HP OpenView Select Identity

Installation Guide

Software Version: 3.3

UNIX® (Sun Solaris, HP-UX, Red Hat Enterprise Linux) and Windows® Operating Systems

April 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

Select Identity uses software from the Apache Jakarta Project including:

- Commons-beanutils.
- Commons-collections.
- Commons-logging.
- Commons-digester.
- Commons-httpclient.

- Element Construction Set (ecs).
- Jakarta-poi.
- Jakarta-regexp.
- Logging Services (log4j).

Additional third party software used by Select Identity includes:

- JasperReports developed by SourceForge.
- iText (for JasperReports) developed by SourceForge.
- BeanShell.
- Xalan from the Apache XML Project.
- Xerces from the Apache XML Project.
- Java API for XML Processing from the Apache XML Project.
- SOAP developed by the Apache Software Foundation.
- JavaMail from SUN Reference Implementation.
- Java Secure Socket Extension (JSSE) from SUN Reference Implementation.
- Java Cryptography Extension (JCE) from SUN Reference Implementation.
- JavaBeans Activation Framework (JAF) from SUN Reference Implementation.
- OpenSPML Toolkit from OpenSPML.org.
- JGraph developed by JGraph.
- Hibernate from Hibernate.org.
- BouncyCastle engine for keystore management, bouncycastle.org.

Trademark Notices

Support

Please visit the HP OpenView web site at:

http://www.managementsoftware.hp.com/

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

You can also go directly to the support web site at:

http://support.openview.hp.com/

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://support.openview.hp.com/access_level.jsp

To register for an HP Passport ID, go to:

https://passport.hp.com/hpp2/newuser.do

contents

1

Welcome to Select Identity

HP OpenView Select Identity is the first truly scalable solution for managing identity within and between large enterprises. The Select Identity solution automates the process of provisioning and managing user accounts and access privileges across platforms, applications, and corporate boundaries. Along with robust workflow, user self-service, reporting, and delegated administration capabilities, Select Identity is the most comprehensive identity management system available.

Select Identity is designed to address the formidable challenges of managing identity within complex, multi-organizational business processes. Traditional identity management systems employ the user-centric model of roles to distribute access to users. In an extended enterprise, roles proliferate exponentially to accommodate the large number of complex business relationships that exist between users, organizations, resources and security policies.

Select Identity's Contextual Identity Management (CIM) ™ is a dramatic advancement in identity management. CIM provides a service-centric approach to managing identity. In any company, its employees, customers, and partners participate in a number of services or business processes that comprise the operation of the company. For example, these processes might include "order processing" or "accounts receivable." Each Service may consist of a number of applications or resources that require unique access privileges depending on the Service, its participants, and corporate policy. CIM incorporates these complex relationships and leverages them to automate the tasks associated with managing identity, including the following:

- Provisioning accounts and privileges
- Approving workflows
- Delegating administrative rights
- Enforcing security policy
- Reporting

CIM mitigates the limitations of the traditional role and rule-based identity management, enabling scalability throughout the extended enterprise while reducing deployment times and management costs.

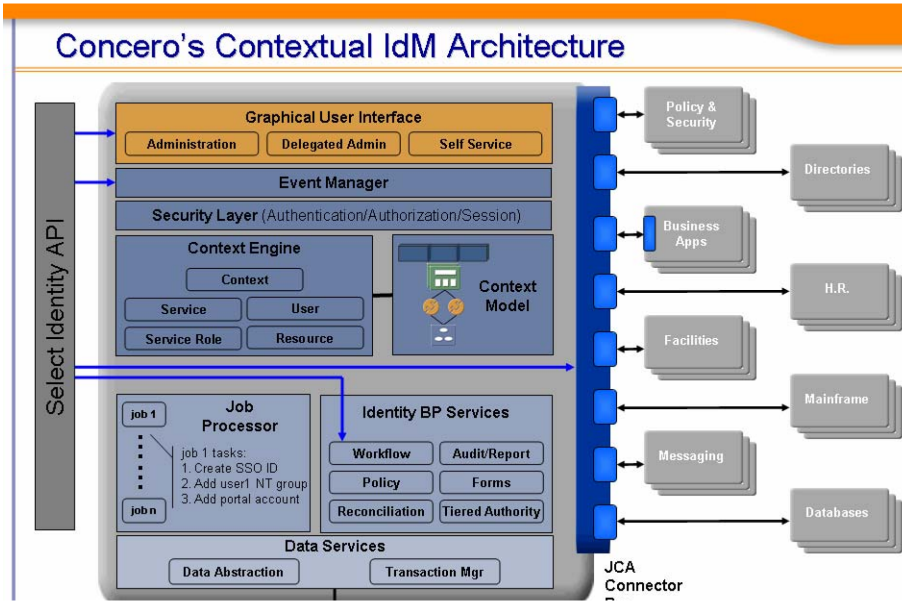
Key features of Select Identity include the following:

- Centralized Management – Provides a single point of control for the management of users and entitlements
- Provisioning – Automates the creation, update, and deletion of accounts and entitlements on information systems across the enterprise
- Extreme Delegation – Enables administrative rights to be distributed to multiple tiers of functional departments, customers, and partners
- User Self-Service – Enables end users to initiate access to services, change passwords, set password hints, and update general identity information through a simple web interface
- Workflow – Automates identity-related processes such as access approval and provisioning, and integrates these processes with other business processes
- Password and Profile Management – Manages and distributes password and user profile information across and between enterprise information systems
- Audit and Reporting – Provides standardized and on-demand reporting on permissions, actions, and user account activity

With Select Identity, provisioning and management of user accounts and privileges is no longer a barrier to realizing the efficiencies and competitive advantage of extending system access to ever greater numbers of employees, customers and partners.

System Architecture

The following illustration provides a high-level view of the Select Identity system and its components.



Concero's Contextual IdM Architecture

Select Identity API

Graphical User Interface

Administration

Delegated Admin

Self Service

Event Manager

Security Layer (Authentication/Authorization/Session)

Context Engine

Context

Service

User

Service Role

Resource

Context Model

job 1

job n

Job Processor

job 1 tasks:
1. Create SSO ID
2. Add user1 NT group
3. Add portal account

Identity BP Services

Workflow

Audit/Report

Policy

Forms

Reconciliation

Tiered Authority

Data Services

Data Abstraction

Transaction Mgr

JCA Connector

Policy & Security

Directories

Business Apps

H.R.

Facilities

Mainframe

Messaging

Databases

All requests to and from the system use the HTTP protocol. Select Identity manages a single, logical identity for each user and administrator. These logical identities are mapped to the users' various accounts on back-end systems and services. The logical identities, as well as their corresponding accounts and privileges, are governed by Select Identity system functions and permissions. Accounts are also governed by security policies that are defined by an administrator based on the access requirements of the company's products and services.

The Context Engine and Identity Business Process Services components of the Select Identity architecture are of particular importance to administrators and personnel responsible for deploying and maintaining the Select Identity system. These components contain the functions that administrators use most. These functions include the following:

- **Context Management**

  Maintains the Context structure that defines identities and access for all users and resources in the extended enterprise.

- **Services**

  Provides a business-centric abstraction over resources, entitlements, and other identity-related entities. Services represent the products and services that you offer to customers, partners, and employees.

- **Service Roles**

  Provides granular control over how groups of users access services.

- **Users**

  Provides consistent account creation and management across products and services.

- **Resources**

  Provides a connection to the physical information systems on which your products and services rely for user account data.

- **Workflow Studio**

  Enables the definition of identity-related business processes that can be executed for access to services or any other event within the Select Identity system.

- **Reconciliation**

  Ensures the proper coordination of provisioning workflow across multiple resources.

- **Auditing and Reporting**

  Provides robust standard and custom reporting facilities over user entitlements and system event history.

- **Forms**

  Automates the creation of electronic forms used by end users to register for access to services, change their passwords, set password hints, and update personal information.

- **Tiered Authority**

  Enables the secure, multi-tiered delegation of administrative tasks, such as management of identity profiles and entitlements, to functional departments, customers, and partners.

Leveraging an open, standard, J2EE Connector Architecture (JCA) bus, Select Identity uses predefined connectors to access back-end system data stores. Connectors are configured during the installation process and are easy to deploy. If you wish to create your own connectors, Select Identity offers a software developer's kit (SDK) that enables you to do so.

Security and Communication

Select Identity encrypts application data in transit and storage. Data that is in-transit is encrypted using SSL. For in-storage encryption, Select Identity uses the standard encryption algorithm, SHA-1 hash. The algorithm guarantees that the same message (input) will produce the same message digest. Thus, at any given time, you can verify that the input (such as a password) is the same as the original value by comparing the hash value. In addition, it is recommended that you tighten your database access control and ensure all passwords are complex.

Select Identity also enables you to generate a keystore, which encrypts and decrypts application data. A keystore is a file that contains security information such as public and private keys, and certificates of trusted Certification Authorities. The private keys are associated with a certificate chain, which authenticates the corresponding public key. By generating the keystore, you add another layer of security to the data that is exchanged in the Select Identity system.

