Setting up an OracleAS® "myPortal" Enterprise Deployment Architecture with the CAI Networks®, Inc WebMux™ Load Balancer

A Step-by-Step Guide

Version 1.2

Oracle® Corporation
CAI Networks®, Inc.

Updated Feb. 7th 2008
Originally Compiled August 17, 2007

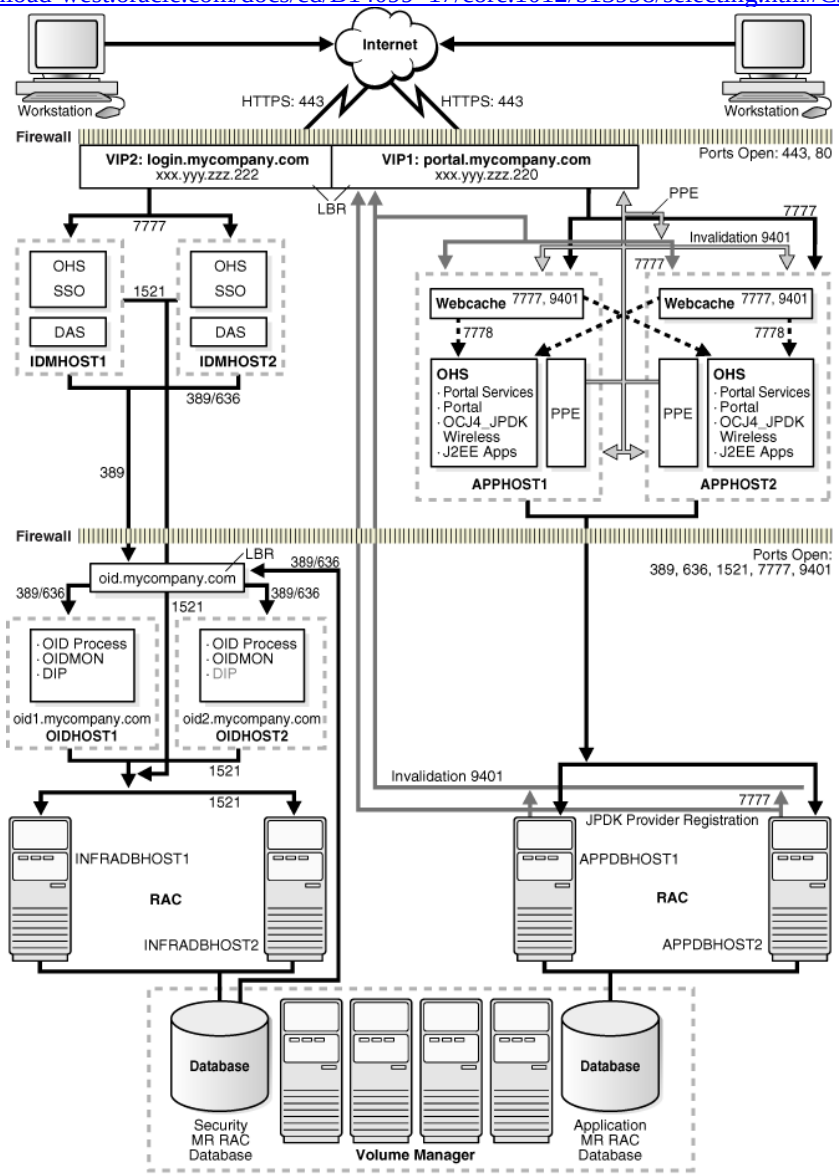
Table of Contents

1. INTRODUCTION

The CAI Networks®, Inc WebMux™ is a high performance network appliance designed for easy deployment of Layer 2-7 load balancing with built-in SSL Termination. This document describes in detail how a basic OracleAS® myPortal Deployment Architecture was set up using the Oracle Application Server® 10g software on Windows 2003 servers and utilizing the WebMux™ as the load balancer, firewall, and SSL accelerator. This document assumes that you are not installing over any existing Oracle® installations (a fresh install) or utilizing any existing Oracle® directory servers or databases. Because of several assumed prerequisites and wrong server references in the original Oracle® documentation, we have decided to include the detailed step by step instructions for setting up each Oracle® server to eliminate the confusion and frustration that may be encountered by the novice user. Our examples will be using a fictitious domain "mycompany.com". Please substitute the appropriate domain name for your setup.

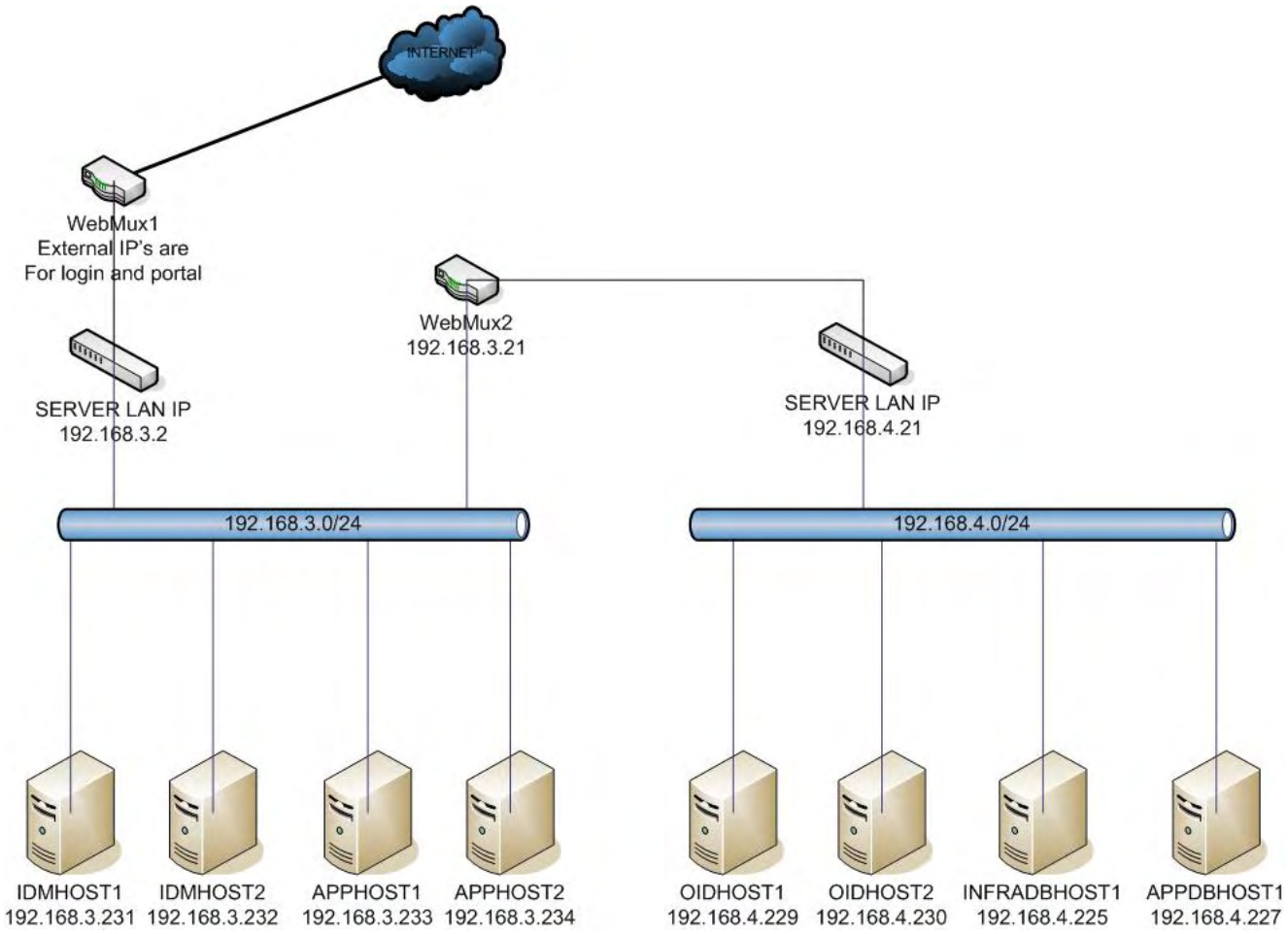
2. NETWORK AND SERVER CONFIGURATION

There will be a total of eight separate servers used in the example setup: IDMHOST1 and IDMHOST2, APPHOST1 and APPHOST2, OIDHOST1 and OIDHOST2, INFRADBHOST1 and APPDBHOST1. This is a slight variation from the example found in the Oracle® documentation at:
http://download-west.oracle.com/docs/cd/B14099_17/core.1012/b13998/selecting.htm#CJACHBBC



The difference is that no actual Real Application Clusters (RAC) database was configured for this deployment example. The databases used in this document are only single node and not recommended for an actual deployment. Setting up an RAC is beyond the scope of this documentation, but we urge you to refer to Oracle®'s extensive documentation repository for further details on that subject. This document will only walk your though setting up the bare minimum number of databases necessary for OracleAS® to work properly.

Here is a simplified topology diagram of the setup we will be doing:



SERVER CONFIGURATION

The following is a list of the servers' network configurations:

Server Name	IP Address	Subnet Mask	Routing	Function
INFRADBHOST1	192.168.4.225	255.255.255.0	Default gw192.168.4.1	DB for infrastructure
APPDBHOST1	192.168.4.227	255.255.255.0	Default gw 192.168.4.1	DB for application servers
OIDHOST1	192.168.4.229	255.255.255.0	Default gw 192.168.4.1 Add route for hosts 192.168.4.225 and 192.168.4.227 to use 192.168.4.1 as gateway.	LDAP server 1
OIDHOST2	192.168.4.230	255.255.255.0	Default gw 192.168.4.1 Add route for hosts 192.168.4.225 and 192.168.4.227 to use 192.168.4.1 as gateway.	LDAP server 2
IDMHOST1	192.168.3.231	255.255.255.0	Default gw 192.168.3.1	SSO server 1
IDMHOST2	192.168.3.232	255.255.255.0	Default gw 192.168.3.1	SSO server 2
APPHOST1	192.168.3.233	255.255.255.0	Default gw 192.168.3.1 Add route for hosts 192.168.3.233 and 192.168.3.234 to use 192.168.3.1 as gateway.	Portal and WebCache 1
APPHOST2	192.168.3.234	255.255.255.0	Default gw 192.168.3.1 Add route for hosts 192.168.3.233 and 192.168.3.234 to use 192.168.3.1 as gateway.	Portal and WebCache 2

EXPLAINATION OF SPECIAL ROUTING RULES

Special routing considerations must be taken because of the way some of the servers communicate with other local servers using IP addresses outside of the local network. As will be detailed later in this document, OIDHOST 1 and 2 are listed under a farm on the WebMux using an IP address outside of their local network. When INFRADBHOST1 queries OIDHOST 1 and 2 via their farm IP on the WebMux™, the WebMux™ will direct the traffic to the OIDHOSTs. Since the WebMux™ does not change the originating IP address, the OIDHOSTs will see that the request came from a client IP that is local. So, it will try reply back to INFRADBHOST1 directly. This will cause a communication breakdown because INFRADBHOST1 is expecting a reply from the external farm IP on the WebMux™ not a local host and, thus, reject the reply. Therefore, it is necessary to force the OIDHOST servers to send their replies back to their default gateway (the WebMux™) so that the WebMux™ will proxy the reply back to INFRADBHOST1. The similar process occurs with APPHOST 1 and 2. As with the way OracleAS® is designed, these servers query themselves and each other using the external farm IP. Likewise, the WebMux™ must proxy the replies back to the servers so that the replies appear to come from the external IP (even though the traffic is really only looping back to itself).

Be sure to list the IP addresses with the proper host names of the servers in the host file of EACH server, including the portal, login, and oid farm IPs and names. Alternately, you can use a DNS for a more centralized control of the names and associated IP addresses.

WEBMUX SETTINGS

Two WebMux™es are required the for the OracleAS® deployment. One in front of the Application/Identity Management Tier (WebMux1) and the other in front of the Data Tier (WebMux2), load balancing OIDHOST1 and OIDHOST2. The firewalls in the diagram have been eliminated since the WebMux™ will act as the firewall for this network.

The WebMux™ settings are as follows (we will start at the bottom tier with WebMux2):

WEBMUX2 (see section 6 for more details)

Mode	NAT
Router LAN IP	192.168.3.21
Router LAN mask	255.255.255.0
External GW	192.168.3.1
Server LAN IP	192.168.4.21
Server LAN mask	255.255.255.0
Server LAN gateway	192.168.4.1
oid.mycompany.com farm	192.168.3.12
servers under “oid” farm	
OIDHOST1	192.168.4.229
OIDHOST2	192.168.4.230

SPECIAL ROUTING CONSIDERATIONS

For proper communication to occur between the different subnets, you must change the WebMux™’s forwarding policy to “accept”.

WEBMUX1

Mode	NAT
Router LAN IP	Your public IP for the WebMux™ (can be the same as one of your farms)
Router LAN mask	refer to your ISP or net admin
External gateway	refer to your ISP or net admin
Server LAN IP	192.168.3.2
Server LAN mask	255.255.255.0
Server LAN gateway	192.168.3.1
login.mycompany.com farm	The public IP for this host.
Servers under the “login” farm	
IDMHOST1	192.168.3.231
IDMHOST2	192.168.3.232
portal.mycompany.com farm	The public IP for this host.
Servers under the “portal” farm	
APPHOST1	192.168.3.233
APPHOST2	192.168.3.234

SPECIAL ROUTING RULES

For proper communication to occur between the different subnets, you must change the WebMux™’s forwarding policy to “accept”. You must add a route on WebMux1 to the 192.168.4.0 network in order for communication between the APPHOST servers and the APPDBHOST1 server to complete. The graphical interface for adding route is through /cgi-bin/route or superuser command line (87 for telnet or 77 for ssh by default) and issue a route command:

route add –net 192.168.4.0/24 gw 192.168.3.12 dev ethb0

As you might notice, 192.168.3.12 is the farm IP for oid.mycompany.com. You can use any IP address on the WebMux2's front network. But it is recommended that you use a farm IP as the gateway because in a redundant WebMux™ set up each WebMux™ will have its own unique device IP, but the farm IPs will float between the two. So, should one WebMux™ go down, your gateway IP will still be valid. You can also create an empty farm so that you have an IP on the front interface of WebMux2, but not actually load balancing any servers for extra security.

The reason why this route needs to be added is because the APPHOST servers point to WebMux1 as its gateway and in turn WebMux1 will continue out to the public internet. The problem comes when APPDBHOST1 on the 192.168.4.0 network tries to query the APPHOST servers using the public IP associated with the portal.mycompany.com name. WebMux1 will correctly send the communication to the APPHOST servers. However, a problem is encountered when the APPHOST servers try to reply back. It sees that the 192.168.4.0 network is not local and will send the reply through it default gateway, WebMux1. WebMux1 will in turn see that 192.168.4.0 network is not in its local network. So, it will continue sending the reply out to its default GW which would be the ISP. Adding the static route for the 192.168.4.0 network pointing to 192.168.3.12 (an IP on WebMux2 which WebMux1 can reach) solves this problem. Once WebMux1 routes the reply to WebMux2, WebMux2 in turn routes the reply back to APPDBHOST1.

Although the WebMux™ is not primarily designed to be a firewall, you can manually issue *iptables* rules from the command line interface to create the desired firewall rules to block unused ports. There are many documentations available on the internet regarding the use of the *iptables* command.

3. SETTING UP INFRADBHOST1 (OracleAS® Metadata Repository)

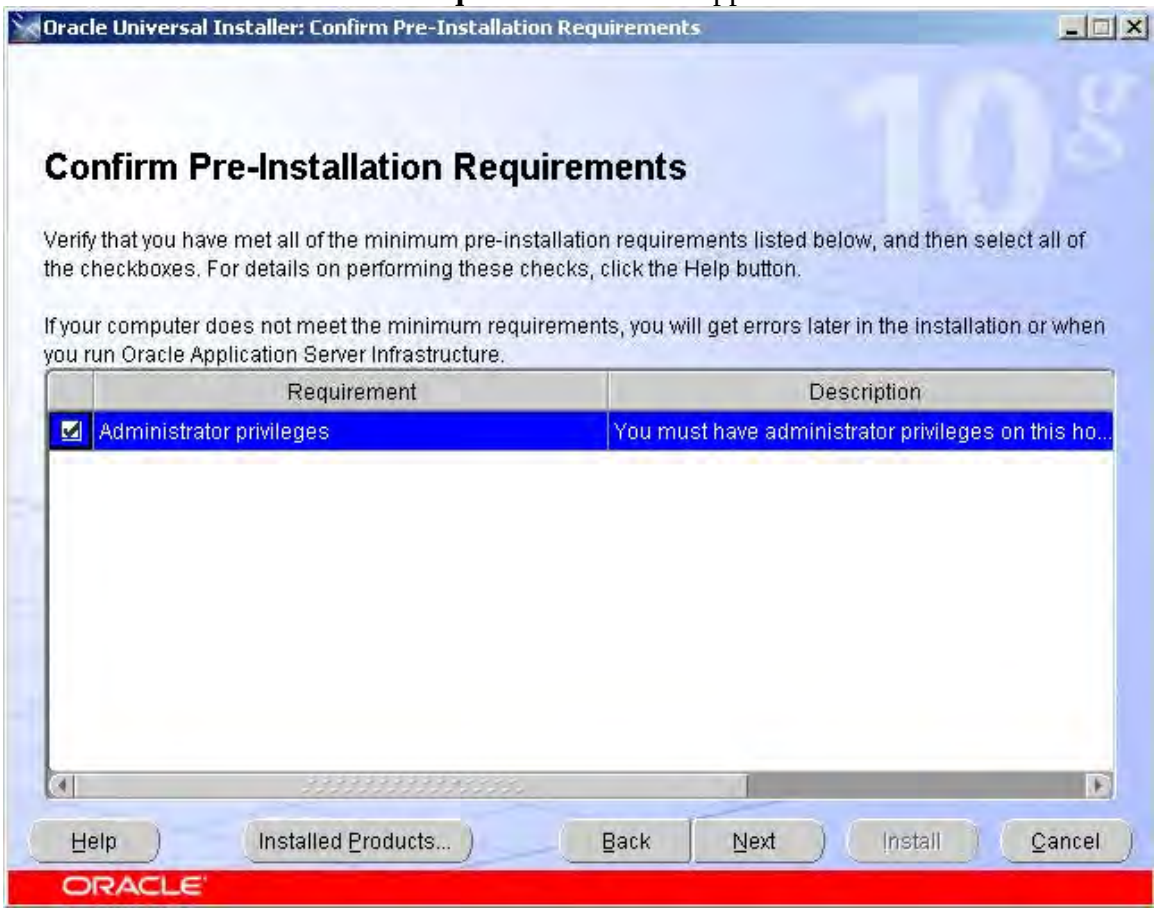
1. Start the Oracle® Universal Installer (double-click *setup.exe*)
   The **Welcome** screen appears.
2. Click **Next**.
3. The **Specify File Locations** screen appears with default locations.
   (we will be using the default locations). Click **Next**.
4. The **Select a Product to Install** screen appears:

Select **Oracle® Application Server Infrastructure 10g**. Click **Next**.

5. The **Select Installation Type** screen appears:

Select **Metadata Repository**. Click **Next**.

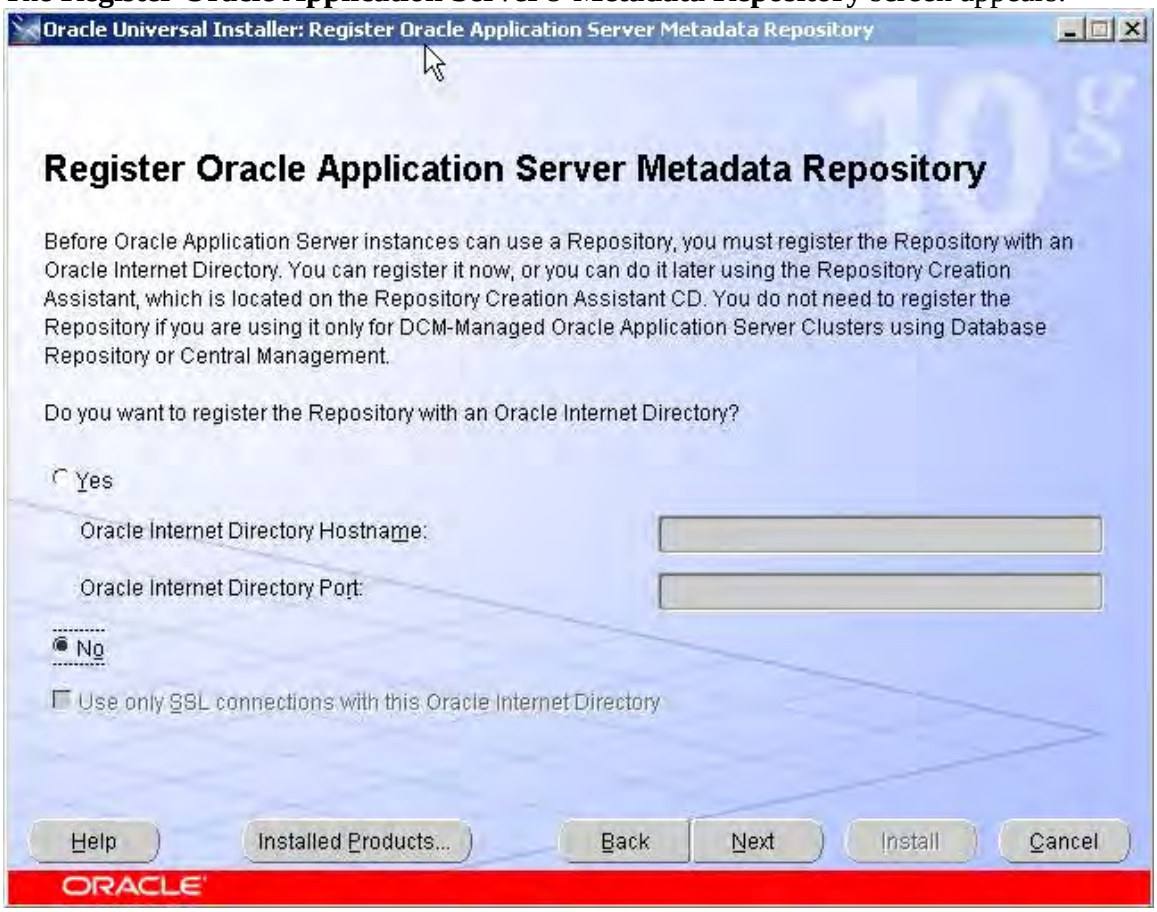
6. The **Confirm Pre-Installation Requirements** screen appears:



Confirm that you have Administrator privileges and click **Next**.

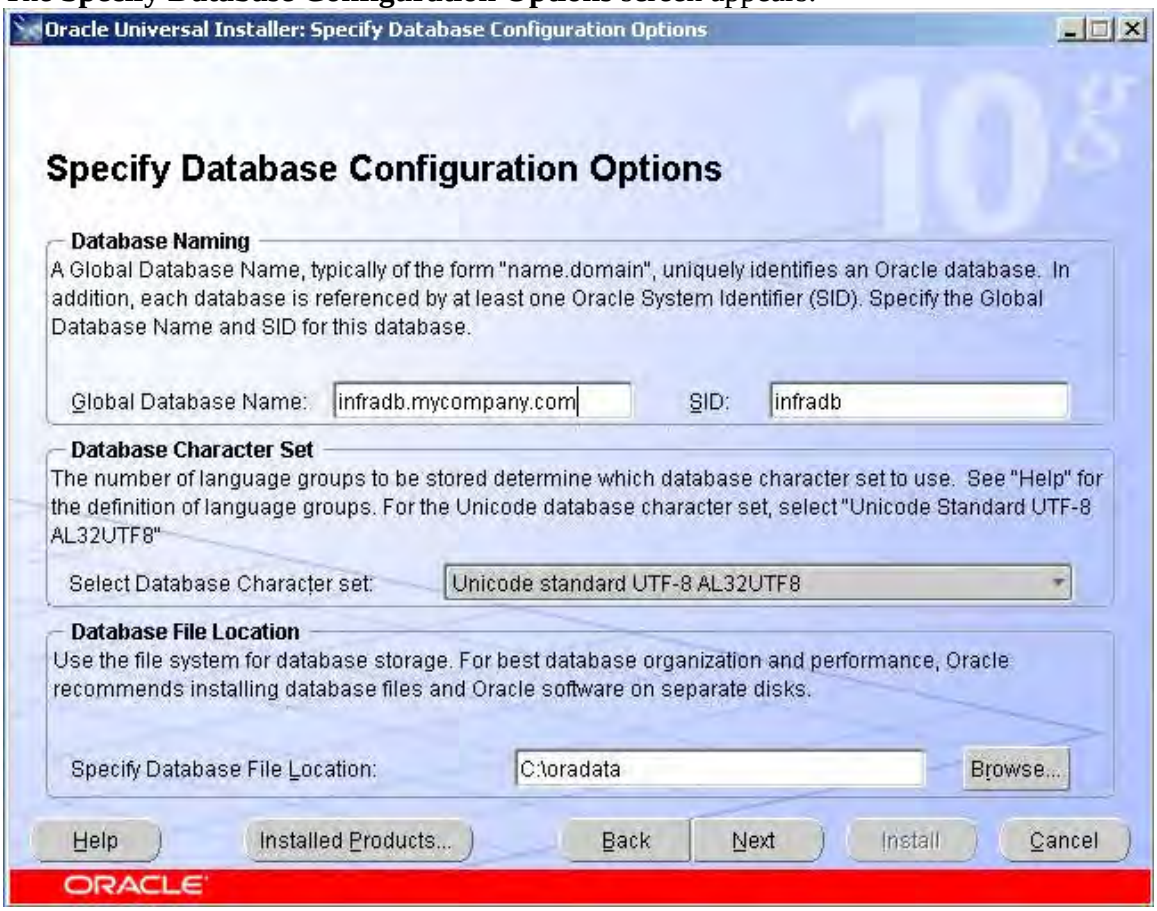
7. The **Select Configuration Options** screen appears. The Oracle Application Server® Metadata Repository should already be selected and cannot be unselected. Click **Next**.

