CA Identity Manager

Installation Guide (WebSphere)

r12.5 SP10

CA Technologies Product References

This document references the following CA products:

- CA Identity Manager
- CA SiteMinder®
- CA Directory
- User Activity Reporting Module (UARM)
- CA Role & Compliance Manager

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA Technologies product documentation, complete our short customer survey, which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Installation Overview

This guide provides instructions for installing CA Identity Manager and also includes information on optional components for installation such as Provisioning and CA SiteMinder.

This section contains the following topics:

Sample CA Identity Manager Installations

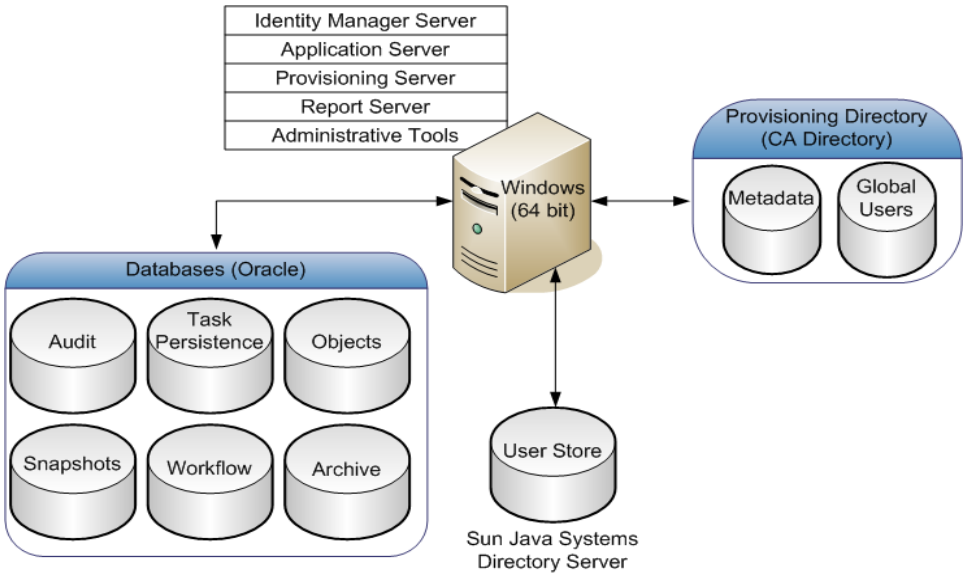
With CA Identity Manager, you can control user identities and their access to applications and accounts on endpoint systems. Based on the functionality you need, you select which CA Identity Manager components to install.

In all CA Identity Manager installations, the Identity Manager Server is installed on an application server. You use the CA Identity Manager Installer to install the other components you need.

The following sections illustrate some examples of CA Identity Manager implementations at a high level.

Example: Single Node Installation

In a single node installation, the Identity Manager Server is installed on one application server node. Also, one copy of each provisioning component is installed, but components can be on different systems. The following figure is an example of a single node CA Identity Manager installation with a Provisioning Server on the same system and a Provisioning Directory on another system:



This example also illustrates choices for platforms. In this case:

- The Identity Manager server is installed on Windows
- The user store is on Sun Java Systems Directory server
- The databases are on Oracle

These platforms are only examples. You can select other platforms instead.

Identity Manager Server

Executes tasks within CA Identity Manager. The J2EE Identity Manager application includes the Management Console (for configuring environments), and the User Console (for managing an environment).

Identity Manager Administrative Tools

Provides tools and samples for configuring and using CA Identity Manager. The tools include Connector Xpress, the Java Connector Server SDK, configuration files, scripts, utilities, and JAR files that you use to compile custom objects with CA Identity Manager APIs and API samples. The Provisioning Manager and WorkPoint Designer are also included with the Administrative Tools.

The default installation location for most Administrative Tools follows:

- **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools
- **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools

However, the default location for Provisioning Manager, which is only installed on Windows, follows:

- C:\Program Files\CA\Identity Manager\Provisioning Manager

Report Server

Uses CA Business Intelligence 3.2. You use this server to include data from the Snapshot Database, which contains information from the Identity Manager object store and the Identity Manager user store. An example of a Snapshot Report is the User Profile report. You can also create non-snapshot reports, which include data from other data sources, such as the Audit Database.

Identity Manager Databases

Store data for CA Identity Manager. The databases store information for auditing, task persistence, snapshots (reporting), workflow, and Identity Manager objects. Each database must be a relational database.

Note: For a complete list of supported relational databases, see the CA Identity Manager support matrix on the CA Support Site.

Identity Manager User Store

Contains users and their information. This store can be a pre-existing user store already in use by the company. This user store can be LDAP or a relational database.

Note: For more information about setting up a user store for CA Identity Manager, see the *Configuration Guide*.

Identity Manager Provisioning Server

Manages accounts on endpoint systems. On the same system or another system, you can also install Connector Servers, which manage Java or C++ based connectors to endpoints.

Identity Manager Provisioning Directory

Specifies the Provisioning Directory schema to CA Directory. This schema sets up the Directory System Agents (DSAs) within CA Directory. The Identity Manager user store can also be the Provisioning Directory.

Identity Manager Provisioning Manager

Manages the Provisioning Server through a graphical interface. This tool is used for administrative tasks such as synchronizing accounts with account templates. The Provisioning Manager is installed as part of the Identity Manager Administrative Tools or can be installed separately from those tools.

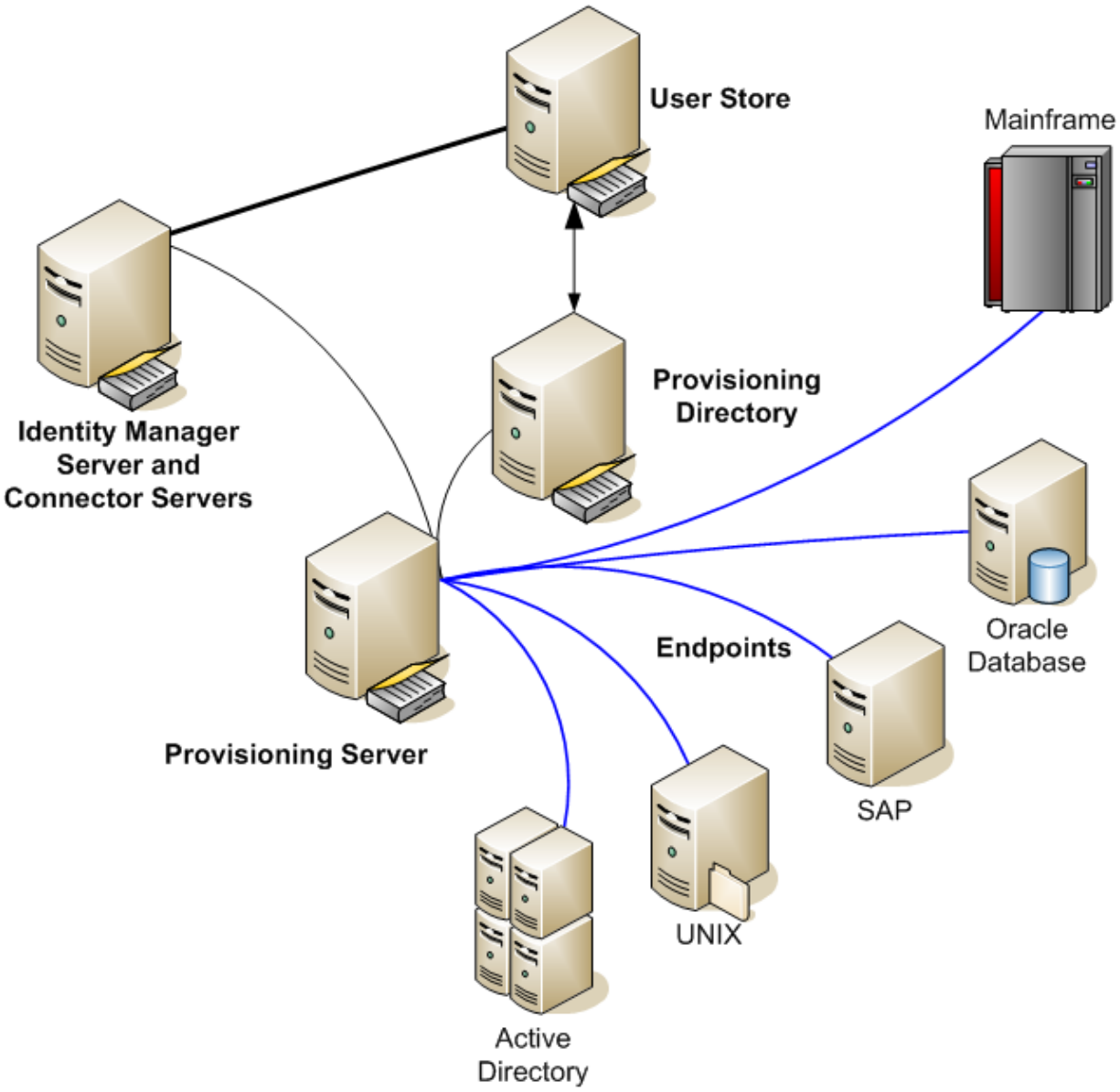
Note: This application runs on Windows only.

Example: Installation with Multiple Endpoints

If you install a Provisioning Server, administrators can provision accounts on endpoints, such as email servers, databases and other applications, to end users. To communicate with the endpoint systems, you install connector servers for endpoint-specific connectors, such as an SAP connector.

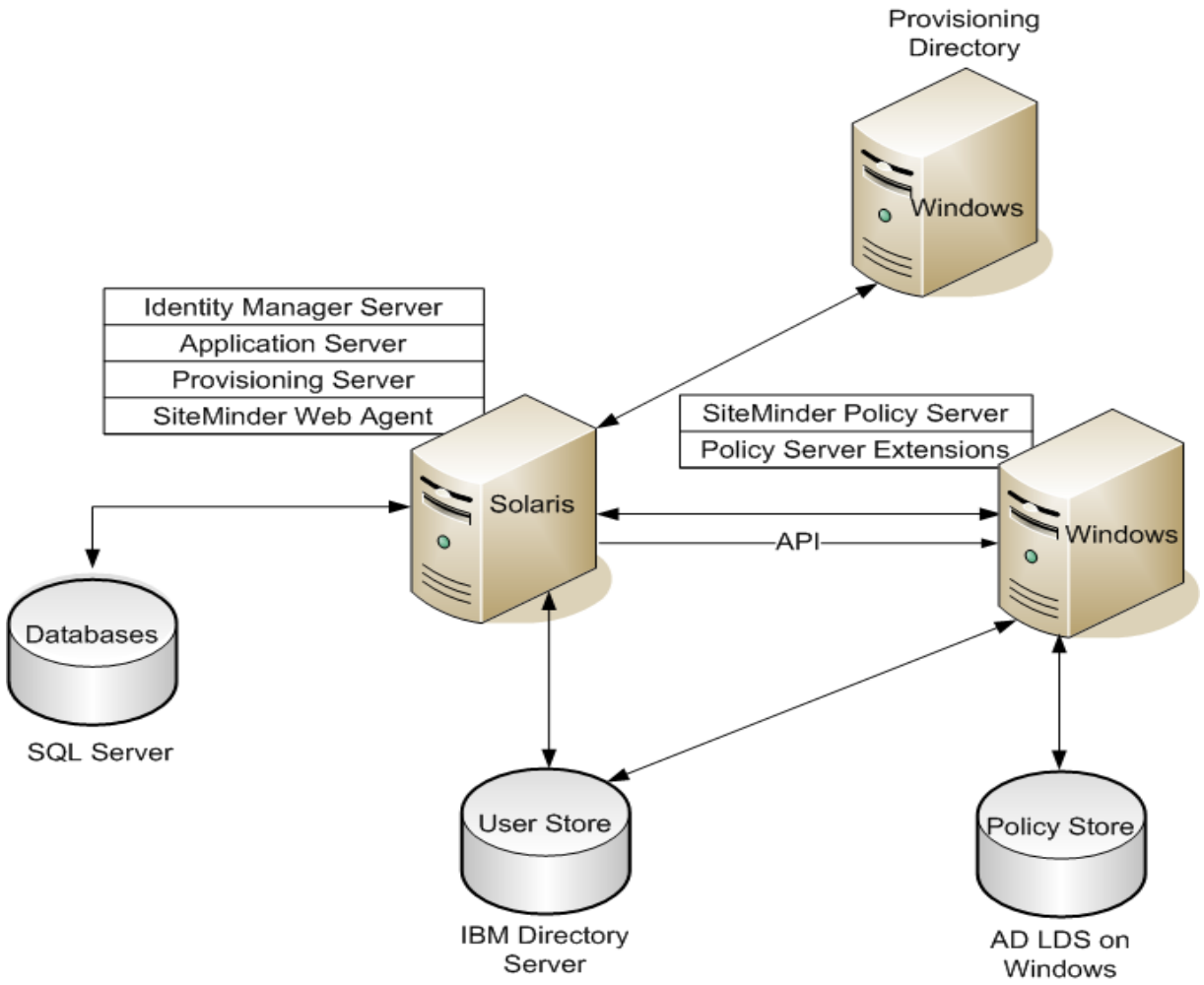
A typical installation scenario involves separate systems for the user store and the Provisioning Directory, which remained synchronized.

This example illustrates the use of CA Identity Manager to provide access to accounts on Active Directory, UNIX, SAP, Oracle, and mainframe systems.



Example: SiteMinder and CA Identity Manager Installation

CA Identity Manager can be integrated with a SiteMinder Policy Server, which provides advanced authentication and protection for your environments. The following figure is an example of a CA Identity Manager installation that uses a CA SiteMinder Policy Server for authentication and authorization:



The SiteMinder elements are defined as follows:

SiteMinder Web Agent

Works with the SiteMinder Policy Server to protect the User Console. Installed on the system with the Identity Manager Server.

SiteMinder Policy Server

Provides advanced authentication and authorization for CA Identity Manager and facilities such as Password Services, and Single Sign-On.

SiteMinder Policy Server Extensions

Enable a SiteMinder Policy Server to support CA Identity Manager. Install the extensions on each SiteMinder Policy Server system in your CA Identity Manager implementation.

The CA Identity Manager components are defined in the previous example on a single node installation; however, in this example, the components are installed on different platforms. The CA Identity Manager databases are on Microsoft SQL Server and the user store is on IBM directory Server. The SiteMinder Policy Store is on AD LDS on Windows, which is one of several supported platforms for a policy store.

High Availability Installation

Before installing CA Identity Manager, consider the goals for your implementation. For example, one goal could be a resilient implementation that consistently provides good performance. Another goal could be scalability.

A high-availability implementation provides the following features:

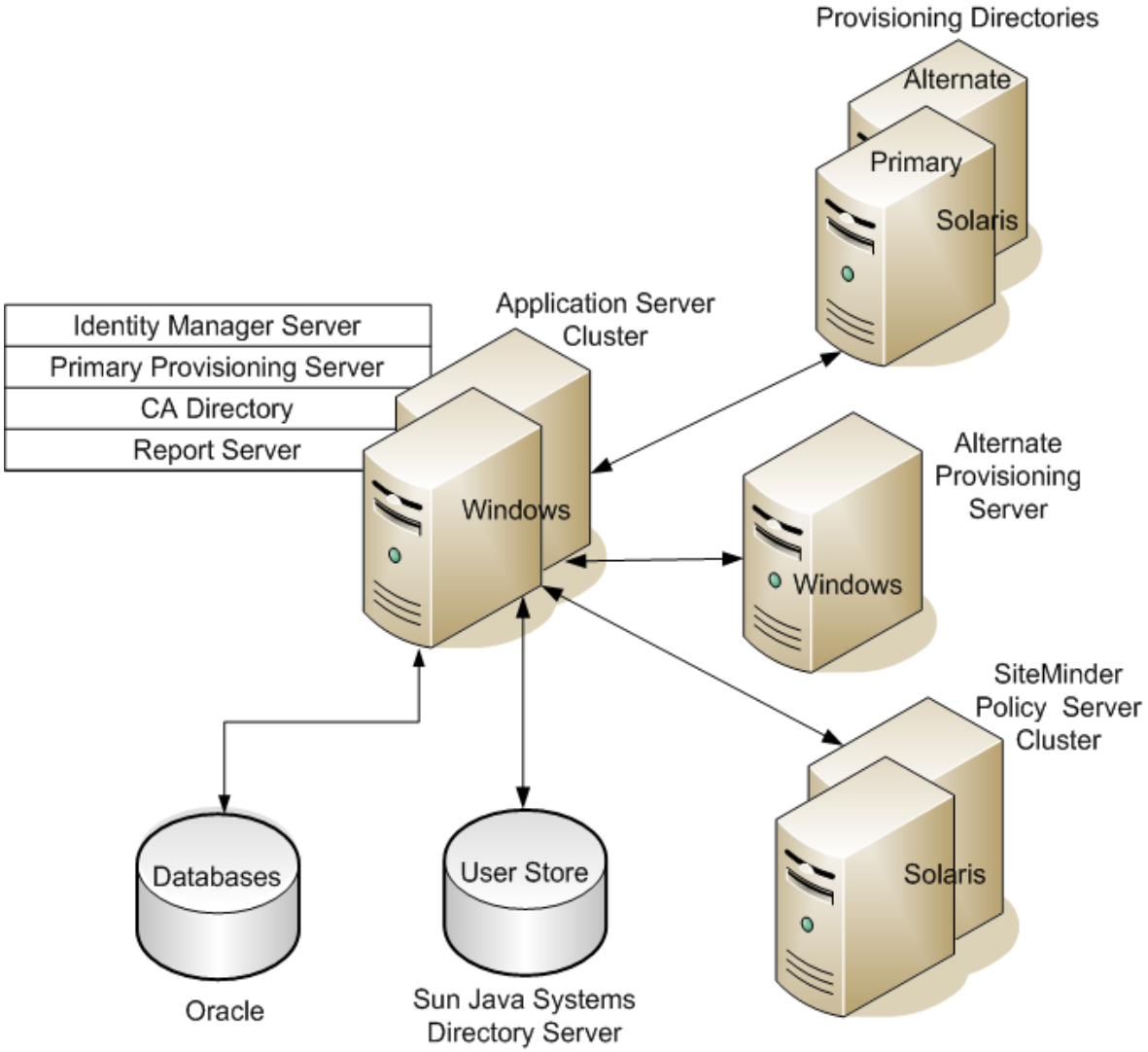
- Failover—Switches to another system automatically if the primary system fails or is temporarily offline for any reason
- Load balancing—Distributes processing and communications activity evenly across a computer network so that performance remains good and no single device is overwhelmed
- Various deployment tiers that provide the flexibility to serve dynamic business requirements

To provide these high-availability features, the following implementation options exist:

- The Identity Manager Server can be installed on an application server cluster to allow failover to any of the node in the cluster, providing uninterrupted access to users. The application server can be a 64-bit format, which provides better performance than a 32-bit application server.
- The Provisioning Server uses a CA Directory router to route traffic to a Provisioning Directory
- CA Identity Manager includes connector servers that you configure per-directory or per-managed systems. Installing multiple connector servers increases resilience. Each connector server is also an LDAP server, similar to the Provisioning Server.

Example: High Availability Installation

The following is an example that provides high availability for the Identity Manager Server, Provisioning Server, Provisioning Directory, and SiteMinder Policy Server. The use of alternate components and clusters provide the high availability features.



In addition to illustrating high availability, this figure shows the different platforms used for the components compared to the SiteMinder (see page 16) illustration. For example, the database uses Oracle instead of Microsoft SQL Server, which appeared in the previous illustration.

Identity Manager Server Architecture

An Identity Manager implementation may span a multi-tiered environment that includes a combination of hardware and software, including three tiers:

- Web Server tier
- Application Server tier
- Policy Server tier (optional)

Each tier may contain a cluster of servers that perform the same function to share the workload for that tier. You configure each cluster separately, so that you can add servers only where they are needed. For example, in a clustered Identity Manager implementation, a group of several systems may all have an Identity Manager Server installed. These systems share the work that is performed by the Identity Manager Server.

Note: Nodes from different clusters may exist on the same system. For example, an application server node can be installed on the same system as a Policy Server node.

Provisioning Components Architecture

Provisioning provides high availability solutions in the following three tiers:

- Client tier

  The clients are the Identity Manager User Console, Identity Manager Management Console and the Provisioning Manager. You can group clients together based on their geographic locations, organizational units, business functions, security requirements, provisioning workload, or other administration needs. Generally, we recommend keeping clients close to the endpoints they manage.

- Provisioning Server tier

  Clients use primary and alternate Provisioning Servers, in order of their failover preference. Client requests continue to be submitted to the first server until that server fails, that is, the connection stays active until the server fails. If a failure occurs, the client reviews the list of configured servers, in order of preference, to find the next available server.

The Provisioning Server can have multiple connector servers in operation. Each connector server handles operations on a distinct set of endpoints. Therefore, your organization could deploy connector servers on systems that are close in the network to the endpoints. For example, if you have many UNIX /etc endpoints, you can have one connector server installed on each server so that each connector server controls only the endpoints on that server where it is installed.

Installing Connector Servers close to the endpoints also reduces delays in managing accounts on endpoints.

- CA Directory tier (Provisioning Directory)

  Provisioning Servers uses a CA Directory router to send requests to primary and alternate Provisioning Directories in order of preference.

Overall Installation Process

To install CA Identity Manager, perform the following steps:

1. Install the prerequisite hardware and software and configure your system as required.
2. Install the Identity Manager Server on a single node or an application server cluster.
3. (Optional) Configure separate databases.
4. (Optional) Install the report server.
5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers for high availability provisioning capabilities.

Note: In this document, each chapter includes a checklist of the steps to install or configure a CA Identity Manager feature or component. That section begins with a How To title.