The connectors that enables you to provision users in external resources are built using JCA (J2EE Connector Architecture) and run within the application server on which Select Identity relies. Communication between Select Identity and the connectors is internal to the application server. The connectors then use the appropriate protocol or means of communication for each resource. For example, the LDAP connector uses the JNDI (Java Naming and Directory Interface) API to address the LDAP stores. For Active Directory (LDAP-based), the connector uses LDAPS (LDAP over SSL). For UNIX-based connectors, provisioning commands are executed through a Telnet session or over SSH.

For agent-based connectors, each agent resides on the resource with which the connector communicates. The messages exchanged between the connector and the agent are based on a non-standard proprietary XML format and encrypted using 128-bit PC1 encryption. The agent communicates internally with the resource application.

Internationalization

The Select Identity application is internationalized in this release, though not localized. This means that Select Identity encodes its data internally using Unicode. The Select Identity server is supported on WebLogic and WebSphere, though only Oracle is supported as its database in a non-US environment with internationalization encoding. In addition, the LDAP connectors are internationalization encoded. The LDAP connectors rely on the JNDI resource provider interface to exchange information with the LDAP resources.

➤ Internationalization is not supported for WebSphere installations at this time.

For more information about the internationalized Select Identity, see Internationalization and Localization on page 89.

Product Documentation

The Select Identity product documentation includes the following:

- Release notes are provided in the top-level directory of the HP OpenView Select Identity CD. This document provides important information about new features included in this release, known defects and limitations, and special usage information that you should be familiar with before using the product.

- For installation and configuration information, refer to the *HP OpenView Select Identity Installation Guide*. All installation prerequisites, system requirements, and procedures are explained in detail in this guide. Specific product configuration and logging settings are included. This guide also includes uninstall and troubleshooting information.
- An *HP OpenView Connector Installation and Configuration Guide* is provided for each resource connector. These are located on the Select Identity Connector CD.
- Detailed procedures for deployment and system management are documented in the *HP OpenView Select Identity Administrator Guide* and Select Identity online help system. This guide provides detailed concepts and procedures for deploying and configuring the Select Identity system. In the online help system, tasks are grouped by the administrative functions that govern them.
- The *HP OpenView Select Identity Workflow Studio Guide* provides detailed information about using Workflow Studio to create workflow templates. It also describes how to create reports that enable managers and approvers to check the status of account activities.
- The *HP OpenView Select Identity External Call Developer Guide* provides detailed information about creating calls to third-party applications. These calls can then be deployed in Select Identity to constrain attribute values or facilitate workflow processes. In addition, JavaDoc is provided for this API. To view this help, extract the `javadoc.jar` file in the `docs/api_help/external_calls/Javadoc` directory on the HP OpenView Select Identity CD.
- If you need to develop connectors, which enable you to connect to external systems for provisioning, refer to the *HP OpenView Select Identity Connector Developer Guide*. This document provides an overview of the Connector API and the steps required to build a connector. The audience of this guide is developers familiar with Java.

  JavaDoc is also provided for the Connector API. To view this help, extract the `javadoc.jar` file in the `docs/api_help/external_calls/Javadoc` directory on the HP OpenView Select Identity CD.
- The *HP OpenView Select Identity Web Service Developer Guide* describes the Web Service, which enables you to programatically provision users in Select Identity. This guide provides an overview of the operations you can perform through use of the Web Service, including SPML examples for each operation.

An independent, web-based help system is available for this API. To view this help, double-click the `index.htm` file in the `docs/api_help/web_service/help` directory on the HP OpenView Select Identity CD.

- The *HP OpenView Select Identity Attribute Mapping Utility User's Guide* describes how to access the Attribute Mapping Utility, provides an overview to the utility's user interface, and describes how to define user and entitlements mappings. This guide is provided on the Select Identity Connector CD and is for use with the SQL and SQL Admin connectors only.

2

System Requirements

The following provides an overview of the Select Identity server installation process:

- Configure the database and load the Select Identity schema
- Configure the web application server for use with Select Identity
- Install Select Identity
- Generate a keystore and configure the Select Identity server; perform this optional procedure if you wish to encrypt and decrypt two-way data in Select Identity using your keystore
- Configure the properties file to customize the settings for your environment

Refer to related sections for system requirements, prerequisite steps, and installation procedures.

➤ Select Identity can leverage the clustering capabilities of the application servers to support high throughput and fault tolerance. Multiple copies of Select Identity can also be installed and can work together when they are connected to the same database. See Clustering Environments on page 122 for details about configuring clustering properties.

Select Identity is supported on the following configurations:

Application Server	Platform	Database
BEA WebLogic Server 8.1.4	Windows Server$^{\text{TM}}$ 2000 with service pack 3 Windows Server$^{\text{TM}}$ 2003	Microsoft SQL Server 2000
	Sun Solaris 9	Oracle 9i
	HP-UX 11i V1	
	Red Hat Enterprise Linux v3	
IBM WebSphere Application Server 5.1.1	Sun Solaris 9, with latest patches	IBM DB2 8.1, with service pack 7 or DB2 8.2
IBM WebSphere Application Server Network Deployment Version 5.1.1	Sun Solaris 9, with latest patches	IBM DB2 8.1, with service pack 7 or DB2 8.2

It is *strongly* recommended that the database server and web application server be installed on separate systems, if possible, for optimal performance and ease of management.

The following sections provide an overview of the *minimum* requirements for the systems required to support Select Identity. Recommend parameters are noted where possible.

Database Server Requirements

Microsoft SQL Server 2000	
Version and edition	Microsoft SQL Server 2000, Enterprise Edition
Operating system	Windows Server 2000 with service pack 3 Windows Server 2003, Standard Edition SP3 Windows Server 2003, Enterprise Edition SP3 Windows Server 2003, Datacenter Edition SP3
Processor	Intel Pentium or compatible, 166 megahertz (MHz) or higher processor
Memory (RAM)	Enterprise Edition: 64 megabytes (MB) of RAM; 128 MB recommended
Disk space	95 - 270 MB of available hard disk space for the server; 250 MB for a typical installation
JDBC driver	BEA MS SQL Server Type 4 driver (classname is weblogic.jdbc.sqlserver.SQLServerDriver)

Oracle 9i	
Version	Oracle Database, version 9.2
Operating system	HP-UX 11i V1 (PA-RISC) Sun Solaris 9 Red Hat Enterprise Linux v3
Processor	Minimal Processor: 330 MHz
Memory (RAM)	512 MB of physical RAM, 1 GB of swap space (or twice the size of RAM)
Disk space	3.5 GB
JDBC driver*	BEA Oracle Type 4 driver (classname is weblogic.jdbc.oracle.OracleDriver)

* If you use the BEA Oracle driver, you must set the `truaccess.repository.oracle.driver.bea` property in the `TruAcccess.properties` file to `yes`, as documented in Customizing TruAccess.properties on page 85.

Web Application Server Requirements

BEA WebLogic	
Version	BEA WebLogic Server, version 8.1 with service pack 4
Operating system	HP-UX 11i V1 (PA-RISC) Microsoft Windows Server 2000 with Service pack 3 Microsoft Windows Server 2003 Standard, Enterprise, and Datacenter Red Hat Enterprise Linux v3 Sun Solaris 9 on SPARC with service pack 4
Processor	1 GHz CPU is recommended
Memory (RAM)	512 MB of RAM minimum 1 GB RAM recommended
Disk space	approximately 820 MB of disk space

IBM WebSphere	
Version	IBM WebSphere Application Server, version 5.1.1+, with latest patches
Operating system	Sun Solaris 9 on SPARC
Processor	Solaris SPARC workstation at 440 MHz, or faster

IBM WebSphere	
Memory (RAM)	512 MB RAM minimum 1 GB RAM recommended
Disk space	Minimum 580 MB free disk space for installation (includes SDK)

DB2 on Solaris	
Version	IBM DB2 8.1, with service pack 7 or DB2 8.2
Operating system	Sun Solaris 9 with the latest patch cluster
Processor	UltraSPARC, 330 MHz
Memory (RAM)	512 MB minimum, 1 GB recommended
Disk space	3 GB
JDBC driver	The DB2 Legacy CLI-based Type 2 JDBC driver (classname is COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource)

When installing in a clustered environment, make sure that the DB2 Runtime Client, version 8.2, is installed on each of the nodes in the cluster. The client should also contain a user directory such as `/export/home/db2inst1`.

Select Identity Interface Requirements

The Select Identity interface requires Microsoft Internet Explorer (IE), version 5.5 or higher, with JavaScript and cookies enabled. No installation steps are required to install the Select Identity interface. The web application server that is configured for Select Identity will serve the Select Identity interface pages.

Ports Required for Firewall Configuration

The Select Identity server uses the following ports for communication. These are the default values, some of which can be changed during installation.

- The application server's (TCP/IP) port for all in-bound communication:
  - 7001 for WebLogic
  - 9080 for WebSphere

  However, if a web server is configured to redirect requests to the Select Identity server, any other TCP/IP port may be used to mask the server URL, including its port.