8. The **Register Oracle Application Server® Metadata Repository** screen appears:



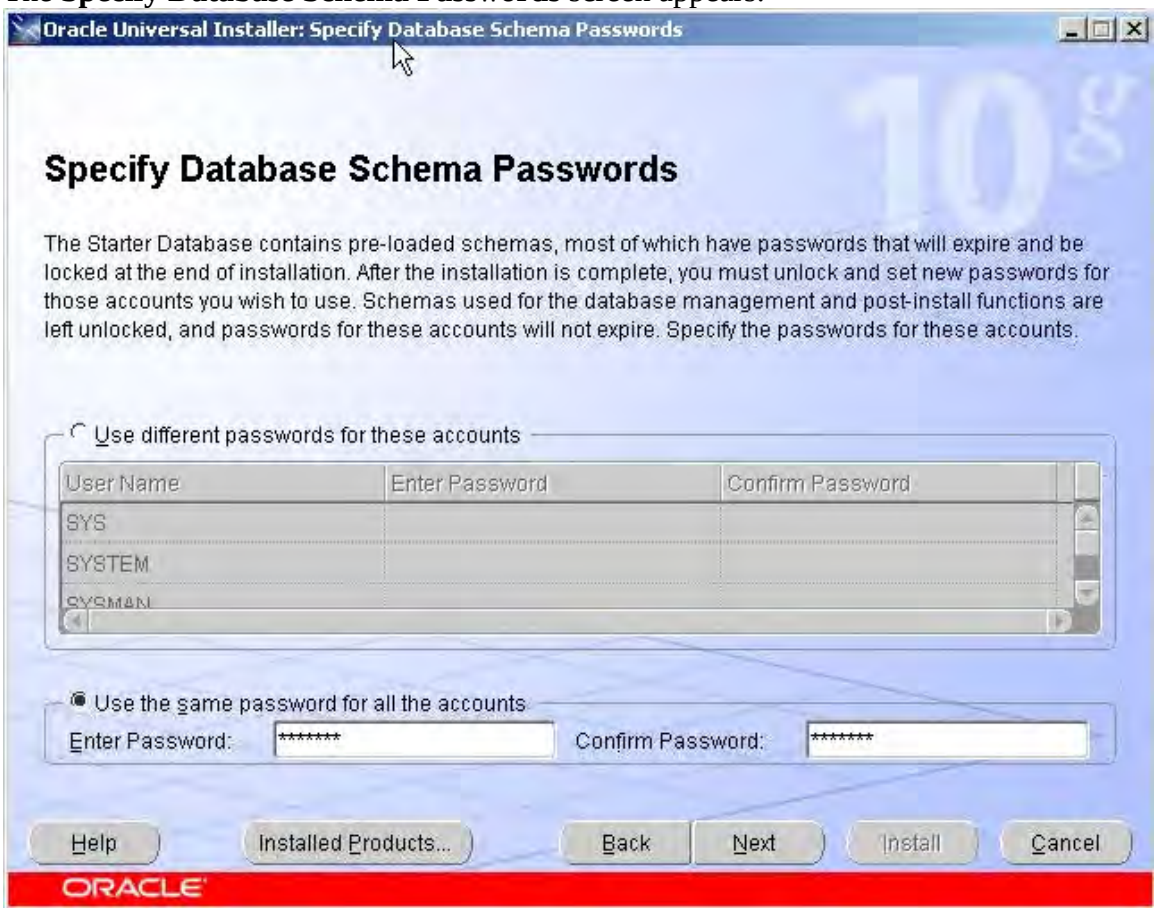
Since there are currently no Oracle® Internet Directories installed, select No, and then click **Next**. (When the first OIDHOST is installed INFRADBHOST1 will be automatically registered).

9. The **Specify Database Configuration Options** screen appears:



Fill out the appropriate Global Database Name. The SID will be automatically filled out. Click **Next**.

10. The **Specify Database Schema Passwords** screen appears:



    You can chose different passwords for each account or use the same password for all accounts. Click **Next**. (We chose to use the same password for all the accounts in our test setup to make things less confusing).
11. The **Summary** screen appears. Click **Install** to begin installation.
12. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
13. Click **Exit**, and confirm your choice to exit.

4. SETTING UP OIDHOST1 (Oracle® Internet Directory)

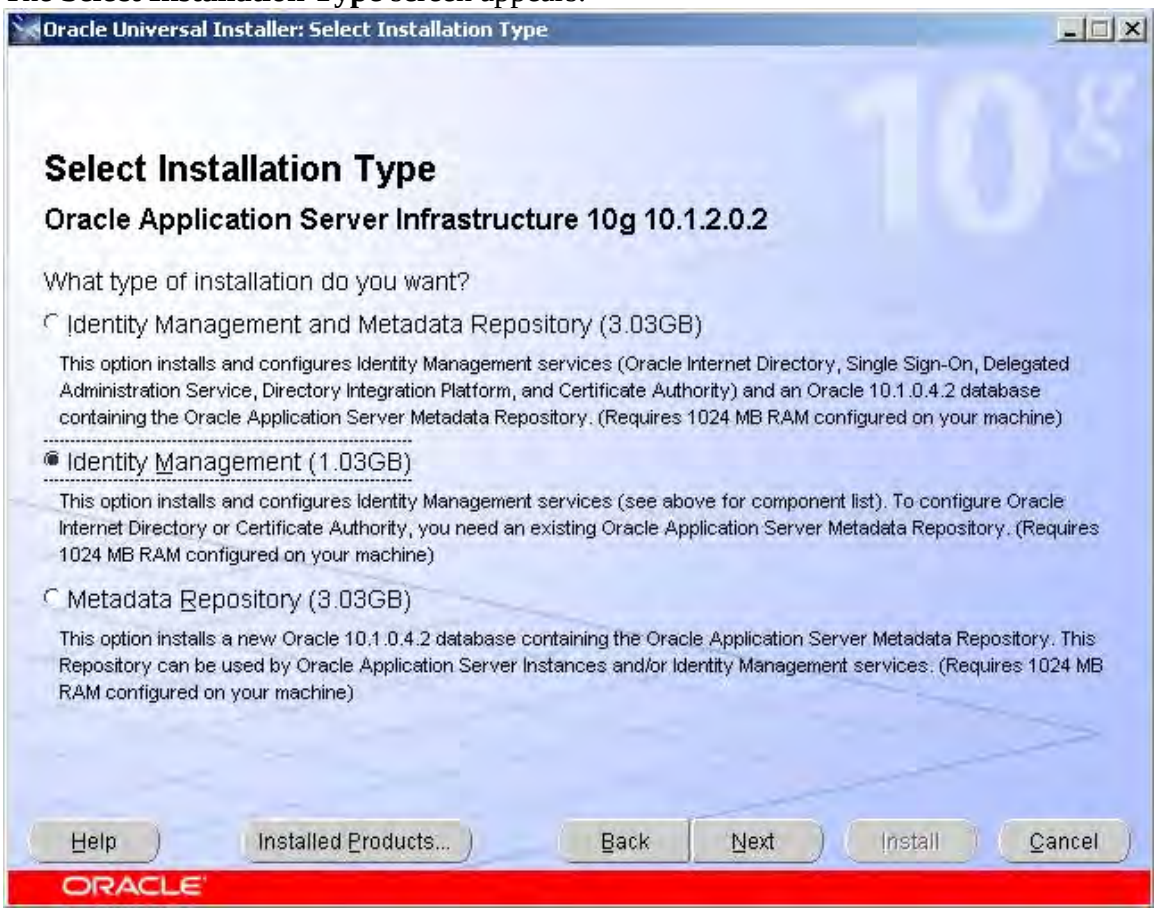
1. Make sure that there are no other services using port 389 and 636 on the computer.
2. Start the Oracle® Universal Installer by double clicking on *setup.exe*
3. The **Welcome** screen appears. Click **Next**.
4. The Specify File Locations screen appears with default locations for:
   - The product files for the installation (Source)
   - The name and path to an Oracle® home (Destination)
     C:\OraHome_1 will be the default destination on all the servers we will be setting up.
5. Click **Next**.
6. At this point, go to the *Disk1\stage\Response* directory of the installation package and copy the staticport.ini file and paste in the *C:\OraHome_1* directory.
7. In the *C:\OraHome_1* directory edit the *staticport.ini* file with the following values:

```
Oracle Internet Directory port = 389
Oracle Internet Directory (SSL) port = 636
```

8. The **Select a Product to Install** screen appears:

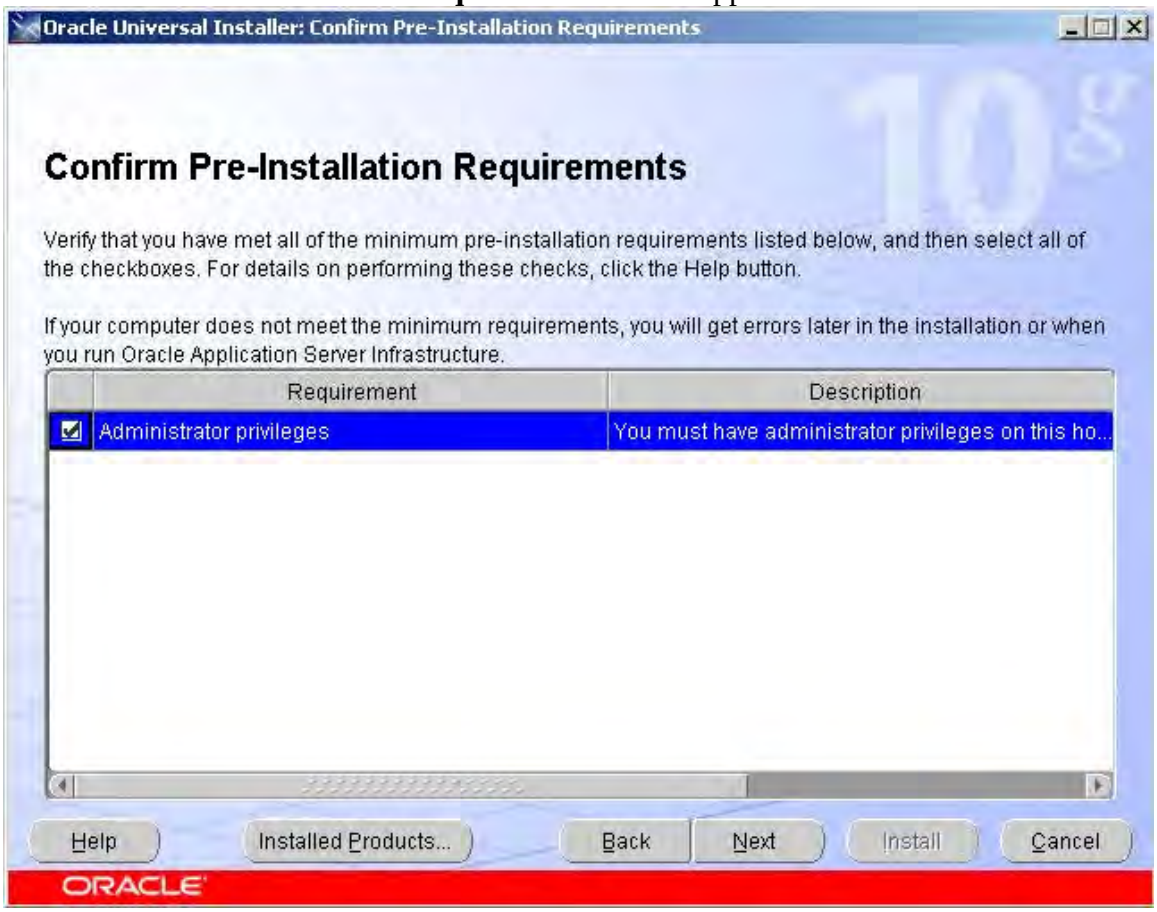
Select **Oracle Application Server Infrastructure 10g**. Click **Next**.

9. The **Select Installation Type** screen appears:



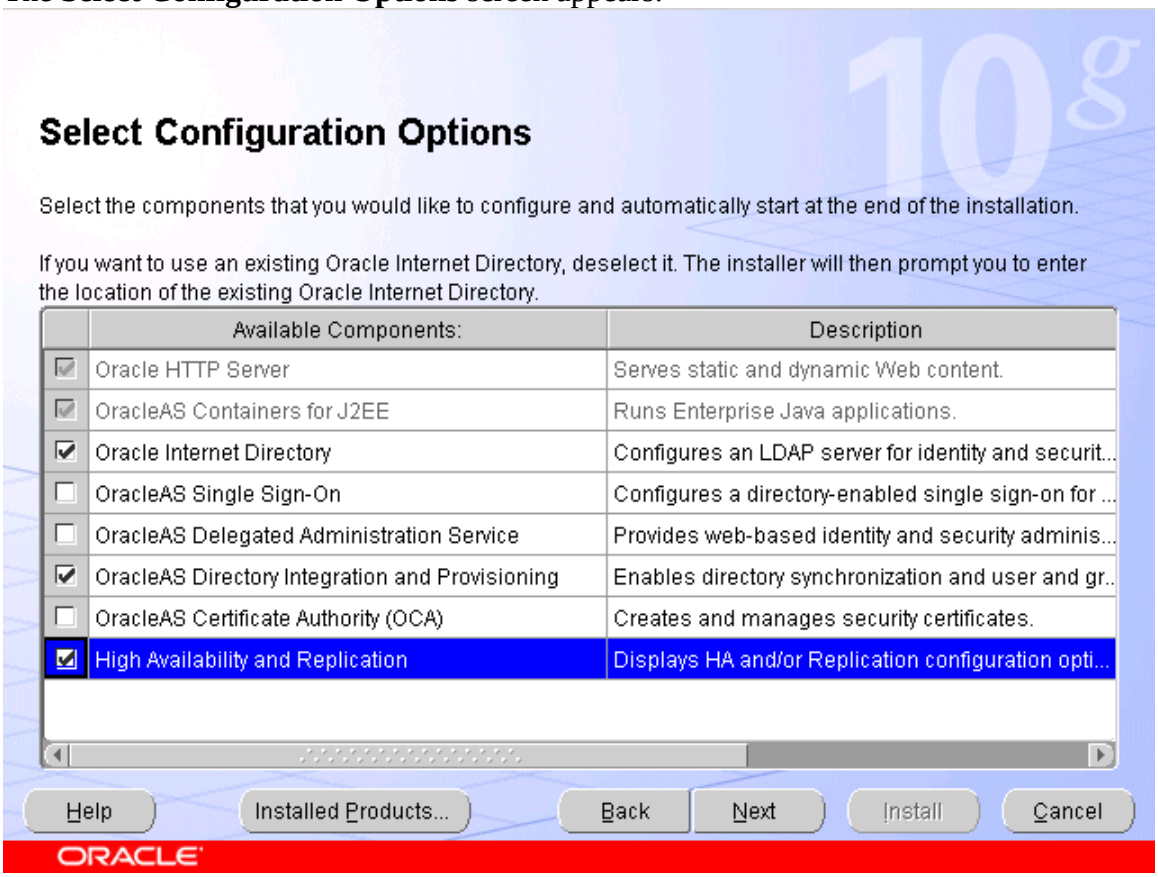
Select **Identity Management** and click **Next**.

10. The **Confirm Pre-Installation Requirements** screen appears:



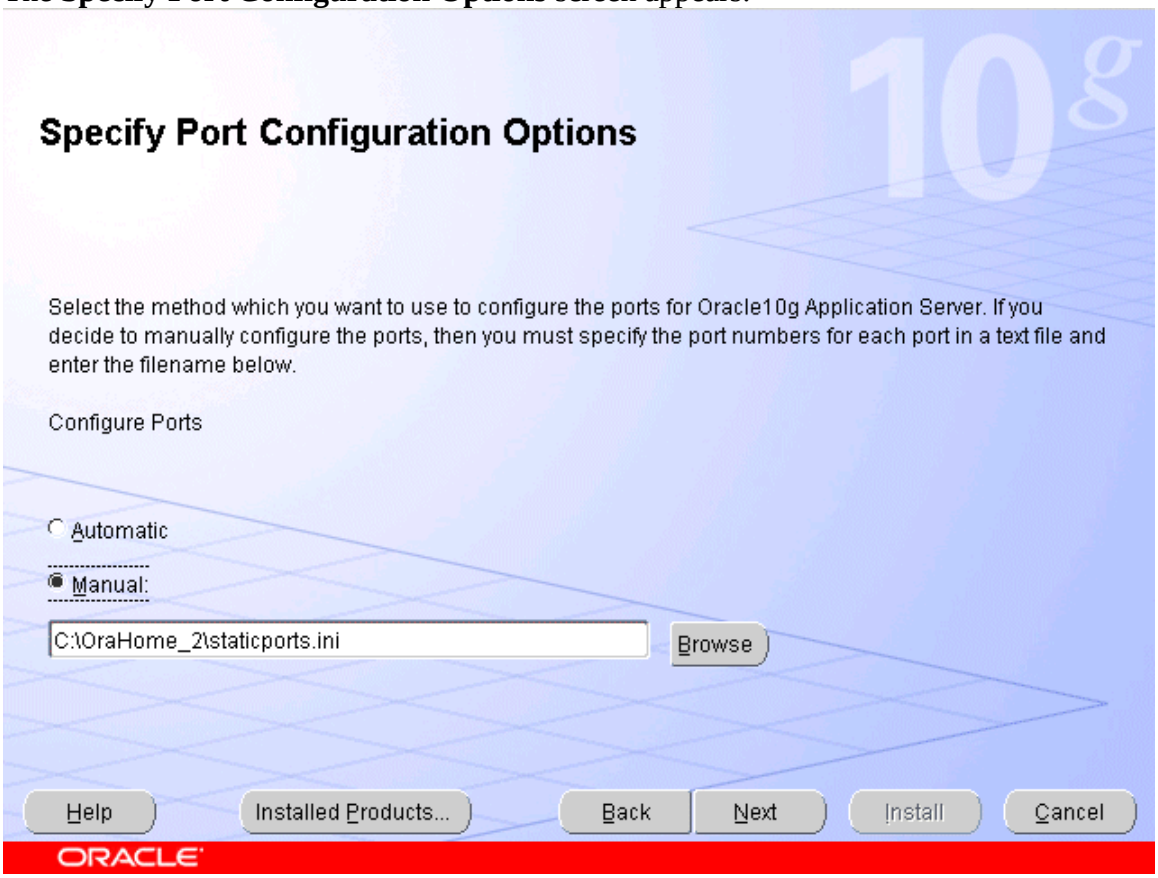
Confirm that you have Administrator privileges and click **Next**.

11. The **Select Configuration Options** screen appears:



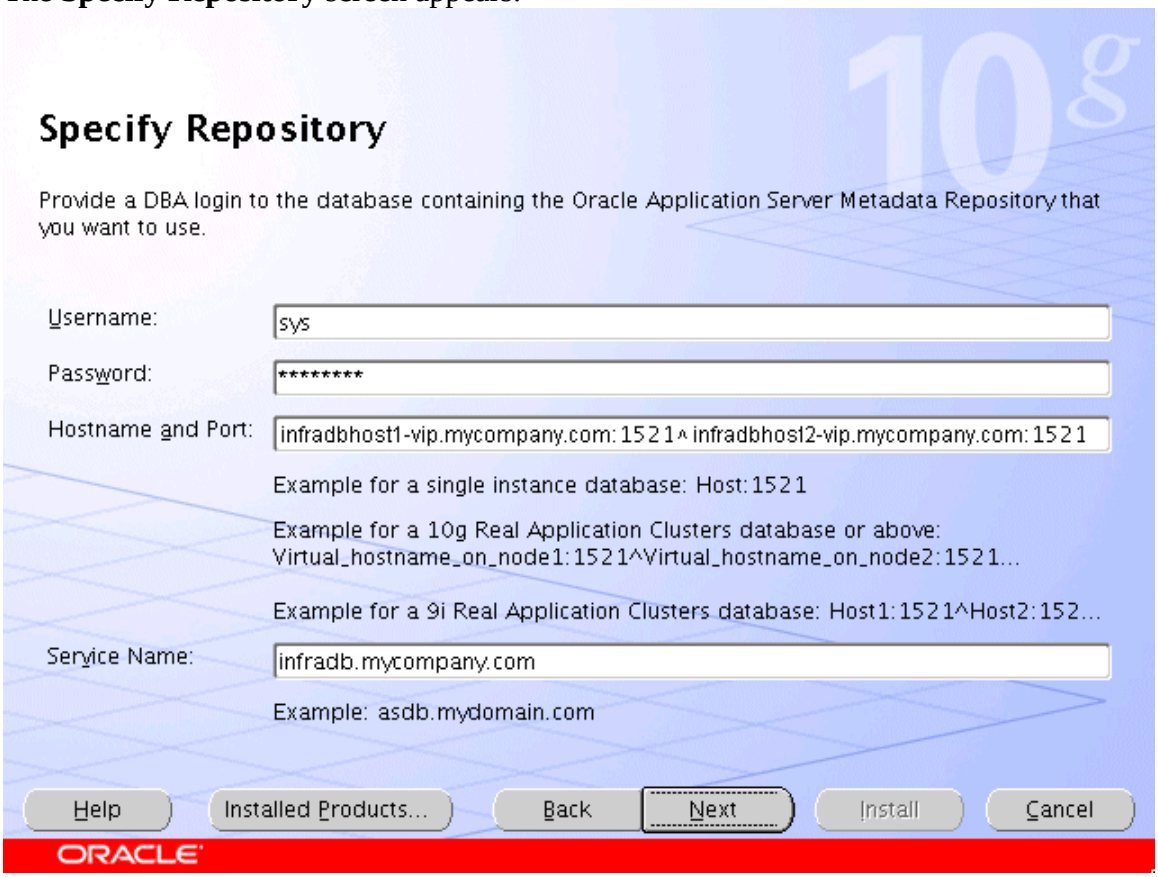
Select **Oracle Internet Directory**, **OracleAS Directory Integration and Provisioning**, and **High Availability and Replication**. Click **Next**.