Chapter 2: Installation Prerequisites

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
X	**1. Install prerequisite hardware and software and configure your system as required.**
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Prerequisite Knowledge

This guide is intended for users who are familiar with Java, J2EE standards, and application server technology. It assumes that you have the following technical knowledge:

- An understanding of J2EE application servers and multi-tier architecture
- Experience with managing the application server, including tasks such as starting the application server
- Experience with managing a relational database
- (Optional) Familiarity with SiteMinder concepts, terms, and Policy Server configuration tasks

How to Install Prerequisite Components

To install the prerequisite hardware and software for CA Identity Manager:

✓	Step
	1. Make your system meets the hardware requirements.
	2. Install CA Directory.
	3. (Optional) Create a FIPS key.
	4. (Optional) Integrate with SiteMinder.
	5. Create a database.
	6. Set up the application server.
	7. Meet IPv6 requirements if installing on IPv6 systems.
	8. Check Provisioning requirements if installing on Solaris.
	9. Complete the Installation Worksheets with information you need for the CA Identity Manager installation program.

Check Hardware Requirements

Identity Manager Server

These requirements take into account the requirements of the application server installed on the system where you install the Identity Manager Server.

Component	Minimum	Recommended
CPU	Intel (or compatible) 2.0 GHz (Windows or Red Hat Linux), SPARC 1.5 GHz (Solaris) or POWER4 1.1 GHz (AIX)	Dual core Intel (or compatible) 3.0 GHz (Windows or Red Hat Linux), Dual core SPARC 2.5 GHz (Solaris) POWER5 1.5 GHz (AIX)
Memory	4 GB	8 GB
Available Disk Space	4 GB	8 GB
Temp Space	2 GB	4 GB

Component	Minimum	Recommended
Swap/Paging Space	2 GB	4 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor and operating system for intermediate and large deployments, dual core	64-bit processor and operating system, quad core

Provisioning Server or a Standalone Connector Server

Component	Minimum	Recommended
CPU	Intel (or compatible) 2.0 GHz (Windows) SPARC 1.5 GHz (Solaris)	Dual core Intel (or compatible) 3.0 GHz (Windows) SPARC 2.0 GHz (Solaris)
Memory	4 GB	8 GB
Available Disk Space	4 GB	8 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor and operating system for intermediate and large deployments, dual core	64-bit processor and operating system, quad core

Provisioning Directory

Component	Minimum	Recommended
CPU	Intel (or compatible) 1.5 GHz (Windows) SPARC 1.0 GHz (Solaris)	Dual core Intel (or compatible) 2.5 GHz (Windows) SPARC 1.5 GHz (Solaris)
Memory	4 GB	8 GB

Component	Minimum	Recommended
Available Disk Space	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per datafile (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per datafile, (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per datafile, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per datafile, total 8 GB	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per datafile (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per datafile, (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per datafile, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per datafile, total 8 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor, 64-bit operating system, and CA Directory (64 bit version) for intermediate and large deployments	64-bit processor and operating system

All Components on One System

Hosting the entire CA Identity Manager product on a single physical system is not recommended for production environments. However, to do so, the hardware requirements are as follows:

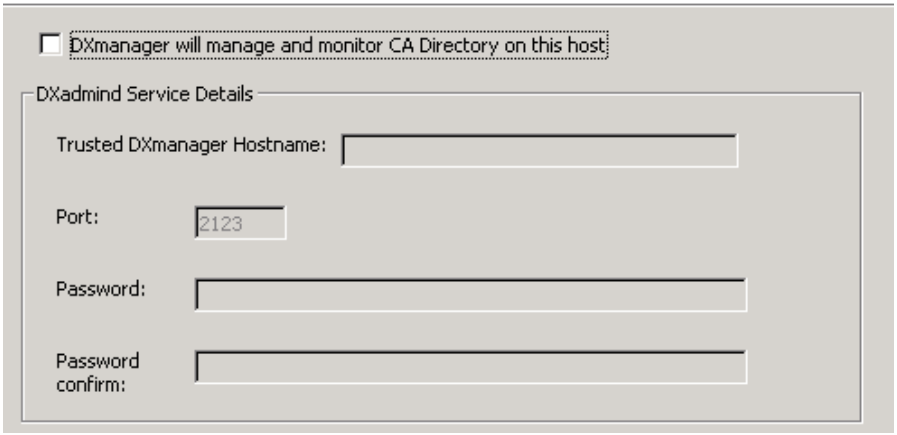
Component	Minimum
CPU	Intel (or compatible) 3.1 GHz (Windows) SPARC 2.5 GHz (Solaris)
Memory	8 GB
Available Disk Space	6 to 14 GB depending on the number of accounts
Processor	64-bit processor and operating system for intermediate and large deployments, quad core
Swap/Paging Space	6 GB

Install CA Directory

Before you install CA Identity Manager, install CA Directory using the following steps:

1. Install CA Directory on the system where you plan to install the Provisioning Directory. A supported version of CA Directory is included on your installation media. For details on installation, download the CA Directory documentation from the support site.

   Note: When the installer asks about installing DXadmind for DXManager, you can safely clear this option. The Provisioning Directory does not use DXManager.



2. Install a second copy of CA Directory on the system where you plan to install the Provisioning Server. This installation is for routing purposes, so that the Provisioning Server can communicate with the remote Provisioning Directory.

Important! We recommend that you disable all antivirus software before installation. If antivirus software is enabled while installation takes place, problems can occur. Remember to re-enable your antivirus protection after you complete the installation.

Create a FIPS 140-2 Encryption Key

When you run the CA Identity Manager installer, you are given the option of enabling FIPS 140-2 compliance mode. For CA Identity Manager to support FIPS 140-2, all components in a CA Identity Manager environment must be FIPS 140-2 enabled. You need a FIPS encryption key to enable FIPS 140-2 during installation. A Password Tool for creating a FIPS key is located in the installation media at PasswordTool\bin.

Important! Use the same FIPS 140-2 encryption key in all installations and be sure that you safeguard the key file once generated by the Password Tool.

(Optional) Integrate with SiteMinder

A SiteMinder Policy Server is an optional component that you install as described in the *SiteMinder Installation Guide*. If you plan to make the Policy Server highly available, you configure it as a Policy Server cluster. You also install JCE libraries to enable communication with CA Identity Manager.

To install a Policy Server

1. Install the SiteMinder Policy Server. For details, see the *CA SiteMinder Policy Server Installation Guide*.
2. If you plan to make the Policy Server highly available, install it on each node that should be in the Policy Server cluster.

   Note: Each Policy Server in the cluster uses the same policy store.
3. Verify that you can ping the systems that host the Policy Server from the system where you plan to install the Identity Manager Server.

To install the Identity Manager Extensions for SiteMinder

Before installing the Identity Manager server, you add the extensions to each Policy Server. If the Policy Server is on the system where you plan to install the Identity Manager server, you can install the extensions and the Identity Manager server simultaneously. If so, omit this procedure.

1. Stop the SiteMinder services.
2. Install the Identity Manager Extensions for SiteMinder. Do one of the following:
   - **Windows:** From your installation media, run the following program in the top-level folder:
     `ca-im-r12.5sp*N*-win32.exe`
   - **UNIX:** From your installation media, run the following program in the top-level folder:
     `ca-im-r12.5sp*N*-sol.bin`

   sp*N* represents the current SP release of CA Identity Manager.
3. Select Extensions for SiteMinder.
4. Complete the instructions in the installation dialog boxes.

To install JCE Libraries

As of r12.5 SP7, the Identity Manager server requires the Java Cryptography Extension (JCE) libraries if you are also using CA SiteMinder.

Before installing the Identity Manager Server, download and install the Java Cryptography Extension Unlimited Strength Jurisdiction Policy Files. Select the one that works with your application server and JDK. The download ZIP file includes a ReadMe text file with installation instructions.

Create the Database

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. Install a supported version of Oracle or Microsoft SQL Server and create a database.

When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, after installing CA Identity Manager, you can configure separate databases for auditing, snapshots (reporting), workflow, and task persistence. To create these databases, see the chapter on Separate Database Configuration.

WebSphere Application Server

The Identity Manager Server is a J2EE application that is deployed on a supported application server. When using WebSphere as the CA Identity Manager application server, perform the following procedures.

Upgrade WebSphere

CA Identity Manager r12.5 SP9 works with Websphere 6.1 (for an upgrade) or Websphere 7 (for a new installation or a migration of CA Identity Manager).

If you need a new version of the IBM WebSphere, install the WebSphere server as described in IBM's documentation. During the installation, perform these actions:

- Select the appropriate plug-in for your Web Server.
- Select the Server and Client options.
- Install the latest FixPack to the server and the required JDK.

Note: For a complete list of supported platforms and versions, see the CA Identity Manager support matrix on CA Support.

Disable Global Security

We recommend that you disable security before the insallation. Verify that the Security Enabled option is unchecked. This action will ensure that the profile can be created without a problem.

Verify WebSphere

Use the following tests to verify that WebSphere is working:

- Test whether the WebSphere application server is installed correctly by accessing IBM's snoop utility at the following URL:

  http://*hostname*:*port*/snoop

  For example:

  http://MyServer.MyCompany.com:9080/snoop

  If WebSphere is installed correctly, the Snoop Servlet—Request Client Information page is displayed in the browser.

- If you have a web server installed, test that the WebSphere application server plug-in is installed correctly. Use IBM's snoop utility without including the application server port in the URL:

  http://*hostname*/snoop

  For example:

  http://MyServer.MyCompany.com/snoop

  If WebSphere is installed correctly, the same Snoop Servlet—Request Client Information page is displayed in the browser. This means that profile was created and has at least one server which is configured with the plug-in.

For additional help with WebSphere, contact IBM customer support.

Configure WebSphere for CA Identity Manager

Perform the following steps to ensure that your CA Identity Manager installation succeeds on WebSphere.

1. Save any changes to the WebSphere configuration via the Admin Console (Save to Master Configuration).
2. Shut down the application server.

3. Remove the contents of the following folders:
   - Temp Directory:
     - Windows: %temp%
     - Unix: /tmp/*
   - *Websphere_home*/profiles/WAS_PROFILE/temp/*
   - *Websphere_home*/profiles/WAS_PROFILE/wstemp/*
   - *Websphere_home*/profiles/WAS_PROFILE/tranlog/*
   - *Websphere_home*/profiles/WAS_PROFILE/configuration/*
   - *Websphere_home*/deploytool/itp/configuration/org.*, leaving only config.ini in this directory
4. In the *Websphere_home*/profiles/WAS_PROFILE/properties/soap.client.props file, set com.ibm.SOAP.requestTimeout to 1800 or higher.

Note: For more information, see your WebSphere documentation.

Important! Restart your WebSphere application server before starting the CA Identity Manager installation.

Enable XA Transactions for Microsoft SQL Server

If you are using WebSphere with Microsoft SQL Server, enable XA transactions on Microsoft SQL Server. CA Identity Manager needs an XA data source for the database transactions to work properly.

To enable XA Transactions for Microsoft SQL Server

1. Download the SQL Server JDBC Driver version 2.0 from Microsoft.

   Note: The download may first present an HTM file that is a license agreement for you to approve.
2. Run the program to install the JDBC driver.
3. Perform the following two procedures included in the Microsoft topic Understanding XA Transactions:
   - Running the MS DTC Service
   - Configuring the JDBC Distributed Transaction Components

   In performing these procedures, verify the following are true:
   - When you run the xa_install.sql script, make sure you get a script complete message. You can ignore the drop table errors, which appear the first time that you run the script.
   - When you add the user to the SqlJDBCXAUser role, add that user to the master database.

Configure SSL

If you upgraded your application server and you are using a user directory with SSL, be sure that SSL is configured on your application server before the upgrade.

Solaris Requirements for the Provisioning Server

Provisioning Server prerequisites on Solaris

Check /etc/system and verify the following minimum IPC kernel parameter values:

- set msgsys:msginfo_msgmni=32
- set semsys:seminfo_semmni=256
- set semsys:seminfo_semmns=512
- set semsys:seminfo_semmnu=256
- set semsys:seminfo_semume=128
- set semsys:seminfo_smmsl=128
- set shmsys:shminfo_shmmni=128
- set shmsys:shminfo_shmmin=4

Solaris 9 or 10 Requirements

Before installing provisioning software on Solaris 9 or 10, download and install the required patches.

To download the Sun Studio 10 patches for the Provisioning SDK

1. Go to the following URL:
   `http://developers.sun.com/prodtech/cc/downloads/patches/ss10_patches.html`
2. Download and install patch 117830.

   Note: Sun Studio 11 does not require patching.

To download Solaris 9 patches for all Provisioning components

1. Go to the following URL:

   `http://search.sun.com/search/onesearch/index.jsp`
2. Download and install 9_recommended.zip

IPv6 Support

CA Identity Manager supports IPv6 on the following operating systems:

- Solaris 9 or higher
- Windows XP SP2 or higher
- Windows 2003 SP2 or higher
- Windows 2008 or higher

Note: The Java Connector Server does not support IPv6 on Microsoft Windows platforms. No JDK is available to work with IPv6 as of release time for CA Identity Manager r12.5 SP9. If a JDK is released that works with IPv6, the CA Identity Manager support matrix will be updated on CA Support.

IPv6 JDK Requirements on WebSphere

JDK 1.5 SR9 is required to support IPv6 on WebSphere.

IPv6 Configuration Notes

Note the following before configuring an Identity Manager Environment that supports IPv6:

- For CA Identity Manager to support IPv6 addresses, all components in the CA Identity Manager implementation, including the operating system, JDK, directory servers, and databases must also support IPv6 addresses.
- If CA Identity Manager integrates with SiteMinder, the Web Server plug-in for the application server must also support IPv6.
- When you connect to SiteMinder or any database from CA Identity Manager using a JDBC connection, specify the hostname not the IP address.
- The Report Server can be installed on a dual-stack host, which supports IPv4 and IPv6, but the communication to the server must be IPv4.
- When you configure a connection to the Report Server in the Management Console, the server name must be in IPv4 format.
- CA Identity Manager does not support IPv6 link local addresses.
- In an IPv4/6 environment, if you want to configure CA Directory DSAs to listen on multiple addresses, add the additional addresses to your DSA knowledge files. For more information, see the CA Directory documentation.

Provisioning Directory on Windows 2008 with Pure IPv6 Not Supported

Due to a Sun Java Systems limitation, installing the Provisioning Directory on a Windows 2008 server with the IPv6 networking service uninstalled is not supported.

To work around this issue, install the IPv6 service on the system and leave it disabled.

Complete the Installation Worksheets

The CA Identity Manager installation program asks you for information about previously installed software and the software that you are installing. Ensure that you provide hostnames (and not IP addresses) in the installer screens.

Note: Use the following **Installation Worksheet** to record this information. We recommend that you complete the worksheet before starting the installation.

Provisioning Directory

Record the following Provisioning Directory and Provisioning Server information you need during the CA Identity Manager installation.

Field Name	Description	Your Response
Provisioning Directory Hostname	The hostname of the Provisioning Directory system if it is remote. You need the hostnames for the primary and any alternate Provisioning Directories.	
Shared Secret	The special password for the Provisioning Directory. Use the same password for the primary and any alternate Provisioning Directories.	
Provisioning Server Hostname	The host names of the primary and any alternate Provisioning Servers.	

WebSphere Information

Record the following WebSphere information you need during the CA Identity Manager installation:

Field Name	Description	Your Response
WebSphere Install Folder	The location of the application server home directory.	
Server Name	The name of the system on which the application server is running.	
Profile Name	The name of the profile you want to use for CA Identity Manager.	
Cell Name	The name of the cell in which the application server is located.	
Node Name	The name of the node in which the application server is located.	
Cluster Name	The cluster name for high-availability implementations. This is only needed if you plan on installing CA Identity Manager in a clustered environment.	
Access URL and port	The application URL and port number of the system that will host the Identity Manager Server (system that will host the application server).	

Database Connection Information

An Oracle or Microsoft SQL Server database must already be configured and working. Record the following database information you need during the CA Identity Manager installation:

Field Name	Description	Your Response
Database Type	The database type (vendor/version) of the database created for task persistence, workflow, audit, reporting, object storage, and task persistence archive.	
Host Name	The hostname of the system where the database is located. **Note:** Be sure that you provide a hostname and *not* an IP address.	
Port Number	The port number of the database.	
Database Name	The database identifier.	
Username	The username for database access. **Note:** This user must have administrative rights to the database unless you plan to import the schema manually.	
Password	The password for the user account with administrative rights.	

Login Information

Record the following passwords you need during the Provisioning Components installation.

Field Name	Description	Your Response
Username	A username that you create to log into the provisioning components.	
Provisioning Server password	A password for this Server.	

Field Name	Description	Your Response
C++ Connector Server password	A password needed for this server. Each C++ Connector Server can have a unique password.	
Provisioning Directory password	A password used by Provisioning Server to connect to Provisioning Directory. For an alternate Provisioning Server, enter the Provisioning Directory password created for the primary Provisioning Server.	

SiteMinder Information

If you plan to use a SiteMinder Policy Server to protect CA Identity Manager, record the following information:

Field Name	Description	Your Response
Policy Server Host Name	The hostname of the SiteMinder Policy Server.	
SiteMinder Administrator Name	The administrator username for the SiteMinder Policy Server.	
SiteMinder Administrator Password	The administrator user password for the SiteMinder Policy Server.	
SiteMinder Folder (Solaris Only)	The location of SiteMinder on the system with a SiteMinder Policy Server installed.	
SiteMinder Agent Name	The name of the SiteMinder Agent that CA Identity Manager will use to connect to SiteMinder.	
SiteMinder Shared Secret	The shared secret for the above Agent.	

Chapter 3: Single Node Installation

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Single node installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

CA Identity Manager Components

In a single node installation, you install one copy of each component, but use two or more systems for where you install them.

Note: If you intend to install multiple copies of components for high availability, see the chapters on installation on a cluster and high-availability provisioning installation.

Install one of each of the following components on a system at your site:

- Identity Manager Server—Installs the server that provides the core functionality of the product.
- Identity Manager Administrative Tools—Installs tools such as the Provisioning Manager, which runs on a Windows system, the SDK for the Java Connector Server, and Connector Xpress.

  Connector Xpress manages dynamic connectors, maps them to endpoints, and establishes routing rules. Dynamic connectors allow provisioning and management of SQL databases and LDAP directories
- Identity Manager Provisioning Server—Enables provisioning in CA Identity Manager. Installation of this server includes the C++ Connector Server, which manages endpoints that use C++ connectors.
- Java Connector Server—Manages endpoints that use java connectors. The Java Connector Server is registered with the Provisioning Server when you install it.

  Note: You can instead install the Java Connector Servers separate from the Provisioning Server. See the *Java Connector Server Implementation Guide*.
- Identity Manager Provisioning Directory Initialization—Configures a CA directory instance to store provisioning data. Use the installation program on each system where CA Directory is installed.
- Extensions for SiteMinder—Extends the SiteMinder Policy Server if you are using it to protect CA Identity Manager. Install these extensions on the same system as the Policy Server before you install the Identity Manager Server.

How to Perform a Single Node Installation

Use the following checklist to perform a basic installation of CA Identity Manager:

✓	Step
	1. Install CA Identity Manager components on the systems required.
	2. Verify the Identity Manager Server starts.
	3. Configure Provisioning Manager if installed on a remote system.
	4. Install optional provisioning components.

Install CA Identity Manager Components

For a production environment, use separate systems for data servers. For example, we recommend that the Provisioning Directory and a database (SQL or Oracle) are on a separate system from the Identity Manager Server and the Provisioning Server. If you are installing SiteMinder, you may also prefer to have it on a separate system. The Administrative Tools can be installed on any system.

Use the CA Identity Manager installer to perform the installation on the systems required. In the procedures that follow, the step to run the installer refers to this program in your installation media's top-level folder:

- **Windows:**
  `ca-im-release-win32.exe`
- **UNIX:**
  `ca-im-release-sol.bin`

release represents the current release of CA Identity Manager.

For each component that you install, be sure that you have the required information for installer screens, (see page 32) such as host names and passwords. If any issues occur during installation, check the installation logs (see page 151).

To install the Extensions for SiteMinder

1. Log into the system where SiteMinder is installed as a Local Administrator (for Windows) or root (for Solaris).
2. Stop the SiteMinder services.
3. Run the installer and select Extensions for SiteMinder.

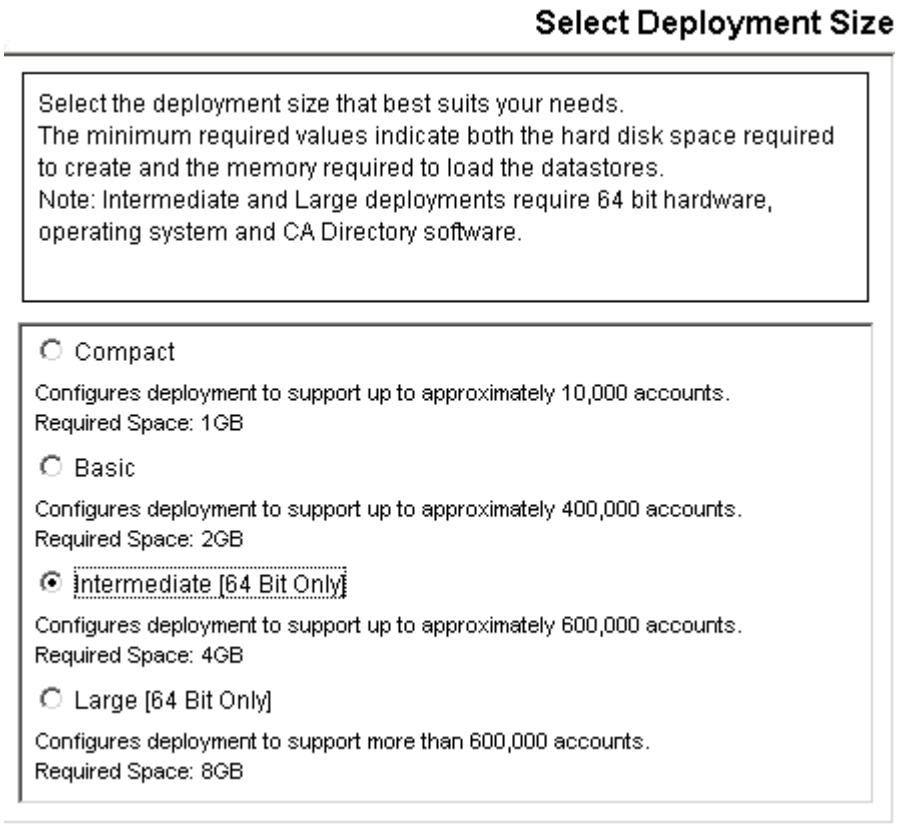
To install the Identity Manager Server

1. If you have installed SiteMinder on a separate system, be sure that you have installed the extensions for SiteMinder there also.
2. Log in to the system where the application server is installed as a Local Administrator (for Windows) or root (for Solaris).
3. Stop the application server.
4. Run the installer and select the Identity Manager Server.