- The JDBC port, which depends on the database server:
  - 1521 for Oracle
  - 1433 for SQL Server
  - 6789 for DB2

If you are installing connectors, additional ports are needed to send requests from the connector to the target resource. For example, the LDAP connectors use port 389 (LDAP) or 636 (LDAPS), and the UNIX connectors port 23 (Telnet) or 22 (SSH). Refer to the documentation for the target resource to determine what the standard communication port is for each.

➤ If you are installing in a clustered environment, each of the servers in the cluster may use different HTTP transport ports. This may require firewall configuration. Select Identity recommends configuring a web server to mask the web container ports.

3

Configuring the Database Server

To enable Select Identity to store its data in a database, you must load the schema into the chosen database server. Before loading the schema, you or the database administrator must ensure that the database server meets the *minimum* requirements. Then, create a database and user that Select Identity can use to access the database server. The following sections provide requirements and procedures for each database server.

Microsoft SQL Server

You can create a database for use by Select Identity by running SQL scripts.

➤ Make sure that the SQL Server database is configured to be case-insensitive.

An important step in configuring the database takes place after you configure the web application server. You must ensure that the `truaccess.repository.type` entry in the `TruAccess.properties` file is set to **`mssql`**. The default setting is for oracle. See Customizing TruAccess.properties on page 85 for more information about this file.

Complete the following to create a SQL Server database:

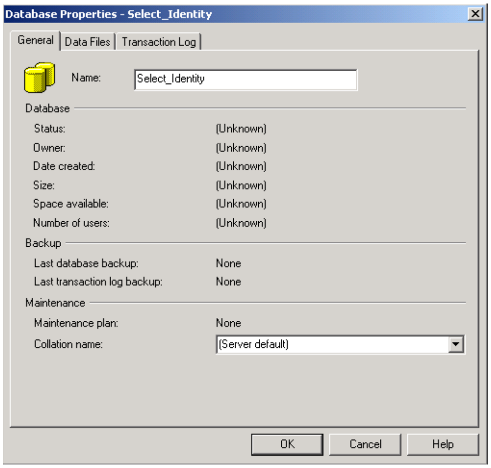
1. Create a directory on the server that will serve as Select Identity's home directory on the SQL Server system, such as `C:\Select_Identity` (on Windows).

2. Copy the `concero_ddl.sql` and `concero_dml.sql` files from the `Database` directory on the Select Identity CD to the Select Identity home directory on the SQL Server system.

1. Log in to the Microsoft SQL Server Enterprise Manager interface.

2. In Enterprise Manager, expand **Microsoft SQL Server** → **SQL Server Group** → ***server***, where ***server*** is the name of the SQL Server instance.

3. Right-click **Databases**, and select **New Database...**.



Database Properties - Select_Identity

Name: Select_Identity

Database

Status: (Unknown)

Owner: (Unknown)

Date created: (Unknown)

Size: (Unknown)

Space available: (Unknown)

Number of users: (Unknown)

Backup

Last database backup: None

Last transaction log backup: None

Maintenance

Maintenance plan: None

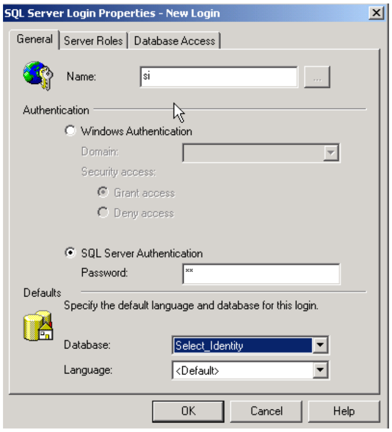
Collation name: (Server default)

4. Enter a name for the database, such as **`Select_Identity`**. Click **OK** to finish creating the database.

5. Create a database user that can be used to manage the Select Identity database. Complete the following steps to do so:

   a. Select the **Microsoft SQL Server** → **SQL Server Group** → ***server*** → **Security** folder in the Enterprise Manager tree.

b Create a new login for the new database by right-clicking on **Logins** and selecting **New Login**. The SQL Server Login Properties dialog displays.



SQL Server Login Properties - New Login

Name: si

Authentication

○ Windows Authentication

Domain:

Security access:

● Grant access

○ Deny access

● SQL Server Authentication

Password: **

Defaults

Specify the default language and database for this login.

Database: Select_Identity

Language: <Default>

c On the **General** tab, enter a user name such as **`SI`**, enter a password, and select **SQL Server Authentication** as the authentication type.

d Select the new database (Select_Identity) from the Database list. Keep remaining default settings.

e Click **OK**. You are prompted to confirm your password.

f Select the **Database Access** tab on the SQL Server Login Properties dialog.

g Select the **Permit** check box next to the Select Identity database user.

h Assign the **db_owner** and **public** permissions to the new user.

i Click **OK** to save your settings.

6 Create the schema for the Select Identity database by following these steps:

a Launch the SQL Query Analyzer by selecting **Tools –> SQL Query Analyzer**.

b Select the new database (`SI`) from the DB drop-down list.

c Load the `concero_ddl.sql` SQL script from the Select Identity home directory you created in Step 2 on page 24.

– Click the Open icon.

- Locate the Select Identity home directory.
- Select the `concero_ddl.sql` file.
- Click **Open**.

d. Run the script by clicking the **Execute Script** or play button.

e. Verify that an error message is not displayed.

7. Insert the required default data into the Select Identity database by performing the following:

   a. Clear the previous script by clicking the **Clear Query Window** button.

   b. Load the `concero_dml.sql` SQL script from the directory you created in Step 2 on page 24.

   c. Run the script clicking the **Execute Script** button.

      Messages in the console indicate that rows are being created.

   d. Verify that an error message is not displayed.

   e. Close the SQL Query Analyzer and the Microsoft SQL Server Enterprise Manager.

The next step is to configure the web application server before installing Select Identity, as described in Configuring the Application Server on page 30.

Oracle

You can create a database for use by Select Identity by running SQL scripts.

An important step in configuring the database takes place after you configure the web application server. You must ensure that the `truaccess.repository.type` entry in the `TruAccess.properties` file is set to **`oracle`**. See Customizing TruAccess.properties on page 85 for more information about this setting.

Complete the following to create the database:

1. Copy the `oracle_concero_ddl.sql` and `oracle_concero_dml.sql` files from the `Database` directory on the Select Identity CD to a directory on the Oracle server.

2 Launch SQL Plus. You can perform the following steps using the Enterprise Manager Console. However, the SQL Plus steps documented below are valid on Solaris and Windows.

3 Create tablespace into which you will load the Select Identity tables.

```
CREATE TABLESPACE tablespace_name

DATAFILE 'install_dir/oracle/oradata/SID/
tablespace_name.dbf'

SIZE 10M

AUTOEXTEND ON NEXT 10M

MAXSIZE unlimited;
```

where *tablespace_name* is the chosen name for the Select Identity tablespace. You will reference this name when creating the database user. This command creates 10MB of tablespace then automatically extends the tablespace as needed.

4 Create a user to be used by Select Identity to access the tables:

```
CREATE USER user_name PROFILE DEFAULT

IDENTIFIED BY password DEFAULT TABLESPACE tablespace_name

ACCOUNT UNLOCK;

GRANT CONNECT TO user_name;

GRANT RESOURCE TO user_name;
```

where *user_name* is the name of the database user to be created, *password* is the user's password, and *tablespace_name* is the name of the tablespace to be used assigned as the user's default tablespace.

➤ The `oracle_concero_ddl.sql` script, which is run in the following step, inserts tables into the user's default tablespace. If you do not assign the Select Identity tablespace as the user's default tablespace, you must edit the script to reference the Select Identity tablespace.

5 Change to the newly created user by entering the following command:

```
CONNECT user_name/password
```

6 Create the schema for the Select Identity database by performing the following steps:

   a Execute the schema creation script by running the following:

   @*path*/oracle_concero_ddl.sql

   where *path* is the full path to the file. If Oracle is installed on Windows, you must include the drive letter in this path.

   b Verify that no error message is displayed.

7 Insert the required default data into the Select Identity database:

   a Run the data creation script by entering the following command:

   @*path*/oracle_concero_dml.sql

   where *path* is the full path to the file. If Oracle is installed on Windows, you must include the drive letter in this path.

   b Verify that no error message is displayed.

The next step is to configure the web application server before installing Select Identity, as described in Configuring the Application Server on page 30.

DB2

Complete the following steps to configure DB2 for use with Select Identity.

An important step in configuring the database takes place after you configure the web application server. You must ensure that the `truaccess.repository.type` entry in the `TruAccess.properties` file is set to **`db2`**. See Customizing TruAccess.properties on page 85 for more information about this setting.

1 Use the DB2 Control Center to create a database instance and database for use by Select Identity. The login account that you use to create the database must be the schema owner.

2 Using the database name, login ID, password, and instance port number, log in to the database where you will load the Select Identity schema.

3 Create a directory in the home directory of the database.

4 Copy the `db2_concero_ddl.sql` and `db2_concero_dml.sql` files from the `Database` directory on the Select Identity CD to the directory you created.