12. The **Specify Port Configuration Options** screen appears:



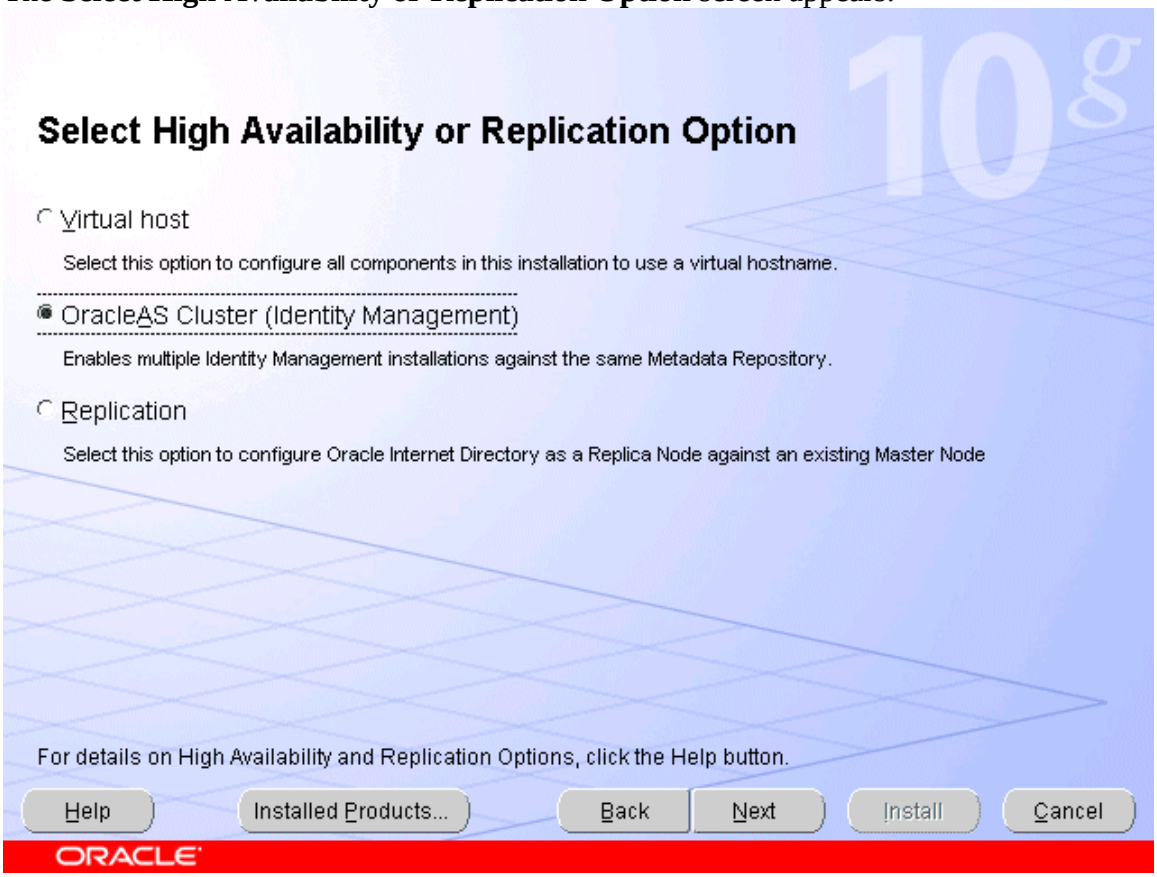
Select manual and select the location of the edited staticports.ini file which should be in *C:\OraHome_1*. Click **Next**.

13. The **Specify Repository** screen appears:



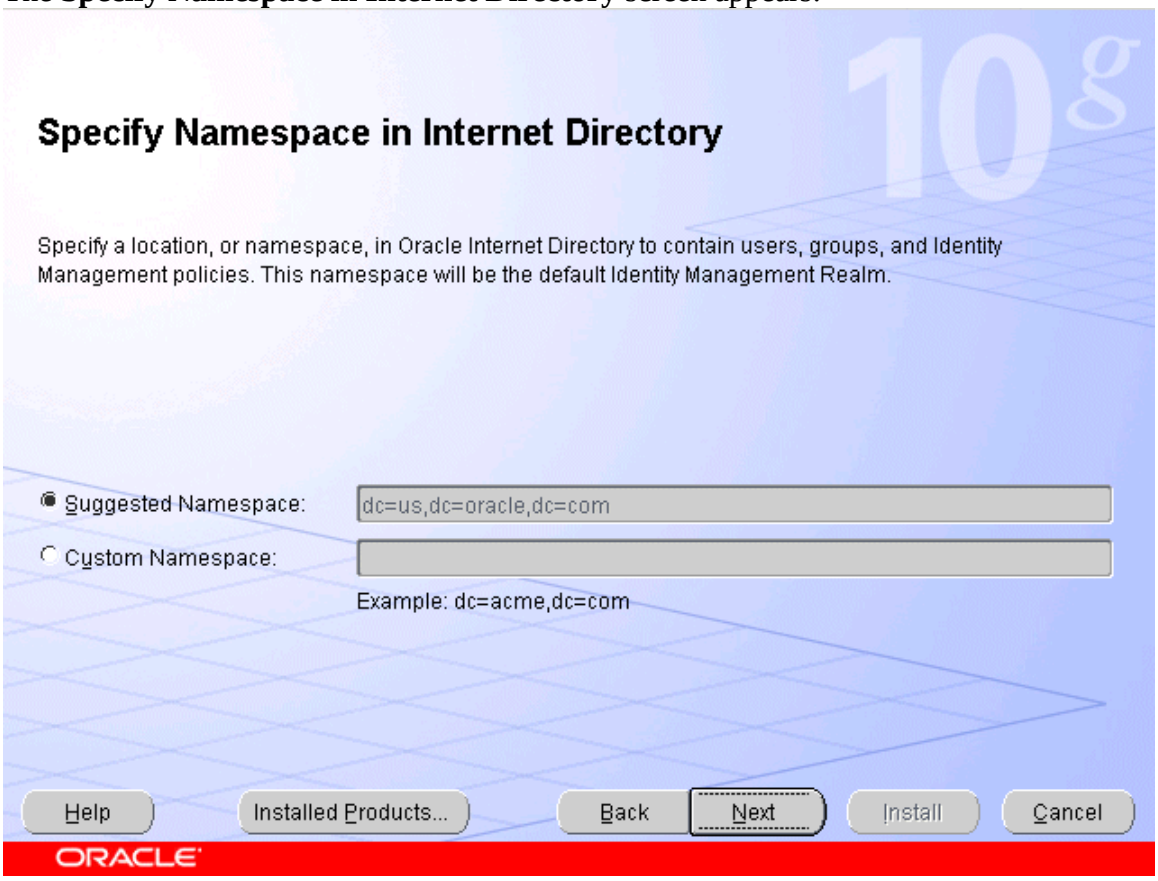
Enter *sys* as the username and enter the password you assigned to it when you set up INFRADBHOST1. In the hostname and port enter only *infradbhost1:1521* since there is only a single instance database. Enter the Global Database Name you assigned when you set up INFRADBHOST1 in the Service Name section. Click **Next**.

14. The **Select High Availability or Replication Option** screen appears:



Select **OracleAS Cluster (Identity Management)**. Click **Next**.

15. The **Specify Namespace in Internet Directory** screen appears:



Select the **Suggested Namespace** and click **Next**.

16. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **oidhost1** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
17. The **Summary** screen appears. Click **Install** to begin installation.
18. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
19. Click **Exit**, and confirm your choice to exit.

5. SETTING UP OIDHOST2 (Oracle® Internet Directory)

1. Follow instructions 1 through 13 in section 4 of setting up OIDHOST1.
2. After clicking **Next** on step 13 a dialog box opens, prompting you to synchronize the system time of the primary Oracle® Internet Directory computer and the system time on the computer on which you are installing. Synchronize the system time on the computers and click **OK**.
3. The **Specify ODS Password** screen appears:

By default it is the ias_admin password. Click **Next**.

4. Since there is no existing Oracle® Internet Directories other than the one just installed in the previous section, the installer automatically detects it and asks you for the password:

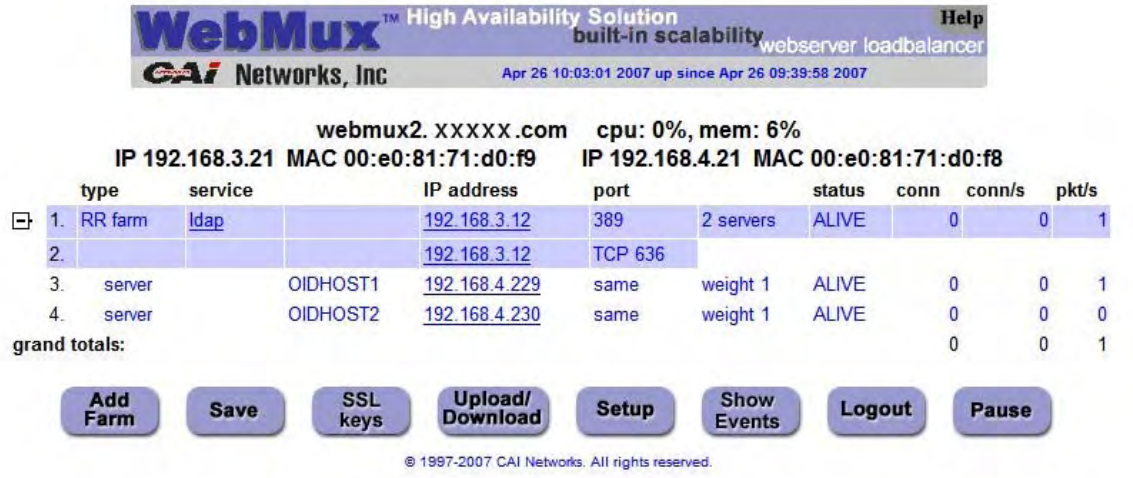
The password is the same that you assigned for the SYS user. Click **Next**.

5. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **oidhost2** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
6. The **Summary** screen appears. Click **Install** to begin installation.
7. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
8. Click **Exit**, and confirm your choice to exit.

6. CREATING THE OID FARM ON WEBMUX2

1. Login the WebMux™ web admin as superuser.
2. Click on the Add Farm button at the bottom of the status screen.
3. In the Add Farm screen, enter 192.168.3.12 as the IP address. Optionally, you can enter oid.mycompany.com in the Label field.
4. In the port number field enter 389.
5. For the service, select LDAP – lightweight directory access protocol.
6. Select Round Robin in the scheduling method.
7. Select NONE for SSL termination.
8. Leave the SSL port field blank.
9. Select NO for the block non-SSL access and NO for the tag-SSL terminated HTTP requests.

10. Click Confirm.
11. Back at the status screen, click on the IP address of the OID farm you just created.
12. In the Modify Farm screen, click on the Add Addr. Port button.
13. In the Add IP address/port screen enter 192.168.3.12 as the IP address.
14. In the port number field enter 636. Optionally, in the Label field you can enter a label to remind yourself such as “secure port”.
15. Click confirm and you will be back in the status screen.
16. You will notice that the OID farm has another 192.168.3.12 address listed with port 636.
17. This allows the OID farm to serve both ports 389 and 636 for the OIDHOST servers.
18. Click on the TOP farm IP link and this time click the Add Server button in the Modify Farm screen.
19. In the IP address field enter 192.168.4.229 for OIDHOST1. Optionally, you can enter OIDHOST1 in the Label field. Click confirm.
20. Repeat step 18 and this time enter 192.168.4.230 for OIDHOST2. Optionally, you can enter OIDHOST2 in the Label field. Click confirm.
21. You should now have a farm with the IP address 192.168.3.12 and the two Oracle® Internet Directories listed under them.



7. TESTING THE OID FARM

1. From INFRADBHOST1 open a command prompt and go to *C:\OraHome_1\bin*
2. Ensure that you can connect to each Oracle® Internet Directory using this command:

   ldapbind –p 389 –h oidhost1

   ldapbind –p 389 –h oidhost2

   If you are unable to connect be sure that your host file or DNS has the proper entries to resolve those host names.

3. Ensure that you can connect to the oid farm using this command:

   ldapbind –p 389 –h oid.mycompany.com

If you are unable to connect be sure that your host file or DNS has the proper entries to resolve the host name. Be sure that you have added the proper routing rules as stated in the SERVER CONFIGURATION of Section 2 for the OIDHOSTs.

8. SETTING UP IDMHOST1 (Identity Management)

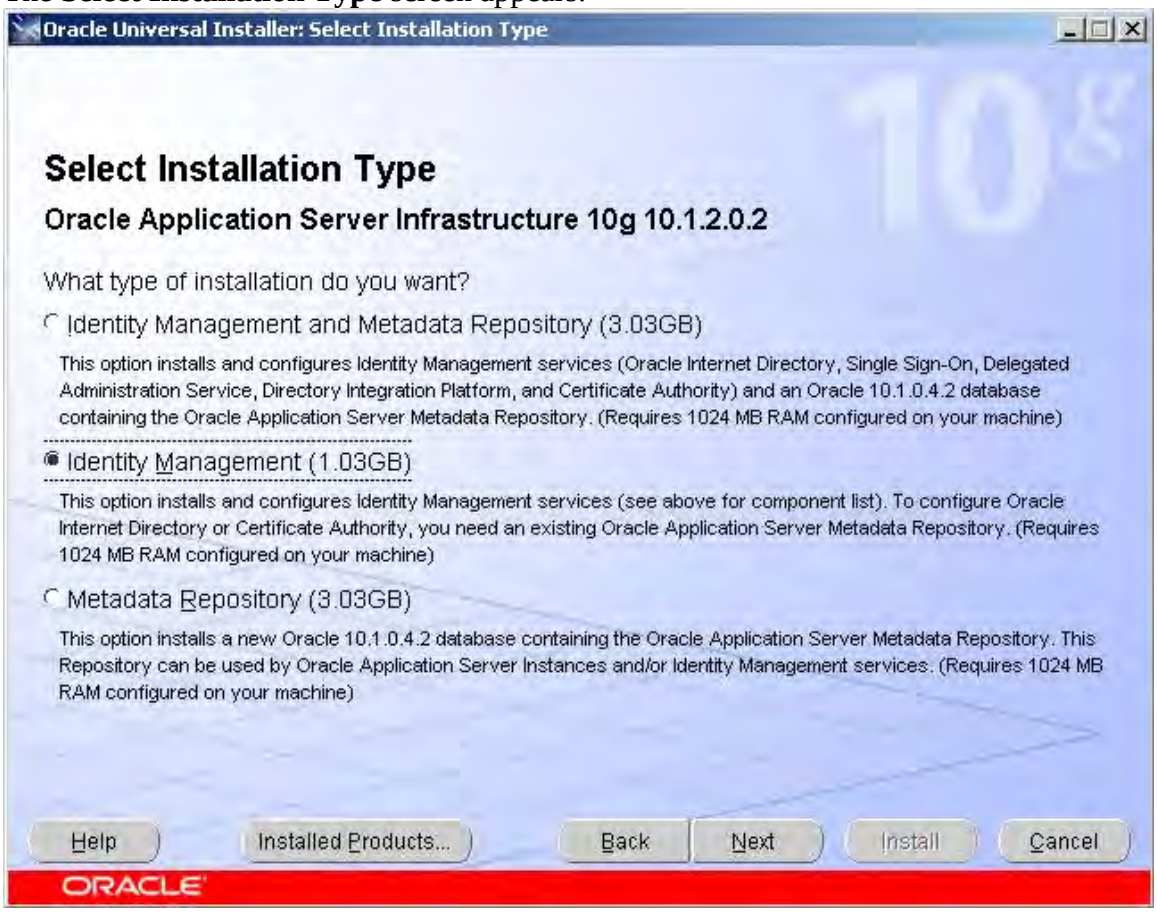
1. Start the Oracle® Universal Installer by double clicking on *setup.exe*
2. The **Welcome** screen appears. Click **Next**.
3. The Specify File Locations screen appears with default locations for:
   - The product files for the installation (Source)
   - The name and path to an Oracle® home (Destination)
     C:\OraHome_1 will be the default destinations on all the servers we will be setting up.
4. Click **Next**.
5. At this point, go to the *Disk1\stage\Response* directory of the installation package and copy the staticport.ini file and paste in the *C:\OraHome_1* directory.
6. In the *C:\OraHome_1* directory edit the *staticport.ini* file with the following values:

```
Oracle HTTP Server port = 7777
Oracle HTTP Server Listen port = 7777
Application Server Control port = 1810
```

7. The **Select a Product to Install** screen appears:

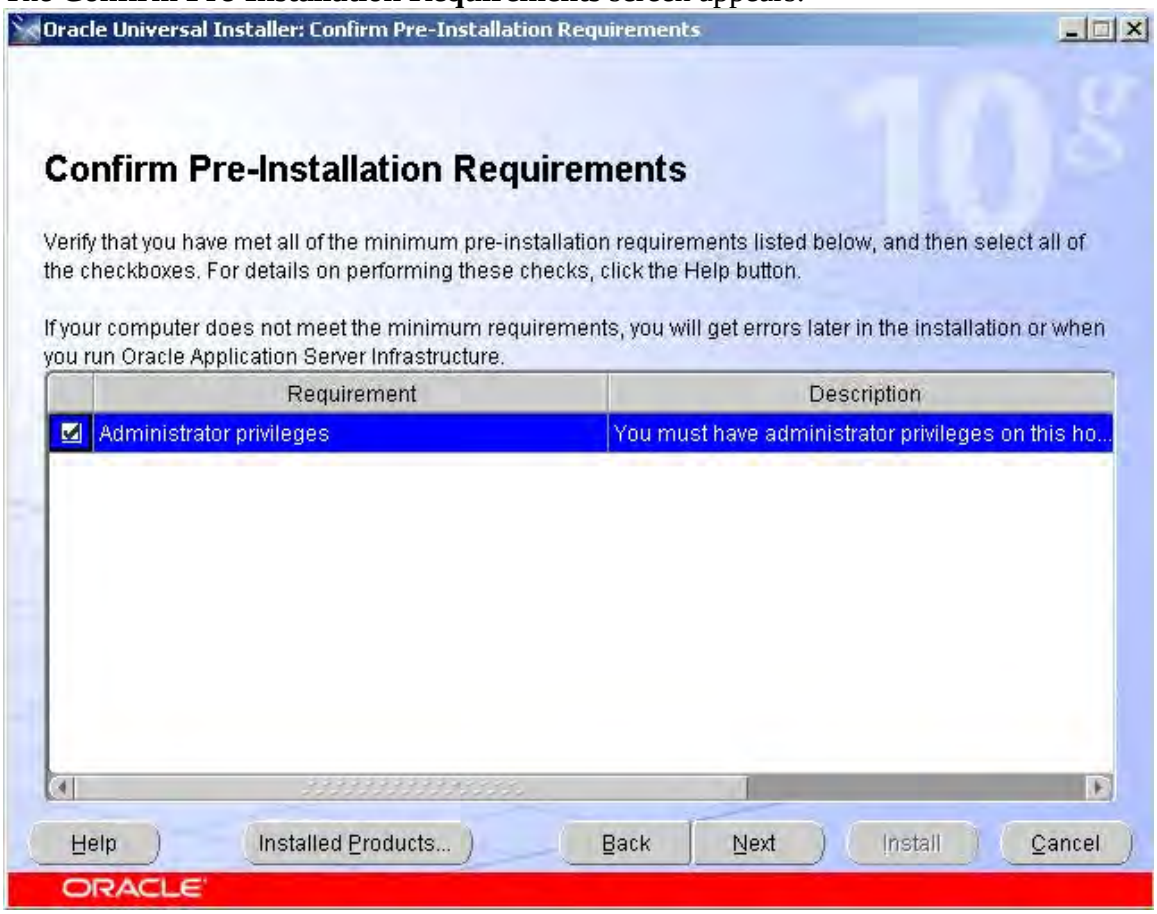
Select **Oracle Application Server Infrastructure 10g**. Click **Next**.

8. The **Select Installation Type** screen appears:



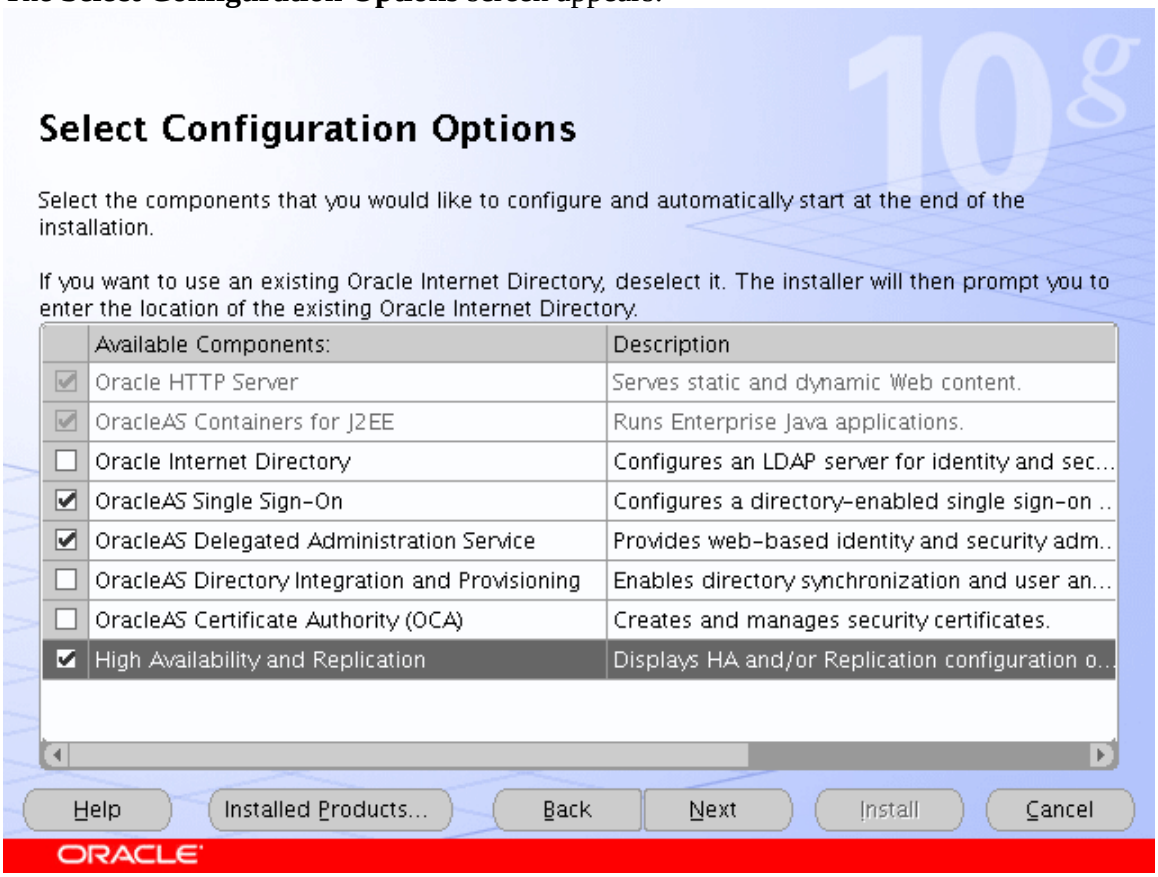
Select **Identity Management** and click **Next**.

9. The **Confirm Pre-Installation Requirements** screen appears:



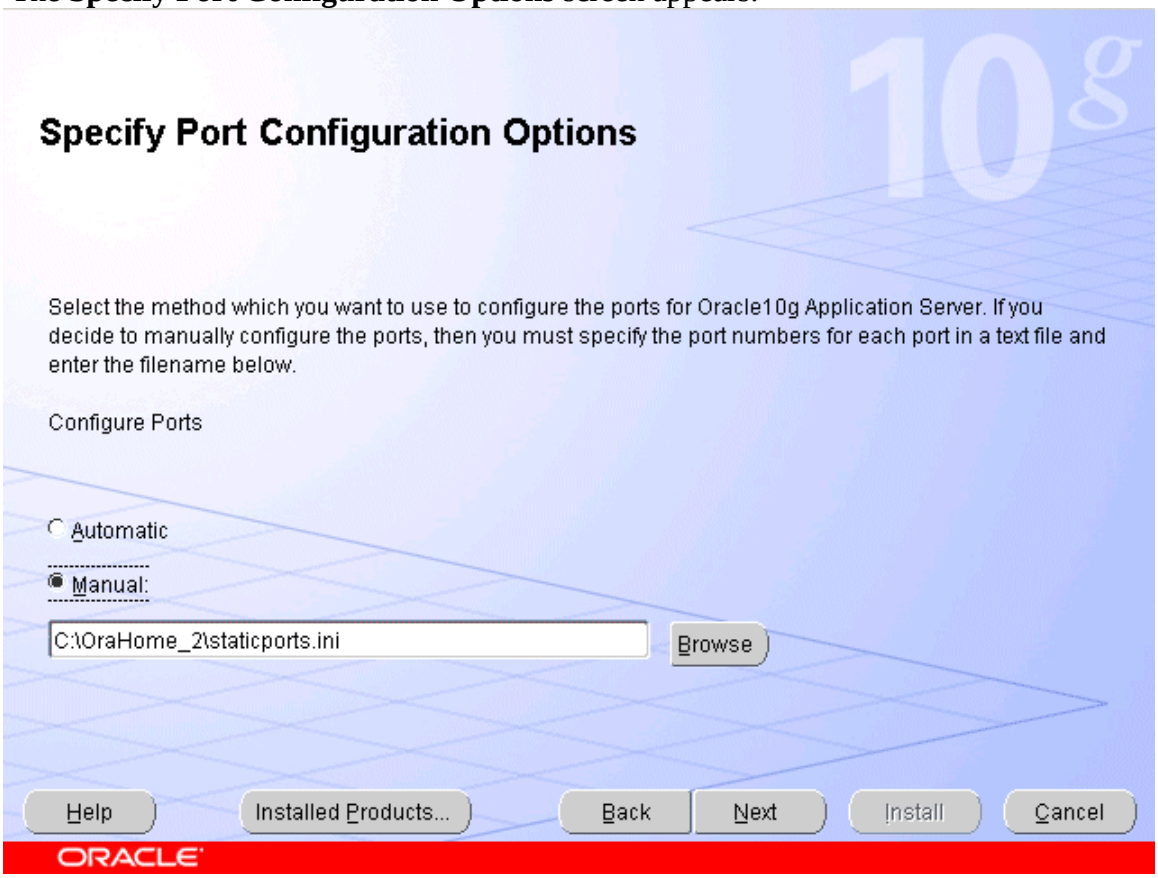
Confirm that you have Administrator privileges and click **Next**.