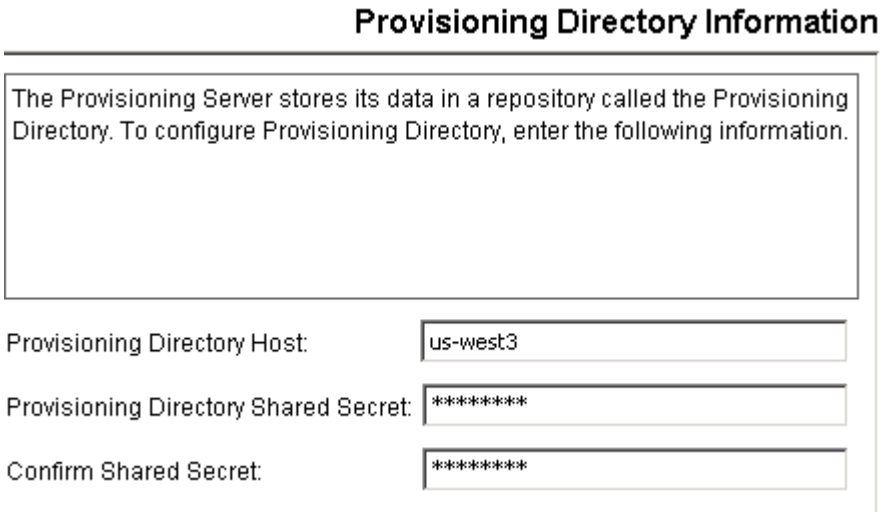
To install the Provisioning Directory

1. Log into the system as a Local Administrator (for Windows) or root (for Solaris).
2. Be sure that CA Directory is already installed on the system.
3. Run the installer and select the Identity Manager Provisioning Directory Initialization.
4. Answer the question about deployment size. Consider the following guidelines, while allowing room for future growth:
   - Compact—up to 10,000 accounts
   - Basic—up to 400,000 accounts

- Intermediate (64 bit only)—up to 600,000 accounts
- Large (64 bit only)—more than 600,000 accounts



5. When you enter any password or shared secret in the installation, be sure to provide a password that you can recall when needed.



To install the Provisioning Server

1. Log into the system as a Local Administrator (for Windows) or root (for Solaris).
2. Be sure that CA Directory is already installed and you have the details of the remote Provisioning Directory.
3. Run the installer and select the Identity Manager Provisioning Server.

Configure Workflow for Your Profile

If you have not used the default WebSphere profile for your installation, you configure workflow for the WebSphere Server.

To configure workflow for your WebSphere Profile

1. Start the WebSphere Console.
2. Navigate to Servers, Server Types, Application Servers, *server_name*.
3. Under Communications, Expand Ports.
4. Make note of the port used for the BOOTSTRAP_ADDRESS.
5. Edit Workpoint-client.properties file under iam_im.ear/config.

6. Locate the WebSphere section in this file.
   `# java.naming.provider.url=iiop://localhost:2809`
7. Replace 2809 with the profile's port that is used for the BOOTSTRAP_ADDRESS.
8. Restart this server.

Verify the Identity Manager Server Starts

To verify access to CA Identity Manager

1. Start CA Identity Manager as follows:
   - **Windows:**

     For WebSphere 6.1, click Navigate to Start, Programs, IBM WebSphere, Application Server 6.x, Profiles, Profile_Type, Start the Server

     For WebSphere 7, click Start, Programs, IBM WebSphere, Application Server Network Deployment V7.0, Profiles, Profile Name

     Note: To view status information, use the First Steps console, which you access from the same location as the Start the Server command mentioned above. In the First Steps console, select Start the Server.
   - **UNIX:**

     a. Navigate to *websphere_home*/profiles/*profile_name/*bin from the command line.

     b. Enter the following command:

     startserver *websphere_server*

     When you see a message that resembles the following, the server has completed its startup process:

     `Server server1 is open for e-business`
2. Access the Management Console and confirm the following:
   - You can access the following URL from a browser:

     http://*im_server*:*port*/iam/immanage

     For example:

     http://MyServer.MyCompany.com:*port-number/*iam/immanage
   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.
3. Verify that you can access an upgraded environment using this URL format:

   http://*im_server*:*port*/iam/im/*environment*

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP file includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you must install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Vista Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

The Identity Manager installer installs all connectors by default. However, in some cases, install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA Option for Password Reset/Unlock (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you configure communication to the server.

Note: To install the Provisioning Manager, install the Identity Manager Administrative Tools on a Windows system.

To configure a remote Provisioning Manager

1. Log into the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA, Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.
4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.
6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Chapter 4: Installation on a WebSphere Cluster

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Single node installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

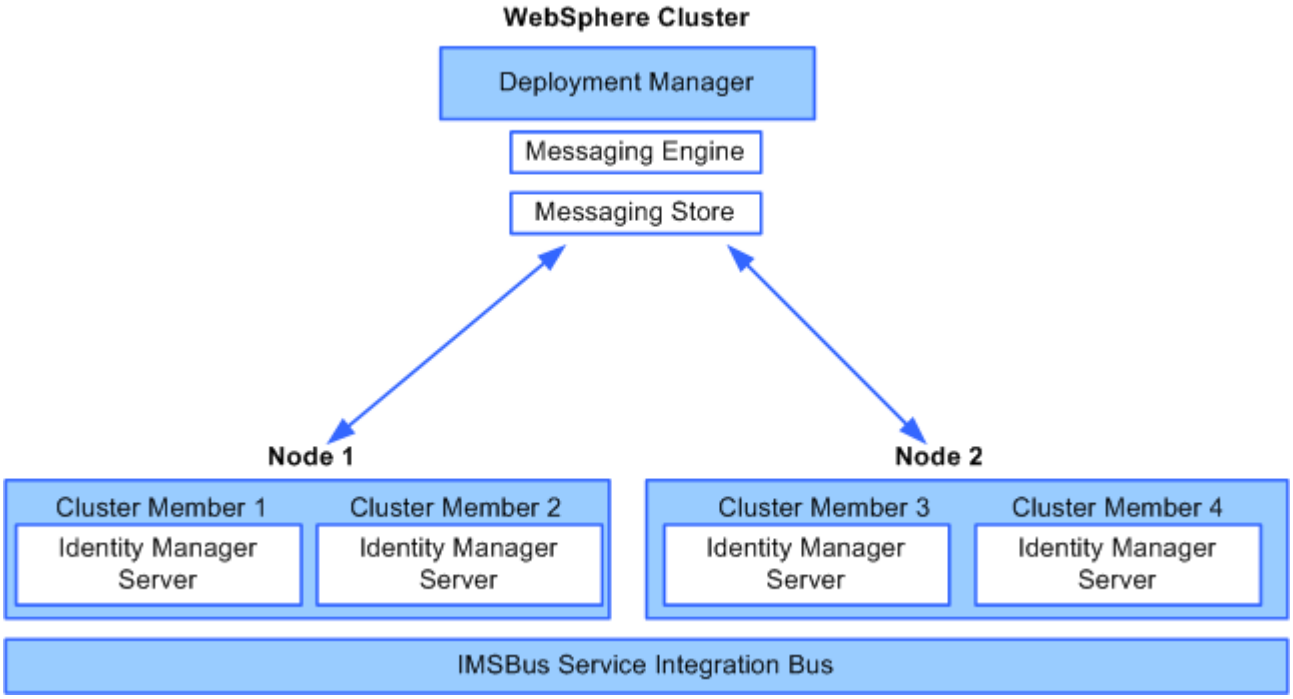
WebSphere Cluster Setup

When you install software for a WebSphere cluster, you set up the following:

- One WebSphere Deployment Manager—Manages the other federated profiles in the cell through node agents.
- One or more nodes—Each node contains one or more cluster members (also called servers), which run the Identity Manager Server.
- Node agent—A process that manages communication between the Deployment Manager and the federated profile.
- Service Integration Bus—Groups resources in WebSphere to simplify administration. The WebSphere cluster is added as a member of the bus.
- Message Engine—Provides messaging functionality for members of the service integration bus. One message engine exists for the cluster.
- Message Store—Stores messages and transaction status for the message engine.
- A Web Server—Distributes the load to the appropriate server and, if SiteMinder is installed, protects access to the cluster members.

The following figure shows the relationship between the Deployment Manager, message engine, message store, nodes, and cluster members. The Identity Manager Server is installed from the Deployment Manager system to each cluster member.

Note: For more information about these components, see the WebSphere v7 System Management and Administration Redbook.



WebSphere Cluster Prerequisites

Before you configure CA Identity Manager on a WebSphere cluster, you should be familiar with the concepts and procedures for creating a WebSphere cluster. See the IBM WebSphere documentation for more information about WebSphere clusters.

Install WebSphere 7 on each Node

On each system that you have used for a cluster member, install WebSphere 7.

To install WebSphere on each Cluster Member system

1. Install the IBM WebSphere Application Server Network Deployment software on each cluster member.
2. Use the Profile Creation Wizard to create a default profile for each node.

   You use this profile to configure a connection to the Deployment Manager.

3. Start each node as follows:
   a. Navigate to *was_home*\WebSphere\AppServer\bin on the system where the managed node is located.
   b. Execute the startNode.bat\.sh command.
4. Confirm that a single cell has all the nodes associated with it at this location:
   was_home/profiles/*Deployment_Manager_Profile*/config/cells/*Cell_Name*/Nodes/
   You should see all federated nodes displayed as folder names.

Creation of profiles may sometimes fail if the bootstrap ports (default: 2809) are not unique. You can check for an error message in the pctLog.txt file in the created profiles' logs folder. For example:

```
(Oct 10, 2007 6:45:55 PM), Install,
com.ibm.ws.install.ni.ismp.actions.ISMPWSProfileLaunchAction, err, INSTCONFFAILED:
Cannot complete required configuration actions after the installation. The
configuration failed. The installation is not successful. Refer to C:\Program
Files\IBM\WebSphere\AppServer\logs\wasprofile\wasprofile_create_CustomIMFromNode.
log for more details.
```

Inspecting the wasprofile_create_CustomIMFromNode.log shows that this failure was due to Bootstrap ports that is not unique.

Create the Cluster with One Member

You now configure the cluster with a single member. The other cluster members are added in a subsequent procedure after you install CA Identity Manager.

To create the cluster with one member

1. In the Administrative Console, verify that the nodes show a Synchronized status.
2. Use the Create New Cluster wizard to create the cluster with one member.
   Note the cluster name and the server node name that you create in using this wizard. The server node is the cluster member node.
3. Stop the cluster member, but leave the Node Agents running.

How to Install CA Identity Manager on a WebSphere Cluster

The following procedures describe how to install CA Identity Manager on a WebSphere cluster.

✓	Step
	1. Run the installation from the deployment manager.
	2. Add cluster members.
	3. Assign the core group policy.
	4. Configure workflow for cluster members.
	5. Configure the proxy plug-in.

Objects Created by the Installation

You install Identity Manager as described in the following procedure. During the installation, the following EARs are installed on the cluster domain:

- iam_im.ear
- ca-stylesr5.1.1.ear

When you supply a cluster name during the installation, these primary resources are configured:

- Distributed queues/topics targeted to the cluster
- Connection factories targeted to the cluster
- Data sources targeted to cluster
- iam_im-IMSBus, the Service Integration Bus for CA Identity Manager
- Message engine store for the cluster
- Core group policies used by the message engine

Run the Installation from the Deployment Manager

Once you have created the WebSphere cluster, you can install CA Identity Manager on it. Installer fields that require a hostname and port number should *not* use localhost.

Note: At previous releases of CA Identity Manager, creating a message store and message engine was a manual process. At this release, you create an empty message store database and supply that database name when you run the CA Identity Manager installer. WebSphere then populates the message store table, creates the message engine, and deploys the CA Identity Manager application ear and binaries to each node in the cluster.

To install CA Identity Manager on the Deployment Manager system

1. Perform these steps if you are using Microsoft SQL server:
   a. Open SQL Management Studio.
   b. Locate the user who owns the message store database.
   c. Set that user's default schema to dbo.
2. Log into the system with the Deployment Manager.
   - On Windows, log in as the Windows Administrator.
   - On UNIX, log in as root.
3. Stop the first cluster member, the only cluster member that you have configured so far.
4. Start the Node Agent for that cluster member.
5. Stop the WebSphere Deployment Manager.
6. On the system that hosts the Deployment Manager, run the CA Identity Manager installation.
   - Windows: From your installation media, run the following program:
     `ca-im-release-win32.exe`
   - UNIX: From your installation media, run the installation program. For example, for Solaris:
     `ca-im-release-sol.bin`

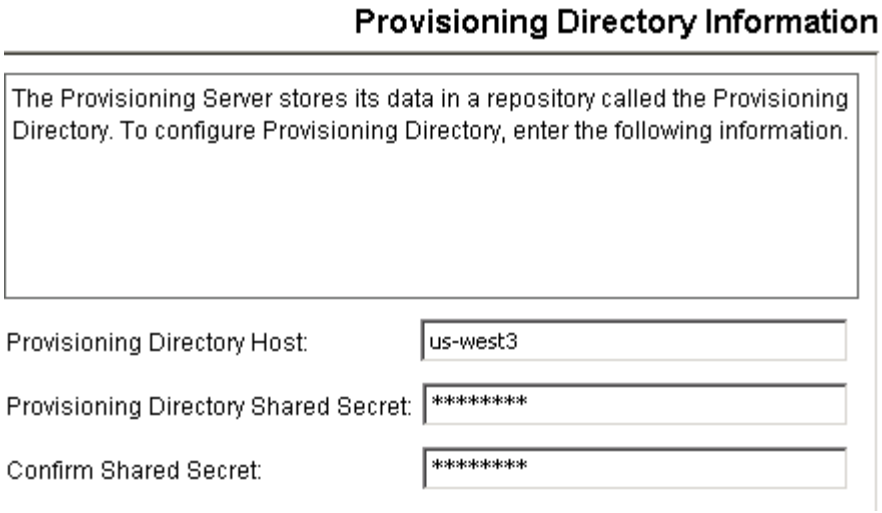
   release represents the current release of CA Identity Manager.

   Important! Be sure that you have the collected the information needed by the installer, such as user names, host names, and ports.

7. Complete the Select Components section by including the Identity Manager Server and any other components that you need on this system.

Note: If you see options to upgrade the workflow database and migrate task persistence data, enable those options. They appear in some scenarios when your previous installation was CA Identity Manager r12.

8. When you enter any password or shared secret in the installation, be sure to provide a password that you can recall when needed.



9. Complete the other sections based on your requirements for the installation.

   The WebSphere section includes these fields:

   WebSphere Install Folder

   The folder or directory where WebSphere is installed. You find this location in the Windows or UNIX file system.

Server Name

The first cluster member in the WebSphere cluster. You find this name in the WebSphere console.

Profile Name

The deployment manager profile. You find this name in the Windows or UNIX file system at the path:

was_home/profiles/*Deployment_Manager_Profile*/config/cells/

Cell Name

The deployment manager's cell which can be found in the WebSphere console.

Node Name

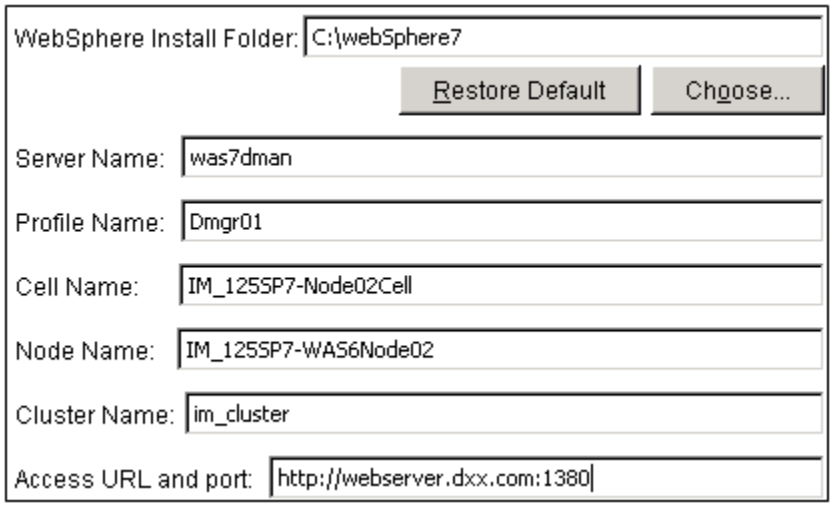
A node that contains the Server Name you supplied on this screen. You find this name in the WebSphere console.

Cluster Name

The name of the cluster. You find this name in the WebSphere console.

Access URL and port

The URL and port number of the Web Server used for load balancing.



10. Complete the Message Store section. The installer creates a JDBC data source as the Message Engine message store based on the following information you provide:

    - Hostname
    - Port
    - Database name

      Enter the message store database.

- Username

  Enter the user who owns the message store database.

- Password
- Schema name

  For Microsoft SQL Server, enter dbo.

  For Oracle, enter the user who owns the message store database.

If any issues occur during installation, inspect the installation logs.

Important! Do not start the cluster yet, as it will not function. Complete the remaining procedures, which conclude with the steps to start the cluster.

Add Cluster Members

You can now add members to the cluster using the first cluster member as a template.

To add cluster members

1. In the Administrative Console for the Deployment Manager, go to Servers, Clusters.
2. Add a cluster member, selecting one of the nodes for which you created a profile.
3. Copy sqljdbc.jar (for Microsoft SQL Server) or ojdbc14.jar (for Oracle) to the cluster member from the deployment manager system.

   On the deployment manager system, the JAR file is in the WAS_INSTALL_ROOT/lib directory. You copy it to the same folder on the system for this cluster member.
4. Repeat this procedure for each cluster member added to the cluster.

Assign the Core Group Policy

To enable high availability and workload management in the cluster, a core group policy now exists for the message engine. This policy, IMSPolicy, defines the preferred cluster member to use for the message engine. If that cluster member fails, the message engine switches to another cluster member, but returns to the preferred cluster member when it becomes available again.

Perform the following procedure once for each cluster member to add cluster members to this policy. For more information about this topic, see Setting up Preferred Servers in the Default Messaging Provider section of the WebSphere v7 System Management and Administration Redbook.

To assign the core group policy to cluster members

1. In the WebSphere Console, locate the IMSpolicy.

   It is under Core Group, Default Core Group, Policies.

2. Select Preferred Servers.

   A list of Core Group Servers appears.

3. Add each cluster member under Preferred Servers.

   Do not select node agents or the Deployment Manager.

   The first cluster member in the list is the one that the messaging engine uses by default. Move the cluster member up or down in the list until they appear in the order in which they should be used.

4. Click OK to save the changes.

Configure Workflow for Cluster Members

From the Deployment Manager system where you installed CA Identity Manager, you configure workflow for each cluster member.

To configure workflow for cluster members

1. Start the WebSphere Console.
2. Navigate to Servers, Server Types, Application Servers, *server_name.*
3. Under Communications, Expand Ports.
4. Make a note of the value for the BOOTSTRAP_ADDRESS port.
5. Edit the workpoint-client.properties file under iam_im.ear/config.

6. Locate the WebSphere section in this file.
7. Replace 2809 (the default port) with the profile's port that is used for the BOOTSTRAP_ADDRESS.
8. Repeat this procedure for each cluster member.
9. Restart the cluster members.

Configure the Proxy Plug-In for the Web Server

You install the proxy plug-in so that WebSphere can communicate with the web server.

To configure the proxy plug-in for the web server

1. See the WebSphere v7 System Management and Administration Redbook for instructions about installing the proxy plug-in for the web server. The chapter on Session Management discusses this plug-in.
2. Restart the Web server to activate the plug-in.
   - For IIS Web Servers—In the master WWW service, be sure that the WebSphere plug-in (sePlugin) appears after the SiteMinder Web Agent plug-in and that the WebSphere plug-in started successfully.
   - For Sun Java System Web Servers—Be sure that the WebSphere plug-in (libns41_http.so) is loaded after the SiteMinder Web Agent plug-in (NSAPIWebAgent.so)

     For Sun Java System 6.0 Web Servers, check the order of plug-ins in <*sun_java_home*>/https-*instance*/config/magnus.conf.

     For Sun Java System 5.x Web Servers, copy the following lines from <*iplanet_home*>/https-*instance*/config/magnus.conf to <*iplanet_home*>/https-*instance*/config/obj.conf

     ```
     Init fn="load-modules" funcs="as_init,as_handler,as_term"
     shlib="/export/WebSphere/AppServer/bin/libns41_http.so"
     Init fn="as_init"
     bootstrap.properties="/export/WebSphere/AppServer/config/cells/plugin-cfg
     .xml"
     ```

     Add the following after AuthTrans fn="SiteMinderAgent" in the obj.conf file:

     ```
     Service fn="as_handler"
     ```
   - For Apache Web Servers— In the Dynamic Shared Object (DSO) Support section of *Apache_home*/config/httpd.conf, be sure that the SiteMinder Web Agent plug-in (mod2_sm.so)is loaded before the WebSphere plug-in (mod_ibm_app_server_http.so).

