar capabilities.
5. In the **Resources** section, you add the SSO servers to the pool.
 - a) In the **Member Address** box, type the IP address of the Oracle server.
In our example, we type **192.168.203.11**.
 - b) In the **Service** box, type the service number you want to use for this device, or specify a service by choosing a service name from the list (for example **7777**).
In our example, we type **7777**.
 - c) The **Member Ratio** and **Member Priority** boxes are optional.
 - d) Click the Add button (>>) to add the member to the **Current Members** list.
 - e) Repeat steps a-d for each SSO server you want to add to the pool.
In our example, we repeat once for **192.168.203.12**.
6. The other fields in the Add Pool screen are optional. Configure these fields as applicable for your network. (For additional information about configuring a pool, click the **Help** button.)

7. Click the **Done** button.
8. In the Pool screen, from the Pool Name list, click the name of the pool you just created.
In our example, we click **oracle_http_pool**.
9. Click the Persistence tab.
The Persistence screen for the pool opens.
10. In the Persistence Type section, click the option button for **Active HTTP Cookie**.
11. From the Method list, select **Insert**.
12. Setting a Expiration for the cookie is optional.
13. Click the **Apply** button.

To create the SSO pool from the command line

To create the pool from the command line, use the following syntax

```
b pool <pool name> { [lb_method <load balancing method>]member <IP address>:<port> persist
  cookie cookie_mode insert }
```

In our example, we type:

```
b pool oracle_sso_pool { lb_method rr member 192.168.203.11:7777 member
  192.168.203.12:7777 persist cookie cookie_mode insert }
```

Creating the SSO virtual server

To create the SSO virtual server using the Configuration utility

1. In the navigation pane, click **Virtual Servers**.
The Virtual Servers screen opens.
2. Click the **Add** button.
The Add Virtual Server screen opens.
3. Enter the IP address and service for the virtual server, then click the **Next** button.
In our example, we use **192.168.201.11** with service of **7777**.
The Configure Basic Properties screen displays. Click the **Next** button again.
4. Click the **Pool** option button, and from the list, select the pool you created in the *Creating the SSO pool* section above.
In our example, we select **oracle_sso_pool**.
5. Click the **Done** button.
For additional information about configuring a virtual server, click the **Help** button.

To create the SSO virtual server from the command line

Use the bigpipe **virtual** command as shown below. You can use standard service names in place of port numbers. If you have DNS configured, you can also use host names in place of IP addresses.

```
b virtual <virtual IP address>:<port> use pool <pool_name>
```

In our example, we use:

```
b virtual 192.168.201.11:7777 use pool oracle_sso_pool
```

Configuring a health monitor

We recommend you configure at least an ICMP health monitor for all the Oracle devices in this configuration. If you want to configure more advanced monitors for specific devices, we recommend using the template for the HTTP Extended Content Verification (ECV) monitor. The HTTP ECV monitors nodes (IP address and port combinations), and can be configured to use **send** and **recv** statements in an attempt to retrieve explicit content from nodes. The following procedures show you how to configure both the ICMP and ECV monitors.

We recommend you configure the health monitors from the Configuration utility. For information on how to configure the health monitor from the command line, see the *BIG-IP Reference Guide*.

To configure an ICMP health monitor using the BIG-IP Configuration utility.

1. In the navigation pane, click **Monitors**.
The Network Monitors screen opens.
2. From the Network Monitors screen, click the Basic Associations tab.
The Basic Association screen opens.
3. In the **Node Address** section, from the list, select **ICMP**.
4. In the **Node** column, locate the Oracle node addresses, and click a check in the **Add** box for each node address.

In our example, we check the Add box for all of the Oracle nodes.