5 Connect to DB2 database:

```
db2 connect to db_name user user_name using user_pwd
```

6 Run the DDL and DML scripts, which create the schema and populate the database with Select Identity data:

```
db2 -v -td@ -f db2_concero_ddl.sql
```

```
db2 -v -td@ -f db2_concero_dml.sql
```

7 Verify that no error message is displayed.

The next step is to configure the web application server before installing Select Identity, as described in Configuring the Application Server on page 56.

4

Installing on WebLogic Servers

Select Identity relies on the web application server to serve the Select Identity interface pages, communicate with the database server to store and retrieve data, and send email based on an action performed through the Select Identity interface. The following sections describe how to install and configure Select Identity on a WebLogic server.

The following procedures assume that WebLogic is already installed and configured, and that at least one domain has been created.

➤ You may need to increase the JTA time-out to a minimum of 300 seconds on WebLogic for Reconciliation to work properly.

Configuring the Application Server

You can configure a BEA WebLogic application server for use with Select Identity.

➤ Select Identity supports clustered servers through the application server layer. See the application server documentation for information on clustered servers. See Using the Installation Wizard for Clustered Servers on page 40 for Select Identity installation procedures.

Single Server Prerequisites

To configure a single WebLogic server for use with Select Identity, complete the following steps.

These steps assume that WebLogic is installed and running.

1 Ensure that the system where WebLogic is installed meets the *minimum* requirements, as documented in System Requirements on page 17.

2 If necessary, upgrade the Java encryption level that WebLogic uses to 128-bit. To determine whether you need to upgrade the encryption level of your server, note the size of the `local_policy.jar` and `US_export_policy.jar` files in the `$BEA_HOME/jdk142_05/jre/lib` directory for HP-UX and Solaris systems, or in the `$BEA_HOME/jrockit81sp4_142_05/jre/lib/` for Linux systems.

   If they are 3KB, you need to upgrade the encryption. The files for 128-bit encryption are 5KB.

   Refer to **http://java.sun.com/j2se/1.4.2/download.html** (click **DOWNLOAD** next to Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 1.4.2 under Other Downloads). Refer to the readme file that is downloaded to confirm which files to replace.

3 Copy the following files from the `library` directory on the Select Identity product CD to `$BEA_HOME/jdk142_05/jre/lib/ext` for HP-UX and Solaris systems, or to `$BEA_HOME/jrockit81sp4_142_05/jre/lib/ext` for Linux systems:

   `commons-logging.jar`

   `commons-httpclient-2.0.jar`

   `commons-digester-1.3.jar`

Clustered Server Prerequisites

These steps assume that theWebLogic administration server is installed and running and that the cluster and its nodes are defined. To facilitate the copying of files to multiple servers, use NFS or a shared drive. Files should be copied to the same location on each node in the cluster.

1 Ensure that the systems where WebLogic is installed meet the *minimum* requirements, as documented in System Requirements on page 17.

2 The cluster installation is run from the administrative server in the cluster. The node managers for each node should be running during the installation and the managed servers should be stopped.

3 If necessary, upgrade the Java encryption level that WebLogic uses to 128-bit. To determine whether you need to upgrade the encryption level of your server, note the size of the `local_policy.jar` and `US_export_policy.jar` files in the `jdk142_05/jre/lib/security` directory for HP-UX and Solaris systems, or in the `jrockit81sp4_142_05/jre/lib/security` directory for Linux systems. If they are 3KB, you need to upgrade the encryption. The files for 128-bit encryption are 5KB.

   Refer to **http://java.sun.com/j2se/1.4.2/download.html** (click **DOWNLOAD** next to Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 1.4.2 under Other Downloads). Refer to the readme file that is downloaded to confirm which files to replace.

4 Copy the following files from the `library` directory on the Select Identity product CD to `$BEA_HOME/jdk142_05/jre/lib/ext` for HP-UX and Solaris systems or to `$BEA_HOME/jrockit81sp4_142_05/jre/lib/ext/` for Linux systems, on the administrative server and each node in the cluster:

   `commons-logging.jar`

   `commons-httpclient-2.0.jar`

   `commons-digester-1.3.jar`

5 Copy the following files to the administrative server and each node in the cluster from the Select Identity product CD. You can choose any location within the WebLogic installation path. You will need to reference each node location when installing Select Identity.

   `TruAccess.properties` – located in the `WebLogic/properties` directory on the Select Identity CD.

   `connector.jar` – located in the `WebLogic/Connector` directory on the Select Identity CD.

Installing on the Application Server

This section provides procedures for installing Select Identity using the installation wizard or by manually configuring the application server. Before completing these steps, be sure to complete the procedures included in the Configuring the Application Server section; the wizard and manual steps require this.

Using the Installation Wizard for a Single Server

The following sections describe how to install Select Identity on the WebLogic application server. The installation wizard performs the following by default:

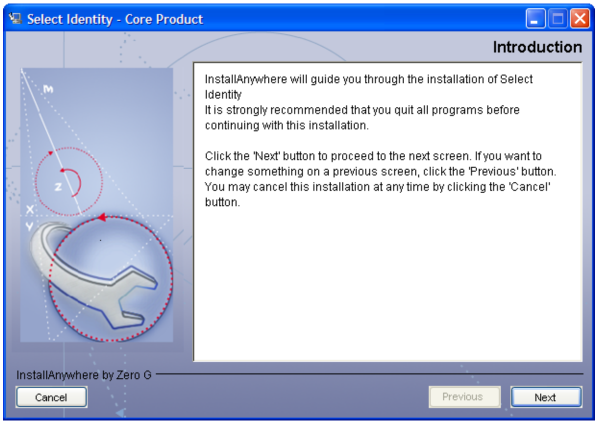
- Copies the server files into the specified home directory
- Creates a JDBC connection pool (on WebLogic) called SI_Connection_Pool
- Creates a data source called SI_Data_Source
- Creates a mail session called SI_MAIL_Session
- Deploys the EAR file
- Configures the Select Identity server with your specified settings

Complete the following steps to configure WebLogic for use with Select Identity and to install Select Identity on the WebLogic Server:

1. Log on to the operating system as the same user that was used to install WebLogic.
2. *On Windows:*
   Insert the Select Identity CD, navigate to the `SI33\install\WebLogic\Windows` directory, and double-click the `install.exe` file. The InstallAnywhere wizard displays.

   On UNIX:
   Mount the CD, change directories to `SI33\install\WebLogic\Solaris`, `Linux` or `HPUX`, and run the `install.bin` executable.

The InstallAnywhere wizard displays.



Select Identity - Core Product

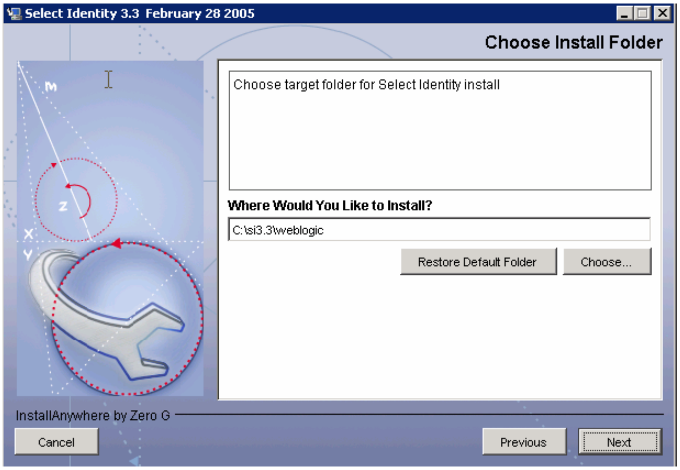
Introduction

InstallAnywhere will guide you through the installation of Select Identity
It is strongly recommended that you quit all programs before continuing with this installation.

Click the 'Next' button to proceed to the next screen. If you want to change something on a previous screen, click the 'Previous' button. You may cancel this installation at any time by clicking the 'Cancel' button.

3 Click **Next**. The License Agreement page displays.

4 Read the license agreement and select **I accept the terms of the License Agreement** option, if you agree. If you do not select this option, you cannot install the software. Then, click **Next**. The Choose Install Folder page displays.



Select Identity 3.3 February 28 2005

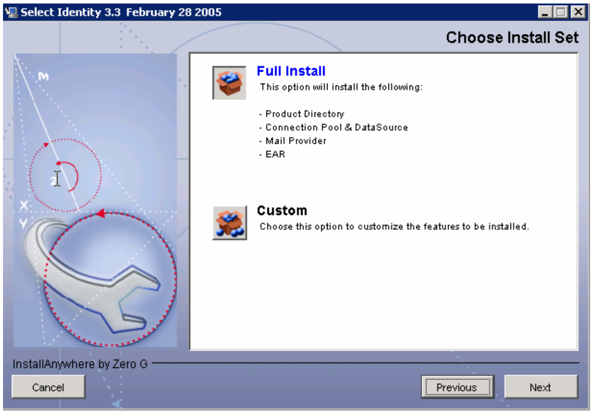
Choose Install Folder

Choose target folder for Select Identity install

Where Would You Like to Install?

C:\si3.3\weblogic

5 Enter or browse to a path where you would like to install the Select Identity files. This folder will serve as the Select Identity home directory. Then, click **Next**. The Choose Install Set page displays.