Start the WebSphere Cluster

To start the WebSphere cluster, you start the Deployment Manager and then start each managed node.

To start the WebSphere cluster

1. Start a Policy Server that supports CA Identity Manager.

   Note: If you have a Policy Server cluster, only one Policy Server should be running while you create Identity Manager directories, create or modify Identity Manager environments, or change WorkPoint settings.

2. Run the Deployment Manager.

3. On the first managed node, complete the following steps:

   a. Navigate to *was_home*\WebSphere\AppServer\bin.

   b. Execute the startNode.bat\.sh command.

      The first managed node starts.

4. Repeat step 3 on each node in the cluster.

5. Start each cluster member in Servers, Clusters, *cluster_name*, Cluster Members in the WebSphere Administrative Console on the Deployment Manager.

6. Verify that the messaging engine for the cluster is running in Service integration, Buses, iam_im-IMSBus, Messaging Engines in the WebSphere Admin Console on the Deployment Manager.

7. If you have installed a SiteMinder Web Agent, start the Web Server where you installed the SiteMinder Web Agent and the application server proxy plug-in.

Verify the Clustered Installation

When you have completed all steps and started the cluster, check that the installation was successful.

To verify the clustered installation

1. Start the databases used by the Identity Manager server.

2. Start any extra Policy Servers and CA Identity Manager nodes that you stopped.

3. Access the Management Console and confirm the following:
   - You can access the following URL from a browser:

     http://*im_server*:*port*/iam/immanage

     For example:

     http://MyServer.MyCompany.com:*port-number/*iam/immanage
   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.
4. Verify that you can access an upgraded environment using this URL format:

   http://*im_server*:*port*/iam/im/*environment*

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you configure communication to the server.

Note: To install the Provisioning Manager, install the Identity Manager Administrative Tools on a Windows system.

To configure a remote Provisioning Manager

1. Log into the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA, Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.
4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.
6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP file includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you must install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Vista Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

The Identity Manager installer installs all connectors by default. However, in some cases, install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA Option for Password Reset/Unlock (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

Chapter 5: Separate Database Configuration

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
X	**3. (Optional) Create separate databases.**
	4. (Optional) Install the Report Server.
	5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Create Separate Databases

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, for scalability purposes, you may want to create a separate database to replace any one of the existing database schemas initially created by CA Identity Manager during installation.

You can create a database instance for the following:

- Workflow
- Auditing
- Task Persistence
- Object Store
- Snapshots (reporting)
- Archive (task persistence archive)

Important! The Windows default locations for CA Identity Manager database schema files are the following:

- Workflow: See the section, Run the CreateDatabase script.
- Auditing: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Task Persistence: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Object Store: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Snapshots (reporting): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\imrexport\tools\db
- Archive (task persistence archive): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db

How to Create Separate Databases

To create separate databases for CA Identity Manager:

✓	Step
	1. Create a Microsoft SQL Server or Oracle database instance for CA Identity Manager.
	2. Create JDBC resources.
	3. Edit the data source.
	4. Set connection pool properties.
	5. (Optional) Run the SQL scripts.

Create an MS SQL Server Database Instance

To create an Microsoft SQL Server Database Instance

1. Create a database instance in SQL server.
2. Create a user and grant this user the necessary rights (such as public and db_owner rights) to the database by editing the properties of the user.

   Note: The user must have at least select, insert, update, and delete permissions for all of the tables created by the .sql script for creating the database, and must be able to execute all of the stored procedures (if applicable) defined in these scripts. For example, the user must have these permissions on the tables defined in the following default location:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
3. While editing the user's properties, set the database you just created as the default database for the user.
4. Ensure the Authentication setting has a value of SQL Server on the Security tab of the SQL Server Properties dialog for the server where the database is installed.

Note: For complete information about Microsoft SQL Server, see your Microsoft SQL Server documentation.

Create an Oracle Database Instance

To create an Oracle Database Instance

1. Create a new tablespace.
2. Create a new user.
3. Grant the user rights to the new database.
   - Create/alter/drop tables
   - Create/alter/drop view
   - Create/alter/drop INDEX
   - Create/replace/drop stored procedures
   - Create/replace/drop functions
   - Create/drop sequence
   - Create/replace/drop triggers
   - Create/replace/drop types
   - Insert/select/delete records
   - CREATE SESSION / connect to database
4. Give DBA rights to the user.

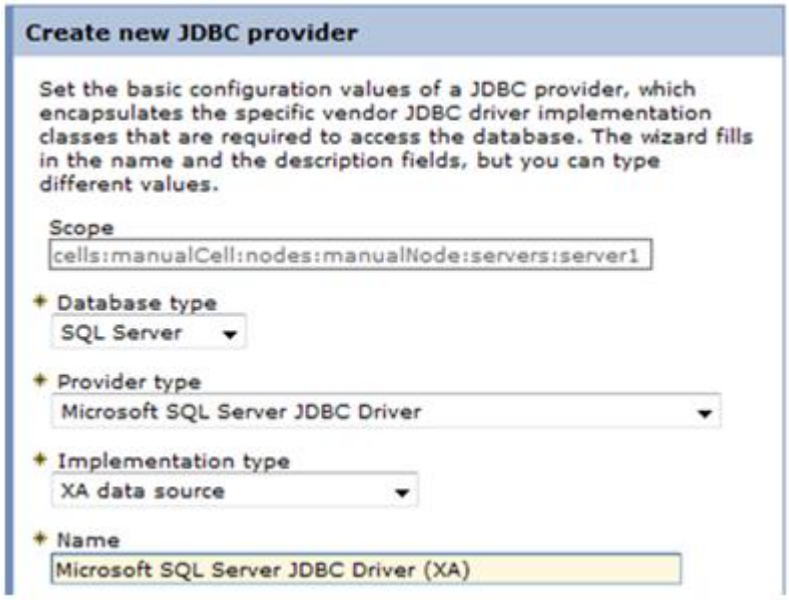
Note: For complete information about Oracle, see your Oracle documentation.

Create JDBC Resources

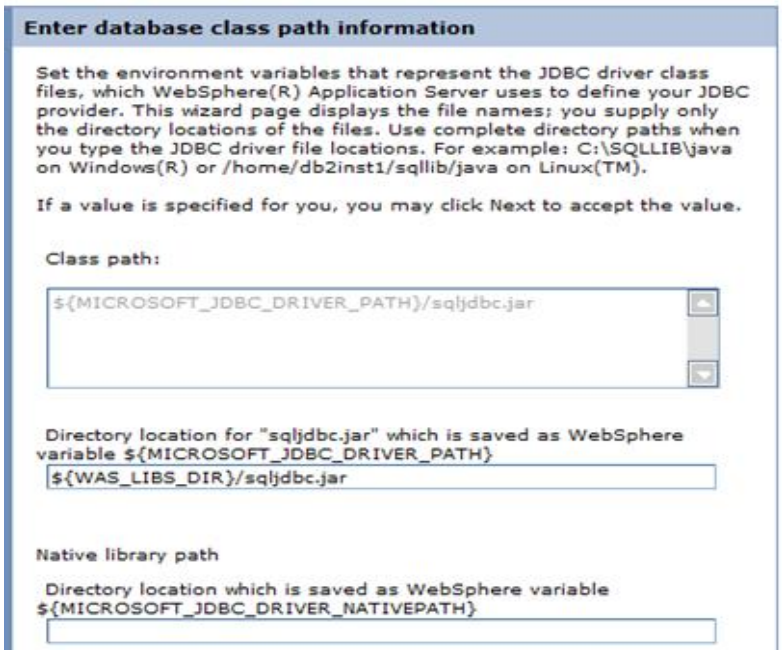
Follow these steps:

1. In the WebSphere Administrative Console, click Resources, JDBC, JDBC Providers.
2. For Scope, select Node=manualNode, Server=*server-name*.
3. Click New.

4. Complete the Create New JDBC provider page with your choices for your database. The following example shows Microsoft SQL Server as the JDBC provider.



5. Fill in the database class page information. The directory location for Microsoft SQL Server appears in the following example.

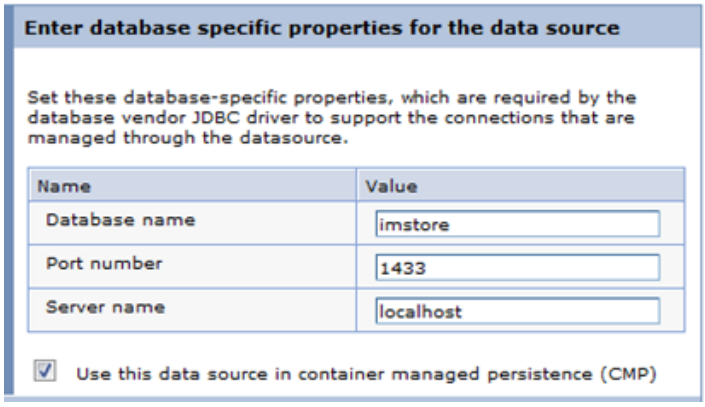


6. Validate the Summary page and click Finish.

Edit the Data Source

Follow these steps:

1. Within the WebSphere Administrative Console, click Resources, JDBC, Data sources.
2. For Scope, select Node=manualNode, Server=*server- name*.
3. Click New to create the data source as follows:
   - For Data source name, enter iam_im Object Store Data Source
   - For JNDI name, enter iam/im/jdbc/jdbc/objectstore
4. Select the JDBC provider.
5. Enter the database specific properties for your environment.

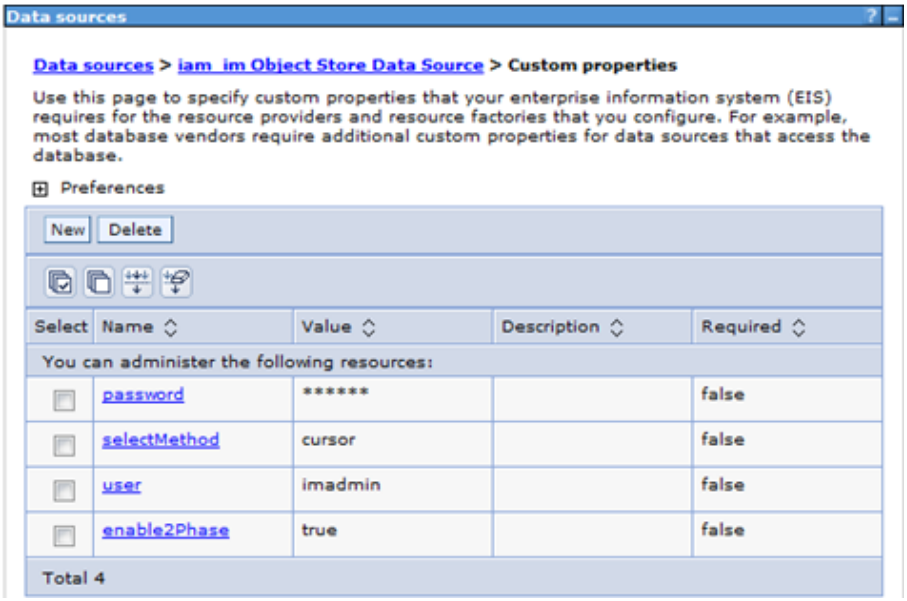


6. For Setup security aliases, accept the defaults.
7. On the Summary page, click Finish.
8. Save changes directly to the master configuration.
9. Add custom properties to the Data Source using the following steps:
   a. From the Data sources page, select iam_im Object Store Data Source.
   b. Under Additional Properties, select Custom properties.

c. Depending on your database, add the following properties

- **SQL:** user=<username>, password=<password>, enable2Phase=true, selectMethod=cursor
- **Oracle**: user=<username>, password=<password>

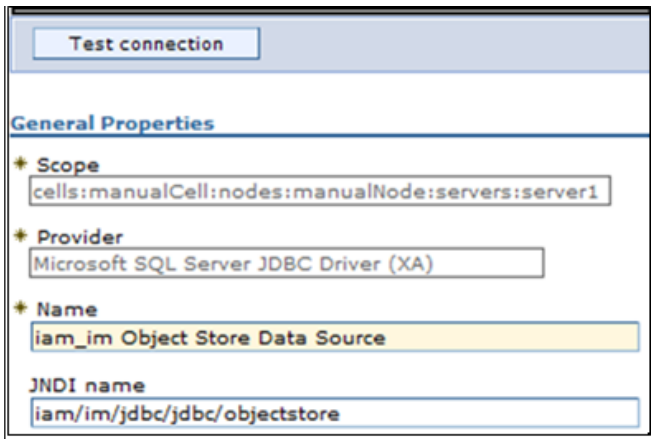
Note: Be sure that the JDBC provider is created as XA.



10. Save changes directly to the master configuration

    The database schema (SQL scripts) are automatically applied when you restart CA Identity Manager.

11. Test the data source connection.



Failures are typically classpath or credentials related. If the test connection passed, then the Data Source configuration is complete.

12. Four additional data sources need to be configured. Repeat this procedure, but use the data sources and JNDI names in the following table:

Data Source	JNDI Name
iam_im Task Persistence Data Source	iam/im/jdbc/jdbc/idm
iam_im Workflow Data Source	iam/im/jdbc/jdbc/WPDS
iam_im Snapshots Data Source	iam/im/jdbc/jdbc/reportsnapshot
iam_im Archive Data Source	iam/im/jdbc/jdbc/archive

Set Connection Pool Properties

The default connection pool values need to be edited for all data sources to ensure proper performance. Set the connection pool properties as follows:

- Connection timeout: 10
- Maximum connections: 200
- Minimum connections: 5
- Reap time: 150
- Unused timeout: 300
- Aged timeout: 300
- Purge policy: FailingConnectionOnly

Run the SQL Scripts

SQL scripts are automatically run against the databases when CA Identity Manager starts, however if you want to run the SQL scripts yourself, perform the following steps before restarting the application server:

These scripts are installed with the Identity Manager Administrative Tools.

To run the SQL scripts

1. Do one of the following:
   - Microsoft SQL Server: Open the Query Analyzer tool and select the script you need.
   - Oracle: Open the SQL prompt for the script you need.
2. Select one of the following scripts (shown with the default Windows locations) depending on what the database was created for:
   - Task Persistence:
     - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
     - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\oracle9i\idm_db_oracle.sql
     - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/taskpersistence/oracle9i/idm_db_oracle.sql
   - Auditing:
     - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\sqlserver\ims_mssql_audit.sql
     - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\oracle\ims_oracle_audit.sql
     - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/auditing/oracle/ims_oracle_audit.sql
   - Snapshots:
     - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
     - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql
     - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/imrexport/db/oracle/ims_oracle_report.sql
   - Workflow: See the Section "Run the SQL Scripts for Workflow."
3. Run the script file.
4. Verify that no errors appeared when you ran the script.

Run the Script for Workflow

CA Identity Manager includes SQL scripts for setting up a new workflow database instance.

To run the CreateDatabase script

1. Add the path to the sqljdbc.jar to the DB_CLASSPATH attribute in the CreateDatabase.bat or .sh script before you run it.
2. From a command prompt, run CreateDatabase.bat or sh. The default location for this script is:

   Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\Workpoint\install.

   UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/Workpoint/install.

   A command prompt window and the WorkPoint application open.
3. Select the database type from the drop-down.
4. Use the following guidelines to fill in fields in the configuration utility:
   - For the JDBC Class parameter, enter:

     Oracle: oracle.jdbc.driver.OracleDriver

     SQL Server: com.microsoft.sqlserver.jdbc.SQLServerDriver
   - For the JDBC URL, enter:

     Oracle: jdbc:oracle:thin:@*wf_db_system*:1521:*wf_oracle_SID*

     SQL Server: jdbc:sqlserver://*wf_db_system*:1433; databaseName=*wf_db_name*
   - For the Database User ID parameter, enter the workflow user you created when creating the workflow database.
   - For the Password parameter, enter the password you created for the workflow user.
   - For the Database ID, enter WPDS
5. Accept the default check box selections.
6. Click the Initialize button.

   When the configuration is complete, a message that resembles the following appears in the Command Prompt window:

   ```
   The create database process finished with 0 errors.
   ```
7. Restart the application server.

Chapter 6: Manual EAR Deployment

This section contains the following topics:

How to Deploy Manually

To manually deploy CA Identity Manager r12.5 SP9 on WebSphere 7, you perform the following steps, which are explained in this chapter.

✓	Step
	1. Review the prerequisites.
	2. Create the primary resources.
	3. Assign the core group policy if you have a cluster.
	4. Generate the EAR files.
	5. Deploy the ca-styles5.1.1.ear.
	6. Deploy the iam_im.ear.
	7. Create Policy Server and workflow objects. (Required for all installations.)
	8. Create Message Driven Bean listener bindings.
	9. Edit the user_console.war.
	10. Connect to SiteMinder, if it is installed.

✓	Step
	11. Connect to RCM, if it is installed.
	12. Create a Provisioning Server shared secret.
	13. Perform post-deployment steps for the cluster.

Prerequisites to Manual Deployment

Review the following prerequisites before manually deploying CA Identity Manager r12.5 SP9.

- Use the Create databases (see page 65) procedure to create the required JDBC resources, edit data sources, and set connection pool properties.
- Verify that you have met the WebSphere prerequisites (see page 27).
- Create a WebSphere cluster (see page 48) if you need high availability for the Identity Manager server.

Create the Primary Resources

To create the JMS resources and the service integration bus, run a JACL script located in the WebSphere-tools folder. Based on your situation, use the single node or cluster procedure that follows.

To create the primary resources for a single node

1. Open a command line and move to the following location:
   `websphere_home/profiles/profile_name/bin`
2. Run the imssetup.jacl as follows:
   `wsadmin -f websphere_tools/imssetup.jacl myNodeName myServerName`
3. To validate the resources were created, review the resource settings from the Webphere Administrative Console. Specifically:
   a. Check under Service Integration, Buses.

b. Check under Resources, JMS for the following items:

- Queue connection factories
- Topic connection factories
- Queues
- Topics
- Activation specifications

Each CA Identity Manager resource begins with an iam prefix.

To create the primary resources for a cluster

1. Copy the CreateCoreGroupPolicy.jacl from WAS_ROOT/bin to the Deployment Manager profile/bin folder.
2. Edit the IMSCoreGroupPolicy.properties for the following variables:
   - $WAS_CLUSTER$ - The cluster name. The entire string that contains this variable corresponds to the messaging engine name.
   - WAS_NODE - The node where the cluster member is created; it can be different from Deployment Manager Node name.
   - WAS_SERVER - The name of the cluster member.
3. Open a command line and move to the following location:
   `websphere_home/profiles/profile_name/bin`
4. Run the imsSetupCluster.jacl as follows:

   wsadmin -f *websphere_tools*/imsSetupCluster.jacl *NodeName ClusterMemberName ClusterName SchemaName*

   Note: SchemaName parameter is a string passed to imsSetupCluster.jacl to specify the schema name for the messaging engine associated with each cluster member. This string can be changed later.
5. To validate the resources were created, review the resource settings from the Webphere Administrative Console. Specifically:

   a. Check under Service Integration, Buses.

   b. Check under Resources, JMS for the following items:

   - Queue connection factories
   - Topic connection factories
   - Queues
   - Topics
   - Activation specifications

   Each CA Identity Manager resource begins with an iam prefix.

Assign the Core Group Policy

To enable high availability and workload management in the cluster, a core group policy now exists for the message engine. This policy, IMSPolicy, defines the preferred cluster member to use for the message engine. If that cluster member fails, the message engine switches to another cluster member, but returns to the preferred cluster member when it becomes available again.

Perform the following procedure once for each cluster member to add cluster members to this policy. For more information about this topic, see Setting up Preferred Servers in the Default Messaging Provider section of the WebSphere v7 System Management and Administration Redbook.

To assign the core group policy to cluster members

1. In the WebSphere Console, locate the IMSpolicy.

   It is under Core Group, Default Core Group, Policies.

2. Select Preferred Servers.

   A list of Core Group Servers appears.

3. Add each cluster member under Preferred Servers.

   Do not select node agents or the Deployment Manager.

   The first cluster member in the list is the one that the messaging engine uses by default. Move the cluster member up or down in the list until they appear in the order in which they should be used.

4. Click OK to save the changes.

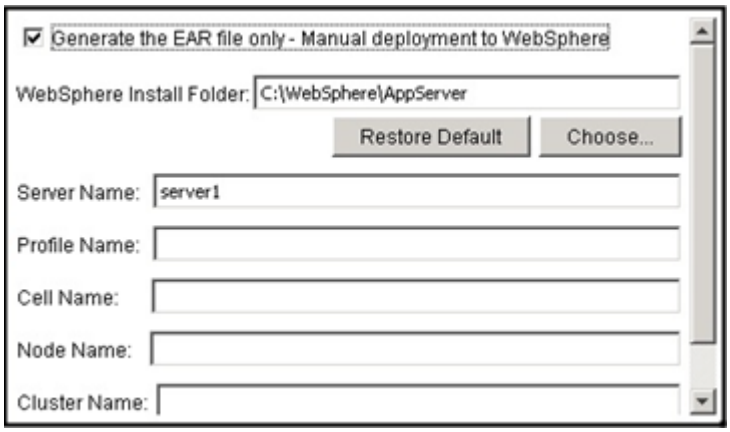
Generate the EAR Files

Follow these steps:

Run the CA Identity Manager installer and select the Generate EAR file only option. The installer creates the following EAR files:

- WAS_IMR12.ear: This is the compressed iam_im.ear (the EAR file of the CA Identity Manager application).
- WAS_caStyles.ear: This is the compressed castyles5.1.1.ear (the EAR file for the CA Identity Manager style sheets).