10. The **Select Configuration Options** screen appears:



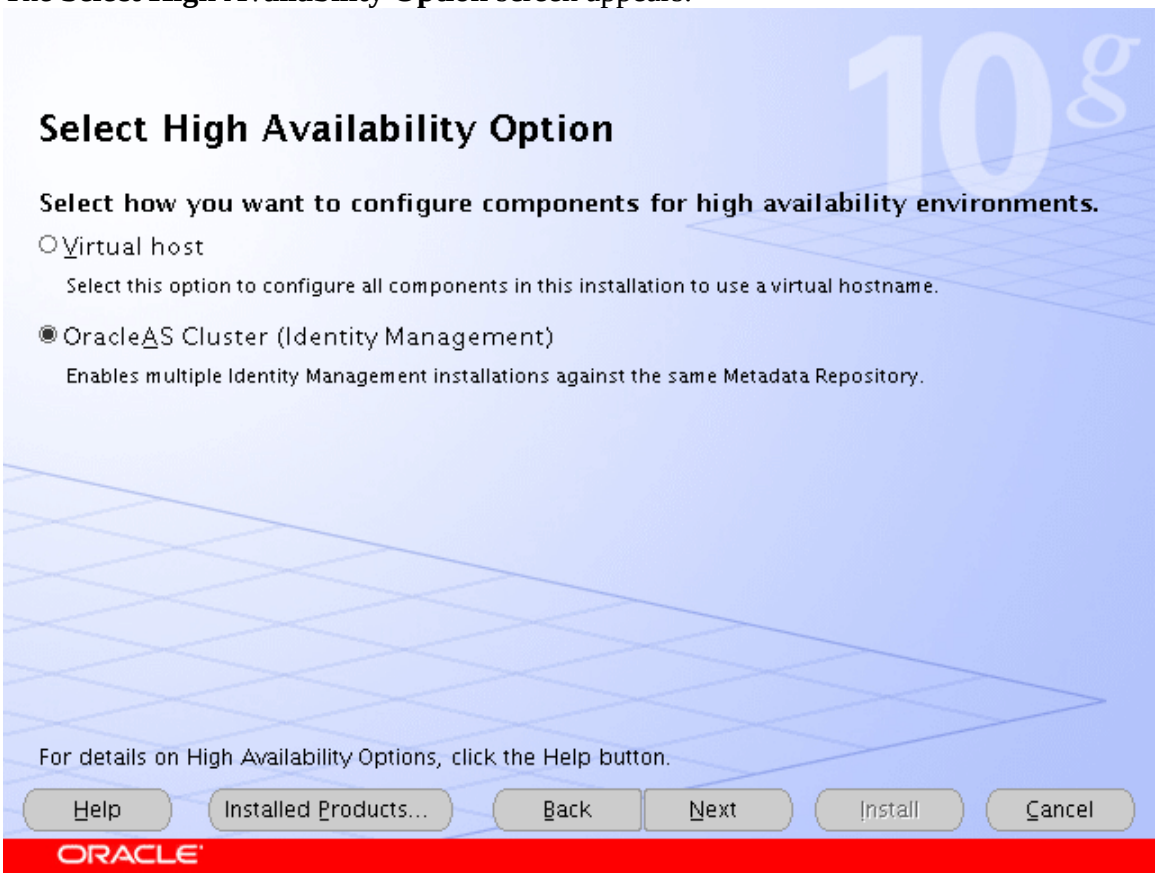
Select **OracleAS Single Sign-On, Oracle Delegated Administration Services**, and **High Availability and Replication**. Click **Next**.

11. The **Specify Port Configuration Options** screen appears:



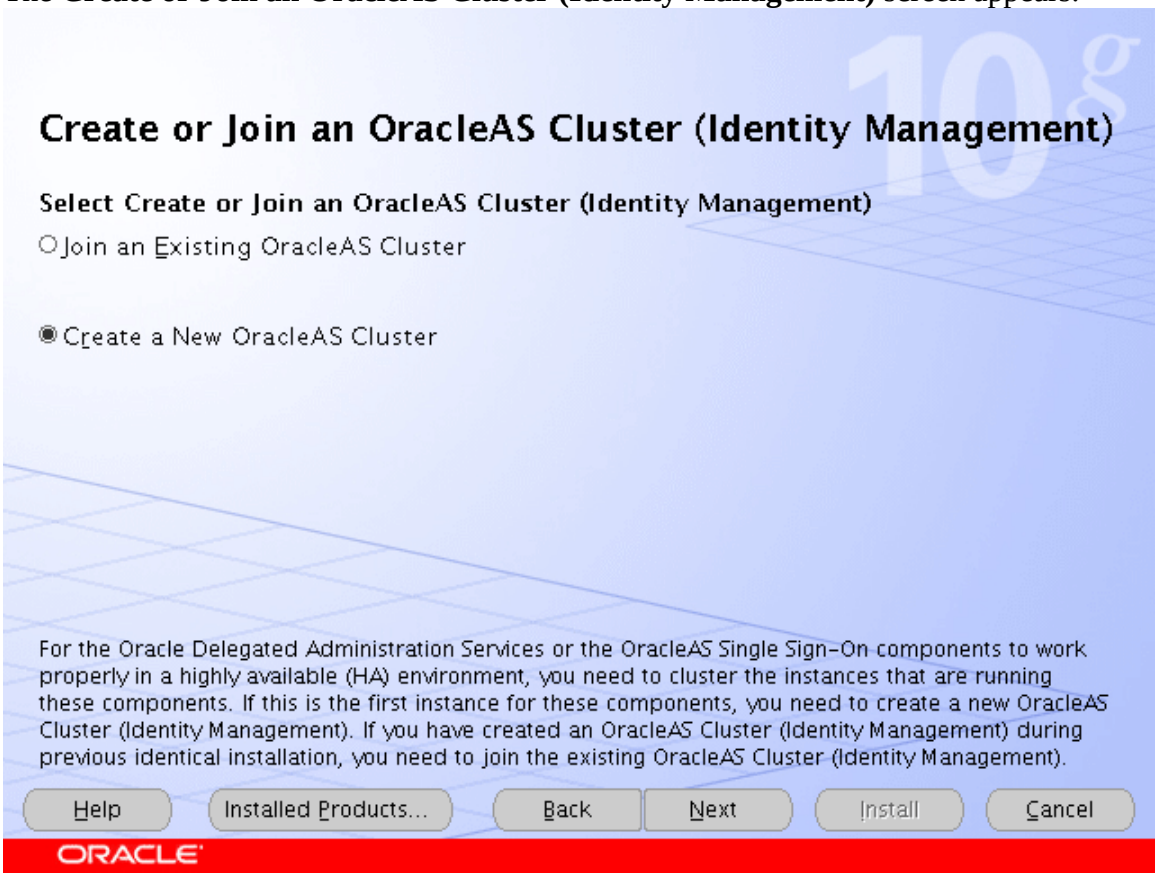
Select manual and select the location of the edited staticports.ini file which should be in *C:\OraHome_1* (not correctly shown in the image). Click **Next**.

12. The **Select High Availability Option** screen appears:



Select **OracleAS Cluster (Identity Management)** and click **Next**.

13. The **Create or Join an OracleAS Cluster (Identity Management)** screen appears:

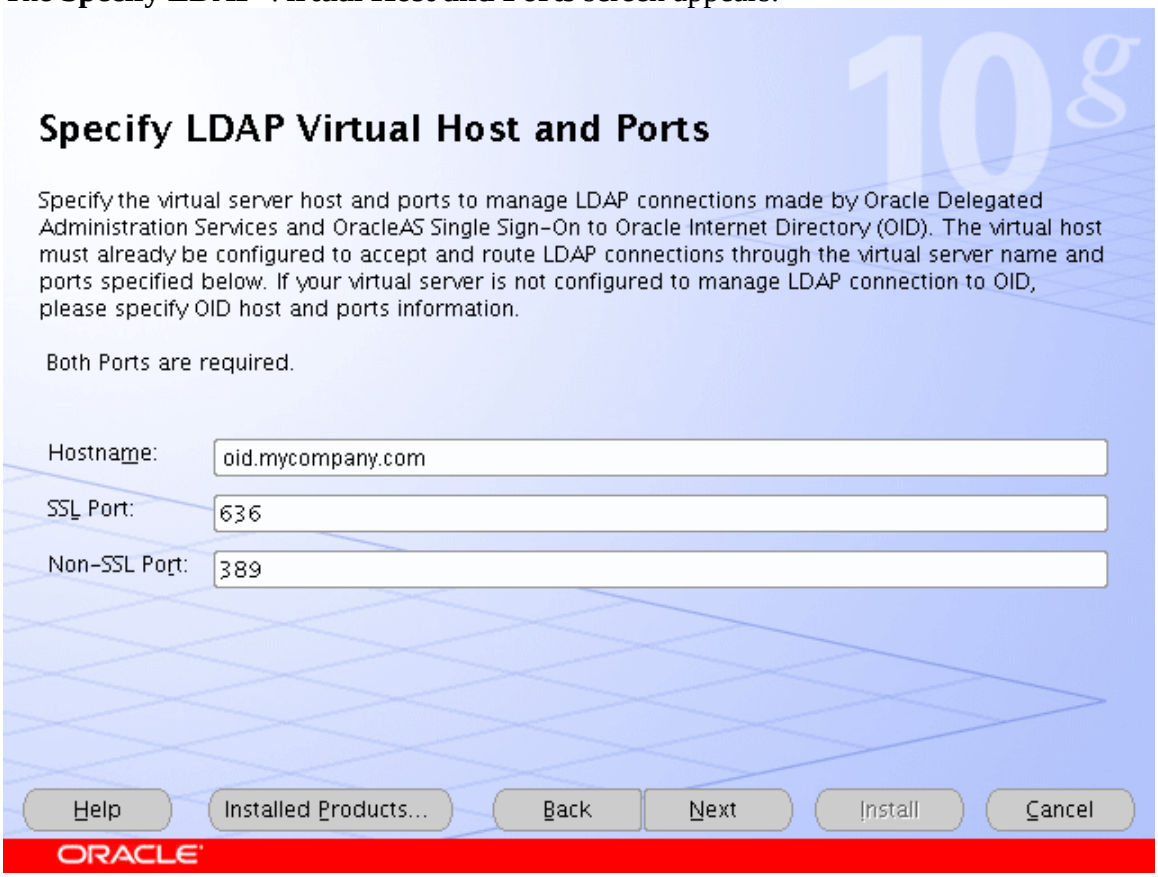


Select **Create a New OracleAS Cluster** and click **Next**.

14. The **Specify New OracleAS Cluster Name** screen appears:

Enter **imcluster** and click **Next**.

15. The **Specify LDAP Virtual Host and Ports** screen appears:

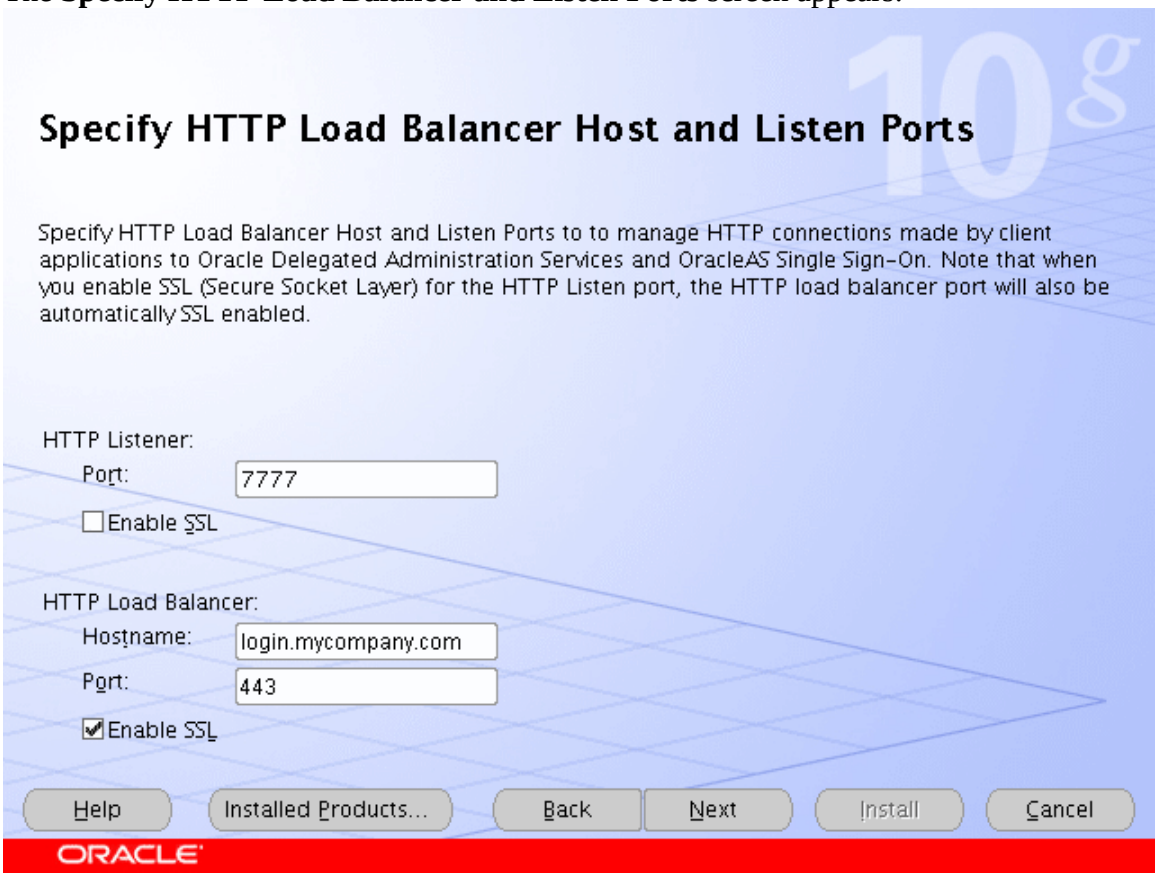


Enter the farm name of the OID servers (oid.mycompany.com) and the ports as shown. Click **Next**.

16. The **Specify OID Login** screen appears:

Enter the password for cn=orcladmin and click **Next**.

17. The **Specify HTTP Load Balancer and Listen Ports** screen appears:



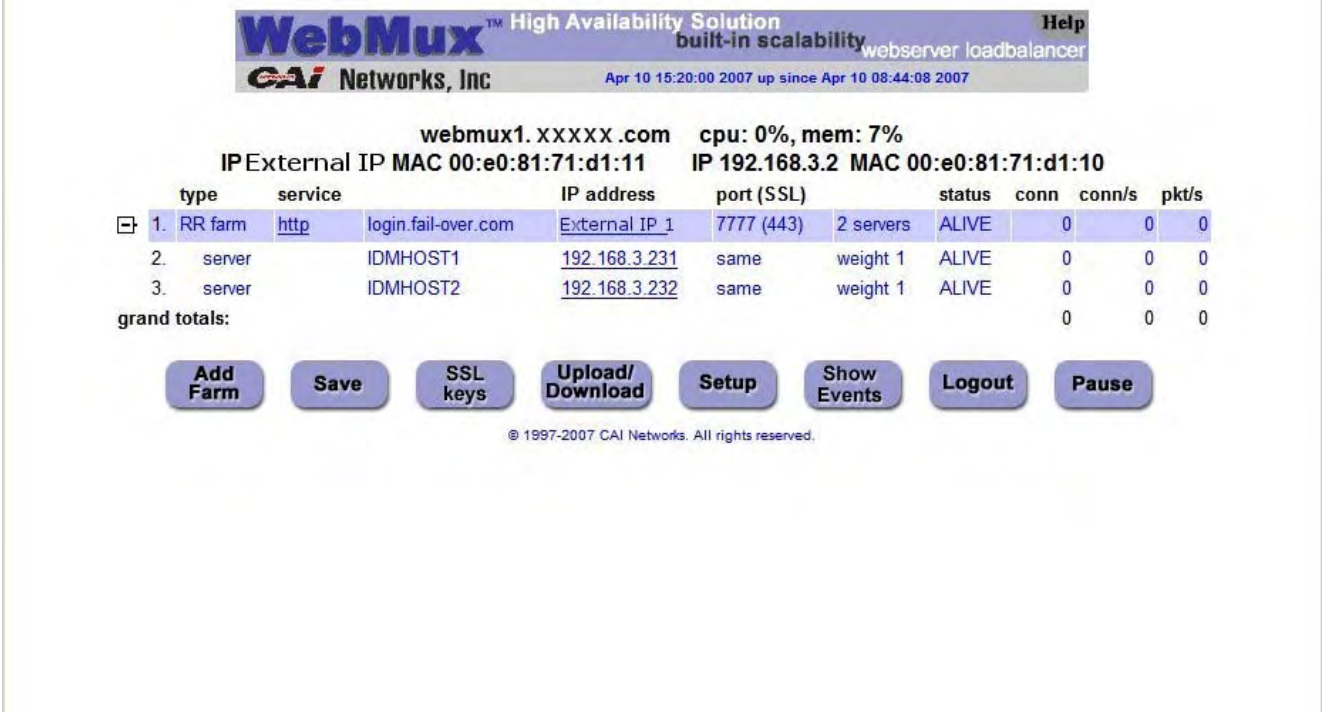
Enter the fields as shown. Click **Next**.

18. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **idmhost1** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
19. The **Summary** screen appears. Click **Install** to begin installation.
20. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
21. Click **Exit**, and confirm your choice to exit.

9. SETTING UP (login.mycompany.com) FARM ON WEBMUX1

1. Log in WebMux1 web admin as “superuser”
2. Click on the Add Farm button at the bottom of the status screen.
3. In the Add Farm screen enter the public IP you are using for login.mycompany.com. Optionally, you can enter login.mycompany.com in the Label field. In the port number field enter 7777. Select “HTTP – hypertext transfer protocol (TCP)”. Select Round Robin for the scheduling method. Select the certificate you want to you use in the SSL termination field. Ensure that the SSL port field shows 443. Select NO for “block non-SSL access” and NO for “tag-SSL terminated HTTP requests”. Click Confirm.
4. Back at the status screen click on the IP address of the newly created farm.
5. In the Modify Farm screen, click on Add Server.
6. In the IP address field enter 192.168.3.231 for IDMHOST1. Optionally, you can enter IDMHOST1 in the Label field. Click confirm.

7. Repeat step 4 and this time enter 192.168.3.232 for IDMHOST2. Optionally, you can enter IDMHOST2 in the Label field. Click confirm.
8. You should now have the login.mycompany.com farm with the two IDMHOSTs listed under them. (IDMHOST2 would be showing dead because the server is not yet up).
9. The port (SSL) column should be showing 7777 (443).



10. TESTING THE IDENTITY MANAGEMENT COMPONENTS WITH ORACLE® INTERNET DIRECTORY

1. Stop all components on OIDHOST1, using this command:

   C:\OraHome_1\opmn\bin\opmnctl stopall

2. Ensure that all components on OIDHOST2 are running:

   C:\OraHome_1\opmn\bin\opmnctl status

3. Access the following URLs:

   https://login.mycompany.com/pls/orasso

   https://login.mycompany.com/oiddas

11. SETTING UP IDMHOST2 (Identity Management)

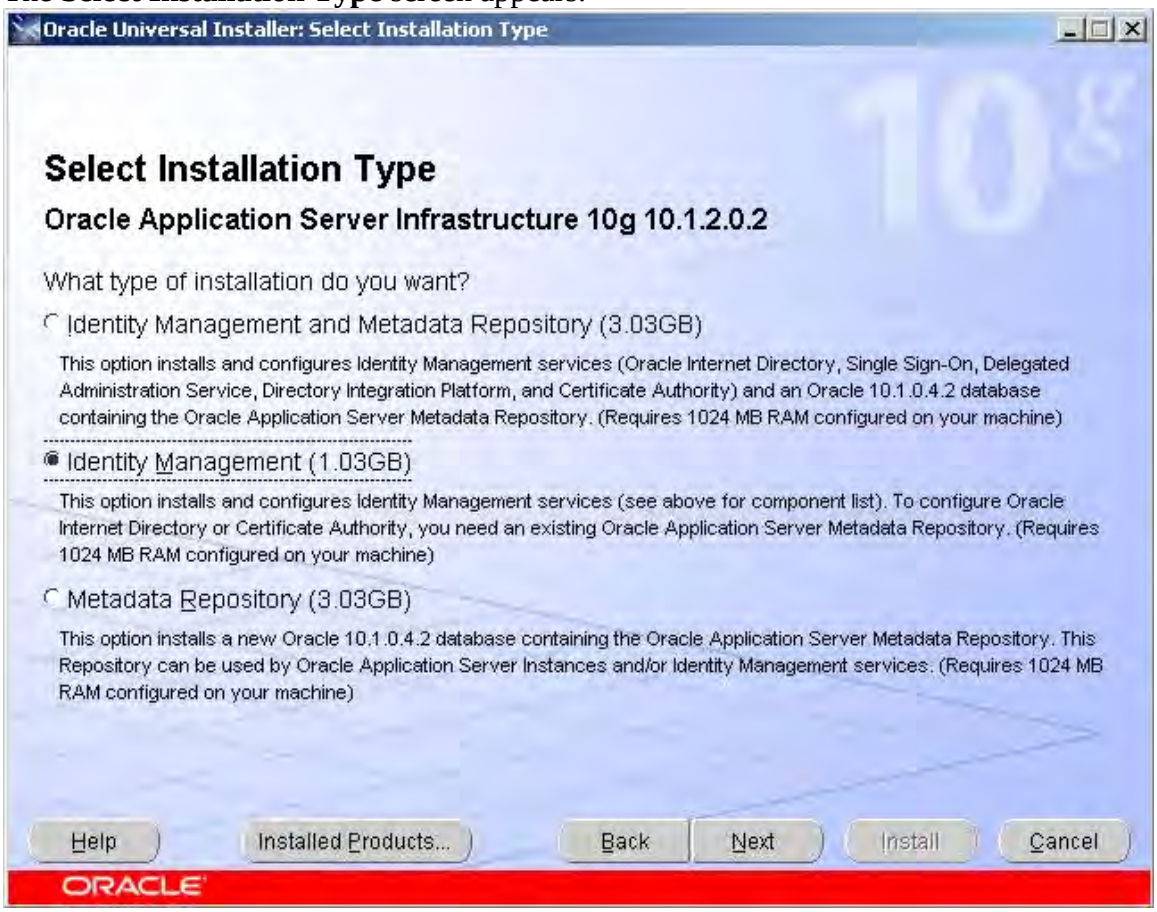
1. Start the Oracle® Universal Installer by double clicking on *setup.exe*
2. The **Welcome** screen appears. Click **Next**.
3. The Specify File Locations screen appears with default locations for:
   - The product files for the installation (Source)
   - The name and path to an Oracle® home (Destination)
     C:\OraHome_1 will be the default destinations on all the servers we will be setting up.
4. Click **Next**.
5. At this point, go to the *Disk1/stage/Response* directory of the installation package and copy the staticport.ini file and paste in the *C:\OraHome_1* directory.
6. In the *C:\OraHome_1* directory edit the *staticport.ini* file with the following values:

```
Oracle HTTP Server port = 7777
Oracle HTTP Server Listen port = 7777
Application Server Control port = 1810
```

7. The **Select a Product to Install** screen appears:

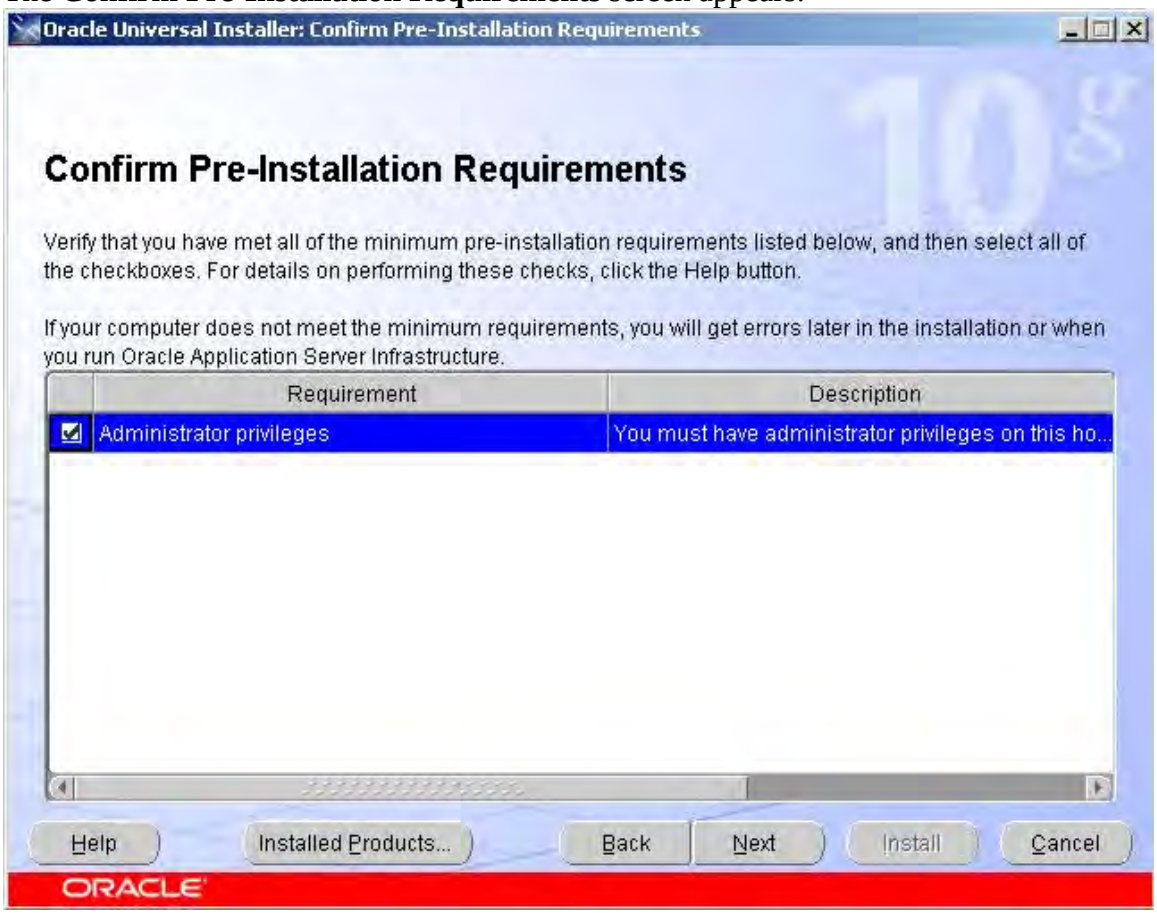
Select **Oracle Application Server Infrastructure 10g**. Click **Next**.