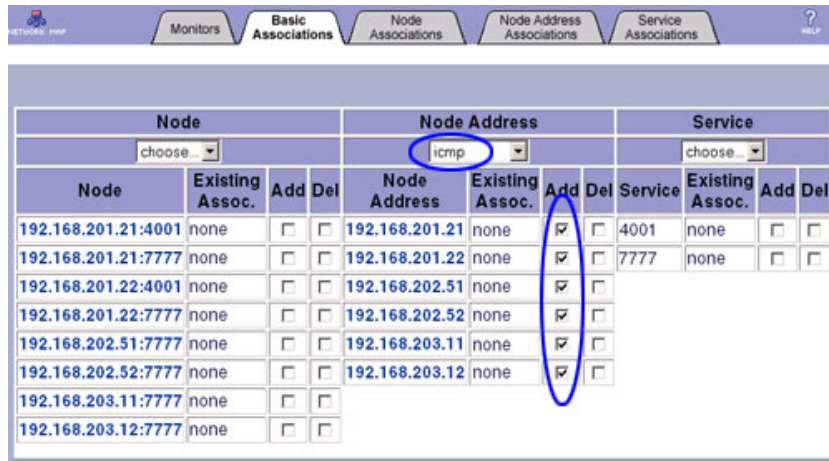


Figure 1.10 Associating the Oracle monitor with the nodes

5. Click **Apply**.

You now see the **ICMP** in the Existing Associations column of the **Node Address** section for each of the Oracle devices. (see Figure 1.11).

For additional information associating a monitor, click the **Help** button.

Node Address			
choose...			
Node Address	Existing Assoc.	Add	Del
192.168.201.21	icmp	<input type="checkbox"/>	<input type="checkbox"/>
192.168.201.22	icmp	<input type="checkbox"/>	<input type="checkbox"/>
192.168.202.51	icmp	<input type="checkbox"/>	<input type="checkbox"/>
192.168.202.52	icmp	<input type="checkbox"/>	<input type="checkbox"/>
192.168.203.11	icmp	<input type="checkbox"/>	<input type="checkbox"/>
192.168.203.12	icmp	<input type="checkbox"/>	<input type="checkbox"/>

Figure 1.11 Nodes associated with the ICMP monitor

You can now configure the optional HTTP ECV monitor. In this example, we configure an HTTP ECV monitor for our Midtier forms servers. We recommend you create individual HTTP ECV monitors for each Oracle Application Server 10g Application Server instance.

1. In the navigation pane, click **Monitors**.
The Network Monitors screen opens.
2. Click the **Add** button.
The Add Monitor dialog box opens.
3. In the Add Monitor screen, type the name of your monitor (it must be different from the monitor template name), in our example, we type **oracle_forms_monitor**.
In the **Inherits From** box, select the **http** monitor template from the list. Click the **Next** button.

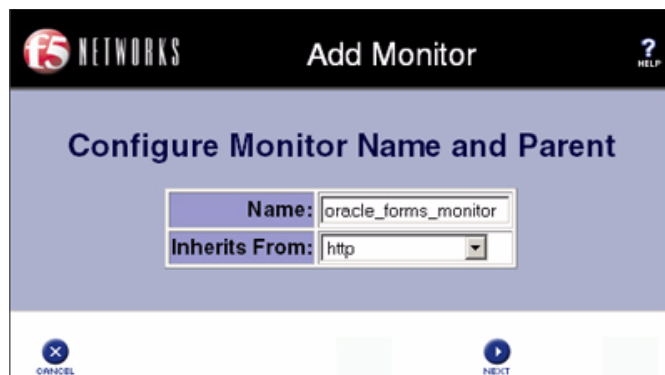


Figure 1.12 Creating the monitor in the BIG-IP Configuration utility

4. In the Configure Basic Properties section, type an **Interval** and **Timeout** value. We recommend at least a 1:3 +1 ratio between the interval and the timeout (for example, the default setting has an interval of **5** and an timeout of **16**). We recommend a slightly higher ratio. In our example, we enter **30** for the Interval and **91** for the Timeout.
Click the **Next** button.
The Configure ECV HTTP Monitor screen opens.
5. In the Configure ECV HTTP Monitor screen, you can add a Send String and Receive Rule specific to that application. Complete the relevant information, and click the **Done** button.
The Add Monitor dialog box closes, and you return to the Network Monitors screen.
6. From the Network Monitors screen, click the Basic Associations tab.
The Basic Association screen opens.
7. In the **Node** section, from the list, select the name of the monitor you created in Step 3. In our example, we select **oracle_forms_monitor**.

8. In the **Node** column, locate the Oracle nodes relevant to this monitor, and click a check in the **Add** box for each node.

In our example, we check the **Add** box for our Forms servers:
192.168.202.51:7777 and **192.168.202.51:7777**.

9. Click **Apply**.
You now see the **oracle_forms_monitor** in the Existing Associations column of the **Node** section for each of the Oracle Forms devices.

For additional information associating a monitor, click the **Help** button.