These files are installed in the location that you specify during installation.



The installation also creates the following folders:

- *install_location*\IAM Suite\WebSphere-ear - Contains EAR files and exploded backup files
- *install_location*\IAM Suite\WebSphere-tools - Contains JACL scripts and other tools

Deploy the castylesr5.1.1.ear File

After generating the EAR files, you start by deploying the ca-styles5.1.1.ear.

Follow these steps:

1. In the WebSphere Administrative Console, click Applications, New Application, New Enterprise Application.
2. Supply the location of the ca-stylesr5.1.ear file.

3. Keep all default settings.
4. Under Select installation options, select the following options:
   - Distribute application
   - Create MBeans for resources
5. On the Map modules to servers page:
   - Verify that the cell and server name are listed.
   - Select the Module CA Styles r5.1.1.
6. On the Map virtual hosts for Web modules page, select Web module CA Styles R5.1.1.
7. Select default_host under Virtual host column.
8. Click Next, then Finish.

   The application is installed.
9. Save directly to the master configuration.
10. Click Applications, Application Types, WebSphere enterprise applications.
11. Select castyles5.1.1 and click Start.
12. Verify that the status field changed to Started.

Deploy the iam_im.ear

Two options exist for deploying the iam_im.ear to WebSphere 7. You can use a JACL script or you can use the WebSphere Administrative Console.

Deploy the iam_im.ear with a JACL Script

The simplest method to deploy the iam_im.ear is to use a JACL script.

Follow these steps:

To deploy the iam_im.ear with JACLs, perform the following steps.

1. Copy the compressed iam_im.ear file to the following directory.

   websphere_home/profiles/*profile_name*/bin
2. Open a command line and move to the preceding directory.
3. Run one of the following commands:

   Single Node: wsadmin -f *WebSphere-tools*/imsinstall.jacl *path_to_EAR*

   Cluster: wsadmin -f *WebSphere-tools*/imsinstall.jacl *path_to_EAR ClusterName*

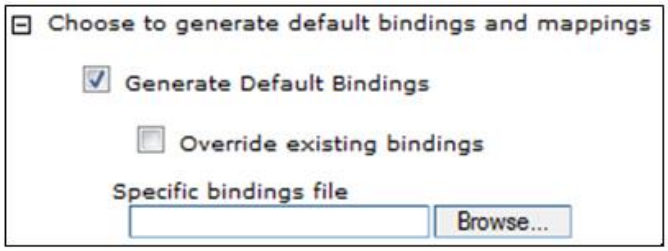
Important! If this procedure succeeds, continue to Edit the user_console.war (see page 84). If this procedure fails, use the procedure, Deploy the iam_im.ear from the WebSphere Administrative Console (see page 79).

Deploy the iam_im.ear from the WebSphere Administrative Console

If deploying the iam_im.ear using a JACL script did not work, use this procedure instead.

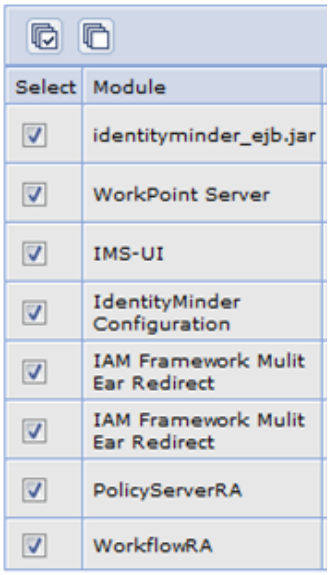
Follow these steps:

1. Log into the WebSphere Administrative Console.
2. Click Applications, New Applications, New Enterprise Application.
3. Click Install.
4. Supply the location of the EAR file that you generated.
5. Fill in the dialog as follows:
   a. Select Fast Path.
   b. Expand the Choose to generate default bindings and mappings.
   c. Select Generate Default Bindings.



6. On the Installation Options page, no changes are needed.

7. On the Map modules to servers page:
   a. Fill in the Clusters or Servers field as appropriate.
   b. Select all modules.



8. On the Map virtual hosts to Web modules page, select all modules.
9. Verify the Summary page appears similar to the following example:

Options	Values
Precompile JavaServer Pages files	No
Directory to install application	
Distribute application	Yes
Use Binary Configuration	No
Deploy enterprise beans	Yes
Application name	iam_im
Create MBeans for resources	Yes
Override class reloading settings for Web and EJB modules	No
Reload interval in seconds	
Deploy Web services	No
Validate Input off/warn/fail	warn
Process embedded configuration	No
File Permission	.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID	Unknown
Allow dispatching includes to remote resources	No
Allow servicing includes from remote resources	No
Business level application name	
Asynchronous Request Dispatch Type	Disabled
Allow EJB reference targets to resolve automatically	No
Cell/Node/Server	Click here

10. Click Finish to deploy the ear.

    Note: This step takes some time to complete.
11. Save the installation directly to the master configuration.

Create Policy Server and Workflow Objects

If you succeeded in deploying the iam_im.ear by using the JACL script, you can omit this procedure.

Follow these steps:

1. In the WebSphere Administrative Console, click Servers, Server Types, WebSphere application servers.
2. On the Application servers page, click the *server-name*.
3. Under Applications, click Installed applications.
4. Click iam_im on the page that appears.
5. Under Modules, click Manage Modules.
6. Click PolicyServerRA in the list of modules.

7. Under Additional Properties, click Resource Adapter.
8. Under Additional Properties, click J2C connection factories.
9. Click New to create the object with the following values:

   Name: iam_im-PolicyServerConnection

   JNDI Name: iam/im/rar/nete/rar/PolicyServerConnection

10. In the messages box at the top of the screen, save directly to master configuration.
11. Follow these steps to create the Workflow connector object:
    a. Return to the Manage Modules page.

       To go to that page, repeat steps 1 through 5 or click Manage Modules in the breadcrumbs.

    b. Click WorkflowRA.

    c. Under Additional Properties, click Resource Adapter.

    d. Under Additional Properties, click J2C connection factories.

    e. Create the Workflow connector object with these values:

       Name: iam_im-Workflow

       JNDI Name: iam/im/rar/Workflow

Create Message Driven Bean Listener Bindings

If you succeeded in deploying the iam_im.ear by using the JACL script, you can omit this procedure.

Follow these steps:

1. In the WebSphere Administrative Console, go to Applications, Application Types, WebSphere Enterprise Applications.
2. Click iam_im.
3. Under Enterprise Java Bean Properties, select Message Drive Bean listener bindings.
4. Under Listener bindings, click Activation Specification.
5. For the identityminder_ejb.jar, fill in these values for each EJB module:

EJB	Listener Bindings
SubscriberMessageEJB	Target Resource JNDI Name: iam/im/ACT Destination JNDI name: iam/im/jms/queue/com.netegrity.ims.msg.queue
ServerCommandsEJB	Target Resource JNDI Name: iam/im/ServerCommand Destination JNDI name: iam/im/jms/topic/topic/ServerCommandTopic

EJB	Listener Bindings
RuntimeStatusDetailEJB	Target Resource JNDI Name: iam/im/jms/RuntimeStatusDetailQueue Destination JNDI name iam/im/jms/queue/queue/RuntimeStatusDetailQueue

6. For the WorkPoint Server, fill in these values for each EJB module:

EJB	Listener Bindings
ServerAutomatedActivityMDBean	Target Resource JNDI Name: iam/im/jms/wpServAutoActActSpec Destination JNDI name: iam/im/jms/queue/queue/wpServAutoActQueue
EventMDBean	Target Resource JNDI Name: iam/im/jms/wpEventActSpec Destination JNDI name: iam/im/jms/queue/queue/wpEventQueue
UtilityMDBean	Target Resource JNDI Name: iam/im/jms/wpUtilActSpec Destination JNDI name: iam/im/jms/queue/queue/wpUtilQueue

7. Click OK.
8. Save your changes directly to the master configuration.
9. Restart WebSphere.

Edit the user_console.war

Use this procedure to reset the class loader order in the user_console.war file.

Follow these steps:

1. Click Application, WebSphere enterprise applications, iam_im.
2. Under Modules, click Manage modules.
3. Click IMS-UI.
4. Set the class loader order to the following choice:

   Classes loaded with local class loader first (parent last)

5. Save directly to the master configuration
6. Restart the WebSphere application server.

Connect to SiteMinder

To connect to a SiteMinder Policy Server, perform the following steps. For a cluster, perform these steps on each cluster member.

Follow these steps:

1. On the WebSphere application server system, navigate to *was_home*/bin/.
2. Edit the startServer.sh file. Add the following path to the SMPS variable under the Start CA IAM Suite section:

   was_home/profiles/*profile_name*/installedApps/*profile_name*/iam_im.ear/library
3. Start the WebSphere application server.
4. In the WebSphere Administrative Console, go to Application servers, *your_server*, Install Applications, IdentityMinder, Manage Modules, policyserver.rar, IdentityMingerPolicyServerRA, J2C connection factories.
5. Click on the object with the following JNDI name:

   nete/rar/PolicyServerConnection
6. Click on Custom Properties.
7. Set the following properties:
   - Enabled = true
   - Set ConnectionUrl = hostname of the SiteMinder system
   - Username = SiteMinder administrative user
   - AdminSecret = SiteMinder administrative user password
   - AgentName = SiteMinder Agent name
   - AgentSecret = SiteMinder Agent secret
8. In the SiteMinder Administrative UI, create an Agent configuration object for the Agent protecting your WebSphere resources.

   Note: For more details on creating an Agent configuration, see the *SiteMinder Policy Server Configuration Guide*.
9. Navigate to the following location:

   was_home\config\cells*cellname*\applications\iam_im.ear\deployments\IdentityMinder\user_console.war\WEB-INF

10. Edit the web.xml file and set enabled=false for the AgentFilter and the FrameworkAuthFilter. For example:

```
   <filter-name>AgentFilter</filter-name>
     <filter-class>com.netegrity.proxy.AgentFilter</filter-class>
        <init-param>
            <param-name>EnableAgent</param-name>
            <param-value>false</param-value>
        </init-param>
 <filter-name>FrameworkAuthFilter</filter-name>

    <filter-class>com.netegrity.webapp.authentication.FrameworkLoginFilter</f
ilter-class>
        <init-param>
            <param-name>Enable</param-name>
            <param-value>false</param-value>
        </init-param>
```

11. Run the CA Identity Manager installer on the SiteMinder system and install the Extensions for SiteMinder.

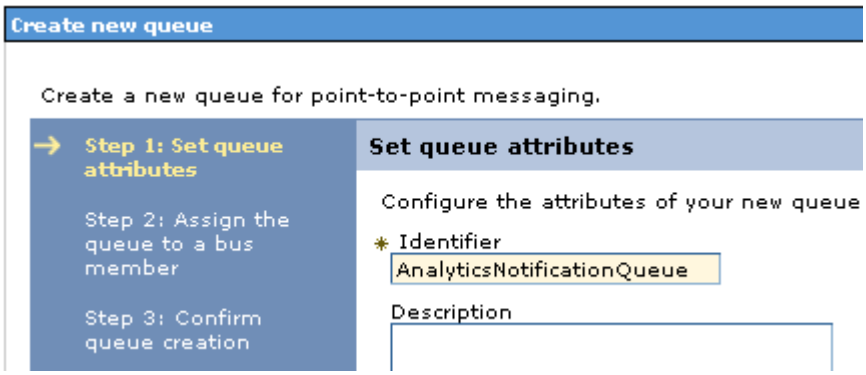
Connect to RCM

If you have Role and Compliance Manager (RCM) in your installation, configure WebSphere to connect to RCM.

Follow these steps:

1. Log in to the WebSphere Administrative Console.
2. Create a queue as a bus destination as follows:
   a. Click Service Integration, Buses.
   b. Click iam_im-IMSBus.
   c. Under Destination Resources, click Destinations.
   d. Click New.
   e. Click Queue.

f. For Identifier, enter AnalyticsNotificationQueue.



3. Create a JMS queue as follows:
   a. Click Resources, JMS, Queues.
   b. Click New.
   c. Click Default messaging provider.
   d. Supply these values for the following fields:

      Name

      AnalyticsNotificationQueue

      JNDI Name

      iam/im/jms/queue/analytics/AnalyticsNotificationQueue

      Bus Name

      IMSBus

      Queue Name

      AnalyticsNotificationQueue

4. Create an activation specification for the queue as follows:
   a. Click Resources, JMS, Activation Specifications.
   b. Click New.
   c. Click Default messaging provider.
   d. Supply these values:

      Name

      AnalyticsNotificationQueueActSpec.

      JNDI Name

      iam/im/jms/analytics/AnalyticsNotificationQueue/ActSpec

Destination JNDI name

iam/im/jms/queue/analytics/AnalyticsNotificationQueue

This name must match the JNDI Name created in Step 3.

Create a Provisioning Server Shared Secret

You need to create a shared secret to communicate with the Identity Manager server.

Follow these steps:

1. Generate an encrypted shared secret using the Password Tool.
2. Update the Provisioning Server shared secret in the systemWideProperties.properties file.

Perform Post-Deployment Steps for the Cluster

If you are performing manual EAR deployment to a cluster, perform the following procedures that apply to deployment on a cluster.

Add Cluster Members

You can now add members to the cluster using the first cluster member as a template.

To add cluster members

1. In the Administrative Console for the Deployment Manager, go to Servers, Clusters.
2. Add a cluster member, selecting one of the nodes for which you created a profile.
3. Copy sqljdbc.jar (for Microsoft SQL Server) or ojdbc14.jar (for Oracle) to the cluster member from the deployment manager system.

   On the deployment manager system, the JAR file is in the WAS_INSTALL_ROOT/lib directory. You copy it to the same folder on the system for this cluster member.
4. Repeat this procedure for each cluster member added to the cluster.

Assign the Core Group Policy

To enable high availability and workload management in the cluster, a core group policy now exists for the message engine. This policy, IMSPolicy, defines the preferred cluster member to use for the message engine. If that cluster member fails, the message engine switches to another cluster member, but returns to the preferred cluster member when it becomes available again.

Perform the following procedure once for each cluster member to add cluster members to this policy. For more information about this topic, see Setting up Preferred Servers in the Default Messaging Provider section of the WebSphere v7 System Management and Administration Redbook.

To assign the core group policy to cluster members

1. In the WebSphere Console, locate the IMSpolicy.

   It is under Core Group, Default Core Group, Policies.

2. Select Preferred Servers.

   A list of Core Group Servers appears.

3. Add each cluster member under Preferred Servers.

   Do not select node agents or the Deployment Manager.

   The first cluster member in the list is the one that the messaging engine uses by default. Move the cluster member up or down in the list until they appear in the order in which they should be used.

4. Click OK to save the changes.

Configure Workflow for Cluster Members

From the Deployment Manager system where you installed CA Identity Manager, you configure workflow for each cluster member.

To configure workflow for cluster members

1. Start the WebSphere Console.
2. Navigate to Servers, Server Types, Application Servers, *server_name.*
3. Under Communications, Expand Ports.
4. Make a note of the value for the BOOTSTRAP_ADDRESS port.
5. Edit the workpoint-client.properties file under iam_im.ear/config.

6. Locate the WebSphere section in this file.
7. Replace 2809 (the default port) with the profile's port that is used for the BOOTSTRAP_ADDRESS.

8. Repeat this procedure for each cluster member.
9. Restart the cluster members.

Configure the Proxy Plug-In for the Web Server

You install the proxy plug-in so that WebSphere can communicate with the web server.

To configure the proxy plug-in for the web server

1. See the WebSphere v7 System Management and Administration Redbook for instructions about installing the proxy plug-in for the web server. The chapter on Session Management discusses this plug-in.
2. Restart the Web server to activate the plug-in.
   - For IIS Web Servers—In the master WWW service, be sure that the WebSphere plug-in (sePlugin) appears after the SiteMinder Web Agent plug-in and that the WebSphere plug-in started successfully.
   - For Sun Java System Web Servers—Be sure that the WebSphere plug-in (libns41_http.so) is loaded after the SiteMinder Web Agent plug-in (NSAPIWebAgent.so)

     For Sun Java System 6.0 Web Servers, check the order of plug-ins in <*sun_java_home*>/https-*instance*/config/magnus.conf.

     For Sun Java System 5.x Web Servers, copy the following lines from <*iplanet_home*>/https-*instance*/config/magnus.conf to <*iplanet_home*>/https-*instance*/config/obj.conf

     ```
     Init fn="load-modules" funcs="as_init,as_handler,as_term"
     shlib="/export/WebSphere/AppServer/bin/libns41_http.so"
     Init fn="as_init"
     bootstrap.properties="/export/WebSphere/AppServer/config/cells/plugin-cfg
     .xml"
     ```

     Add the following after AuthTrans fn="SiteMinderAgent" in the obj.conf file:

     ```
     Service fn="as_handler"
     ```
   - For Apache Web Servers— In the Dynamic Shared Object (DSO) Support section of *Apache_home*/config/httpd.conf, be sure that the SiteMinder Web Agent plug-in (mod2_sm.so)is loaded before the WebSphere plug-in (mod_ibm_app_server_http.so).

Start the WebSphere Cluster

To start the WebSphere cluster, you start the Deployment Manager and then start each managed node.

To start the WebSphere cluster

1. Start a Policy Server that supports CA Identity Manager.

   Note: If you have a Policy Server cluster, only one Policy Server should be running while you create Identity Manager directories, create or modify Identity Manager environments, or change WorkPoint settings.

2. Run the Deployment Manager.

3. On the first managed node, complete the following steps:

   a. Navigate to *was_home*\WebSphere\AppServer\bin.

   b. Execute the startNode.bat\.sh command.

      The first managed node starts.

4. Repeat step 3 on each node in the cluster.

5. Start each cluster member in Servers, Clusters, *cluster_name*, Cluster Members in the WebSphere Administrative Console on the Deployment Manager.

6. Verify that the messaging engine for the cluster is running in Service integration, Buses, iam_im-IMSBus, Messaging Engines in the WebSphere Admin Console on the Deployment Manager.

7. If you have installed a SiteMinder Web Agent, start the Web Server where you installed the SiteMinder Web Agent and the application server proxy plug-in.

Verify the Clustered Installation

When you have completed all steps and started the cluster, check that the installation was successful.

To verify the clustered installation

1. Start the databases used by the Identity Manager server.

2. Start any extra Policy Servers and CA Identity Manager nodes that you stopped.

3. Access the Management Console and confirm the following:
   - You can access the following URL from a browser:

     http://*im_server*:*port*/iam/immanage

     For example:

     http://MyServer.MyCompany.com:*port-number/*iam/immanage
   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.
4. Verify that you can access an upgraded environment using this URL format:

   http://*im_server*:*port*/iam/im/*environment*

Chapter 7: Report Server Installation

This section contains the following topics:

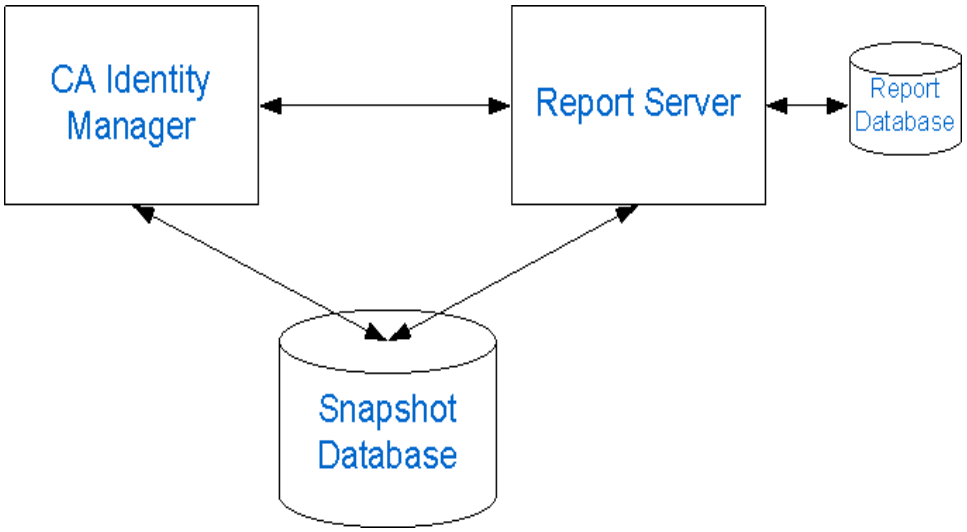
Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
X	**4. (Optional) Install the Report Server.**
	5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Reporting Architecture

In CA Identity Manager, the reporting setup requires the three major components in the following diagram:



Note: The Snapshot Database in this illustration graphic could also be the Audit Database or Workflow Database.

Report Server

Also known as CA Business Intelligence, this server generates reports, communicating directly with CA Identity Manager and the Snapshot Database.

Report Database

The database where the CA Report Server (Business Objects) stores its own data.

CA Identity Manager

CA Identity Manager allows you to export CA Identity Manager object data to the Report Database.

Snapshot Database

A separate database containing the snapshot data of objects in CA Identity Manager

Important! The Report Server uses Business Objects Enterprise. If you already have a Report Server in your environment and want to use it with CA Identity Manager, the minimum version required by CA Identity Manager is CA Business Intelligence 3.2.

Reporting Considerations

Consider the following before installing the Report Server:

- Installing the Report Server can take up to two hours.
- If JBoss is installed on the computer where you are installing the Report Server, port conflicts may occur. If Apache Tomcat is the web server, you can locate JBoss port information in the following files:
  - jboss-service.xml

    Default location: *jboss_home*\server*server_configuration*\conf
  - server.xml

    Default location:
    jboss_home\server*server_configuration*\deploy\jboss-web.deployer

  jboss_home

  Specifies the JBoss installation path.

  server_configuration

  Specifies the name of your server configuration.