8. The **Select Installation Type** screen appears:



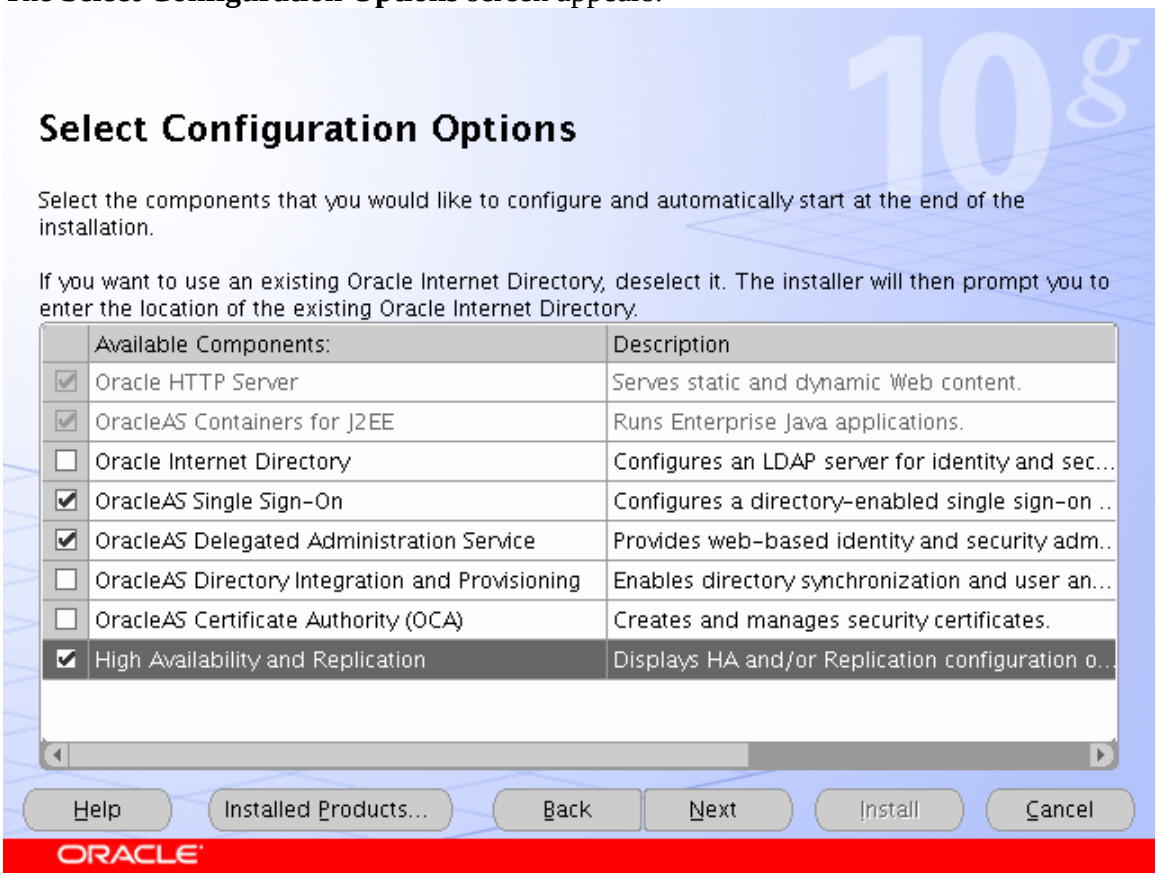
Select **Identity Management** and click **Next**.

9. The **Confirm Pre-Installation Requirements** screen appears:



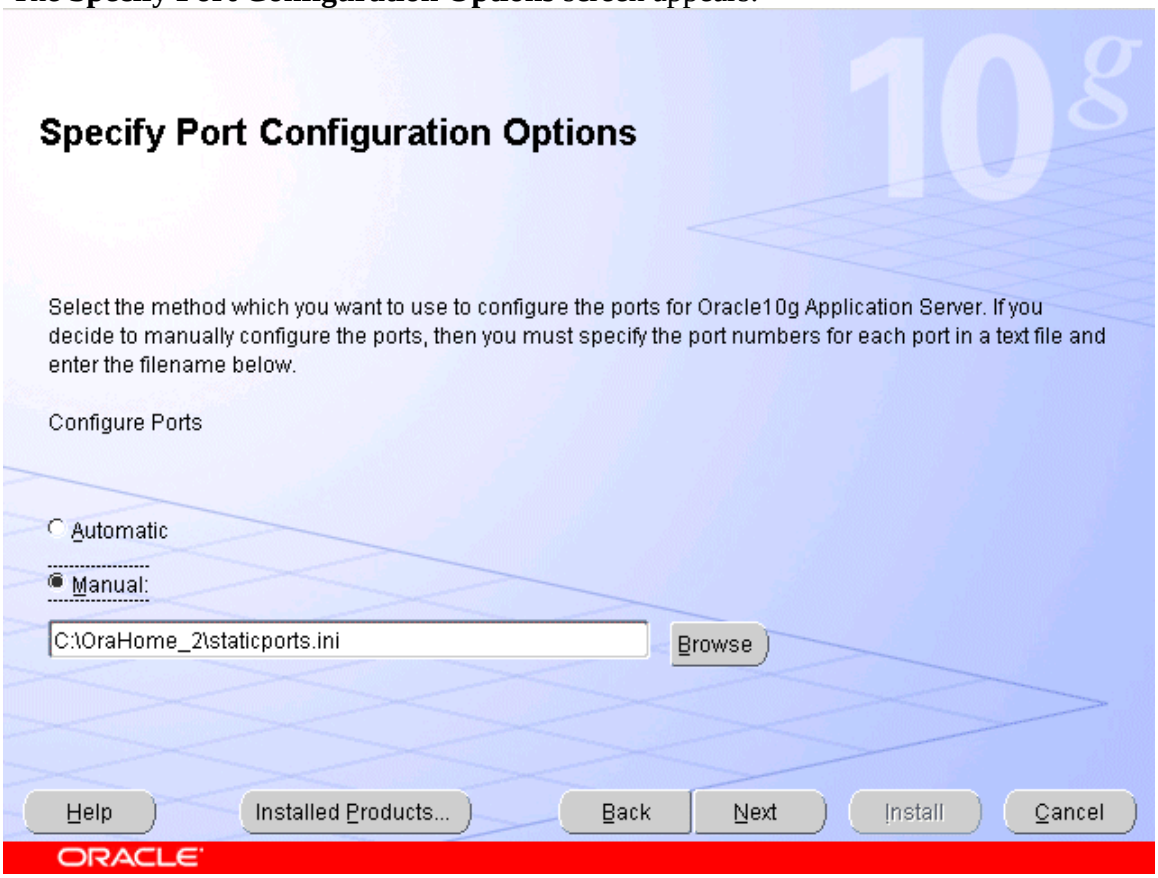
Confirm that you have Administrator privileges and click **Next**.

10. The **Select Configuration Options** screen appears:



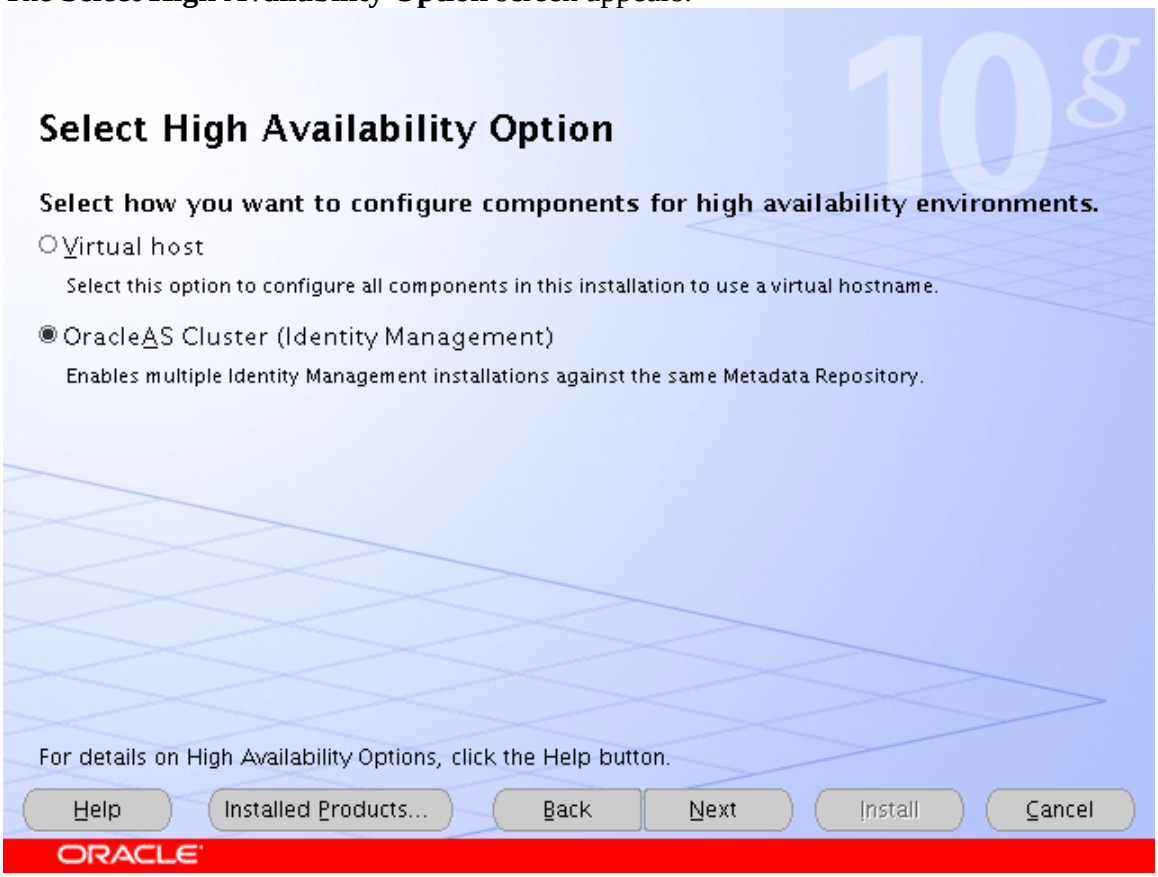
Select **OracleAS Single Sign-On, Oracle Delegated Administration Services**, and **High Availability and Replication**. Click **Next**.

11. The **Specify Port Configuration Options** screen appears:



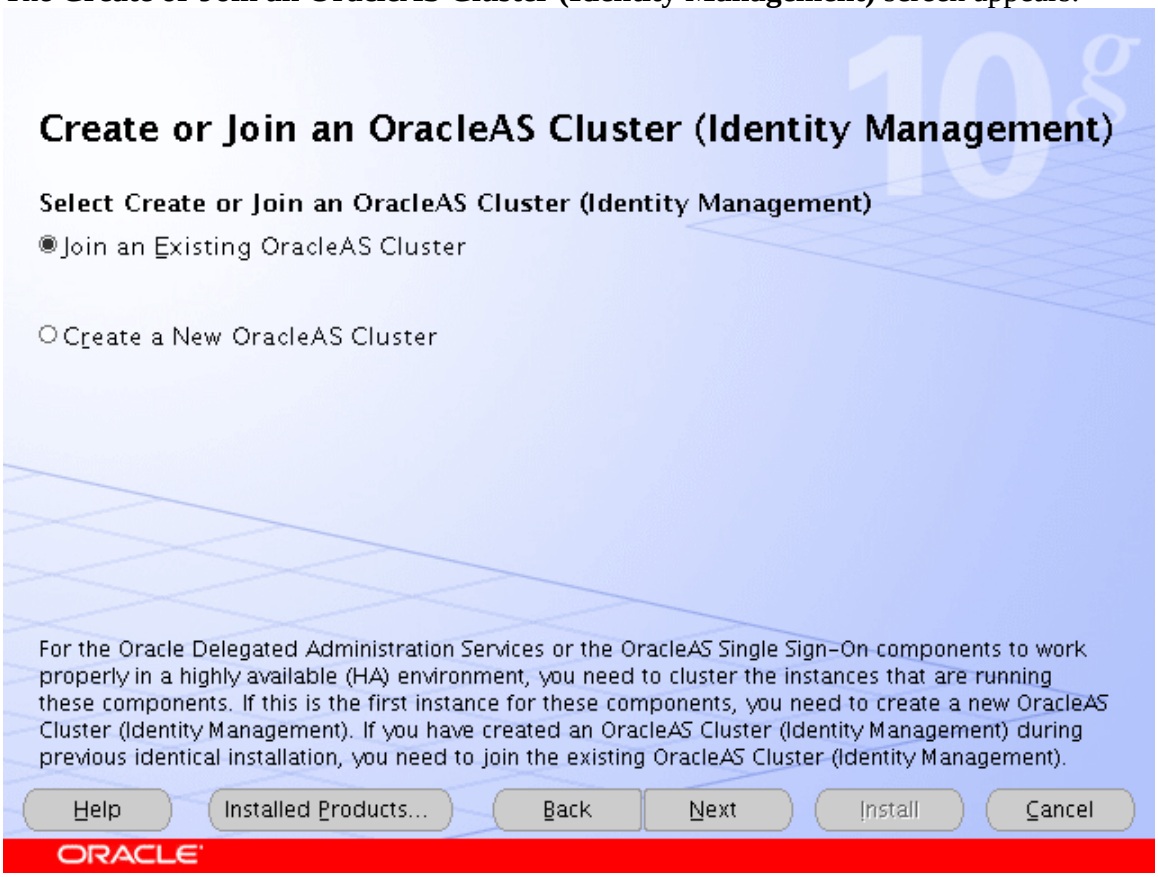
Select manual and select the location of the edited staticports.ini file which should be in *C:\OraHome_1* (not correctly shown in the image). Click **Next**.

12. The **Select High Availability Option** screen appears:



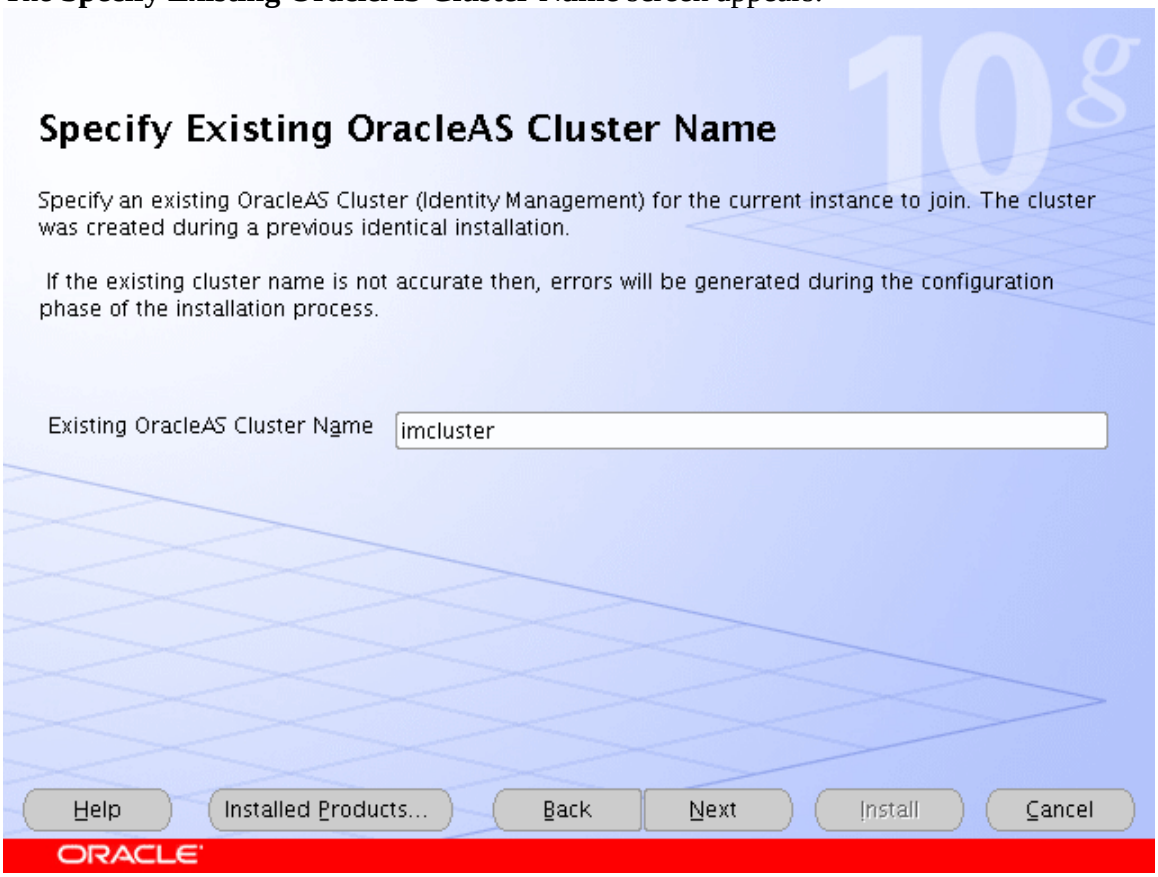
Select **OracleAS Cluster (Identity Management)** and click **Next**.

13. The **Create or Join an OracleAS Cluster (Identity Management)** screen appears:



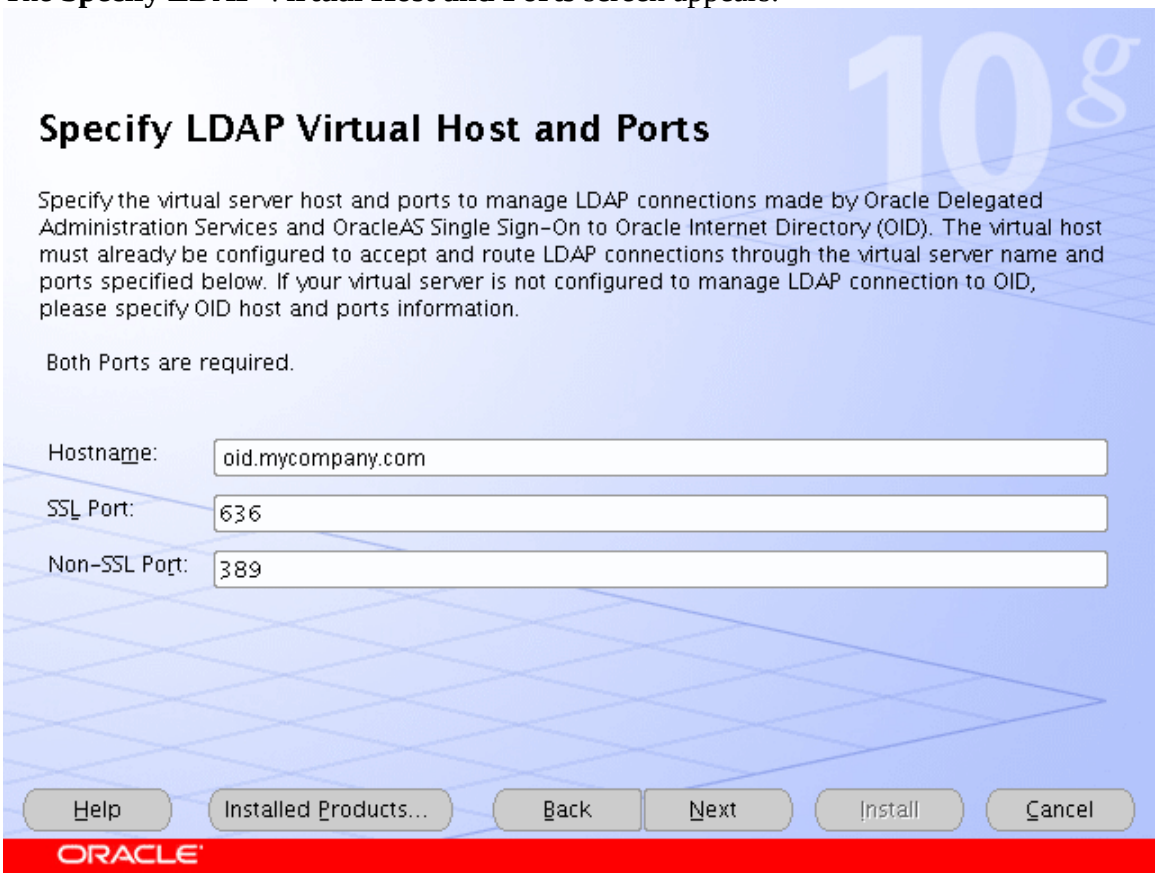
Select **Join an Existing OracleAS Cluster** and click **Next**.

14. The **Specify Existing OracleAS Cluster Name** screen appears:



Enter **imcluster**. Click **Next**.