Select Identity 3.3 February 28 2005

Choose Install Set

Full Install

This option will install the following:

- Product Directory
- Connection Pool & DataSource
- Mail Provider
- EAR

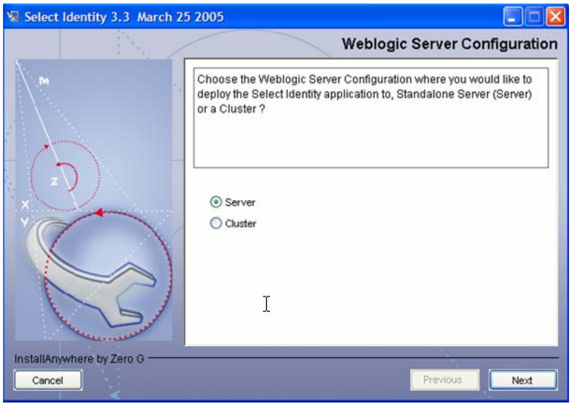
Custom

Choose this option to customize the features to be installed.

6 Select **Full Install** to accept all of the default configuration or **Custom** to choose your components.

7 Click **Next**. The Pre-Installation Summary page displays.

8 Review the chosen settings. If you wish to change a setting, click **Previous**. When you are satisfied with the settings, click **Install**. The Weblogic Server Configuration page displays.



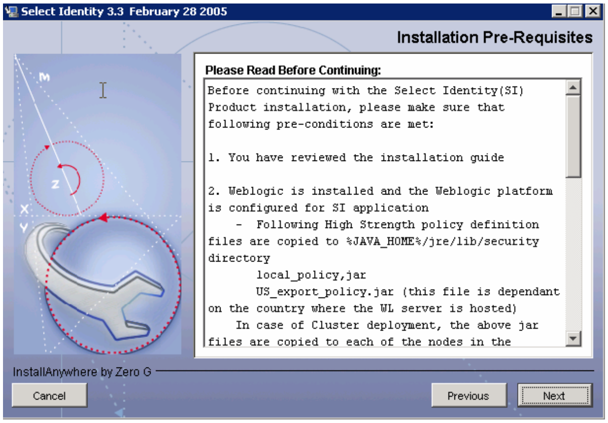
Select Identity 3.3 March 25 2005

Weblogic Server Configuration

Choose the Weblogic Server Configuration where you would like to deploy the Select Identity application to, Standalone Server (Server) or a Cluster ?

● Server
○ Cluster

9 Select Server for Standalone Server installation and click **Next**. The Installation Pre-Requisites page displays.



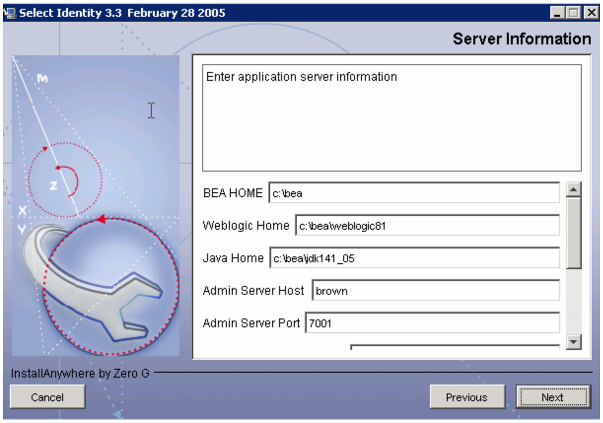
Select Identity 3.3 February 28 2005

Installation Pre-Requisites

Please Read Before Continuing:

```
Before continuing with the Select Identity(SI)
Product installation, please make sure that
following pre-conditions are met:

1. You have reviewed the installation guide

2. Weblogic is installed and the Weblogic platform
is configured for SI application
    -  Following High Strength policy definition
files are copied to %JAVA_HOME%/jre/lib/security
directory
       local_policy,jar
       US_export_policy.jar (this file is dependant
on the country where the WL server is hosted)
    In case of Cluster deployment, the above jar
files are copied to each of the nodes in the
```

10 Verify that the pre-requisites have been met before continuing. Click **Next**. The Server Information page displays.



Select Identity 3.3 February 28 2005

Server Information

Enter application server information

BEA HOME: c:\bea

Weblogic Home: c:\bea\weblogic81

Java Home: c:\bea\jdk141_05

Admin Server Host: brown

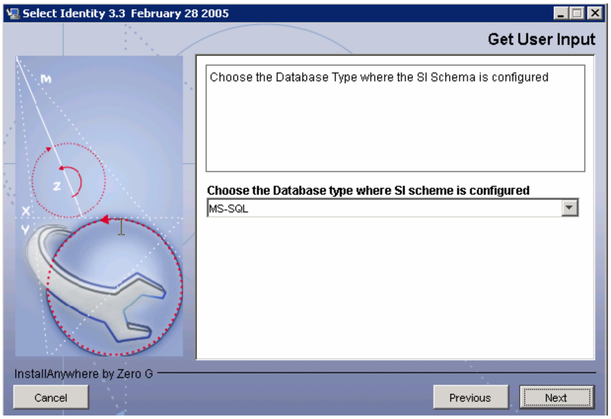
Admin Server Port: 7001

11 Specify settings that are set in the WebLogic Server, as follows:

- WebLogic Home — The directory where WebLogic is installed
- Java Home — The directory where the JDK is installed
- Server Host — The name of the local host (where WebLogic is installed)
- Server Port — The port used by Select Identity

- Server Domain — The domain configured in WebLogic where Select Identity will be installed
- Admin Server Login Name — The user name of the WebLogic administrative account
- Admin Server Password — The password of the Admin account
- Server Name — The name of the server created in WebLogic and on which you are installing Select Identity

Then, click **Next**. The Database Type Selection page displays.



Select Identity 3.3 February 28 2005

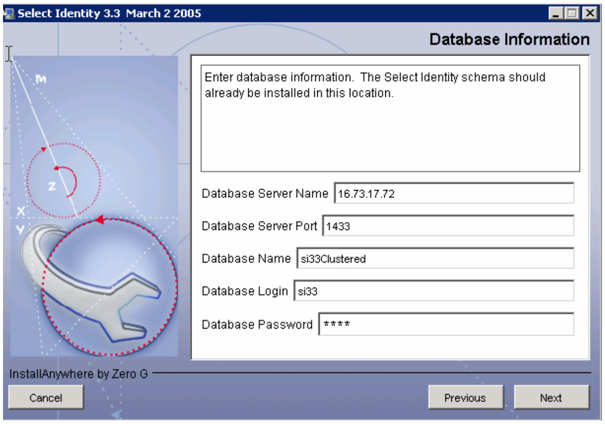
Get User Input

Choose the Database Type where the SI Schema is configured

Choose the Database type where SI scheme is configured

MS-SQL

12 Select **Oracle** or **Microsoft SQL Server** based on the database you have installed for use with Select Identity, then click **Next**. The Database Information page displays.



Select Identity 3.3 March 2 2005

Database Information

Enter database information. The Select Identity schema should already be installed in this location.

Database Server Name 16.73.17.72

Database Server Port 1433

Database Name si33Clustered

Database Login si33

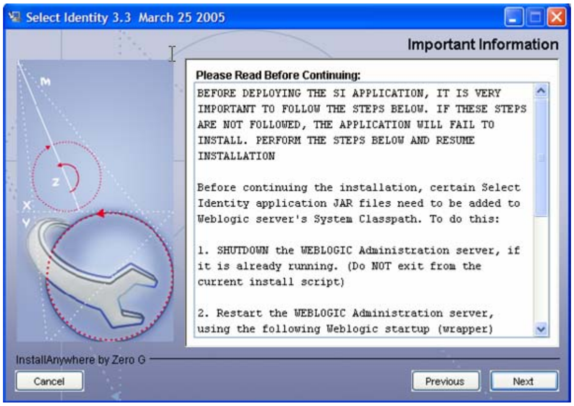
Database Password ****

13 Specify the settings for the database where Select Identity will store its data. This database must be configured, as described in Configuring the Database Server on page 23. Set the following:

- Database Server Name — The name of the database server
- Database Server Port — The port of the database server
- Database Name — The name of the database created for Select Identity's use
- Database Login — The user name of the database account that can be used by Select Identity to access the database
- Database Password — The user password

Then, click **Next**. The EMail Information page displays.

14 Specify the name of the SMTP host that Select Identity will use when sending email. Then, click **Next**. The Important Information page displays.