  Default value: default

  Note: Restart JBoss if you make changes to either of these files.

Hardware Requirements

The hardware requirements for the Report Server are based on the operating system:

- For UNIX, see the "Minimum Hardware Requirements" section in *installer-media-root-directory*/Docs/supported-platforms-SP3-Solaris.pdf.
- For Windows, see the "Minimum Hardware Requirements" section in *installer-media-root-directory*/Docs/supported-platforms-SP3-Windows.pdf.

Note: For more information about supported OS versions and databases, see the Business Objects website.

How to Install the Report Server

The following checklist describes the steps to install CA Identity Manager's reporting feature:

✓	Step
	1. Review the report pre-installation checklist.
	2. Gather reporting information.
	3. Open ports required by the Report Server.
	4. Install the Report Server (CA Business Intelligence)
	5. Run the Registry Script.
	6. Copy the JDBC JAR files.
	7. Deploy the default reports.

Note: For more information about configuring reporting after the installation, see the *Administration Guide*.

Reports Pre-Installation Checklist

Print the following checklist to be sure that you meet the minimum system and database requirements before installing the Report Server:

- Be sure that the Windows or UNIX system on which you are installing the Report Server meets the minimum system requirements.
- Be sure that you use MySQL for the Report Database.
- If you create a database instance for the Snapshot Database, run the following scripts on the new database:
  - Microsoft SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
  - Oracle: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql

    To execute these scripts, the database user needs DBA, connect, and resource roles and system privileges to create tables, indexes, sessions and views with global query rewrite permission.

- On UNIX, set the following parameters as global in the local .profile files:
  - ORACLE_BASE: the top-level directory where Oracle is installed.
  - ORACLE_HOME: the path to the Oracle root directory under ORACLE_BASE
  - LD_LIBRARY_PATH: $ORACLE_HOME/lib32:$ORACLE_HOME/lib

    If Oracle is a 64-bit installation, use lib32. Use SQL Plus to connect to the oracle database instance to determine if it is a 64-bit installation.
  - ORACLE_SID: the SID name used in the tnsnames.ora file.
  - JAVA_HOME: the path to the Java root directory. Business Objects installs a JDK in the following location:

    `report_server_home/jre`

    Note: JDK 1.5 is the minimum version supported for reports.
  - PATH: $LD_LIBRARY_PATH:$JAVA_HOME:$JAVA_HOME/bin:$ORACLE_HOME/bin:$PATH
  - LC_ALL: en_US.UTF-8

    Note: Be sure that the CASHCOMP environment variable is empty.
- On UNIX systems:
  - 3 GB of free space is required under /tmp.
  - You need access to a non-root user account to install the Report Server.

    This user should have a home directory in the local file system. For example. the following command creates a user with a local home directory:

    `useradd -u 505 -g 0 -d /export/home/cabi -m cabi`

    Also, add the non-root user to the oinstall group and any group for which the root user is a member.
  - Enter the database server name in the /etc/hosts file if the database server is not on the same system as the Report Server. (If you have DNS, this step is unnecessary.)
  - If you encounter problems, inspect the SDK.log under these locations:

    /opt/CA/SharedComponents/CommonReporting3/ca-install.log

    /opt/CA/SharedComponents/CommonReporting3/CA_Business_Intelligence_InstallLog.log

Reporting Information

Record the following information you need during the Report Server installation:

Field Name	Description	Your Response
Administrator Password	Defines the password to log in to the Business Objects Infoview console.	
User Name	Identify the username for the Report Database.	
Password	Identify the administrative password credentials for the Report Database.	
Pre-Installed Tomcat Information	Identify the path and port numbers for any previous installation of Tomcat. If you do not want to use a previous installation of Tomcat, Report Server installer can install Tomcat.	
Tomcat Port Numbers	The Tomcat connection, redirect, and shutdown ports. **Note:** If you install the Report Server on the same system as the CA Identity Manager, be sure that the Tomcat connection port does not conflict with the port number you specified for the application server URL when installing the CA Identity Manager.	

Open Ports for the Report Server

For CA Identity Manager and the Report Server to communicate successfully, the following ports must be opened.

- The Central Management Server (CMS) port: 6400
- The Report Server web application port:
  - JBoss/Tomcat: 8080
  - WebLogic: 7001
  - WebSphere: 9080

  Note the following:

  - This port is not the application server port for the Identity Manager Server.
  - The web server ports are provided during the Report Server installation. If you use different ports during the installation, those ports must be opened through the firewall when the Report Server is deployed in production.
  - The Report Server does not connect to the application server used by CA Identity Manager.
- All database ports that CA Identity Manager has configured for the reporting and auditing databases. The Identity Manager Server must send database information to the Report Server, so these ports must be opened. For example, if the Snapshot Database is an Oracle database, the Report Server needs the Oracle port open outbound.

Install the CA Report Server

You can install the Report Server on a supported Windows or UNIX system. The following sections detail how to install the Report Server using a Windows and UNIX installation wizard.

Important! For a production environment, install the Report Server on a separate system from the system with the Identity Manager Server. If you want to install the Report Server on the same system as the Identity Manager Server for demonstration purposes, do not choose the default tomcat ports 8080 and 1099 if JBoss is using those ports.

Note: CA Identity Manager supports CA Business Intelligence 3.2 (which is Business Objects XI 3.0 SP3).

Run the Windows Installer

Install the Report Server using the Windows installation wizard (Disk1\InstData\VM\Install.exe) found on the Report Server media.

Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.

To install the Report Server

1. Exit all applications.
2. Download the Report Server and unzip it.
3. Navigate to Disk1\InstData\VM and double-click the installation executable.

   The installation wizard starts.
4. Use the gathered reporting information to install the Report Server.

   Note the following:

   - Select a New install during installation. This choice helps ensure that you use MySQL as the Report Database. If you must set non-default ports to avoid port conflicts, select a Custom install, but be sure to select MySQL for the Report Database.
   - Select Tomcat as the web server, deselecting IIS.
   - If you are installing the Report Server on the same system as CA Identity Manager, select the Tomcat connection port carefully. Verify that it does not conflict with the port number you specified for the application server URL when installing CA Identity Manager. However, we recommend installing the Report Server on a different system than the Identity Manager Server in a production environment.
5. Review the installation settings and click Install.

   The Report Server is installed.

Run the UNIX Installer

Add execute permissions to the install file by running the following command:
`chmod+x /ca-iamreportserver-12.5-solaris/cabiinstall.sh`

Important! The installer may crash if executed across different subnets. To avoid this problem, install the Report Server directly on the host system.

To install the Report Server

1. Log in as the non-root user you created to install the Report Server.
2. Exit all applications.

3. Download the Report Server and untar it.

   Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.

4. Open a command window and navigate to where the install program is located.

5. Enter the following command:

   ```
   /ca-iamreportserver-12.5-solaris/cabiinstall.sh console
   ```

6. Use the gathered reporting information to install the Report Server.

   Note the following:

   - Select a New install during installation. This choice helps ensure that you use MySQL as the Report Database. If you set non-default ports to avoid port conflicts, select a Custom install, but be sure to select MySQL for the Report Database.
   - Select Tomcat as the web server.
   - The installer installs the Report Server to /opt/CA/SharedComponents/CommonReporting3. Specifying another location does not change the installation location. So the /opt/CA directory must have non-root user permissions or the installation fails.

7. Review the installation settings and click Install.

   The Report Server is installed.

Run the Registry Script

For CA Identity Manager to change data sources for reports in the Report Server, run the mergeConnection script.

Note: On a 64-bit system, omit this procedure. The Report Server is a 32-bit application, so you use the 32-bit side of the registry. Open REGEDT32 directly from SysWOW64, and create the MergeConnectionProperties key with the Type REG_SZ and value Yes. Create the key in this location:

```
@HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Business Objects\Suite 12.0\Crystal Reports\DatabaseOptions
```

On the Report Server, the default location for this script is as follows:

- Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\ReportServerTools.
- UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/ReportServerTools.

On Windows, perform the following steps:

1. Run the mergeconnections_3.0.reg script and respond to the prompts that appear.
2. Click Start, Program Files, CA, Report Server, Central Configuration Manager.
3. Start all services, including Tomcat and the BO Server service.

On UNIX, perform the following steps:

1. Check for Windows control characters in the mergeconnections script.

   If you downloaded the software using FTP in binary mode, these characters do not appear in this script. If you used another download method, use the dos2unix command to remove these characters.

2. Copy the mergeconnections_3.0.cf script from the ReportServerTools directory to the following directory

   installation-directory/bobje/enterprise120/generic

3. Source in the environment variables for BusinessObjects Enterprise, as follows:

   source *installation-directory*//bobje/setup/env.sh

4. Run the following script, as follows:

   ./configpatch.sh mergeconnections_3.0.cf

   Select 1 as the option when prompted.

5. Restart crystal processing servers as follows:

   a. Log in as the non root user you used to install the Report Server.

   b. Issue these commands:

   ```
   cd /opt/CA/SharedComponents/CommonReporting3/bobje
   ./stopservers
   ./startservers
   ```

Copy the JDBC JAR Files

To copy the JDBC JAR files

1. Navigate to the jdbcdrivers folder where the CA Identity Manager Admin toolkit is installed. The default location is as follows:
   - Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\lib\jdbcdrivers
   - UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/lib/jdbcdrivers
2. Copy ojdbc14.jar (for Oracle) or sqljdbc.jar (for SQL Server) to the following location:
   - Windows: CA\SC\CommonReporting3\common\4.0\java\lib
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java/lib
3. Open the CRConfig.xml file, found in the following location:
   - Windows: CA\SC\CommonReporting3\common\4.0\java
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java
4. Add the location of the JDBC JAR files to the Classpath. For example:
   - Windows: <Classpath>report_server_home\common\4.0\java\lib\sqljdbc.jar; report_server_home\ common\4.0\java\lib\ojdbc14.jar …</Classpath>
   - UNIX: <Classpath>${BOBJEDIR}/java/lib/sqljdbc.jar:${BOBJEDIR}/java/lib/ojdbc14.jar: …</Classpath>
5. Save the file.
6. Restart the Report Server as follows:
   - For Windows, do the following:
     a. Go to Start, Program Files, BusinessObjects XI 3.1, BusinessObjects Enterprise, Central Configuration Manager.

        The Central Configuration Manager opens.

     b. Select all services and click Restart.
   - For UNIX, do the following:

```
cd /opt/CA/SharedComponents/CommonReporting3/bobje
./stopservers
./startservers
```

Deploy Default Reports

CA Identity Manager comes with default reports you can use for reporting. BIConfig is a utility that uses a specific XML format to install these default reports for CA Identity Manager.

If you are upgrading from a previous version of the Report Server, first remove the CA Identity Manager Reports folder using the Central Management Console. The existing reports do not work. You can then deploy default reports for the new Report Server.

Important! This process updates all default reports. If you customized any default reports, be sure to back them up before performing the update.

To deploy the default reports

1. Gather the following information about the Report Server:
   - Hostname
   - Administrator name
   - Administrator password
   - Snapshot database type
2. Copy all content from the Reports installer-root-directory/disk1/cabi/biconfig folder to the *im_admin_tools_dir*/ReportServerTools folder.
3. Set the JAVA_HOME variable to the 32-bit version of the JDK1.5 you installed.
4. Run one of the following commands:
   - For a Microsoft SQL Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "ms-sql-biar.xml"
     ```

   - For an Oracle Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "oracle-biar.xml"
     ```

   Note: In a UNIX operating environment, be sure that biconfig.sh has execute permissions.
5. View the biconfig.log file found in the location where you ran the command in Step 4.
6. Verify that the default reports installed successfully. Inspect the end of the log file for status; a successful install appears as follows:

   ```
   ReportingDeployUtility - Reporting utility program terminated and return code =
   0
   ```

BusinessObjects XI 3.x Post-Installation Step

If you run report tasks and receive a "Server Input% not found or server may be down" error message, perform this procedure.

Follow these steps:

1. Log in to the Central Management Console using the username and password you entered during the Report Server installation.
2. Under the main dashboard, select Servers.
3. Under the Server Name column, search for Input File Repository server and double-click the name.
4. In the Server Name text box, enter the following:

   Input.*report_server_hostname*.InputFileRepository
5. Click Save.
6. Under the Server Name column, search for Output File Repository server and double-click the name.
7. In the Server Name text box, enter the following:

   Output.*report_server_hostname*.OutputFileRepository
8. Click Save.
9. Restart *all* the servers by selecting the servers in the Server List.

Verify the Reporting Installation

To verify that reporting has been installed correctly, do the following:

- In the Central Management Console, be sure that all services are running.
- Be sure that your Report Database is running.

Note: For more information on configuring reporting after the installation, see the *Administration Guide*.

Silent Installation

For more information about silent installation of the Report Server, see the *CA Business Intelligence Installation Guide.* The Report Server documentation is available in one of the following locations when you extract the Report Server installer files:

- **Windows**: *install_root_directory*\Docs\CABI_Impl_ENU.pdf
- **UNIX:** *install_root_directory/*Docs/ENU/CABI_Impl_ENU.pdf

How to Uninstall Reporting

You uninstall the Report Server when it is no longer required on the system. For more information, see the CA Business Intelligence documentation.

After uninstalling the Report Server, Remove Leftover Items (see page 106).

Remove Leftover Items

The following sections detail the items you must manually remove after uninstalling the Report Server to keep the system as clean as possible and to prevent a reinstallation of the Report Server to the same system from failing.

Remove Windows Items

To remove leftover Report Server items on Windows

1. Navigate to *report_server_home*.

   report_server_home specifies the Report Server installation path.
2. Open the BusinessObjects Enterprise 12 folder, and delete the following folders:
   - Data
   - java
   - Logging
   - Samples
   - Web Content
   - Web Services
   - win32x86
3. Return to the Report Server folder.
4. Open the common folder.
5. Open the 4.0 folder, and delete the following folders:
   - crystalreportviewers115
   - java

You have completed removing leftover items.

Remove UNIX Items

To remove leftover Report Server items on UNIX

1. Navigate to the following location from a command prompt:

   /opt/CA/SharedComponents

2. Delete the CommonReporting3 folder.

You have completed removing leftover items.

Chapter 8: High Availability Provisioning Installation

Based on the guidelines in this chapter, you implement high availability for provisioning components by installing alternate Provisioning Servers and Provisioning Directories, and connector servers for C++ and Java connectors.

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
X	**5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.**

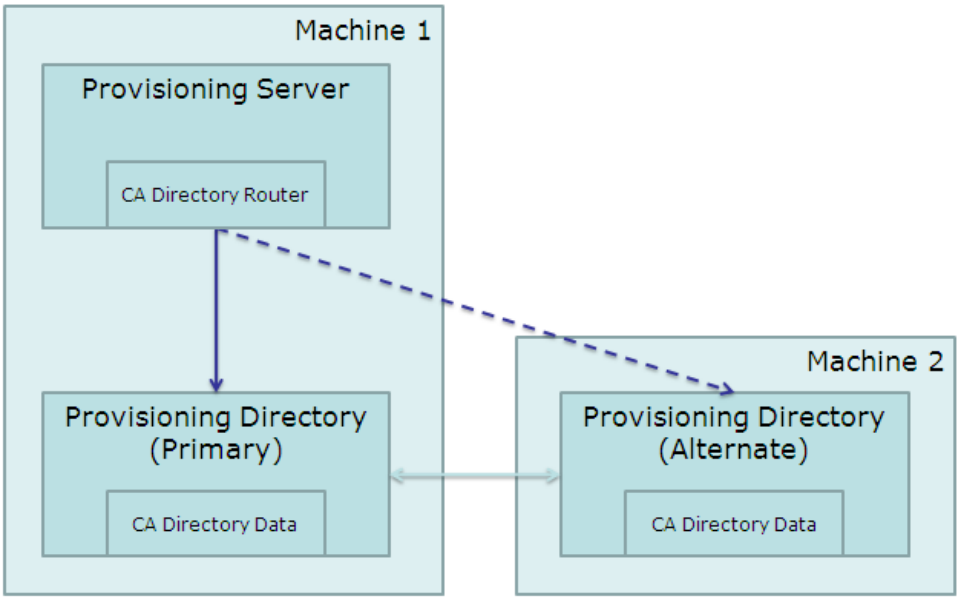
How to Install High Availability Provisioning Components

The following table describes the steps involved in installing provisioning components for high availability:

✓	Step
	1. Install primary and alternate Provisioning Servers and provisioning directories for load balancing and failover.
	2. Install several connector servers for load balancing and failover.
	3. Enable clients of the provisioning server to fail over.

Install Provisioning Directories

To support failover, you can install primary and alternate Provisioning Directories. For example, you may have one system with the Provisioning Server on it and the primary Provisioning Directory. A second system has the alternate Provisioning Directory. If the primary Provisioning Directory fails, the alternate Provisioning Directory is assigned automatically.



You install alternate Provisioning Directories if they were not configured during the installation.

To install Provisioning Directories

1. Install the primary Provisioning Directory using the Provisioning Directory installer from where you unpacked the install package.

   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup

   If you have already installed a primary Provisioning Directory during the installation, you can omit step 1.

2. Perform the prerequisite configuration for the new Provisioning Directories.
3. Install one or more alternate Provisioning Directories.

Perform Prerequisite Configuration for New Provisioning Directories

You use the High Availability Configuration command before you use the Provisioning Directory installation program.

To Perform Prerequisite Configuration for New Provisioning Directories

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the highavailability sub-directory where you unpacked the install package. For example:

   Installed-Provisioning-Directory-location\Provisioning\Provisioning Directory\highavailability

3. Enter this command:
   `highavailability.bat`

   The command displays a summary of the current configuration: the domain name, the hostname of each Provisioning Server and Provisioning Directory, and which one is the Primary Provisioning Directory.

4. Respond to the prompts to provide the hostnames required for each alternate Provisioning Directory that you want to add.

   If you plan to install alternate Provisioning Servers, you can add their hostnames now by responding to the prompts.

5. Log in to all other Provisioning Directory and Provisioning Servers and repeat steps 2 through 4.

Install Alternate Provisioning Directories

Once you have performed the prerequisite configuration required, you can install alternate Provisioning Directories.

To install alternate Provisioning Directories

1. Log in as a Local Administrator (for Windows) or root (for Solaris) into the system where you plan to install the alternate Provisioning Directory.
2. Be sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:
   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, data replication between the Provisioning Directories fails.
4. Run the Provisioning Directory installer from where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup
5. Select High Availability, and respond to the questions about the hostnames for systems where other Provisioning Directories are installed and which system is the primary Provisioning Directory.

6. Respond to other questions using the same answers given during the primary Provisioning Directory installation for:
   - Deployment Size
   - Shared Secret
   - FIPS key
7. Respond to this question based on how and when you want to replicate data from the Primary Provisioning Directory:

   ```
   Do you want to start replication to the Provisioning Directory.
   ```

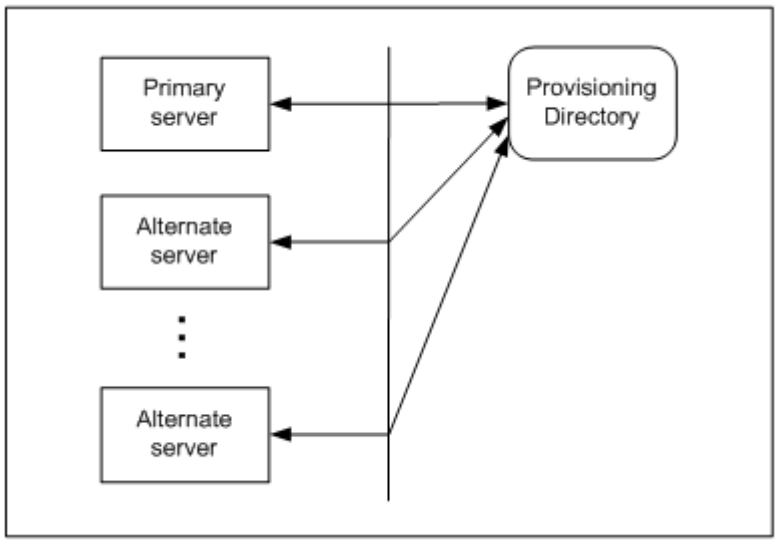
   If you are upgrading from a previous release, you may have a significant amount of data to replicate. You should deselect the checkbox if you do not want replication to start at this time. After the installation, you would then need to copy an LDIF data dump or online backup files from an existing Provisioning Directory and load the data or start the DSAs manually, which will start automatic replication.

Important! If alternate Provisioning Directory installation failed, data replication may have occurred before the failure. If so, the master and alternate Provisioning Directories have a record that replication occurred. If you now reinstall the alternate Provisioning Directory, that data is not replicated again. Instead, use the High Availability Configuration command on the primary and alternate Provisioning Directories to remove and add back the alternate Provisioning Directory before you reinstall it.

Provisioning Servers

Multiple Provisioning Servers share the workload of a provisioning domain, providing performance, scalability, and high availability. The first Provisioning Server installed is called the primary Provisioning Server. Additional servers are called alternate Provisioning Servers.

As shown in this illustration, you can configure multiple alternate Provisioning Servers for one primary Provisioning Server.



In this illustration, three Provisioning Servers are configured to serve the provisioning domain. All servers are configured to use the Provisioning Directory of the primary Provisioning Server installation.

Router DSA for the Provisioning Server

The Provisioning Server communicates through a CA Directory router DSA, and not directly to the Provisioning Directory. The router DSA, imps-router, is installed with the Provisioning Server installer. This DSA accepts requests from the Provisioning Server and routes them to the appropriate Provisioning Directory DSA (impd-co, impd-main, impd-inc, or impd-notify) depending on the prefix.

In a high-availability installation, the imps-router DSA has connection information for Provisioning Directory DSA on at least one alternate Provisioning Directory system. If a primary Provisioning Directory DSA becomes unavailable, the router DSA attempts to use an alternate DSA.