15. The **Specify LDAP Virtual Host and Ports** screen appears:

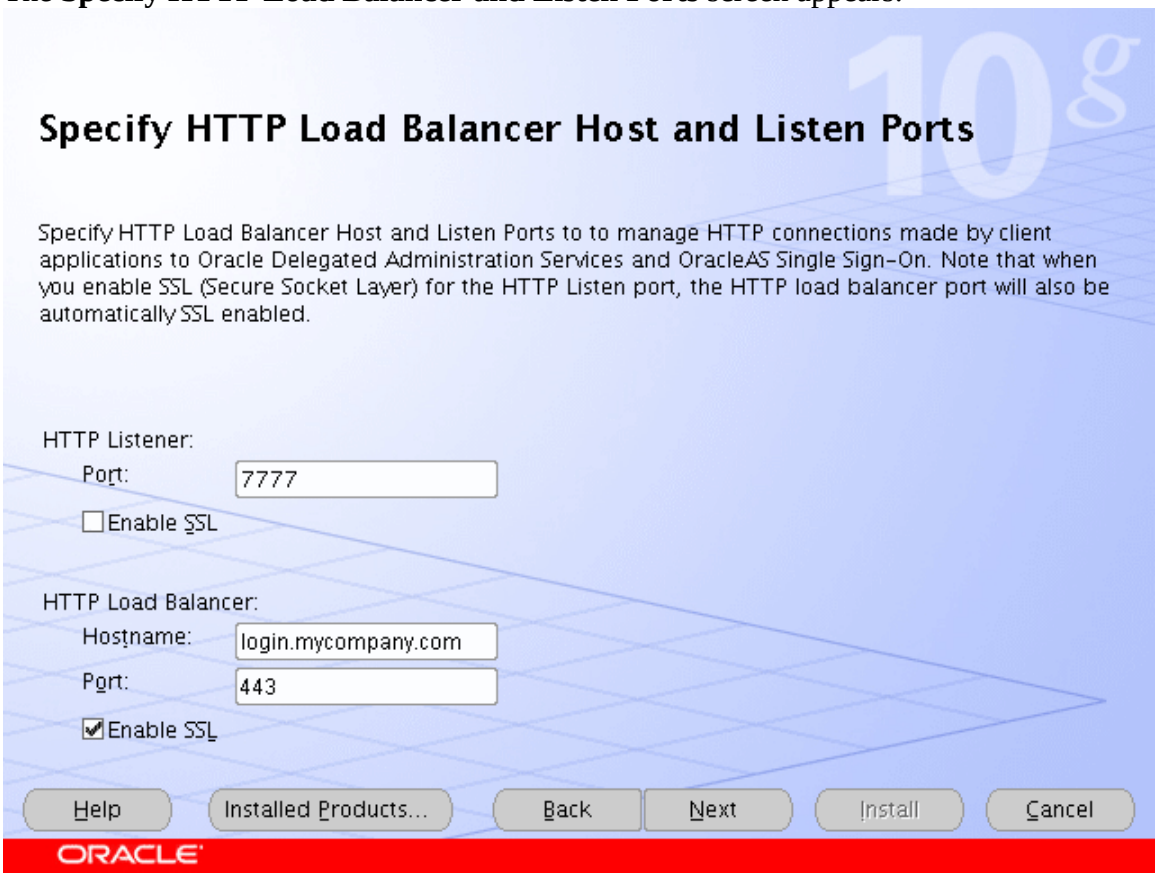


Enter the farm name of the OID servers (oid.mycompany.com) and the ports as shown. Click **Next**.

16. The **Specify OID Login** screen appears:

Enter the password for cn=orcladmin and click **Next**.

17. The **Specify HTTP Load Balancer and Listen Ports** screen appears:



Enter the fields as shown. Click **Next**.

18. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **idmhost2** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
19. The **Summary** screen appears. Click **Install** to begin installation.
20. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
21. Click **Exit**, and confirm your choice to exit.

12. TESTING THE IDENTITY MANAGEMENT COMPONENTS

1. Stop all components on IDMHOST1, using this command:

   C:\OraHome_1\opmn\bin\opmnctl stopall

2. Ensure that all components on IDMHOST2 are running, using this command:

   C:\OraHome_1\opmn\bin\opmnctl status

3. Access the following URLs from two browsers:

   https://login.mycompany.com/pls/orasso

https://login.mycompany.com/oiddas

4. Start all components from IDMHOST1, using this command:

   C:\ORAHOME_1\opmn\bin\opmnctl startall

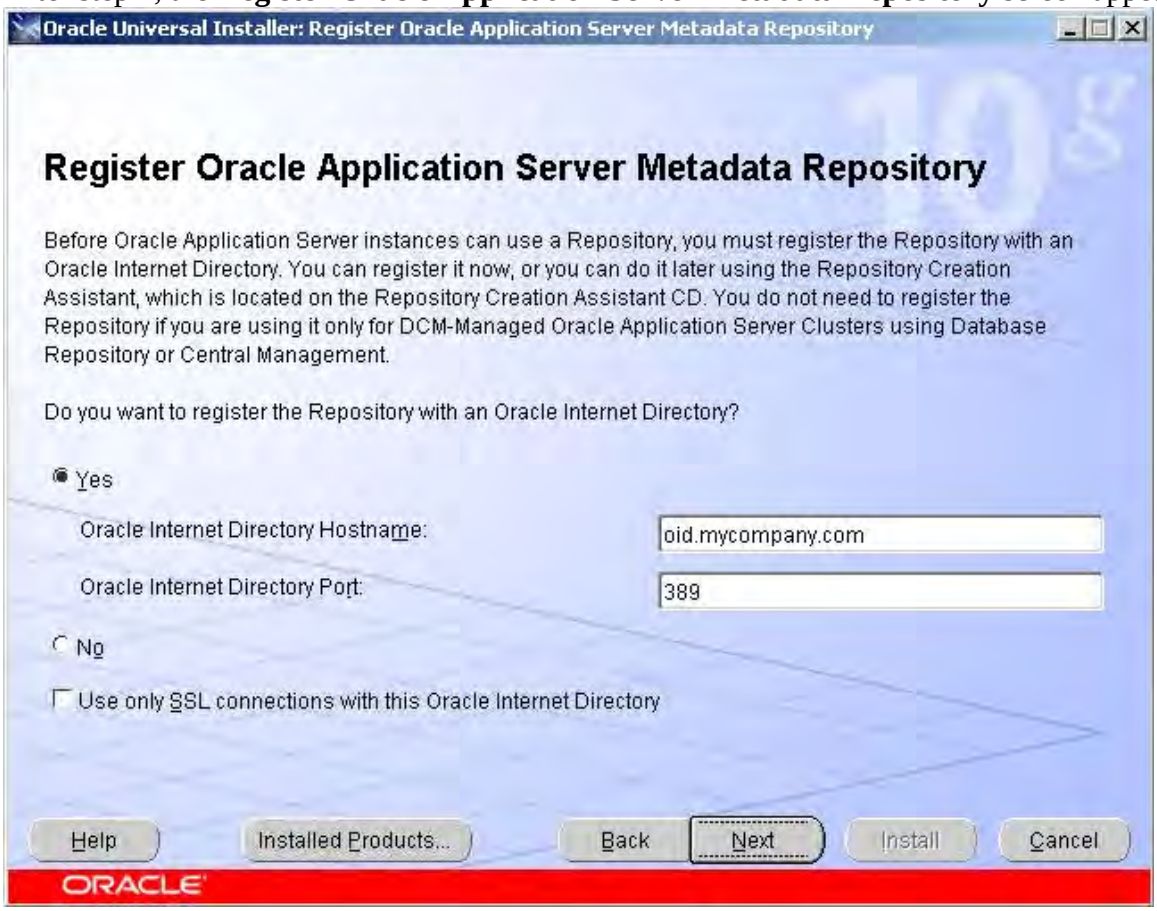
5. Stop all components on IDMHOST2, using this command:

   C:\ORAHOME_1\opmn\bin\opmnctl stopall

6. Ensure that the login session is still valid for the orasso and oiddas logins.

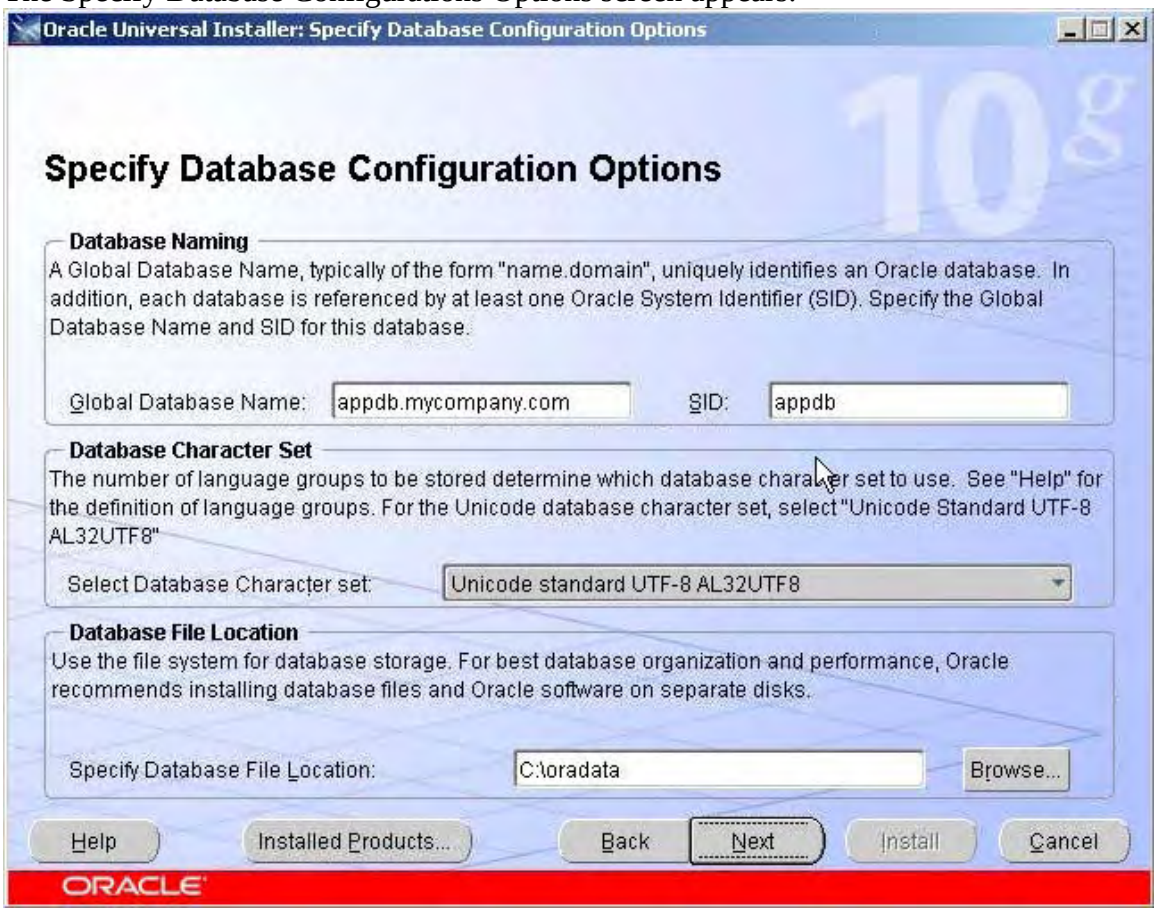
13. SETTING UP APPDBHOST1 (Application Metadata Repository)

1. Follow steps 1 through 7 in Section 3 (Setting up Infradbhost1).
2. After step 7, the **Register Oracle Application Server Metadata Repository** screen appears:



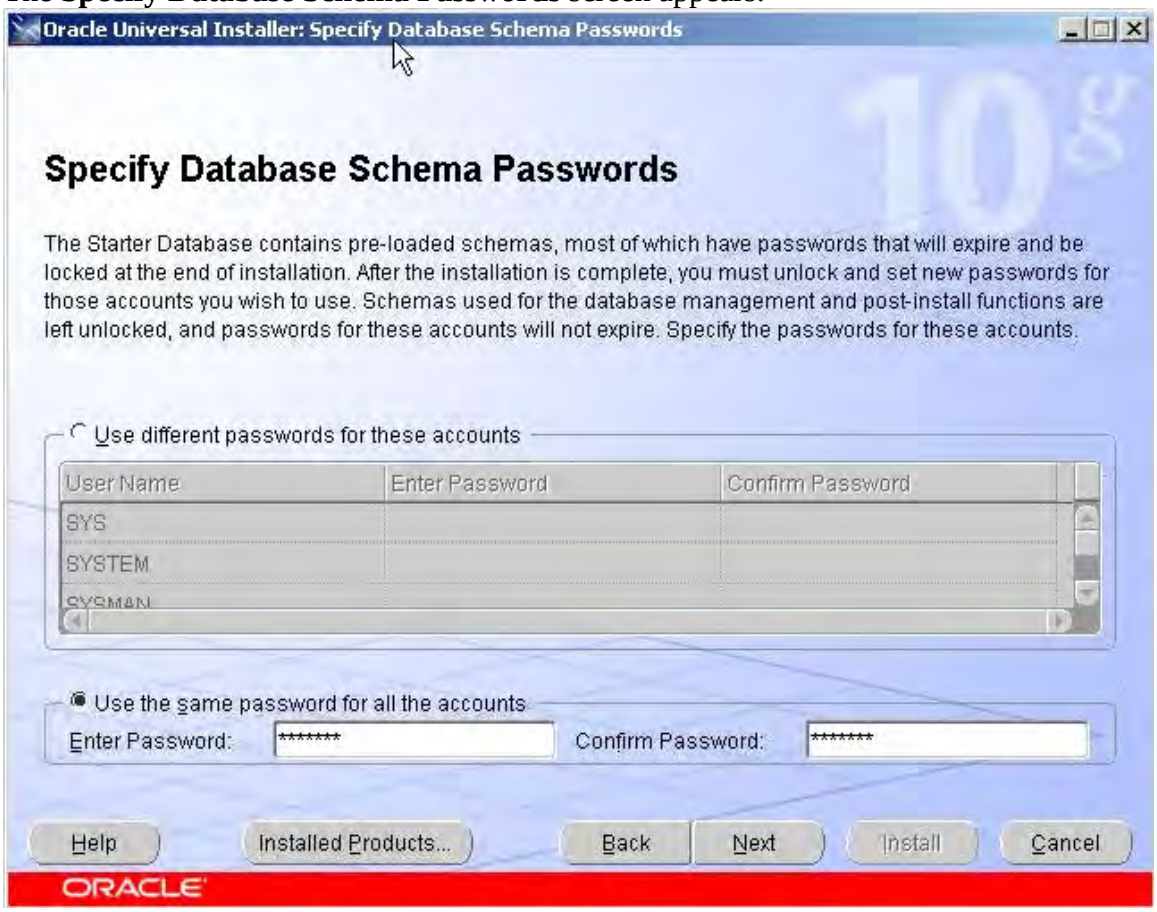
This time you will be registering the installation in the Oracle Internet Directory. Select **Yes**. Enter the OID farm name (oid.mycompany.com) and port 389. Click **Next**.

3. The Specify Database Configurations Options screen appears:



Fill out the appropriate Global Database Name. The SID will be automatically filled out. Click **Next**.

4. The **Specify Database Schema Passwords** screen appears:



   You can chose different passwords for each account or use the same password for all accounts. Click **Next**. (We chose the same password for all the accounts in our test setup).
5. The **Summary** screen appears. Click **Install** to begin installation.
6. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
7. Click **Exit**, and confirm your choice to exit.

14. SETTING UP APPHOST1 (Application Server, Portal, Web Cache)

1. Start the Oracle® Universal Installer by double clicking on *setup.exe*
2. The **Welcome** screen appears. Click **Next**.
3. The Specify File Locations screen appears with default locations for:
   - The product files for the installation (Source)
   - The name and path to an Oracle® home (Destination)
     C:\OraHome_1 will be the default destinations on all the servers we will be setting up.
4. Click **Next**.
5. At this point, go to the *Disk1\stage\Response* directory of the installation package and copy the staticport.ini file and paste in the *C:\OraHome_1* directory.
6. In the *C:\OraHome_1* directory edit the *staticport.ini* file with the following values:

```
Oracle HTTP Server port = 7777
Oracle HTTP Server Listen port = 7778
Web Cache HTTP Listen port = 7777
Web Cache Administration port = 9400
Web Cache Invalidation port = 9401
Web Cache Statistics port = 9402
Application Server Control port = 1810
```

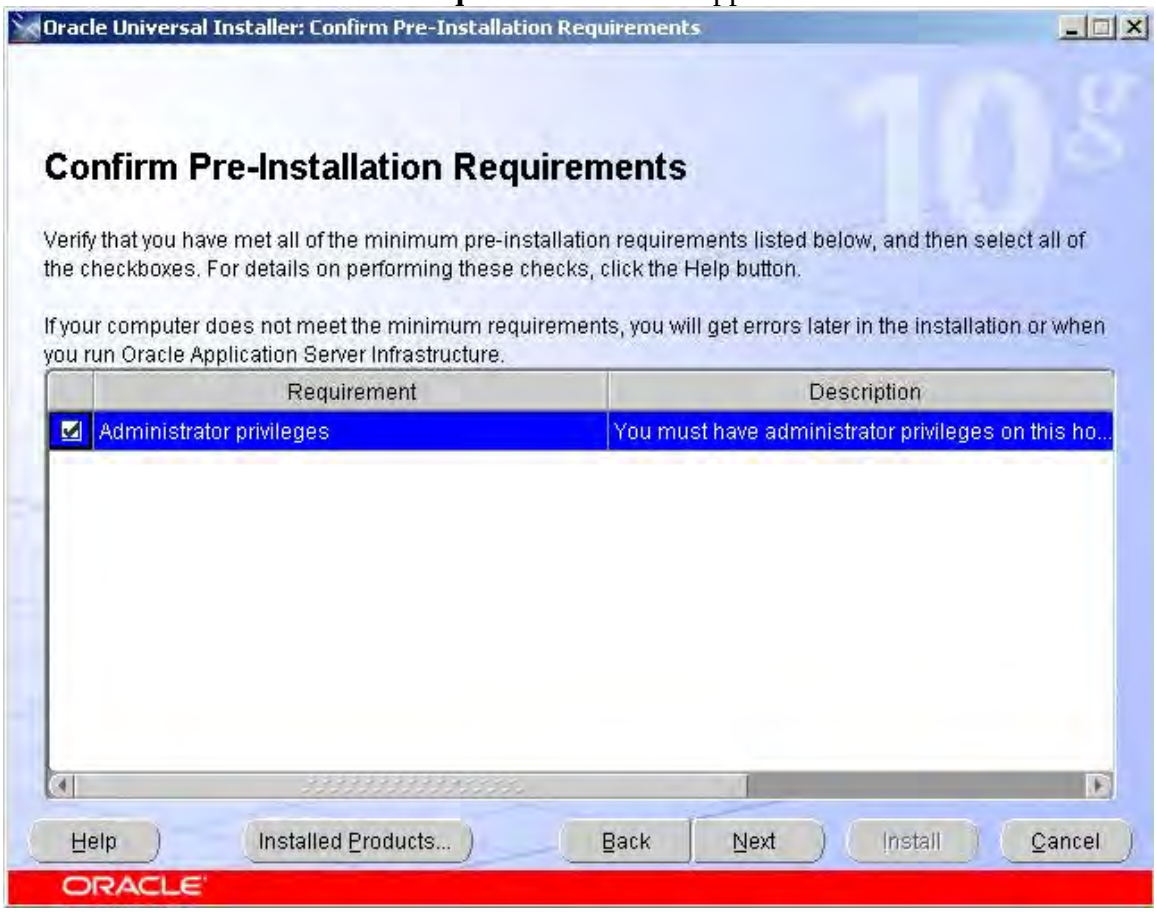
7. The **Select a Product to Install** screen appears:

Select **Oracle Application Server 10g** and click **Next**.

8. The **Select Installation Type** screen appears:

Select **Portal and Wireless**. Click **Next**.

9. The **Confirm Pre-Installation Requirements** screen appears:

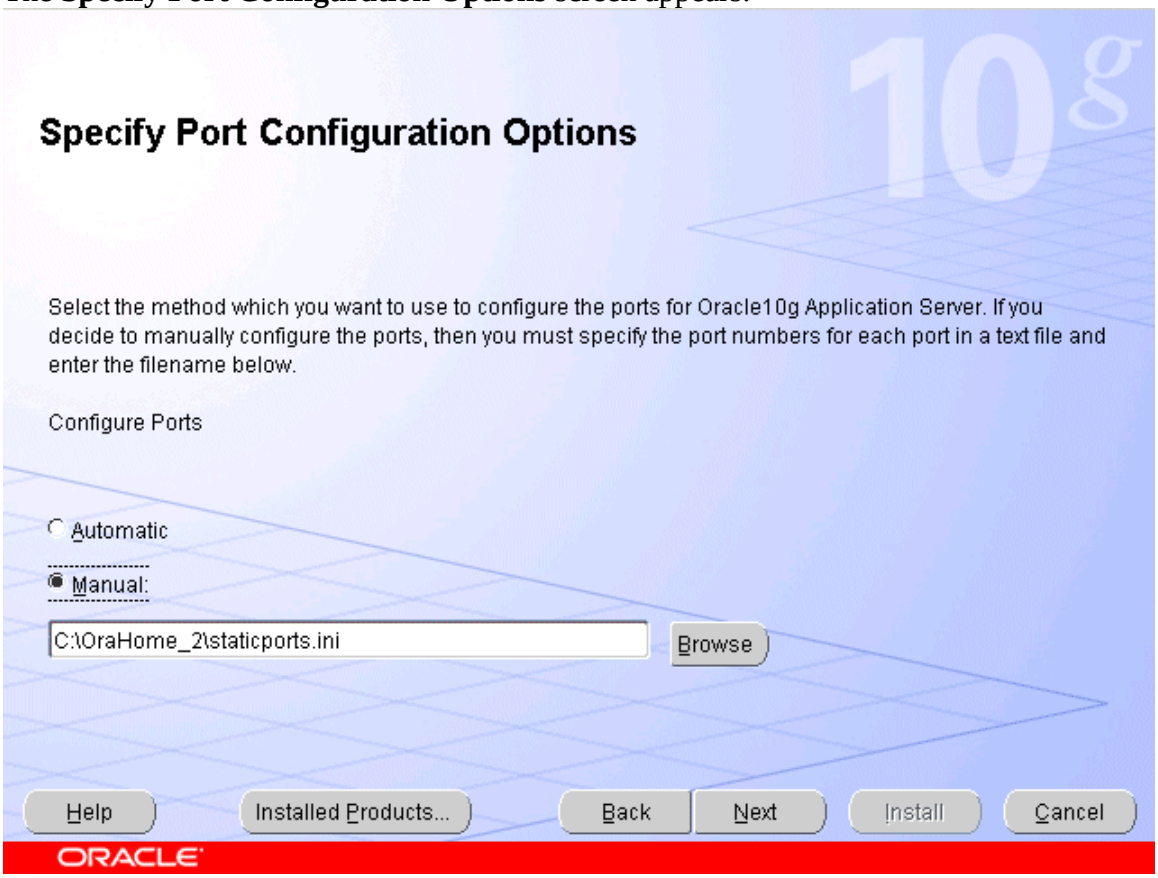


Confirm that you have Administrator privileges and click **Next**.

10. The **Select Configuration Options** screen appears:

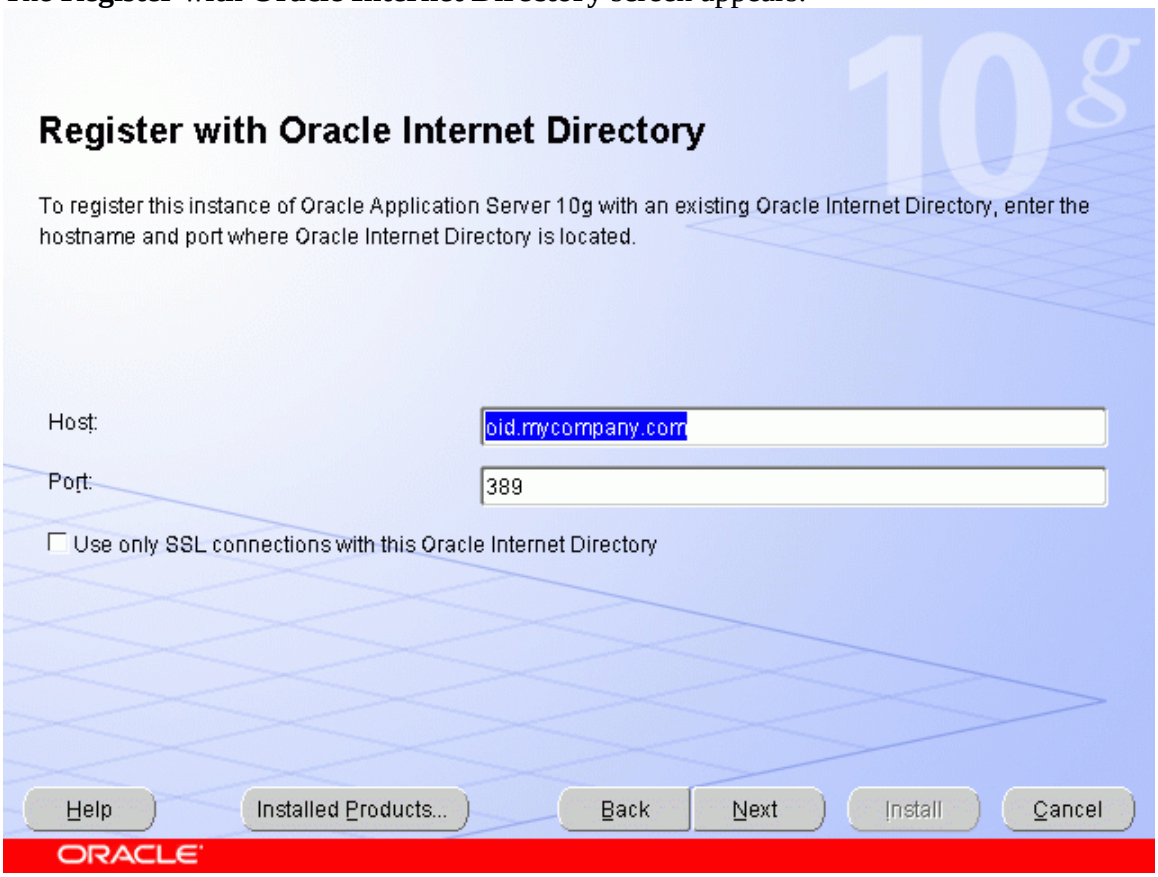
Select **Oracle Application Server 10g Portal**. Click **Next**.