Please Read Before Continuing:

```
BEFORE DEPLOYING THE SI APPLICATION, IT IS VERY
IMPORTANT TO FOLLOW THE STEPS BELOW. IF THESE STEPS
ARE NOT FOLLOWED, THE APPLICATION WILL FAIL TO
INSTALL. PERFORM THE STEPS BELOW AND RESUME
INSTALLATION

Before continuing the installation, certain Select
Identity application JAR files need to be added to
Weblogic server's System Classpath. To do this:

1. SHUTDOWN the WEBLOGIC Administration server, if
it is already running. (Do NOT exit from the
current install script)

2. Restart the WEBLOGIC Administration server,
using the following Weblogic startup (wrapper)
```

15 Follow the directions listed on this page before continuing. The required steps are:

a SHUTDOWN the WEBLOGIC Administration server, if it is already running. (Do NOT exit from the current install script.)

b Restart the WEBLOGIC Administration server, using the following Weblogic startup (wrapper) script created by the installer:

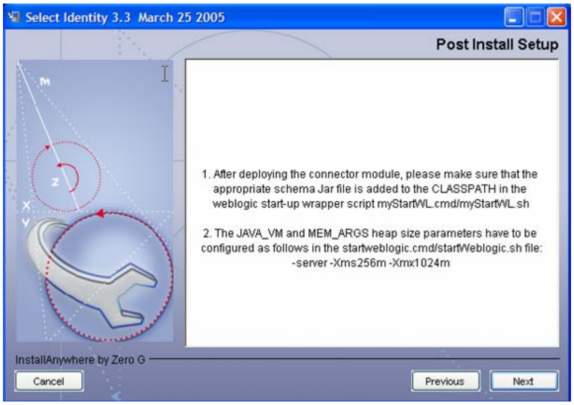
```
Windows: C:/si3.3/weblogic/scripts/myStartWL.cmd
Unix:   /opt/si3.3/weblogic/scripts/myStartWL.sh
```

c On UNIX, use the following command to execute the script:

```
sh /opt/si3.3/weblogic/scripts/myStartWL.sh
```

d Make sure that Weblogic server is started successfully, and resume installation at this point.

16 Click **Next** and the wizard configures the application server. Once complete, the Post Install Setup page displays.



1. After deploying the connector module, please make sure that the appropriate schema Jar file is added to the CLASSPATH in the weblogic start-up wrapper script myStartWL.cmd/myStartWL.sh

2. The JAVA_VM and MEM_ARGS heap size parameters have to be configured as follows in the startweblogic.cmd/startWeblogic.sh file:
-server -Xms256m -Xmx1024m

17 Perform steps listed on this page, if necessary, then click **Next**. The Install Complete page displays.

18 Click **Done** to close the wizard.

After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebLogic, this file resides in the `/$BEA_HOME/jdk142_05/jre/lib` directory on HP-UX and Solaris systems, or in the `$BEA_HOME/jrockit81sp4_142_05/jre/lib` on Linux systems.

⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

Using the Installation Wizard for Clustered Servers

The following procedures install Select Identity in a WebLogic cluster. The installation wizard performs the following by default:

- Copies the server files into the specified home directory on the administration server and each node in the cluster
- Creates a JDBC connection pool called SI_Connection_Pool
- Creates a data source called SI_Data_Source
- Creates a mail session called SI_MAIL_Session
- Deploys the EAR file
- Configures the Select Identity server with your specified settings

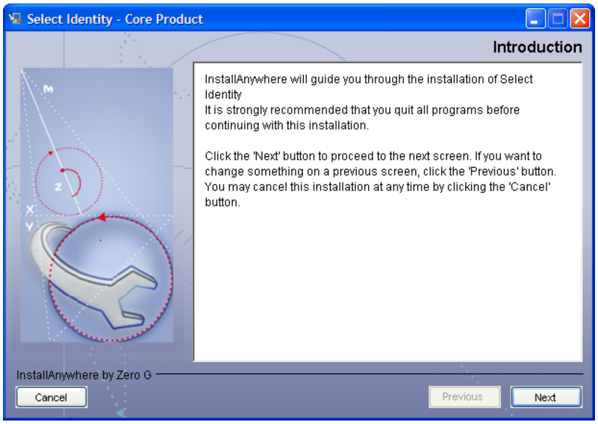
The cluster installation is run from the administrative server in the cluster. The node managers for each node should be running during the installation and the managed servers should be stopped.

Complete the following steps to configure WebLogic for use with Select Identity and to install Select Identity on the WebLogic Server:

1. Log on to the operating system as the same user that was used to install WebLogic.
2. *On Windows:*
   Insert the Select Identity CD, navigate to the `SI33\install\WebLogic\Windows` directory, and double-click the `install.exe` file. The InstallAnywhere wizard displays.

   On UNIX:
   Mount the CD, change directories to `SI33\install\WebLogic\Solaris, Linux or HPUX`, and run the `install.bin` executable.

The InstallAnywhere wizard displays.

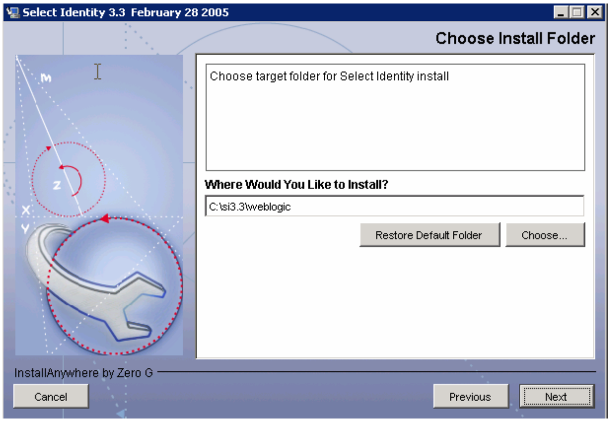


InstallAnywhere will guide you through the installation of Select Identity
It is strongly recommended that you quit all programs before continuing with this installation.

Click the 'Next' button to proceed to the next screen. If you want to change something on a previous screen, click the 'Previous' button. You may cancel this installation at any time by clicking the 'Cancel' button.

3 Click **Next**. The License Agreement page displays.

4 Read the license agreement and select **I accept the terms of the License Agreement** option, if you agree. If you do not select this option, you cannot install the software. Then, click **Next**. The Choose Install Folder page displays.

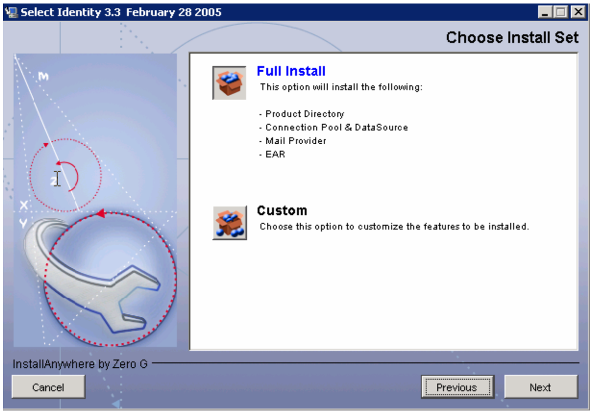


Choose target folder for Select Identity install

Where Would You Like to Install?

C:\si3.3\weblogic

5 Enter or browse to a path where you would like to install the Select Identity files. This folder will serve as the Select Identity home directory. Then, click **Next**. The Choose Install Set page displays.



Select Identity 3.3 February 28 2005

Choose Install Set

Full Install
This option will install the following:

- Product Directory
- Connection Pool & DataSource
- Mail Provider
- EAR

Custom
Choose this option to customize the features to be installed.

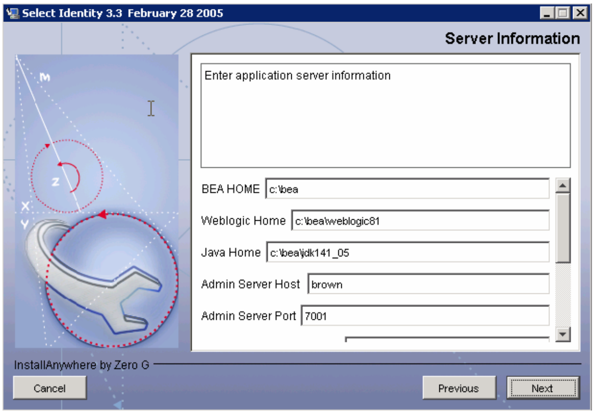
6 Select **Full Install** to accept all of the default configuration settings or **Custom** to choose your components.

7 Click Next. The Pre-Installation Summary page displays. Review the information before continuing.

8 Click **Install**. Select the **Cluster** installation option.

9 Click **Next**. The Installation Prerequisites page displays. Review the information and make sure that all prerequisites are met before continuing.

10 Click **Next**. The Server Information page displays.



Select Identity 3.3 February 28 2005

Server Information

Enter application server information

BEA HOME: c:\bea

Weblogic Home: c:\bea\weblogic81

Java Home: c:\bea\jdk141_05

Admin Server Host: brown

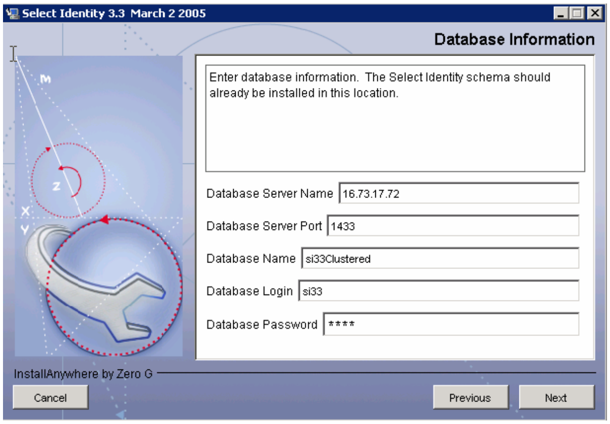
Admin Server Port: 7001

11 Specify settings for the WebLogic administrative server. This information is used to communicate with the node manager for each node in the cluster. Settings include:

- BEA Home — The directory where WebLogic is installed
- Java Home — The directory where the JDK is installed
- Server Host — The name of the local host (where WebLogic is installed)
- Server Port — The port used by Select Identity
- Domain Name — The application domain of the cluster in which Select Identity will be installed
- Admin Server Login Name — The user name of the WebLogic administrative account
- Admin Server Password — The password of the Admin account
- Cluster Name — The name of the cluster created in which you are installing Select Identity

12 Click **Next**. Choose the database that you installed for this environment. This database must be configured, as described in Configuring the Database Server on page 23.