The imps-router DSA has been assigned ports 20391, 20391, 20393 (for address, SNMP, and console respectively).

Note: In previous releases of this software, the etrustadmin DSA used port 20391. Any connections to 20391 on the Provisioning Directory system fail unless the Provisioning Directory and Provisioning Server are on the same system. Therefore, reroute these connections to port 20391 on the Provisioning Server system.

For CA Directory DSAs running on one system to communicate with DSAs on another system, they must have connection information for each other. So during Provisioning Directory installation, you identify each Provisioning Server that can connect to it.

Install Provisioning Servers

To support failover, you can install primary and alternate Provisioning Servers. If you have already installed a Provisioning Server, you can omit step 1.

To install Provisioning Servers

1. Install the primary Provisioning Server using the Provisioning Server installer from where you unpacked the install package.
   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup
2. Perform prerequisite configuration for the new Provisioning Servers.
3. Install one or more alternate Provisioning Servers.
4. Enter the alternate Provisioning Server host and port number when you enable provisioning in the Identity Manager Management Console. For details, see the *Configuration Guide*.

Perform Prerequisite Configuration for New Provisioning Servers

To configure knowledge files, you use the High Availability Configuration command on each system with a Provisioning Directory.

To Perform Prerequisite Configuration for New Provisioning Servers

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the highavailability sub-directory where you unpacked the install package. It is a sub-directory of where you install the Provisioning Directory or Provisioning Server. For example:

   ```
   cd C:\\Program Files\Provisioning Directory\highavailability
   ```

3. Enter this command:

   `highavailability.bat`

   The command displays a summary of the current configuration: the domain name, and the hostname of each Provisioning Server and Provisioning Directory.

4. Respond to the prompts to provide the hostnames required for each Provisioning Server that you want to add.

   If you plan to also install alternate Provisioning Directories, you can add their hostnames now by responding to the command prompts.

5. Log in to each system that will host a Provisioning Directory and repeat Steps 2 through 4.

Install Alternate Provisioning Servers

Once you have performed the prerequisite configuration involving the highavailability command, you can install one or more Provisioning Servers.

To install alternate Provisioning Servers

1. Log in as a Local Administrator (for Windows) or root (for Solaris) on each system that will host an alternate Provisioning Server.
2. Make sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:

   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, the Provisioning Server will not route any custom schema.

4. Run the Provisioning Server installer from where you unpacked the install package.

   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup

5. Complete the instructions in the installer dialog boxes.

   You can select a check box during installation to configure Provisioning Directory high availability. If you choose this option, you must supply the hostnames of any alternate Provisioning Directories and specify the primary Provisioning Directory.

Configure Provisioning Server Failover

For CA Identity Manager to distinguish the primary from the alternate Provisioning Server, you create server definitions in JIAM in the Management Console. You create these definitions in the directory object associated with the Identity Manager directory for your environment. During initialization, CA Identity Manager reads any failover server definitions defined in that object, adding them to the JIAM failover server definitions.

Note: For details on setting up server definitions, see the *Configuration Guide*.

Connector Servers

With the Connector Server Framework (CSF), you can run multiple Connector Servers and configure the Provisioning Servers to communicate with Connector Servers in specific contexts.

As a result, the Provisioning Server can:

- Support Connector Servers on different platforms to manage endpoint types that are unavailable on the platform where the Provisioning Server is installed.
- Communicate with multiple Connector Servers, which each manage a different set of endpoint types or endpoints. Therefore, endpoint types or endpoints can be managed on a parallel basis to achieve load balancing.

Connector Server Framework

The use of several Connector Servers is called the Connector Server Framework. The Connector Server Framework has two important characteristics:

- Scalability - multiple connector servers may share the load of working on a set of endpoints.

  For example, a lengthy exploration of an endpoint on one connector server does not influence the ability to operate on an endpoint that is being controlled by another Connector Server

- Communication channel security - communication between Provisioning Server and connector server is encrypted using TLS.

  If an endpoint type uses a proprietary protocol to communicate between the connector server and endpoints of that protocol, the extent of use of the proprietary protocol may be limited to a local network, or even to just local communication inside one server.

When deciding on an implementation strategy, consider these factors so that you protect the Connector Servers in your organization against unauthorized access:

- The Connector Server may be configured to disclose passwords in clear text.

  Any person with access to the system running the Connector Server and with sufficient privileges to modify the configuration of the Connector Server and to restart the Connector Server can make the Connector Server log passwords appear in clear text.

  The Connector Server is based on the open source slapd process. The instructions to make a slapd process log incoming passwords in clear text are in the public domain, for example, by looking at the manual pages at http://www.openldap.org

- The Connector Server is only protected by a bind password.

  The Connector Server trusts any client who connects to it and is able to provide the proper credentials, such as Bind DN and Bind Password. The Connector Server does not know if the connection comes from a Provisioning Server or not. Any user with internal access may disclose the bind password, then connect to the Connector Server from another server, and so have administrator privileges over the endpoints controlled by the Connector Server.

- The Connector Server is not protected against brute force attacks on the bind password

  Unlike the Provisioning Server, the Connector Server is not protected against repeated attempts at binding with different passwords. An attacker may therefore try to guess the password by brute force attack. Should an attacker succeed in guessing the bind password, then the road is open for the attacker to control the endpoints under control of the Connector Server.

For these reasons you are advised to design your implementation such that

- The same organizational unit is responsible for administrative access to all Provisioning Servers and connector servers.
- Your connector servers are suitably protected by firewalls or similar such that the ports may not be reached by unauthorized means.
- The ability to connect to Provisioning Servers and connector servers on non-TLS ports should be disabled in your production environments.

Load-Balancing and Failover

Failover and load-balancing of connector requests is achieved by each provisioning server based on the CSF configuration defined using csfconfig or Connector Xpress.

Each provisioning server consults the CSF configuration that applies to it and determines which Connector Servers it should use to access each endpoint or endpoint type. Failover and load-balancing occur when there are multiple connectors servers configured to serve the same endpoint or endpoint type.

Failover and load-balancing are unified and cannot be controlled separately. One cannot indicate that a particular connector server is to remain idle except when needed for failover. Instead, a provisioning server that is configured to use two or more connector servers interchangeably will distribute work between these connector servers (load balancing) during normal operation. Should one or more of the Connector Server become unavailable, the remaining connector servers will provide failover support for the unavailable connector servers.

Reliability and Scalability

With the Connector Server Framework (CSF), the Connector Server high availability features increase reliability and scalability.

Reliability is enhanced by having multiple Connector Servers serve a Provisioning Server, so it can continue to function if one or more Connector Servers become unavailable.

For example, if one Connector Server manages the UNIX endpoint type and another manages the Active Directory endpoint type; and the Active Directory Connector Server becomes unavailable, the Provisioning Server can still manage the UNIX endpoint types.

Scalability is achieved by having a mechanism to add more Connector Servers to manage an increasing amount of endpoint types or endpoints. For example, if the number of endpoint types increases to 100, the Provisioning Server can be configured to have 20 Connector Servers, with each Connector Server managing five endpoint types. Or configure 20 Connector Servers with each Connector Server managing overlapping sets of 10 endpoint types to allow for failover and load balancing behaviors as well.

Multi-Platform Installations

The Connector Server Framework is the configuration of Connector Servers that exist on multiple systems, which could be Windows or Solaris systems.

The following use cases are supported:

- Use Case 1
  - Provisioning Server and connector server installed on a Solaris system.
  - A second Connector Server installed on a Windows system, serving the non-multi-platform connectors.
- Use Case 2
  - Provisioning Server and connector server installed on a Windows system.
  - A second Connector Server installed on Solaris system, serving the multi-platform connectors.
  - A third Connector Server installed on a remote Windows system, serving the other connectors.
- Use Case 3
  - Provisioning Server installed on a Windows or Solaris system and a Connector Server installed on the same system.
  - Multiple additional Connector Servers installed on Windows or Solaris systems, serving as endpoint agents. This scenario is important for cases where the connector is using a proprietary or un-secured communication channel. Using this topology, the important segment of network traffic is secured by the standard Provisioning Server to Connector Server communication protocol and not by the proprietary protocol.

Install Connector Servers

Based on the guidelines in this chapter, you make connector servers highly available by installing several instances of Java Connector Servers or C++ Connector Servers, or both.

To install the Java Connector Server

If you plan to install more than one Java Connector Server, see the *Java Connector Server Implementation Guide* for additional guidelines. For a single Java Connector Server, follow these steps:

1. Run the following program where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Connector Server\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ConnectorServer/setup
2. Complete the instructions in the installer dialog boxes.

To install the C++ Connector Server

1. Run the following program where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningServer\setup.bin
2. Complete the instructions in the installer dialog boxes.

   This installation program also gives you the option to install alternate Provisioning Servers. However, for that component, a different procedure (see page 115) applies.

Configure Connector Servers

You configure the Connector Server Framework by using the csfconfig command or by using Connector Xpress. The csfconfig command uses the data in the Windows Registry (or UNIX counterpart created for Provisioning Server) to connect to a Provisioning Server. The csfconfig command must run on the system where one of the Provisioning Server runs.

Using the command, you can:

- Add or modify a Connector Server connection object with information such as the connector server, host, and port.
- Define for which endpoints or endpoint types the connector server is used; possibly varying this definition for alternate provisioning servers.
- Delete the Connector Server connection information object.
- List all connector server connection objects in a domain.
- Show one or all connector server connection objects for one or all connector servers

The csfconfig command uses the authorizations provided by a global user credential, so that global user must have the necessary administrative privileges to manipulate the appropriate ConfigParam and ConfigParamContainer objects.

csfconfig Command

To use the csfconfig command, the command line syntax is:

```
csfconfig [--help[=op]] [operation] [argument]
```

You can use these arguments in any order. The operation argument is required unless you are using the --help argument.

The --help[=op] option provides minimal on-line help. The “=op” argument may be used to list the arguments that are required or optional for the operation. For example, “--help=add” will provide a description of the add operation, while “--help” will provide general information.

If help is requested, other arguments are ignored and no request is sent to the server.

Note: The domain parameter can be omitted as it is always the domain used in the whole installation.

The following operations are available.

add

Add a new CS connection object. A name will be generated by this operation if one is not specified by the user. Required arguments: auth, host, pass. Optional arguments: authpwd, br-add, desc, domain, name, port, usetls, debug.

addspec

Adds a branches specialization for one provisioning server.

When you have installed alternative provisioning servers, sometimes a connector server is not to be used by all of these Provisioning Servers. Or sometimes different provisioning servers will want to use the same connector servers for different branches (endpoint types or endpoints). A branches specialization is a list of branches that is specific to one provisioning server. Only provisioning servers without a specialization will use the branches specified in the main CS connection object. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, domain, debug.

list

List all CS connection objects. Required arguments: auth. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, usetls, debug.

modspec

Edits a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, br-rem, domain, debug.

remove

Remove an existing CS connection object. Required arguments: auth, name. Optional argument: authpwd, debug.

remspec

Removes a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, server, tls, usetls.

show

Show a specific CS connection object or show all CS connection objects. The output shows the host and port of the connector server if it is available. Required arguments: auth. Optional arguments: authpwd, name, domain, debug.

Each operation takes several arguments in the form “name=value”. Spaces are not allowed before or after the “=” symbol, and if the value contains any spaces, the argument must be quoted appropriately for the platform (Windows or UNIX). Except as noted, the value must be provided, and must be non-empty.

The following arguments are used for the operations as noted above:

auth=<value>

Identify the global user for authentication.

Value format: “name” where name is the global user's name.

authpwd=<value>

Identify a file containing just the global user's password on the first line. If this argument is not specified, the user will be prompted for a password.

Value format: any appropriate operating system file path.

br-add=<value>

Add a new branch. This argument may be specified multiple times to add multiple branches.

Value format: “[[endpoint,]endpoint type][@[domain]]”. Use a branch of “@” by itself to represent all branches. Add “endpoint type” or “endpoint,endpoint type” to identify a specific endpoint type or endpoint.

br-rem=<value>

Remove an existing branch. This argument may be specified multiple times to remove multiple branches.

Value format: same format as specified for br-add.

debug=<value>

Turns on trace logging for the command. Tracing messages are written to the file PSHOME\logs\etaclientYYYYMMDD.log file.

Value format: The value “yes" enables logging.

desc=<value>

Provide an arbitrary description for the object. If not specified in an add operation, it will default to the value of the host argument.

Value format: an arbitrary string.

domain=<value>

Define the default domain. If not specified, the domain specified in the auth argument is used as the default.

As the value can only be the default, this parameter can always be omitted

host=<value>

Define the name of the host on which the Connector Server runs.

Value format: any legal host name or IP address.

name=<value>

The name of the Connector Server object. If not specified during Add, csfconfig will assign a name and display what name was created.

Value format: A case-insensitive string of one or more characters consisting of upper-case English letters (A-Z), lower-case English letters (a-z), digits (0-9), hyphen(-) or underscore(_).

pass[=<value>]

Define the file containing the password for the Connector Server connection object. If the value is not specified, the user will be prompted.

Value format: any appropriate OS file path.

Important! The password you must specify is the password you entered when you installed that Connector Server or you changed subsequent to install by running the pwdmgr utility on that Connector Server system.

port=<value>

Define the port number for the object. This must be a valid number for a port where the Connector Server listens for connections.

Value format: an integer.

server[=<value>]

In addspec, modspec and remspec commands, define the name of the Provisioning Server that is served by the Connector Server . The branches defined in a specialization override, for a particular provisioning server, the branches defined in the CS configuration object by add and modify commands.

Value format: the name of the host where the Provisioning Server is running as returned by the system's hostname command.

Note: The Connector Server configuration objects are stored with the other domain configuration parameters in the provisioning directory. While the Connector Server configuration parameters cannot be viewed or changed with the provisioning manager directly, one can use the provisioning manager (System task, Domain Configuration button) to get a list of known provisioning servers. To do this, open the “Servers” parameter folder and the known provisioning servers will be listed.

usetls[=<value>]

Indicate if TLS should be used to communicate with the Connector Server. The value is optional for the add operation only, in which case it defaults to “yes.” .

Value format: a string “yes” or “no”.

Upon successful completion of the add operation, the name of the newly created Connector Server connection object will be listed. If the name parameter is missing, a name is generated. For example:

```
Created CS object with name = SA000
```

For most operations, successful or not, the status and a message (if any) will be shown. For example:

```
The host name, port number, or TLS flag was successfully changed. The branch settings
were successfully changed.
```

For some errors, such as invalid command line parameters, no status code or server error message is displayed. In these cases, a simple statement of the error will be shown. For example:

```
$ csfconfig add
No authentication information supplied.
For on-line help, use "--help [=<op>]"
```

csfconfig Command Examples

To specify that the UNIX and CA Access Control endpoint types should be served by the Connector Server running on host "sunserver01" and the remaining endpoint types served by a Connector Server running on host "windows02", issue the following commands.

Each command execution prompts you for the etaadmin password.

```
csfconfig add \
auth="etaadmin" \
br-add="UNIX – etc" \
br-add="UNIX – NIS-NIS plus Domains" \
br-add="Access Control" \
host="sunserver01" \
usetls="yes"

csfconfig add \
auth="etaadmin" \
br-add="@" \
host="windows02" \
usetls="yes"
```

C++ Connector Server on Solaris

The C++ Connector Server installed on Solaris can manage only Solaris UNIX ETC and ACC endpoints. For all other Connectors, install the C++ Connector Server on a Windows system and register it with the Provisioning Server installed on Solaris. During installation, specify that this Connector Server is your default C++ Connector Server.

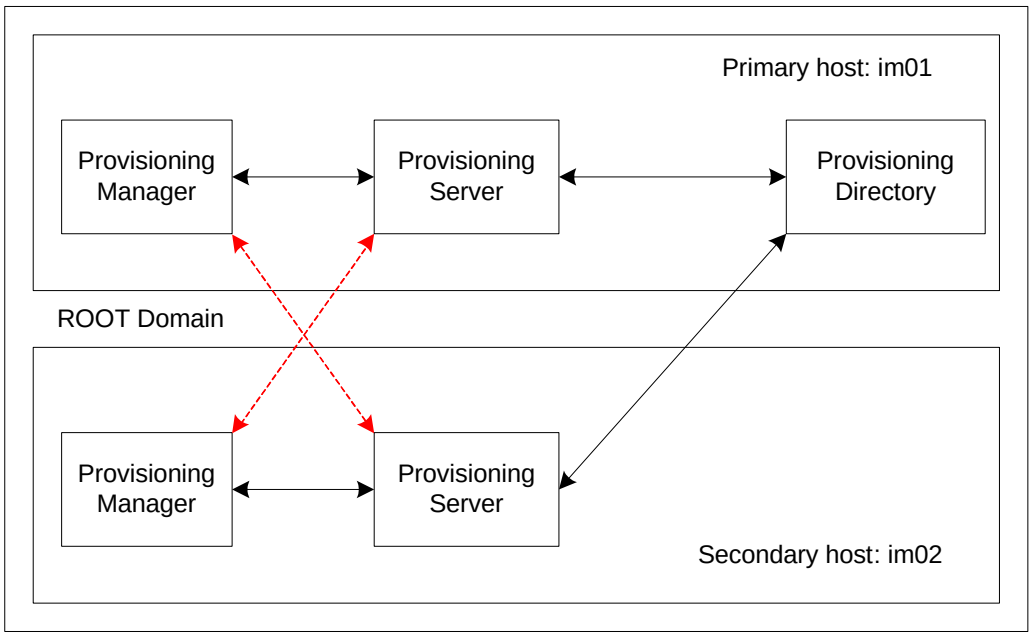
Failover for Provisioning Clients

Client-tier configuration includes the following tasks:

- Configure the Windows client-tier failover
- Configure the Provisioning Manager to communicate with their local Provisioning Servers, and fail over to the remote Provisioning Server

You use the same Provisioning Manager dialog to accomplish both of these tasks, on each server in turn.

The configuration shown in the following illustration lets Provisioning Manager submit identity provisioning requests to one Provisioning Server and fail over to another server:



The Provisioning Manager sends requests to the default Provisioning Server and fails over to another server.

Enable User Console Failover

If the application server for the Identity Manager Server fails, it does not receive Provisioning Server updates. As a result, the Identity Manager User Console does not show provisioning changes. Therefore, you should configure an alternate URL for the Identity Manager Server.

To enable the client-tier failover for the User Console

1. Launch the Provisioning Manager.
2. Click System, Identity Manager Setup.
3. Fill in the host name and port for another system in the cluster.
4. Fill in the environment.

   It must be the same one that is on the primary URL.
5. Click Add.

Enable Provisioning Manager Failover

You can enable Provisioning Manager failover on both the primary and secondary host servers. When this procedure is complete, you will have configured each server for failover to the other.

To enable the Provisioning Manager failover

1. Launch the Provisioning Manager.
2. Select File, Preferences, and select the Failover tab.
3. Mark the Enable Failover check box. By default, the local Provisioning Server is already defined.
4. Click Add.
5. Enter the host name of the remote Provisioning Server.

   For example, on im01, enter the server host for im02. On im02, enter the server host for im01.
6. Enter 20389 for the LDAP port value and 20390 for the LDAP/TLS port value, respectively.
7. Adjust the preference order by moving the entries up and down in the list.
8. Click OK.
9. Restart the Provisioning Manager to enable your changes.

Test the Provisioning Manager Failover

You can test your client failover configuration by performing the following procedure:

To test Provisioning Manager failover

1. Stop the CA Identity Manager - Provisioning Server service on one domain server.
2. Issue one or more operations using Provisioning Manager for this server installation.

   Since you stopped the CA Identity Manager - Provisioning Server service locally, the traffic flows to the failover domain server. If it does not, check your configuration and try the test again.

Appendix A: Uninstallation and Reinstallation

This section contains the following topics:

How to Uninstall CA Identity Manager

To fully uninstall CA Identity Manager, remove CA Identity Manager software components and clean up the CA Identity Manager-specific configuration in your application server. The following checklist describes the steps to uninstall CA Identity Manager:

✓	Step
	1. Delete CA Identity Manager objects with the Management Console.
	2. (Optional) If you used SiteMinder, remove the CA Identity Manager schema from the policy store or remove the Policy Server. For more information, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide*.
	3. Use the highavailability command to uninstall Provisioning Directories and Provisioning Servers from this location: *Unpacked-Install-Package*\Provisioning\Provisioning Directory\highavailability
	4. Uninstall the CA Identity Manager components.
	5. Remove CA Identity Manager configuration information from the application server.

Remove CA Identity Manager Objects with the Management Console

In order to remove objects created automatically by CA Identity Manager when you configure environments and directories, use the Management Console.

1. Open the Management Console:

   http://*im_server*:*port*/iam/immanage
2. Click Environments.
3. Select all of the check boxes for the existing Environments.
4. Click Delete.
5. Click Directories.
6. Select all of the check boxes for the existing Directories.
7. Click Delete.

Remove the CA Identity Manager Schema from the Policy Store

If you were using a SiteMinder Policy Server, remove the CA Identity Manager schema from the policy store.

Remove the CA Identity Manager schema from a SQL Policy Store

On systems where you installed the CA Identity Manager Extensions for SiteMinder, remove the CA Identity Manager schema. The default location for the command to remove the schema follows:

- SQL Server:

  `C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\MicrosoftSQLServer`
- Oracle:

  `UNIX`:
  `/opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas/OracleRDBMS`

  `Windows`: `C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager/tools/policystore-schemas/OracleRDBMS`

Remove the CA Identity Manager schema from an LDAP Policy Store

Note: If you are using Microsoft Active Directory or Microsoft ADAM as a policy store, you do not need to complete this procedure. You cannot remove schema objects from these policy stores. However, you can disable them. For more information, see the documentation for your directory.