11. The **Specify Port Configuration Options** screen appears:



Select manual and select the location of the edited staticports.ini file which should be in *C:\OraHome_1*. Click **Next**.

12. The **Register with Oracle Internet Directory** screen appears:

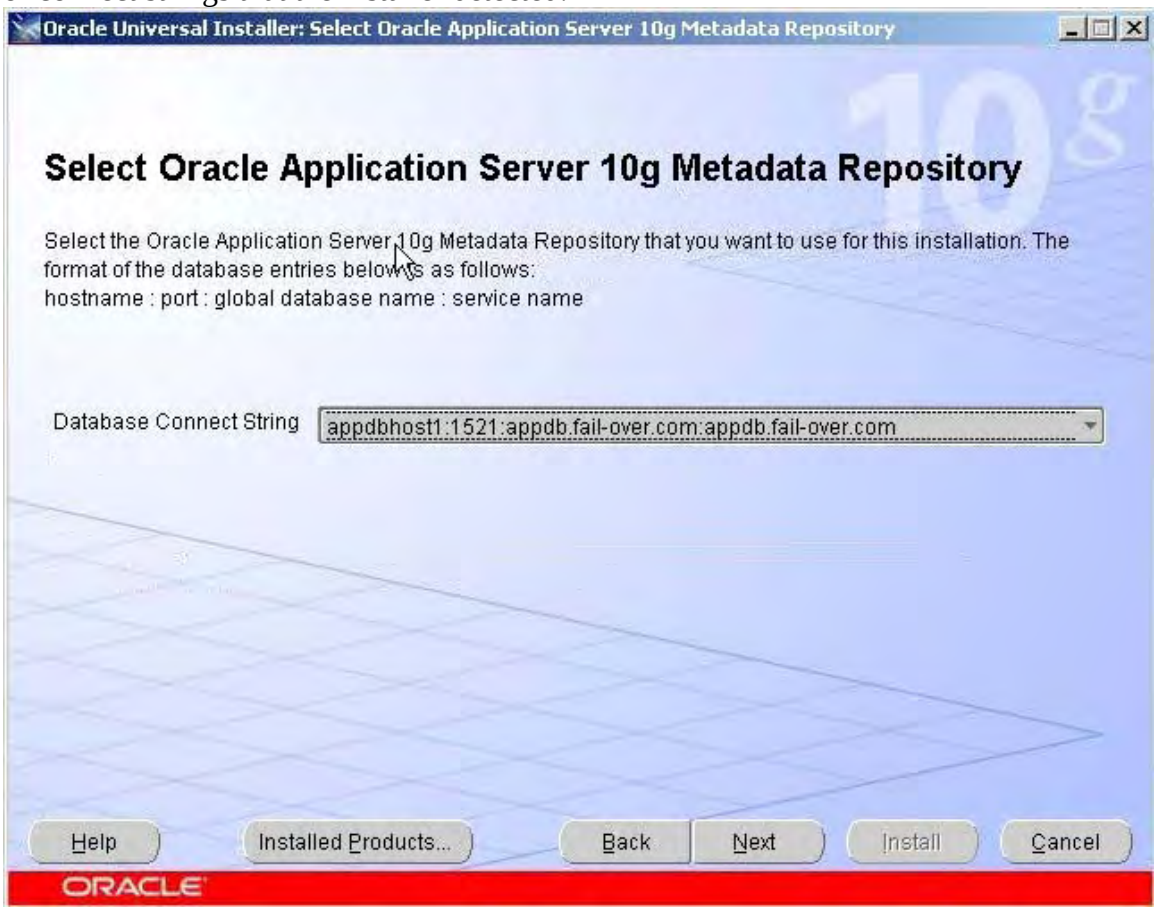


Enter the OID farm name and port 389. Click **Next**.

13. The **Specify OID Login** screen appears:

Enter the password for cn=orcladmin and click **Next**.

14. The **Select OracleAS 10*g* Metadata Repository** screen appears, displaying a drop-down list of connect strings that the installer detected:



Select the connect string for the application Metadata Repository database (on APPDBHOST1) and click **Next**.

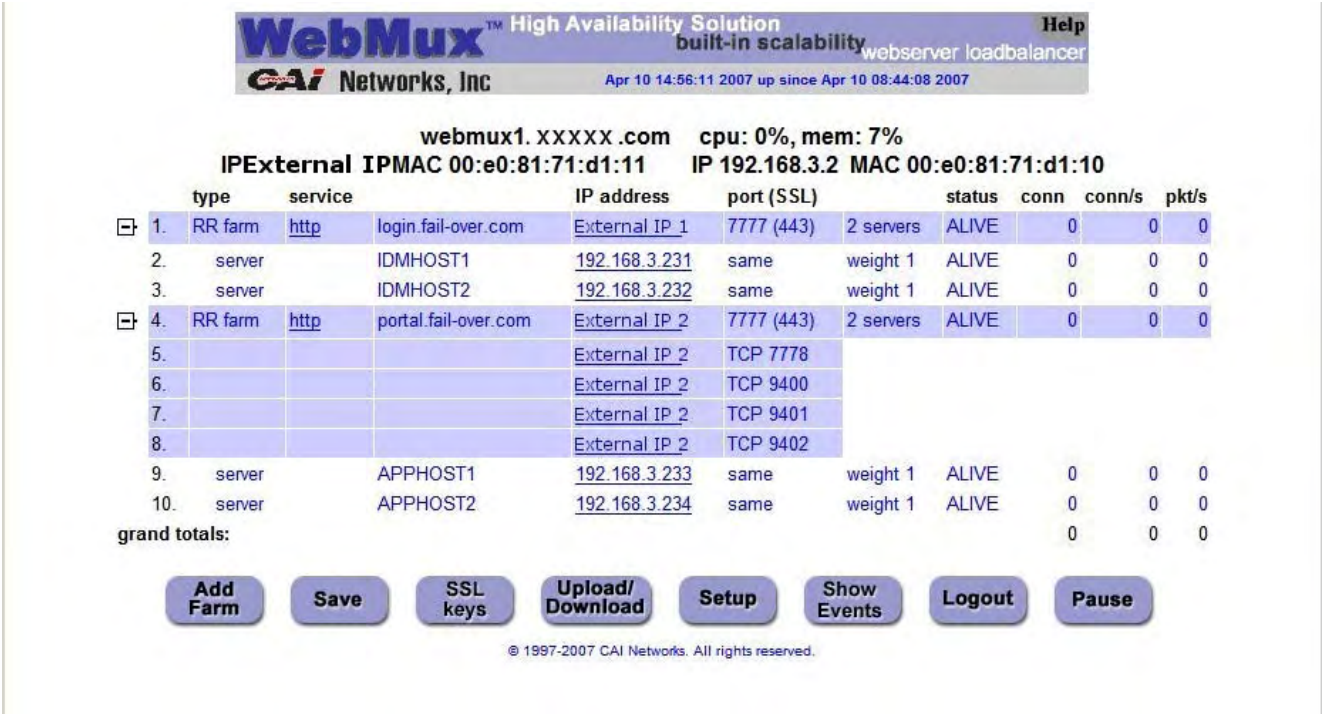
15. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **apphost1** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
16. The **Summary** screen appears. Click **Install** to begin installation.
17. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
18. Click **Exit**, and confirm your choice to exit.
19. Verify that the installation was successful by accessing the OracleAS Portal page at:

    http://apphost1.mycompany.com:7777/pls/portal

15. SETTING UP (portal.mycompany.com) FARM ON WEBMUX1

1. Log in to the WebMux1 web interface as “superuser”.
2. Click on the Add Farm button at the bottom of the status screen.
3. Enter the public IP address you assigned for the portal.mycompany.com host. Optionally, you can enter portal.mycompany.com in the Label field.
4. Enter 7777 in the port number field.
5. Select “HTTP – hypertext transfer protocol (TCP)” for service.

6. Select “Round Robin” for the scheduling method.
7. Select your imported certificate in the SSL termination field.
8. Verify that SSL port shows 443.
9. Select NO for Block non-SSL access to farm.
10. Select no for tag SSL-terminated HTTP requests.
11. Click Confirm.
12. Back at the status screen, click on the farm IP for portal.mycompany.com.
13. Click Add Server.
14. Enter 192.168.3.233 in the IP field.
15. Optionally, you can enter APPHOST1 in the Label Field.
16. Click Confirm.
17. Repeat step 12 to 16 for APPHOST2.
18. Back at the status screen, click on the farm IP for portal.mycompany.com.
19. Click Add Addr.Port.
20. In the IP field enter the same IP for portal.mycompany.com.
21. Enter 7778 in the port number field.
22. You can leave the rest of the fields with their default settings
23. Click confirm.
24. Repeat steps 18 through 23 for ports 9400, 9401, and 9402.



16. Executing the SSL Configuration Tool on APPHOST1

Follow these steps to use the SSL Configuration Tool to configure SSL on APPHOST1:

1. Set the ORACLE_HOME environment variable to the Oracle home in which OracleAS Portal resides.

2. Verify that the Oracle Internet Directory server is running by issuing this command in *ORACLE_HOME*\bin:

   ldapbind -h oid.mycompany.com

3. Create a file, *ORACLE_HOME*\configMyPortal.xml file to include the following:

   ```
   <sslconfig>
     <mid_tier>
       <virtual_address ssl="on" host="portal.mycompany.com" port="443" inv_port="9401"
   ssl_terminate="lbr"/>
         <lbr loopback_port="7777"/>
     </mid_tier>
   </sslconfig>
   ```

4. Issue this command in *ORACLE_HOME*/bin:

   SSLConfigTools -config_w_file *ORACLE_HOME*\configMyPortal.xml -opwd *orcladmin password* -ptl_inv_pwd *webcache invalidation password*

   In the preceding command, *orcladmin password* is the EXISTING Oracle administrator password, and *webcache invalidation password* is the “ias_admin” password by default.

5. Log in to the OracleAS Single Sign-On Administration page as the Administrator, and use the Administer Partner Applications page to delete the entry for the partner application apphost1.apphost1:7777.

6. Configure the OmniPortlet and Web Clipping Provider registration URLs to go through the HTTP port of the Load Balancing Router:

   a. Access the OracleAS Portal page at https://portal.mycompany.com/pls/portal and log in as the portal administrator.
   b. Click the Navigator link.
   c. Click the Providers tab.
   d. Click the Registered Providers link.
   e. Click the Edit Registration link.
   f. Click the Connection tab and change the beginning of the provider registration URL from https://portal.mycompany.com/ to http://portal.mycompany.com:7777/.
   g. Perform steps e and f for the Web Clipping Provider.

7. To prevent access to Oracle Enterprise Manager 10g from the outside, the link provided by OracleAS Portal must be changed back to point to the internal server. To do this, on APPHOST1, issue the following command in ORACLE_HOME/portal/conf:

   ptlconfig -dad portal -em

8. Configure OmniPortlet to use a shared preference store. (By default, the OmniPortlet provider uses the file-based preference store. However, in a multiple middle tier environment, you must use a shared preference store, such as the database preference store DBPreferenceStore.) To configure OmniPortlet to use DBPreferenceStore, perform the following steps:

a. Navigate to the directory:

ORACLE_HOME\j2ee\OC4J_Portal\applications\jpdk\jpdk\doc\dbPreferenceStore.

b. Create a user on the database containing the PORTAL schema, and grant create resource and connect privileges, using the create user and grant connect commands in SQL*Plus.

You will be using the *sqlplus* command to access the appdb database. Open a command prompt and type:

sqlplus sys*/password****@appdb as sysdba***

"password" being the password that you set up for the sys user account on appdb.

Once connected, create the prefstore user and substitute an actual password you want to use in the following command. Do not use the default password of welcome, as this poses a security risk.

create user prefstore identified by password;

grant connect, resource to prefstore;

c. Connect as user prefstore (*sqlplus prefstore/password@appdb*) and execute the jpdk_preference_store2.sql script by issuing this command:

@jpdk_preference_store2

d. Edit the ORACLE_HOME\j2ee\OC4J_Portal\config\data-sources.xml file to add the entry in the subsequent example:

```
<data-source
        class="com.evermind.sql.DriverManagerDataSource"
        name="omniPortletprefStore"
        location="jdbc/UnPooledConnection"
        xa-location="jdbc/xa/XAConnection"
        ejb-location="jdbc/PooledConnection"
        connection-driver="oracle.jdbc.driver.OracleDriver"
        username="prefstore"
        password="password"
        url="jdbc:oracle:thin:@(description=(address_list=
        (address=(host=appdbhost1.mycompany.com)(protocol=tcp)(port=1521))
        (address=(host=appdbhost2.mycompany.com)(protocol=tcp)(port=1521))
        (load_balance=yes)(failover=yes))(connect_data=(service_name= db9i)))"
        inactivity-timeout="30"
/>
```

Substitute "password" with the password you assigned to the prefstore user. The service name is the SID you assigned when you set up the APPDBHOST1 metadata repository.

In the setup specific to this documentation, there is only a single database node. So, in the "url" section, you will have:

url="jdbc:oracle:thin:@(description=(address_list=
(address=(host=appdbhost1.mycompany.com)(protocol=tcp)(port=1521))
(load_balance=no)(failover=no))(connect_data=(service_name= appdb))"
inactivity-timeout="30"

e. Edit the ORACLE_HOME\j2ee\OC4J_Portal\applications\portalTools\omniPortlet\WEB-INF\providers\omniPortlet\provider.xml file to edit the preferenceStore tag as shown in the subsequent example:

```
<provider class="oracle.webdb.reformlet.ReformletProvider">
    <vaultId>0</vaultId>
    <session>true</session>
    <preferenceStore class="oracle.portal.provider.v2.preference.DBPreferenceStore">
        <name>omniPortletprefStore</name>
        <connection>jdbc/PooledConnection</connection>
    </preferenceStore>
```

f. Restart the OC4J_Portal instance.

1. Verify that OmniPortlet and the Web Clipping Provider work properly through the HTTP port of the Load Balancing Router, by accessing the test pages at the following URLs:

   OmniPortlet Provider:

   http://portal.mycompany.com:7777/portalTools/omniPortlet/providers/omniPortlet

   Note:
   If the "No Portlets Available" message appears under the **Portlet** Information section in the **OmniPortlet Provider** test page, then the provider may not be configured correctly. Review Step 1 to ensure correct configuration. The **Portlet Information** section should list the following:
   OmniPortlet
   Simple Parameter Form

   Web Clipping Provider:

   http://portal.mycompany.com:7777/portalTools/webClipping/providers/webClipping

   Note:
   If, while accessing the test pages, you are prompted to examine the site's certificate, accept the certificate.

17. RE-REGISTERING mod_osso ON APPHOST1

1. Access the following URL:

https://portal.mycompany.com/pls/portal

2. Refresh the Portlet Repository so that the Portal Tools portlets appear in the Portlet Builders folder in the Portlet Repository:

    a. Log in as the portal administrator (**orcladmin** or **portal_admin**), and click the **Builder** link.
    b. Click the **Administrator** tab.
    c. Click the **Portlets** sub-tab.
    d. Click the **Refresh Portlet Repository** link in the Portlet Repository portlet.
    e. The refresh operation continues in the background.

18. Verifying Connectivity for Invalidation Messages from the Database to the OracleAS® Web Cache on APPHOST1 through the Load Balancing Router

When a cached OracleAS® Portal object is modified, the OracleAS® Portal metadata repository database sends an invalidation message to OracleAS® Web Cache to invalidate that object. Since the target configuration has two instances of OracleAS® Web Cache, the invalidation message must be load balanced across both OracleAS® Web Cache instances. This is an example of component level load balancing.

Before you proceed with this verification, ensure that messages can be sent from the computer hosting the database to the Load Balancing Router. To do this, issue the following command from APPDBHOST1:

telnet portal.mycompany.com 9401

Verify that no connection failure message is returned.

If you are not able to get a connection, check that you have entered the correct routing rules in WebMux1 and in APPHOST1 (See Section 2).

19. TESTING THE CONFIGURATION ON APPHOST1

1. Perform the following tests:
    a. Access OracleAS® Web Cache and Oracle® HTTP Server through the Load Balancing Router with following URL:

        https://portal.mycompany.com

    b. Test the connection to the Oracle Application Server® Metadata Repository through the Load Balancing Router, by accessing the following URL:

        https://portal.mycompany.com/pls/portal/htp.p?cbuf=test

The response should be test. If this is the result, the Oracle Application Server® middle-tier was able to connect to the OracleAS® Metadata Repository. If it is not, review *APPHOST1_ORACLE_HOME*\Apache\Apache\logs\error_log and *APPHOST1_ORACLE_HOME*\j2ee\OC4J_Portal\application-deployments\portal\OC4J_Portal_default_island_1\application.log for information on how to resolve the error.

c. Test the Oracle AS® Portal using following URL (ensure that you can log in):

https://portal.mycompany.com/pls/portal

d. Verify that content is being cached in OracleAS® Web Cache on APPHOST1, using Web Cache Administrator. Under **Monitoring**, click **Popular Requests**. Select **Cached** from the **Filtered Objects** drop-down list, and click **Update**.

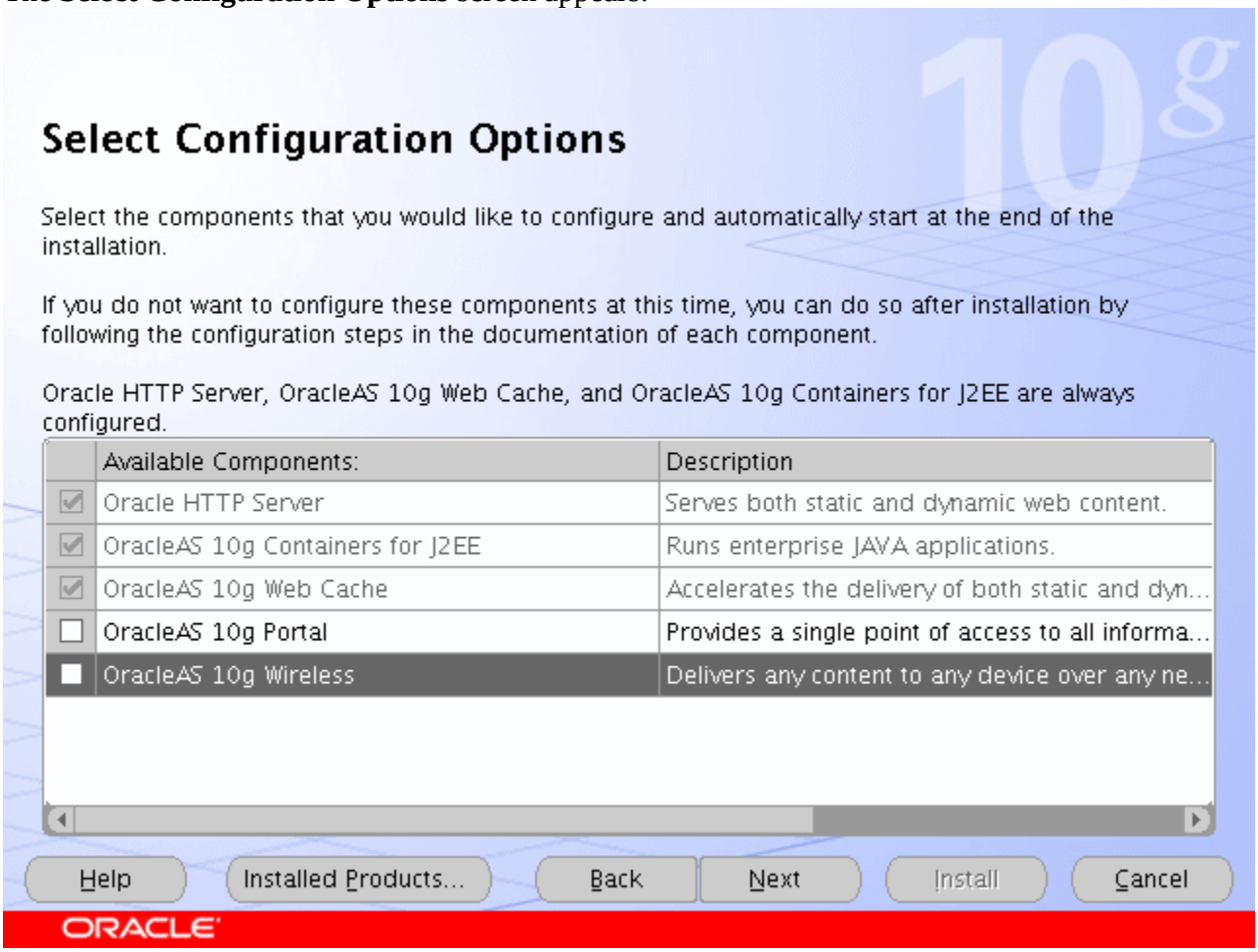
If you accessed OracleAS® Portal, portal content (for example, URLs that contain /pls/portal) will appear. If there is no portal content, open another browser and log in to OracleAS® Portal. Return to the **Popular Requests** page, and click **Update** to refresh the page content.

e. Add a portlet to a page, and then verify that the new content is present. If the new content does not display properly, or if errors occur, then the OracleAS® Web Cache invalidation is not configured correctly.

20. SETTING UP APPHOST2 (Application Server, Portal, Web Cache)

1. Follow steps 1 through 9 in Section 14 “Setting up APPHOST1”
2. Continue to next page…

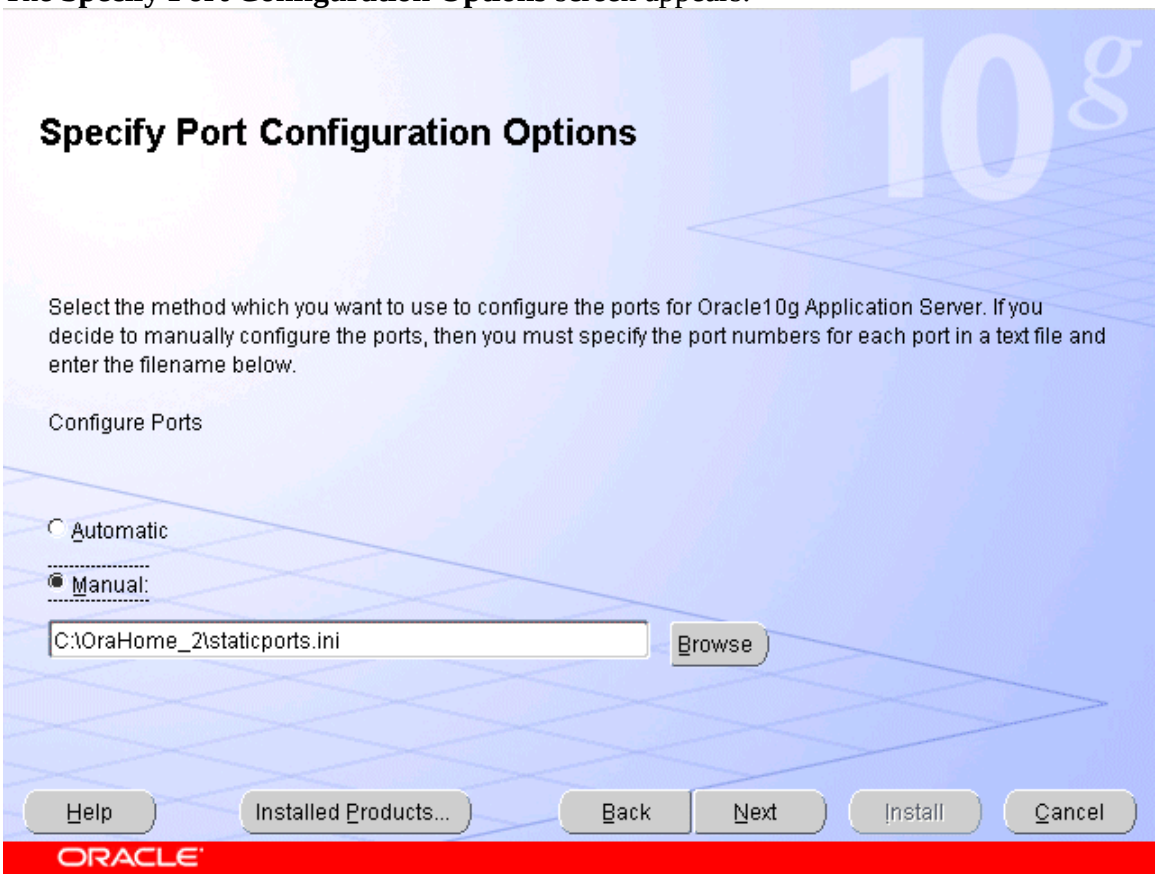
3. The **Select Configuration Options** screen appears:



Do not select any configuration options and click **Next**.