13 Click **Next**. The Database Information page displays.



Database Information

Enter database information. The Select Identity schema should already be installed in this location.

Database Server Name 16.73.17.72

Database Server Port 1433

Database Name si33Clustered

Database Login si33

Database Password ****

14 Specify the settings for the database where Select Identity will store its data. Default information is provided based on your selection on the previous page. Set the following:

- Database Server Name — The name of the database server
- Database Server Port — The port of the database server
- Database Name — The name of the database created for Select Identity’s use
- Database Login — The user name of the database account that can be used by Select Identity to access the database
- Database Password — The user password

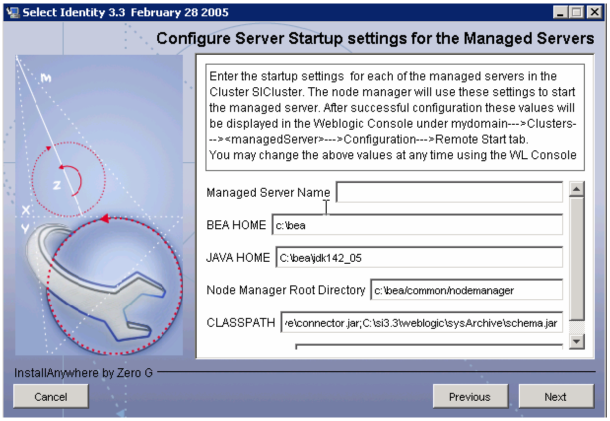
15 Click **Next**. The Email Information page displays.

16 Specify the name of the SMTP host that Select Identity will use when sending email.

17 Click Next. The Important Information page displays. Follow the directions on the page.

18 Click **Next**. The installer verifies the cluster’s administrative server configuration information.

19 Click **Next**. The Managed Server Configuration page displays.



Select Identity 3.3 February 28 2005
InstallAnywhere by Zero G
Cancel
Previous
Next

Configure Server Startup settings for the Managed Servers

Enter the startup settings for each of the managed servers in the Cluster SICluster. The node manager will use these settings to start the managed server. After successful configuration these values will be displayed in the Weblogic Console under mydomain--->Clusters---><managedServer>--->Configuration--->Remote Start tab.
You may change the above values at any time using the WL Console

Managed Server Name

BEA HOME c:\bea

JAVA HOME C:\bea\jdk142_05

Node Manager Root Directory c:\bea/common/nodemanager

CLASSPATH re\connector.jar;C:\si3.3\weblogic\sysArchive\schema.jar

20 Specify settings for the WebLogic administrative server. This information is used to communicate with the node manager for each node in the cluster. Settings include:

- Managed Server Name — Name of the managed server on which you are installing
- BEA Home — Directory where WebLogic is installed
- Java Home — Directory where the JDK is installed.
- Node Manager Root Directory — Location of the node manager root directory
- Class Path — Class path the managed servers will use when starting, which includes the `connector.jar` file, such as:

  ```
  C:\bea\jdk142_05\lib\tools.jar;c:\bea\weblogic81\server\lib\
  weblogic_sp.jar;c:\bea\weblogic81\server\lib\weblogic.jar;C:
  \si3.3\weblogic\sysArchive\connector.jar
  ```

  The path for `connector.jar` was specified on the administrative server configuration page.

- JVM Arguments — The location of the `TruAccess.properties` file, such as

  ```
  -Dcom.trulogica.truaccess.property.file=C:\si3.3\weblogic/
  sysArchive/TruAccess.properties -Djava.awt.headless=true
  ```

21 Click **Next**. The installer verifies the information.

22 Perform Step 19, Step 20, and Step 21 for each node in the cluster.

23 Start all servers in the cluster from the Administrative Console. This enables the node manager to propagate all of the values to each server.

After each server is started, you can verify the values that you defined through the remote start settings for each managed server on WebLogic.

24 When the installer is finished, click **Done** to close the wizard.

After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebLogic, this file resides in the `$BEA_HOME/jdk142_05/jre/lib` directory for HP-UX and Solaris systems, or in the `$BEA_HOME/jrockit81sp4_142_05/jre/lib` directory for Linux systems.

⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file. This must be done for all nodes in the cluster

Using the Manual Installation Process

If you choose to, you can use manual steps to install Select Identity. Installation prerequisites are common to WebSphere standalone, WebSphere cluster, and WebLogic configurations. The following procedures require that you perform the following:

- Copy files from the Select Identity CD
- Create a mail session
- Create a JDBC connection pool (WebLogic only)
- Create a JDBC data source
- Edit the startup script (WebLogic only)
- Deploy the Select Identity EAR file

Creating Directories and Copying Files

You must perform the following steps no matter what version of application server you are using. These steps prepare the application server before you configure it and deploy Select Identity.

1. Create a directory on the application server that will serve as the Select Identity home directory, storing its files and subdirectories. The product installation and connector installations will reference this directory. For example, you could create the `C:\Select_Identity` directory on Windows or `/usr/Select_Identity` on UNIX.
2. Copy the following files from the Select Identity product CD to the new directory. Additional directories may be required based on the configuration of the `TruAccess.properties` file. See Configuring TruAccess.properties on page 113 for details.
   - `application/lmz.ear` for WebLogic
   - `properties/TruAccess.properties`
   - `connector/connector.jar`

   Create a new subdirectory for each connector type that you install and install connector-specific information in its respective directory.
3. If you want to designate a log file for Select Identity messages only, copy `logging.properties` from the `properties` directory on the CD to a directory on the application server, such as a subdirectory of the library directory. This is particularly useful if you have multiple application servers configured in a domain.

   By default, a `logging.properties` file is provided by the application server's JVM. On WebLogic, this file resides in the `$BEA_HOME/jdk142_05/jre/lib` directory for HP-UX and Solaris systems, or in the `$BEA_HOME/jrockit81sp4_142_05/jre/lib` directory for Linux systems.

4 If you copied `logging.properties` from the Select Identity CD, ensure that the `C:\temp\log` directory (on Windows) exists on the application server's system. You can use any valid path on UNIX. The `logging.properties` file specifies this directory as the destination of Select Identity log messages by default.

> ⚠ If this directory does not exist, Select Identity will not start. If you wish to log to a different (existing) directory, edit the `logging.properties` file as described in Logging on page 110.

5 If necessary, create a subdirectory named `ext` in the JDK library subdirectory of the WebLogic server's installation directory.

6 Copy the contents of the `library` directory on the Select Identity CD to the `ext` subdirectory on WebLogic.

7 For easier access to documentation, you can copy the product documentation and help from the `docs` directory on the Select Identity CD to a directory on the application server.

Installing Manually on WebLogic

To configure WebLogic for use with Select Identity, complete the following steps. These steps assume that WebLogic is installed and running.

1 Ensure that the system where WebLogic is installed meets the *minimum* requirements, as documented in System Requirements on page 17.

2 If the WebLogic server is not running, start the server and log in to the WebLogic Server Console as a system user. (Typically, the URL for the console is **http://*host*:7001/console**.)

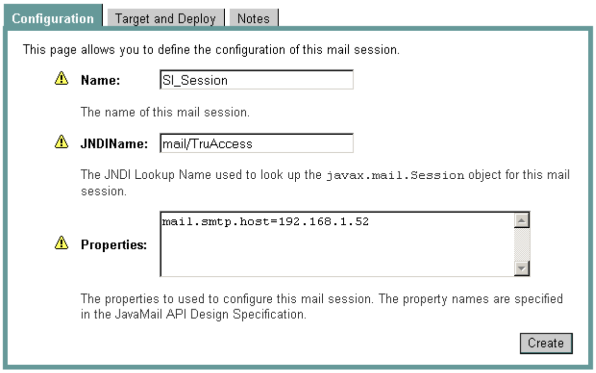
Select **Start** → **Programs** → **BEA WebLogic** → **Configuration Wizard** to create the User Projects directory.

The WebLogic server can be started by selecting **Start** → **Programs** → **BEA WebLogic** → **Projects** → **User Projects** then selecting **Start Server** next to the name of your domain or project.

3 Configure the mail session, as follows:

a Select ***My_domain*** → **Services** → **Mail** from the tree on the left-hand side of the console, where *domain_name* is the domain created during the WebLogic installation. The Mail Sessions page is displayed.

b Click **Configure a new Mail Session** on the Mail Sessions page. The Create a new MailSession page is displayed.