Synchronizing the BIG-IP configuration if using a redundant system

If you are using a redundant BIG-IP configuration, the final step is to synchronize the configuration to the peer BIG-IP device.

To synchronize the configuration using the Configuration utility

1. In the navigation pane, click **System**.
The Network Map screen opens.
2. Click the Redundant Properties tab.
The Redundant Properties screen opens.
3. Click the **Synchronize Configuration** button.

To synchronize the configuration from the command line

Synchronize the configuration from the command line using the **bigpipe config sync** command:

```
bigpipe config sync all
```

The **bigpipe config sync all** command synchronizes the following configuration files:

- The common **bigdb** keys
- All files in **/config** (except **bigip_base.conf**)

Use the **bigpipe config sync** command without the **all** option to synchronize only the boot configuration file **/config/bigip.conf**.

Appendix A: Backing up and restoring the BIG-IP system configuration

We recommend saving your BIG-IP configuration before you begin this configuration. When you save the BIG-IP configuration, it collects the following critical data and compress it into a single User Configuration Set (UCS) file:

- BIG-IP configuration files
- BIG-IP license and passwords
- SSL certificates
- SSH keys

Saving up and restoring the BIG-IP configuration using the Configuration utility

The Configuration Management screen allows you to save and restore all configuration files that you may edit to configure a BIG-IP system. These configuration files are called a User Configuration Set (UCS). The Configuration Management screen contains sections for saving and restoring a configuration. The list boxes in these sections display only files in the `/usr/local/ucs` directory. If you want to save or restore files from another directory, you must type the full path in the box.

To save the BIG-IP configuration using the Configuration utility

1. In the navigation pane, click **System Admin**.
The User Administration screen displays.
2. Click the Configuration Management tab.
The Configuration Management screen displays.
3. In the **Save Current Configuration** section, type the path where you want your configuration file saved or choose a path from the list box. If no path is specified, the BIG-IP saves files to `/usr/local/ucs`. The BIG-IP appends the extension.ucs to file names without it. In our example, we type **pre_oracle_backup.ucs**.
4. Click the **Save** button to save the configuration file.

To restore a BIG-IP configuration using the Configuration utility

1. In the navigation pane, click **System Admin**.
The User Administration screen displays.
2. Click the Configuration Management tab.
The Configuration Management screen displays.

3. In the **Restore a Configuration** section, choose the configuration file you want to restore from the list box, or type the path where your configuration files were saved.
4. Click the **Restore** button.
To check the status of the restoration, click the **View Log** button. You should wait a few moments for the log file to start generating before you click **View Log**. Repeated clicking of this button will update your screen with the most current log file information until the restoration is complete.

Saving and restoring the BIG-IP configuration using the bigpipe command line interface

You can also save and restore your configuration using the **bigpipe** command line.

To save and backup your BIG-IP configuration data from the command line

1. From the command line, use the following syntax:

```
bigpipe config save <filename>
```

Where **<filename>** is the name of the your saved configuration.

In our example, we type:

```
bigpipe config save preLCS_backup
```

Note: By default, BIG-IP saves the UCS to the following location: /usr/local/ucs. F5 Networks recommends that you use this default.

2. Copy the UCS file to a remote location.

Important

Important: The UCS file contains both authorization data and configuration files. It is critical that you safely store this file in a remote location.

To reinstall configuration data from the command line

To reinstall the UCS file that you created and stored during the backup procedure, perform the following steps:

1. Verify that the BIG-IP system has an IP address and route to the remote host on which the UCS file is located.
2. Copy the UCS file from the remote host to the **local /usr/local/ucs** directory.
3. If the hostname of the local system has changed since you created the original UCS file, you must set the hostname back to the original name. To change the hostname, type the following command, where **<fully-qualified-hostname>** is the original hostname:

```
hostname <fully-qualified-hostname>
```

Important: *If you do not set the hostname back to the original name, the BIG-IP device will only perform a partial installation of the configuration files.*

4. Type the following command to decompress the UCS file and save the configuration files on the BIG-IP:

```
bigpipe config install <filename>
```

5. Reboot the system by typing the following:

```
reboot
```

For more information on saving and restoring a configuration file using the command line (including instructions on how to reinstall the BIG-IP system from scratch and a special case if you are using a BIG-IP version 4.5 RMA system), visit the F5 Networks Technical Support web site (requires registration) and refer to Solution 1493.