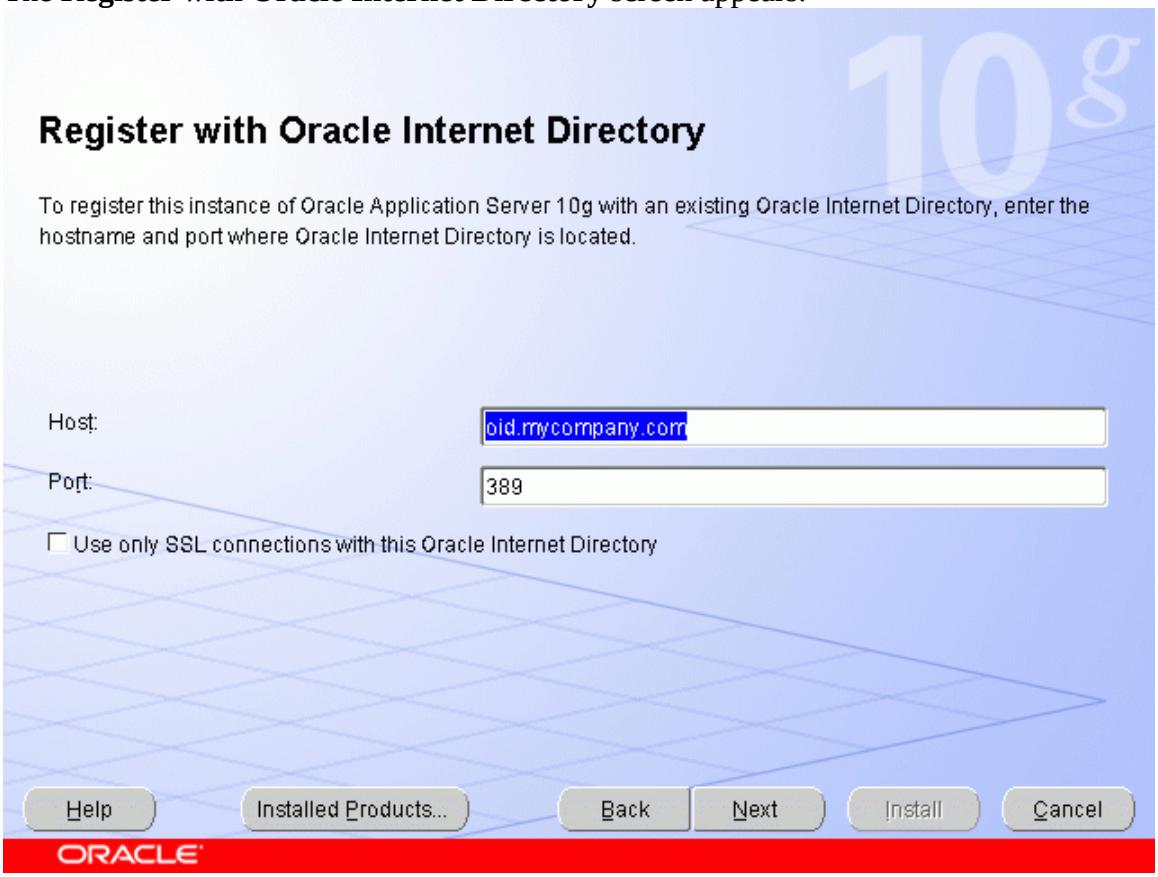
> **Note:**
> Selecting the Oracle Application Server® 10*g* Portal option in this screen now will overwrite the previously created configuration entries. For more information, refer to the *Oracle Application Server® Portal Configuration Guide*, section titled "Configuring OracleAS® Portal During and After Installation".

4. The **Specify Port Configuration Options** screen appears:



Select manual and select the location of the edited staticports.ini file which should be in *C:\OraHome_1* (not correctly shown in the image). Click **Next**.

5. The **Register with Oracle Internet Directory** screen appears:

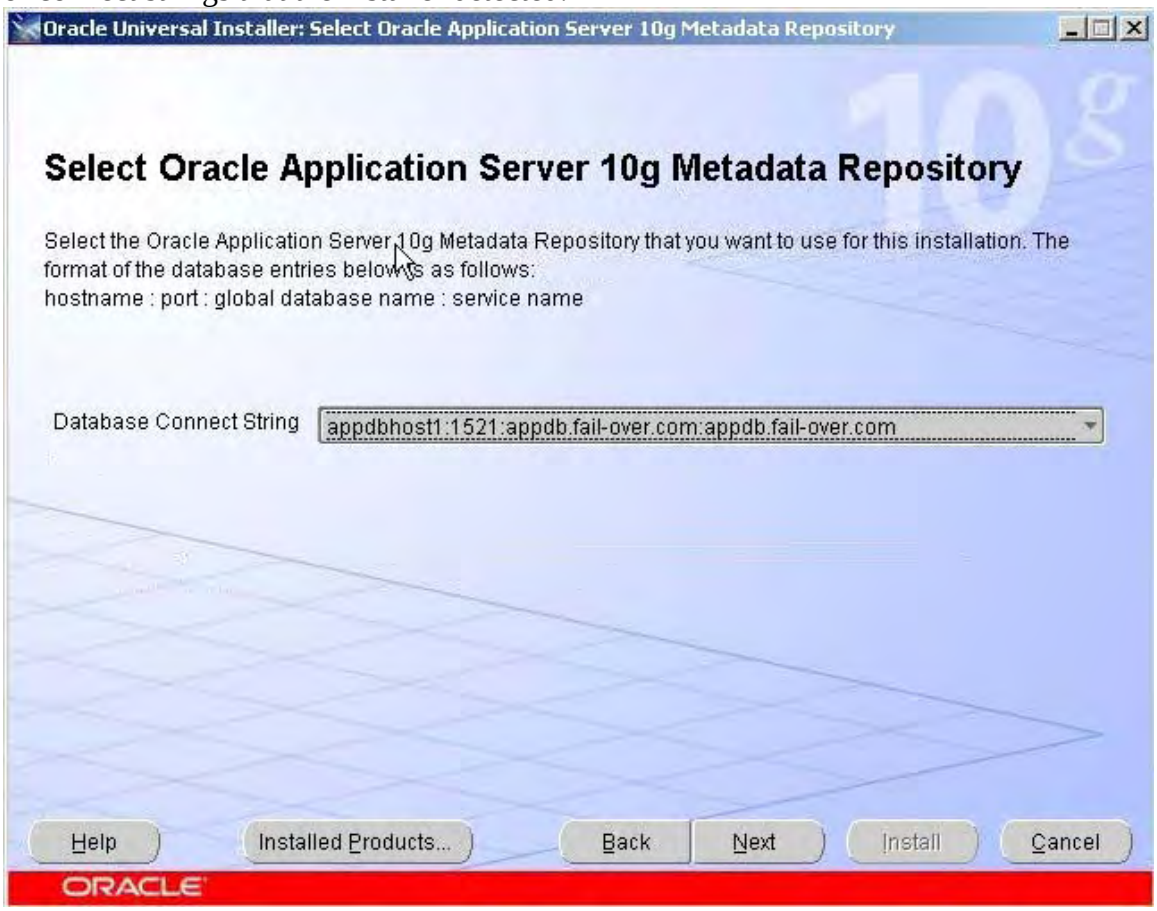


Enter the OID farm name and port 389. Click **Next**.

6. The **Specify OID Login** screen appears:

Enter the password for cn=orcladmin and click **Next**.

7. The **Select OracleAS 10*g* Metadata Repository** screen appears, displaying a drop-down list of connect strings that the installer detected:



Select the connect string for the application Metadata Repository database (on APPDBHOST1) and click **Next**.

8. The **Specify Instance Name and ias_admin Password** screen appears. We will assign **apphost2** as the instance name for this server. Assign your desired password of the ias_admin user for this instance. Click **Next**.
9. The **Summary** screen appears. Click **Install** to begin installation.
10. When the software installation completes the **Configuration Assistants** screen appears and will automatically go through each process. When it completes, the **End of Installation** screen appears.
11. Click **Exit**, and confirm your choice to exit.

21. Enabling Portal on APPHOST2

The first task is to configure OracleAS® Portal, using the Oracle® Enterprise Manager 10*g* Application Server Control Console. Follow these steps to configure OracleAS® Portal, beginning on the Application Server page:

1. Click **Configure Component**.
   The **Select Component** page appears.
2. Select **Portal** from the drop-down list.
   The **Login** page appears.
3. Enter the ias_admin password and click **Finish**.

The configuration process may take 10-20 minutes to complete.

Before you continue with the OracleAS® Portal application server configuration, ensure that the following is configured:

- You are able to resolve portal.mycompany.com from APPHOST2, either with DNS or with an entry in the hosts file, such that it contacts the Load Balancing Router. To ensure you can resolve portal.mycompany.com:

    - Issue this command from APPHOST2:

      nslookup portal.mycompany.com

      The IP address for the Load Balancing Router should be returned.

- You are able to contact port 7777 on portal.mycompany.com from APPHOST2. Issue this command on APPHOST2:

  telnet portal.mycompany.com 7777

  Verify that no connection failure message is returned.

It is important that you have the proper routing rules in APPHOST1 and 2 as specified in the SERVER CONFIGURATIONS of Section 2. Otherwise, communication will break and you will will get random errors or the portal page will not load up at all.

22. Configuring the Oracle® HTTP Server with the Load Balancing Router on APPHOST2

This step associates the components on which OracleAS® Portal depends with the Load Balancing Router, portal.mycompany.com on port 443.

1. Access the Oracle® Enterprise Manager 10*g* Application Server Control Console.
2. Click the link for the APPHOST2 installation.
3. Click the **HTTP Server** link.
4. Click the **Administration** link.
5. Click **Advanced Server Properties**.
6. Open the httpd.conf file.
7. Go to the very end of the file.
8. Perform the following steps:
    a. Add the LoadModule certheaders_module directive:

       LoadModule certheaders_module modules/ApacheModuleCertHeaders.dll

    b. Add the following lines to create a NameVirtualHost directive and a VirtualHost container for **portal.mycompany.com** and port **443**.

```
NameVirtualHost *:7778
<VirtualHost *:7778>
    ServerName portal.mycompany.com
    Port 443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    SimulateHttps On
</VirtualHost>
```

Notes:
The LoadModule directives (in particular, the LoadModule rewrite_module directive) must appear in the httpd.conf file at a location preceding the VirtualHost directives. The server must load all modules before it can execute the directives in the VirtualHost container.

It is a good idea to create the VirtualHost directives at the end of the httpd.conf file.

c. Create a second NameVirtualHost directive and a VirtualHost container for apphost2.mycompany.com and port 7777.

```
NameVirtualHost *:7778
<VirtualHost *:7778>
  ServerName apphost2.mycompany.com
  Port 7777
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

9. Save the httpd.conf file, and restart the Oracle® HTTP Server when prompted.
10. Copy the *APPHOST1_ORACLE_HOME*\Apache\modplsql\conf\dads.conf file to *APPHOST2_ORACLE_HOME*\Apache\modplsql\conf\.
11. Copy the *APPHOST1_ORACLE_HOME*\Apache\oradav\conf\oradav.conf file to *APPHOST2_ORACLE_HOME*\Apache\oradav\conf\.
12. Copy the *APPHOST1_ORACLE_HOME*\Apache\modplsql\conf\cache.conf file to *APPHOST2_ORACLE_HOME*\Apache\modplsql\conf\cache.conf.
13. Save the manual configuration changes to the DCM repository by issuing this command in *APPHOST2_ORACLE_HOME*\dcm\bin:

    dcmctl updateconfig -ct ohs

14. Use the Application Server Control Console to access the mod_plsql configuration pages:
    a. Click on the **HTTP Server** link.
    b. Click on **Administration**
    c. Click on **PL/SQL Properties**
15. Scroll to the bottom and click on the **/pls/portal** link under the DAD section click **Edit**.
16. Click **Apply**.

The required mod_rewrite and mod_oc4j directives are added.

23. Configuring the Parallel Page Engine Loop-Back with the Load Balancing Router on APPHOST2

In this step, you enable (non-SSL) loop-back communication between the Load Balancing Router and the Parallel Page Engines on APPHOST1 and APPHOST2. If the OracleAS® Web Cache on APPHOST1 is down, the Parallel Page Engine can loop back to the OracleAS® Web Cache on APPHOST2 through the Load Balancing Router to reach Portal Services. This is an example of component-level high availability.

Follow these steps to create the loop-back configuration:

1. Open the *APPHOST2_ORACLE_HOME*\j2ee\OC4J_Portal\applications\portal\portalWEB-INF\web.xml file.

2. Locate the Page servlet section and add the lines shown in bold:

```
<servlet>
<servlet-name>page</servlet-name>
  <servlet-class>oracle.webdb.page.ParallelServlet</servlet-class>
      <init-param>
        <param-name>useScheme</param-name>
        <param-value>http</param-value>
      </init-param>
      <init-param>
        <param-name>usePort</param-name>
        <param-value>7777</param-value>
      </init-param>
      <init-param>
        <param-name>httpsports</param-name>
        <param-value>443</param-value>
      </init-param>
</servlet>
```

3. Save the web.xml file.

The configuration now provides component-level high availability, since if the OracleAS® Web Cache on APPHOST1 is down, the Parallel Page Engine can loop back to the OracleAS® Web Cache on APPHOST2, through the Load Balancing Router, to reach Portal Services.

4. Save the manual configuration changes in the Distributed Configuration Management repository by issuing the following command on APPHOST2 in *ORACLE_HOME*/dcm/bin:

   dcmctl updateconfig

5. Restart all components on APPHOST2 by issuing the following command in *ORACLE_HOME*\opmn\bin:

opmnctl stopall
opmnctl startall

24. Modifying the Portal Dependency Settings (iasconfig.xml) File on APPHOST2

The Portal Dependency Settings file iasconfig.xml must contain the correct host, port and farm name to enable access to OracleAS® Portal and perform OracleAS® Web Cache invalidation.

1. Copy the *APPHOST1_ORACLE_HOME*\portal\conf\iasconfig.xml file to *APPHOST2_ORACLE_HOME*\portal\conf\.
2. Overwrite the file on APPHOST2 when prompted.

25. Configuring the Portal Tools Providers on APPHOST2

You must propagate the configuration changes made to Portal Tools providers on APPHOST1 to APPHOST2 by following these steps:

1. Copy the *APPHOST1_ORACLE_HOME*\j2ee\OC4J_Portal\applications\portalTools\omniPortlet\WEB-INF\providers\omniPortlet\provider.xml file to: *APPHOST2_ORACLE_HOME*\j2ee\OC4J_Portal\applications\portalTools\omniPortlet\WEB-INF\providers\omniPortlet\provider.xml
2. Copy the *APPHOST1_ORACLE_HOME*/j2ee/OC4J_Portal/applications/portalTools/webClipping/WEB-INF/providers/webClipping/provider.xml file to: *APPHOST2_ORACLE_HOME*/j2ee/OC4J_Portal/applications/portalTools/webClipping/WEB-INF/providers/webClipping/provider.xml
3. Copy the *APPHOST1_ORACLE_HOME*/j2ee/OC4J_Portal/config/data-sources.xml file to: *APPHOST2_ORACLE_HOME*/j2ee/OC4J_Portal/config/data-sources.xml.
4. Copy the *APPHOST1_ORACLE_HOME*/j2ee/OC4J_Portal/config/jazn-data.xml file to: *APPHOST2_ORACLE_HOME*/j2ee/OC4J_Portal/config/jazn-data.xml
5. Restart the OC4J_Portal instance.

26. Re-registering mod_osso on APPHOST2

1. Back up the *APPHOST2_ORACLE_HOME*\Apache\Apache\conf\osso\osso.conf file.
2. Use FTP binary mode to copy the

   APPHOST1_ORACLE_HOME\Apache\Apache\conf\osso\osso.conf file to *APPHOST2_ORACLE_HOME*\Apache\Apache\conf\osso.

3. Synchronize the DCM repository with the values in the obfuscated osso.conf file by issuing the following command:

   ORACLE_HOME\Apache\Apache\bin\ssotransfer $ORACLE_HOME\Apache\Apache\conf\osso\osso.conf

> **Note:**
> This does not create any new partner applications; it enables the partner application **portal.mycompany.com** forAPPHOST1 and APPHOST2.

4. Issue this command in *ORACLE_HOME*/dcm/bin:

   dcmctl updateconfig

5. Restart the components on APPHOST2 by issuing these commands in *APPHOST2_ORACLE_HOME*/opmn/bin:

   opmnctl stopall
   opmnctl startall

6. Access the following URL:

   https://login.mycompany.com/pls/orasso

7. Log in to the OracleAS Single Sign-On Administration page as the Administrator, and use the **Administer Partner Applications** page to delete the entry for the partner application **apphost2.mycompany.com**.

27. Configuring OracleAS® Web Cache Clusters

To cluster the OracleAS® Web Cache instances, you will perform the configuration steps on APPHOST1 and propagate them to APPHOST2.
From the Oracle® Enterprise Manager Application Server Control, you can access the Web Cache Manager, the graphical user interface provided for editing the configuration stored in the webcache.xml file. Start the Oracle Application Server® instance on APPHOST1, then follow these steps to access the Web Cache Manager from the **System Components** page:

1. Access the Web Cache Administrator at:

   http://apphost1.mycompany.com:9400/webcacheadmin

   The Web Cache Administrator password dialog appears.

2. For the user name, enter ias_admin or administrator, and enter the OracleAS® Web Cache administrator password.

> **Note:**
> At installation time, The OracleAS® Web Cache administrator password is set to the same password as the ias_admin password. The OracleAS® Web Cache administrator password must be identical for all cache cluster members.

3. The **Web Cache Manager** page appears. A scrollable frame on the left side of the window contains groups of configuration elements. To access an element, click its link. The content area of the page is then populated with the values for that element.
4. Click **Clustering** in the **Properties** section.

   The **Clustering** page appears.

5. In the **Cluster Members** table, click **Add**.

   The **Add Cache to Cluster** page appears.

6. Enter the following information for APPHOST2:

   - Host Name: **apphost2.mycompany.com**
   - Admin. Port: **9400**
   - Protocol for Admin. Port: **HTTP**
   - Cache Name: **apphost2.mycompany.com-Webcache**
   - Capacity: **20**

7. Click **Submit**.
8. Click the **Origin Server** link in the **Origin Servers, Sites, and Load Balancing** section.

   The **Origin Server** page appears.

9. Click **Add** under the **Application Web Servers** table.

   The **Add Application Web Server** page appears.

10. Enter the following information:

    - Hostname: **apphost2.mycompany.com**
    - Port: **7778**
    - Routing: **ENABLED**
    - Capacity: **30**
    - Failover Threshold: **5**
    - Ping URL: **/**
    - Ping Interval: **10**
    - Protocol: **HTTP**

11. Click **Submit**.
12. Click the **Site-to-Server Mapping** link in the **Origin Servers, Sites, and Load Balancing** section.

    The **Site-to-Server Mapping** page appears.

13. Select the mapping for the Load Balancing Router site (portal.mycompany.com) from the table and click **Edit Selected**.

    The **Edit/Add Site-to-Server Mapping** page appears.

14. In the **Select Application Web Servers** section, select an application Web server specified in the Origin Servers page for **apphost2.mycompany.com** (**apphost1.mycompany.com** is already mapped).
15. Click **Submit**.
16. Click **Apply Changes**.
17. In the **Cache Operations** page, click **Propagate**.

    The changes are propagated to apphost2.mycompany.com.

18. Click **Restart**.

    OracleAS® Web Cache is restarted on APPHOST1 and APPHOST2. OracleAS® Web Cache on APPHOST1 begins to balance requests to the Oracle HTTP Server and OC4J_Portal instances on APPHOST2.

After the clustering operation is completed, OracleAS® Web Cache on APPHOST1 will start balancing requests to the Oracle HTTP Server and OC4J_Portal instances running on APPHOST2. Repeat the steps in **Section 18 "Testing the Configuration on APPHOST1"** to confirm that the Oracle® HTTP Server and OC4J_Portal instances on APPHOST2 were configured properly.

> **Tip:**
> If these tests yield unsatisfactory or unexpected results, revisit the configuration steps performed to identify the cause. If the site is accepting live traffic, you might find it useful to temporarily remove the new OracleAS® Web Cache instance from the cluster, revisiting the configuration while the new middle tier is completely off-line. After the problem is resolved, you can redo the clustering operation and perform the validation again.

28. Enabling Session Binding on OracleAS Web Cache Clusters

The Session Binding feature in OracleAS® Web Cache is used to bind user sessions to a given origin server to maintain state for a period of time. Although almost all components running in a default OracleAS® Portal middle tier are stateless, session binding is required for two reasons:

- The Web Clipping Studio, used by both the OracleAS® Web Clipping Portlet and the Web Page Data Source of OmniPortlet, uses HTTP session to maintain state, for which session binding must be enabled.
- Enabling session binding forces all the user requests to go to a specific OracleAS® Portal middle-tier, resulting in a better cache hit ratio for the portal cache.

Follow these steps on APPHOST1 or APPHOST2 to enable session binding in OracleAS® Web Cache:

1. Access the Web Cache Administrator at:

   http://apphost1.mycompany.com:9400

   The Web Cache Administrator password dialog appears.

2. Enter the OracleAS® Web Cache administrator password.

> **Note:**
> At installation time, The OracleAS® Web Cache administrator password is set to the same password as the ias_admin password. The OracleAS® Web Cache administrator password must be identical for all cache cluster members.

3. The **Web Cache Manager** page appears. A scrollable frame on the left side of the window contains groups of configuration elements. To access an element, click its link. The content area of the page is then populated with the values for that element.
4. Click the **Session Binding** link in the **Origin Servers, Sites, and Load Balancing** section.

   The **Session Binding** page appears.

5. Select the Load Balancing Router site, portal.mycompany.com:443, from the table and click **Edit Selected**.

   The **Edit Session Binding** window opens.

6. Select **Any Set-Cookie** from the **Please select a session** drop-down list.
7. Select **Cookie-based** from the **Please select a session binding mechanism** drop-down list.
8. Click Submit.
9. Click **Apply Changes**.
10. On the **Cache Options** page, click **Propagate**.
    The changes are propagated to the OracleAS® Web Cache instance on the other computer.
11. Click **Restart**.
    OracleAS® Web Cache is restarted on APPHOST1 and APPHOST2.

29. CONCLUSION

At this point you should have a working portal page at https://portal.mycompany.com/pls/portal. If you enter http://portal.mycompany.com:7777/pls/portal, you should be automatically redirected to the https address.

You can log in as “orcladmin” or “portal_admin”.

We hope that this document allowed you to have a basic but clear understanding as to how the OracleAS® deployment interacts and to be able to advance from this point. Please keep in mind that this document does not discuss any security precaution issues. For greater detail about the OracleAS® myPortal architecture and security measures, please refer to http://download-west.oracle.com/docs/cd/B14099_17/core.1012/b13998/selecting.htm