To remove the CA Identity Manager schema from an LDAP policy store

1. Complete one of the following:
   - If you are using IBM Directory Server as a policy store, in the IBM Directory Server Web Administration user interface, remove the schema file V3.imsschema60 from the Files section of the schema configuration. Then, restart the directory server.

     Note: There are no other steps required to remove the schema from an IBM Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using CA Directory as a policy store, remove the etrust_ims.dxc file from *dxserver_home*\config\schema.

     where *dxserver_home* is the install location of CA Directory.

     Note: There are no other steps required to remove the schema from a CA Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using another LDAP directory as a policy store, skip to Step 2.
2. Navigate to the policystore-schemas folder. These are the default locations:
   - **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas
   - **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas
3. Use the appropriate LDIF schema file from the following table to remove the schema from the directory.

   Note: For more information on removing schema files, see the documentation for your directory.

Directory Type	LDIF File
Novell eDirectory	novell\novell-delete-ims8.ldif
Oracle Internet Directory (OID)	oracle-internet-directory\oracle-internet-directory-delete-ims8.ldif

Directory Type	LDIF File
Sun Java Systems (Sun One, iPlanet)	sunone\sunone-delete-ims8.ldif

Uninstall CA Identity Manager Software Components

Use the instructions in this section to uninstall CA Identity Manager components from each system on which you installed a component. For example, if you installed the Identity Manager Server and the Identity Manager Administrative Tools on separate systems, uninstall components from both systems.

To uninstall CA Identity Manager software components on Windows

1. Go to Start, Control Panel, Add/Remove Programs and select CA Identity Manager.
2. Select CA Identity Manager.
3. Click Change/Remove.

   All non-provisioning components are uninstalled.
4. For any provisioning components, use the individual component installer to uninstall the component.

   Note: Although you install Provisioning Manager with Administrative Tools, you use the Provisioning Manager installer to uninstall this component.

To uninstall CA Identity Manager software components on UNIX

1. Navigate to the following location:

   IM_HOME/install_config_info/im-uninstall
2. Run the following script:

   ```
   sh uninstall.sh
   ```

   Follow the on-screen instructions.
3. For any provisioning components, use the individual component installer to uninstall the component.

Remove CA Identity Manager from WebSphere

After uninstalling CA Identity Manager software, you can remove the CA Identity Manager configuration from your application server by using the WebSphere Administrative Console or by executing scripts from the command line.

To remove CA Identity Manager using the Administrative Console

1. Open the WebSphere Administrative Console using the following URL:

   http://*websphere_server*:9060/admin

2. Select Applications, Enterprise Applications.
3. In the Enterprise Applications screen, select the check box next to CA Identity Manager and click Stop.
4. Select the check box next to CA Identity Manager and click Uninstall.
5. If you installed the SiteMinder EAR and SiteMinder Agent EAR, stop these applications, and uninstall them as described previously.
6. Click Save.
7. Click Save to save changes to the master configuration.
8. Remove the ca-stylesr5.1.1.ear file.

   Note: Only remove the ca-stylesr5.1.1.ear if no other CA product is using it.

To remove CA Identity Manager using the command line

CA Identity Manager includes two scripts that you can use to remove CA Identity Manager from the WebSphere application server:

- Uninstall script (uninstallApp.jacl)—Stops the CA Identity Manager application, then removes it from WebSphere.
- Cleanup script (Ims6Cleanup.jacl) —Removes the CA Identity Manager resources, such as those created by running the uninstallApp.jacl.

  Note: Running the Cleanup script removes resources that are used by all CA Identity Manager installations running on the same application server. If you have CA Identity Manager installations on the same system that you do not want to delete, do not run the Cleanup script. Also, this script does *not* remove any data sources created by CA Identity Manager.

To remove CA Identity Manager using the command line, perform the following procedure.

1. From the command line, navigate to *websphere_home*\bin.
2. Be sure that the WebSphere application server is running.
3. Run the Uninstall script as follows:
   - **Windows:** wsadmin -f uninstallApp.jacl
   - **Unix:** ./wsadmin.sh -f uninstallApp.jacl

4. Run the Cleanup script as follows:
   - **Windows:** wsadmin -f Ims6Cleanup.jacl *websphere_node*
   - **Unix:** ./wsadmin.sh -f Ims6Cleanup.jacl *websphere_node*

   where *websphere_node* is the name of the WebSphere node where CA Identity Manager was installed.
5. Remove the ca-stylesr5.1.1.ear file.

   Note: Only remove the ca-stylesr5.1.1.ear if no other CA product is using it.
6. Remove the service integration bus as follows:
   a. In the WebSphere Administrative Console, go to Service Integration, Buses.
   b. Remove iam_im-IMSBus.
   c. Stop the application server.
   d. Remove the *node_name.server_name*.IMSBus directory under *websphere_home*\profiles*websphere_profile*\databases\com.ibm.ws.sib\

Reinstall CA Identity Manager

You can reinstall any of the CA Identity Manager software components by rerunning the installer. When you run the installer, it detects any CA Identity Manager components installed on the system. You may reinstall the same components that you originally installed on the system or other components that were not originally on the system.

Note: Reinstalling the Identity Manager Administrative Tools replaces all of the files in the Administrative Tools directory. To prevent overwriting custom files, back up the directory where the Administrative Tools are installed.

Appendix B: UNIX, Linux, and Non-Provisioning Installations

For UNIX and LINUX systems and scenarios where no provisioning software is needed, some additional instructions apply.

This section contains the following topics:

UNIX and Console Mode Installation

The examples in this guide provide the Solaris executable name for the installation program. However, you may be installing on AIX or Linux.

- For AIX, use: ca-im-*release*-aix.bin
- For LINUX, use: ca-*release*-linux.bin

release represents the current release of CA Identity Manager

If you are performing an installation in console mode, such as on a UNIX workstation, you add another option to the command line.

- For the main installation, add -i console. For example:
  `./ca-im-12.5-sp*N*-sol.bin -i console`
- For installation of provisioning components, add -console to the setup command.

Red Hat Linux 64-bit Installation

If you plan to install CA Identity Manager on a Red Hat Linux 64-bit system, you need to prepare the system for the installation.

Follow these steps:

1. Create a symbolic link to work around a CryptoJ failure. Use the following command:
   ```
   ln –s /dev/urandom /dev/random
   ```
2. Install four 32-bit packages using the following commands:

   ```
   yum install glibc.i686
   yum install libXext.i686
   yum install libXtst.i686
   yum install ncurses-devel.i686
   ```

Note: The i686 suffix specifies that the library is 32-bit, for the x86 processor.

Alternatively, the dependencies may be resolved using Add/Remove Software, and unchecking the Only Native Packages filter option. Using this approach, you select and install the required i686 architecture dependencies.

The native ksh shell package also needs to be installed. Use the following command:
```
yum install ksh
```

Alternatively, the package dependency may be resolved using Add/Remove Software. Using this approach, you select and install the required i686 architecture dependencies required ksh package.

Non-Provisioning Installation

This guide refers to the Windows and Solaris program names for the installers that provide options to install provisioning software. If you prefer to see no provisioning options, you can use these installers:

- For Windows, use IMWithoutProvisioning\ca-im-web-*release*-win32.bat
- For Solaris, use IMWithoutProvisioning/ca-im-web-*release*-sol.sh

release represents the current release of CA Identity Manager.

Appendix C: Unattended Installation

This section contains the following topics:

How to Run an Unattended Installation

To run the installer in the unattended installation mode

1. Modify the im-installer.properties file.
2. Run the following command:
   - **Windows:**
     `ca-im-12.5-sp01-win32.exe -f im-installer.properties -i silent`
   - **UNIX:**
     `./ca-im-12.5-sp01-sol.bin -f im-installer.properties -i silent`

Modify the Configuration File

To enable an unattended CA Identity Manager installation, modify the settings in the im-installer.properties configuration file using a text editor. The default parameters in the file reflect the information entered during the initial CA Identity Manager installation. Change the default values as needed.

Note the following when modifying the configuration file:

- Make a back-up copy of the installer properties file before modifying the original, since the file holds all of the values you entered during the initial installation or configuration.
- Do not add extra spaces between the parameter name, the equals sign (=), and the attribute value.
- All directory names on Windows must contain either double back slashes or forward slashes, not single back slashes.

Initial Choices

For basic installation choices, enter values for the following parameters:

Parameter	Instructions
DEFAULT_NEW_INSTANCE_DISPLAY_NAME	Enter 'New Installation' if this is a fresh install. For upgrades, this will be blank.
DEFAULT_COMPONENTS	Enter one or more components: ■ Server - Identity Manager Server ■ Exten - Extensions to the Policy Server ■ Admin - Identity Manager Administrative Tools ■ Provision - Provisioning Server ■ Directory - Provisioning Directory To install more than one component, separate components by a comma.
DEFAULT_INSTALL_FOLDER	Enter the directory in which to install the Identity Manager Server.
DEFAULT_GENERIC_USERNAME	Generic login information for CA Identity Manager components that are installed.
DEFAULT_GENERIC_PASSWORD	Generic password information for CA Identity Manager components that are installed.
DEFAULT_FIPS_MODE	Select if you want to enable FIPS 140-2 compliance.
DEFAULT_FIPS_KEY_LOC	Enter the path to the FIPS key location.

The installation program ignores any parameters that do not apply to the component you are installing. For example, if you set DEFAULT_COMPONENTS to Exten, only the DEFAULT_PS_ROOT and DEFAULT_USE_SITEMINDER parameters are used.

Identity Manager Server

If you plan to install the Identity Manager Server, enter values for the following:

Parameter	Instructions
DEFAULT_APP_SERVER	Enter, Weblogic, WebSphere, or JBoss
DEFAULT_APP_SERVER_URL	Enter full URL of the application server hosting CA Identity Manager, including the port.
DEFAULT_JAVA_HOME	Path to JRE or JDK for CA Identity Manager.
Additional Database Parameters	
DEFAULT_DB_HOST	Enter the hostname of the system hosting the CA Identity Manager database.
DEFAULT_DB_PORT	Enter the port of the system hosting the CA Identity Manager database.
DEFAULT_DB_NAME	Enter the name of the CA Identity Manager database.
DEFAULT_DB_USER	Enter the administrative username for the CA Identity Manager database.
DEFAULT_DB_PASSWORD	Enter the password for the administrative user of the CA Identity Manager database.
DEFAULT_DB_TYPE	Enter the type of database used for the CA Identity Manager database.
Additional JBoss Parameter	
DEFAULT_JBOSS_FOLDER	Enter the full pathname of the directory where you installed the JBoss application server. For example, C:\jboss-5.1
Additional WebLogic Parameters	

Parameter	Instructions
DEFAULT_BINARY_FOLDER	Enter the full directory path of the directory where you installed WebLogic. For example: C:\Oracle\Middleware\weblogic\
DEFAULT_DOMAIN_FOLDER	Enter the full path and directory name for the WebLogic domain you created for CA Identity Manager.
DEFAULT_SERVER_NAME	Enter the name of the WebLogic server instance you created for use with CA Identity Manager.
DEFAULT_BEA_CLUSTER	Enter the cluster name for the WebLogic cluster.
Additional WebSphere Parameters	
DEFAULT_WEBSPHERE_FOLDER	Enter the full pathname of the directory where you installed CA Identity Manager Tools for WebSphere.
DEFAULT_WAS_NODE	Enter the name of the node in which the application server is located.
DEFAULT_WAS_SERVER	Enter the name of the system on which the application server is running.
DEFAULT_WAS_CELL	Enter the name of the cell in which the application server is located.
WAS_PROFILE	Enter the location of the WebSphere profile files.
DEFAULT_WAS_CLUSTER	Enter the cluster name for the WebSphere cluster.

If you are using a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_HOST	Enter the fully-qualified domain name of the Policy Server.

Parameter	Instruction
DEFAULT_PS_USER	Enter the user name of the Policy Server administrator.
DEFAULT_PS_PW	Enter the password of the Policy Server administrator.

Provisioning Components

If you install Provisioning, enter the following:

Parameter	Instruction
DEFAULT_CONFIG_REMOTE PROVISIONING	Enter true if you are connecting to a remote Provisioning Directory.
DEFAULT_DEPLOYMENT_SIZE	Enter the size of your Provisioning Directory deployment.
DEFAULT_DIRECTORY_IMPS_HOSTNAMES	Enter the hostnames of all Provisioning Servers that will connect to the Directory.
DEFAULT_DOMAIN_NAME	Enter 'im' unless you have an existing Provisioning domain.
DEFAULT_DIRECTORY_HOST	Enter the hostname of the system with Provisioning Directory installed.
DEFAULT_DIRECTORY_PORT	Enter the port number of the system with the Provisioning Directory installed.
DEFAULT_DIRECTORY_PASSWORD	Enter the password for the Provisioning Directory.

Extensions for SiteMinder

To install the extensions for a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_ROOT	(Solaris Only) Enter the directory where the Policy Server is installed.
DEFAULT_USE_SITEMINDER	Enter true if you are using a SiteMinder Policy Server in your implementation.

Configuration File Format

The im-installer.properties file is located in the CA Identity Manager installation directory. For example:

- **Windows**: C:\Program Files\CA\CA Identity Manager\install_config_info
- **UNIX**: /opt/CA/IdentityManager/install_config_info/im-installer.properties

The following is an example of the im-installer.properties file created during a CA Identity Manager installation:

```
############################################################
### Silent input properties file for the IM R12.5SP7 installer ###
############################################################

# Component list
#   Valid values (comma-separated, one or
more):  Server,Exten,Admin,Provision,Directory
DEFAULT_COMPONENTS=Server,Admin,Provision,Directory

# Install folder
#   All products are installed in subfolders under this folder
#   This is parent product root selected by the user
#   For e.g. C:\\\\Program Files\\\\CA\\Identity Manager
DEFAULT_INSTALL_FOLDER=C:\\Program Files (x86)\\CA\\Identity Manager

#Generic login information
DEFAULT_GENERIC_USERNAME=imadmin
#DEFAULT_GENERIC_PASSWORD=<For silent install, insert generic user password here and
uncomment line.>

# Provisioning Server and Provisioning Directory Information.
#  Configure the Provisioning Server to a remotely installed Provisioning
Directory(true/false)
DEFAULT_CONFIG_REMOTE_PROVISIONING=false

 #Select the deployment type that suits your needs (1,2,3 or 4): 1. Compact 2. Basic
3. Intermediate (64 Bit only) 4. Large (64 Bit only)
DEFAULT_DEPLOYMENT_SIZE=1
DEFAULT_DIRECTORY_IMPS_HOSTNAMES=im-weblogic11
DEFAULT_DOMAIN_NAME=im
DEFAULT_DIRECTORY_HOST=im-weblogic11
DEFAULT_DIRECTORY_PORT=20394
#DEFAULT_DIRECTORY_PASSWORD=<For silent install, insert password to be used with
Provisioning Components here and uncomment line.>
```

```
#FIPS 140-2 Compliance mode (true/false) for Identity Manager, Admin Tools,
Provisioning Manager and Provisioning Server
DEFAULT_FIPS_MODE=false
#Complete path of the FIPS key file. For e.g. C:\\\\Program Files\\\\FIPSkey.dat
DEFAULT_FIPS_KEY_LOC=

#Identity Manager Application Server information
#   App Server
#   Valid values:  JBoss, Weblogic, WebSphere
DEFAULT_APP_SERVER=WebLogic
DEFAULT_APP_SERVER_URL=http://im-weblogic11.ca.com:7001

#Path to JDK for the JBOSS Application Server. No input required for other Application
Servers
DEFAULT_JAVA_HOME=C:\\PROGRA~1\\Java\\JDK16~1.0_2

#JBoss info
DEFAULT_JBOSS_FOLDER=C:\\Oracle\\Middleware
DEFAULT_JBOSS_PROFILE=
DEFAULT_JBOSS_SERVER_ID=

#Weblogic info
DEFAULT_BINARY_FOLDER=C:\\Oracle\\Middleware\\wlserver_10.3
DEFAULT_DOMAIN_FOLDER=C:\\Oracle\\Middleware\\user_projects\\domains\\base_domain
DEFAULT_SERVER_NAME=AdminServer

#For Weblogic 9 & 10 only:
DEFAULT_BEA_CLUSTER=

#WebSphere info
DEFAULT_WEBSPHERE_FOLDER=C:\\Oracle\\Middleware

#WAS_NODE Value: \\installedApps\\ or \\config\\cells\\\\nodes\\. These should be
same.
DEFAULT_WAS_NODE=

#WAS_SERVER Value: \\config\\cells\\\\nodes\\\\servers\\
DEFAULT_WAS_SERVER=

#WAS_CELL Value: \\config\\cells\\
DEFAULT_WAS_CELL=

#WAS_PROFILE Value: \\profiles\\
WAS_PROFILE=

#WAS_CLUSTER Value: \\config\\cells\\\\clusters\\
DEFAULT_WAS_CLUSTER=

#Policy Server info
```

```
DEFAULT_PS_HOST=localhost
DEFAULT_PS_USER=SiteMinder
#DEFAULT_PS_PW=<For silent install, insert PS Admin user password here and uncomment
line.>

#8.1 Migration
# SiteMinder Agent Name
DEFAULT_AGENT_NAME=imadmin
# SiteMinder Shared Secret
#DEFAULT_AGENT_PW=<For silent install, insert PS Shared Secret here and uncomment
line.>
# Automatically migrate. Valid values (true/false)
DEFAULT_MIGRATE_DIR_ENV=
# Directory to export to
DEFAULT_DIR_ENV_EXPORT=

#Policy Server Extensions info
#   Location of CsSmPs-<Instance name> folder
DEFAULT_PS_ROOT=
#You can use the SiteMinder Policy Server and a SiteMinder Web Agent to provide advanced
security
# for CA Identity Manager environments. Valid values (true/false)
DEFAULT_USE_SITEMINDER=false

#Database Info
DEFAULT_DB_HOST=im-weblogic11
DEFAULT_DB_PORT=1433
DEFAULT_DB_NAME=im
DEFAULT_DB_USER=sa
#DEFAULT_DB_PASSWORD=<For silent install, insert database password here and uncomment
line.>

#Following are permissible values: mssql2005 or oracle10
DEFAULT_DB_TYPE=mssql2005

#WAS Message Engine Database Info
DEFAULT_ME_HOST=
DEFAULT_ME_PORT=
DEFAULT_ME_NAME=
DEFAULT_ME_USER=
#DEFAULT_ME_PASSWORD=<For silent install, insert database password here and uncomment
line.>
DEFAULT_ME_SCHEMA=IBMWSSIB

#Upgrading from IM8.1sp2
#  If you have data stores located on separate servers or you wish to install them
on separate
# servers, you can specify them below. Otherwise if you wish to have all the data stores
on
```

```
#  the same server, change the DEFAULT_DB_* properties from above.

#Object Store Datastore Info
#DEFAULT_OS_DB_HOST=
#DEFAULT_OS_DB_PORT=
#DEFAULT_OS_DB_NAME=
#DEFAULT_OS_DB_USER=
#DEFAULT_OS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Task Persistence Datastore Info
#DEFAULT_TP_DB_HOST=
#DEFAULT_TP_DB_PORT=
#DEFAULT_TP_DB_NAME=
#DEFAULT_TP_DB_USER=
#DEFAULT_TP_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Audit Datastore Info
#DEFAULT_AUDIT_DB_HOST=
#DEFAULT_AUDIT_DB_PORT=
#DEFAULT_AUDIT_DB_NAME=
#DEFAULT_AUDIT_DB_USER=
#DEFAULT_AUDIT_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Reporting Snapshot Datastore Info
#DEFAULT_RS_DB_HOST=
#DEFAULT_RS_DB_PORT=
#DEFAULT_RS_DB_NAME=
#DEFAULT_RS_DB_USER=
#DEFAULT_RS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Workflow Datastore Info
#DEFAULT_WF_DB_HOST=
#DEFAULT_WF_DB_PORT=
#DEFAULT_WF_DB_NAME=
#DEFAULT_WF_DB_USER=
#DEFAULT_WF_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

# Automatically Upgrade Workflow DB
DEFAULT_UPGRADE_WF_DB=

# Automatically Migrate Task Persistance
DEFAULT_MIGRATE_TP=
```

Appendix D: Installation Log Files

The log files are stored based on where you unpack the installation package. The following examples may have different top-level directories than these default locations.

This section contains the following topics:

Log Files on Windows

If you encounter issues during CA Identity Manager installation, see this log file:

C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\caiamsuite.log

The Identity Manager Server installer logs are written to the following default locations:

- C:\Program Files\CA\Identity Manager\install_config_info (32-bit system)
- C:\Program Files (x86)\CA\Identity Manager\install_config_info (64-bit system)

The Provisioning installer logs are written to the user's Temp directory and copied to the *Install-Directory*_uninst directory.

Example:

C:\Documents and Settings*user*\Local Settings\Temp\imps_server_install.log

Log files on UNIX

If you encounter any issues while performing a CA Identity Manager installation, see the caiamsuite.log file in this location:

/opt/CA/IdentityManager/

The Identity Manager Server installer logs are written to the following default location:

/opt/CA/IdentityManager/install_config_info

The Provisioning installer logs are written to the user's Temp directory.

Appendix E: Windows Services Started by CA Identity Manager

The following are the services started on Windows when you install and start all components of CA Identity Manager:

- CA Directory *hostname*-impd-co
- CA Directory impd-inc
- CA Directory impd-main
- CA Directory impd-notify
- CA Directory impd-router
- CA Identity Manager Connector Server (C++)
- CA Identity Manager Connector Server (Java)
- CA Identity Manager Provisioning Server
- Enterprise Common Services (Transport)
- Enterprise Common Services GUI Framework
- Enterprise Common Services Store-And-Forward Manager

This list of services may useful to you for troubleshooting purposes.

Index

I

L

M

N

O

P

R

S

T

U

V

W