This page allows you to define the configuration of this mail session.

⚠ Name: SI_Session

The name of this mail session.

⚠ JNDIName: mail/TruAccess

The JNDI Lookup Name used to look up the `javax.mail.Session` object for this mail session.

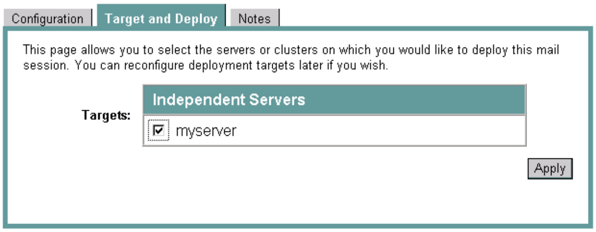
⚠ Properties: `mail.smtp.host=192.168.1.52`

The properties to used to configure this mail session. The property names are specified in the JavaMail API Design Specification.

c On the Configuration tab, provide the following information:

Field	Value
Name	Enter a name for the mail session.
JNDIName	Enter **`mail/TruAccess`**.
Properties	Enter the mail server's IP address. Here is an example: **`mail.smtp.host=192.168.1.52`**.

d Click **Create**. The Target and Deploy tab is displayed.



This page allows you to select the servers or clusters on which you would like to deploy this mail session. You can reconfigure deployment targets later if you wish.

Targets: Independent Servers

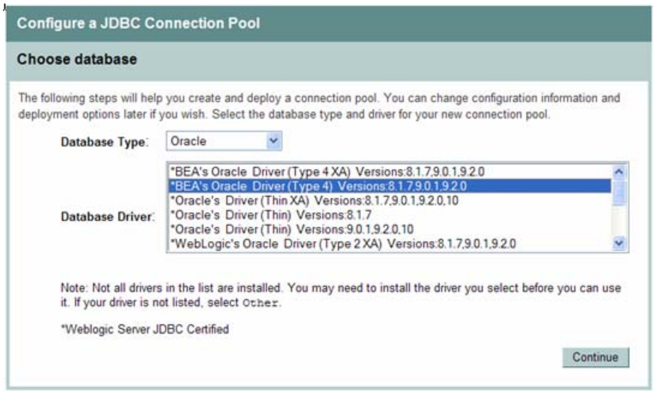
☑ myserver

e Select the WebLogic server designated for Select Identity's use and click **Apply** to finish mail session configuration.

4 Configure a JDBC connection pool to enable WebLogic to communicate with the database server by completing the following steps:

a Select ***My_domain*** → **Services** → **JDBC** → **Connection Pools** from the tree. The JDBC Connection Pool page is displayed.

b Click **Configure a new JDBC Connection Pool** on the JDBC Connection page. The Configure a JDBC ConnectionPool page is displayed.



Configure a JDBC Connection Pool

Choose database

The following steps will help you create and deploy a connection pool. You can change configuration information and deployment options later if you wish. Select the database type and driver for your new connection pool.

Database Type: Oracle

Database Driver:

- *BEA's Oracle Driver (Type 4 XA) Versions:8.1.7,9.0.1,9.2.0
- *BEA's Oracle Driver (Type 4) Versions:8.1.7,9.0.1,9.2.0
- *Oracle's Driver (Thin XA) Versions:8.1.7,9.0.1,9.2.0.10
- *Oracle's Driver (Thin) Versions:8.1.7
- *Oracle's Driver (Thin) Versions:9.0.1,9.2.0.10
- *WebLogic's Oracle Driver (Type 2 XA) Versions:8.1.7,9.0.1,9.2.0

Note: Not all drivers in the list are installed. You may need to install the driver you select before you can use it. If your driver is not listed, select `Other`.

*Weblogic Server JDBC Certified

c Choose the database type and driver. If you installed the driver documented in Configuring the Database Server on page 23, choose **BEA’s MS SQL Server Driver (Type 4) Versions:7.0, 2000** for SQL Server. Choose **BEA’s Oracle Driver (Type 4) Versions:8.1.7,9.0.1,9.2.0** for Oracle.

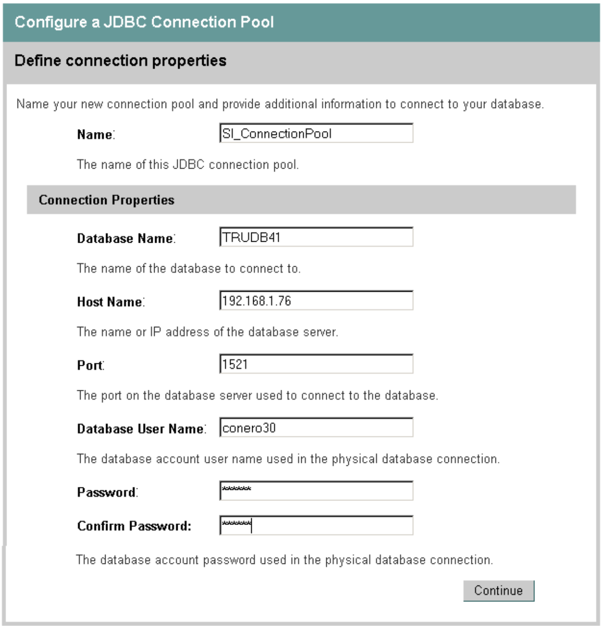
Then, click **Continue**.

d On the Define connection properties page, enter the following information:

Field	Value
Name	Enter a name for connection pool.
Database Name	The name of the database created on the database server for use by Select Identity. For example, Select_Identity.
Host Name	The IP address or host name of the database server.
Port	The database’s port. The default port for Microsoft SQL Server is 1433. For Oracle, the default port is 1521.

Field	Value
Database User Name	The user created for use to administer the Select Identity database.
Password and Confirm Password	Enter the database user's password.

The following is an example:



Configure a JDBC Connection Pool

Define connection properties

Name your new connection pool and provide additional information to connect to your database.

Name: SI_ConnectionPool

The name of this JDBC connection pool.

Connection Properties

Database Name: TRUDB41

The name of the database to connect to.

Host Name: 192.168.1.76

The name or IP address of the database server.

Port: 1521

The port on the database server used to connect to the database.

Database User Name: conero30

The database account user name used in the physical database connection.

Password: ******

Confirm Password: ******

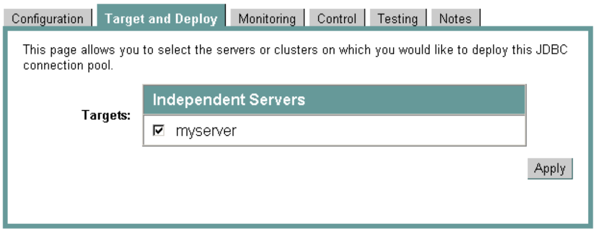
The database account password used in the physical database connection.

e Click **Continue**. WebLogic displays the Test database connection page and constructs the values displayed in the fields on the page.

f Click **Test Driver Configuration** to verify that WebLogic can connect to the database. Or, you can skip this step.

g Click **Create and deploy** to create the JDBC connection pool.

h Click on the newly created connection pool to verify that your server is selected on the Target and Deploy tab. You can also edit the connection properties, if you wish.



This page allows you to select the servers or clusters on which you would like to deploy this JDBC connection pool.

Targets:

Independent Servers

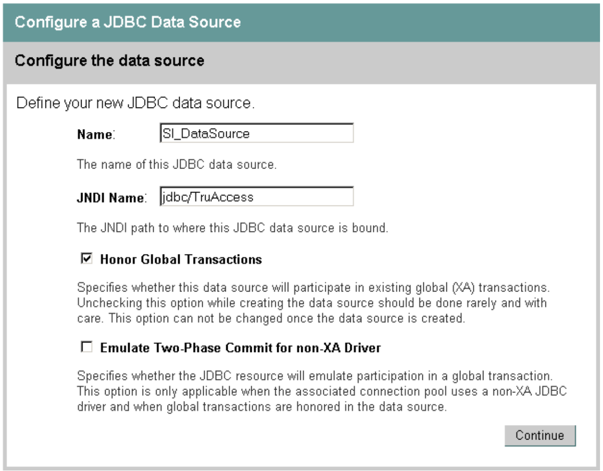
☑ myserver

i Click **Apply** if you change settings.

5 Configure JDBC data sources, as follows:

a Select ***My_domain*** → **Services** → **JDBC** → **Data Sources** from the tree. The data sources configuration page is displayed.

b Click **Configure a new JDBC Data Source** on the JDBC Data Sources page. The Configure a JDBC Data Source page is displayed.



Configure a JDBC Data Source

Configure the data source

Define your new JDBC data source.

Name: SI_DataSource

The name of this JDBC data source.

JNDI Name: jdbc/TruAccess

The JNDI path to where this JDBC data source is bound.

☑ **Honor Global Transactions**

Specifies whether this data source will participate in existing global (XA) transactions. Unchecking this option while creating the data source should be done rarely and with care. This option can not be changed once the data source is created.

☐ **Emulate Two-Phase Commit for non-XA Driver**

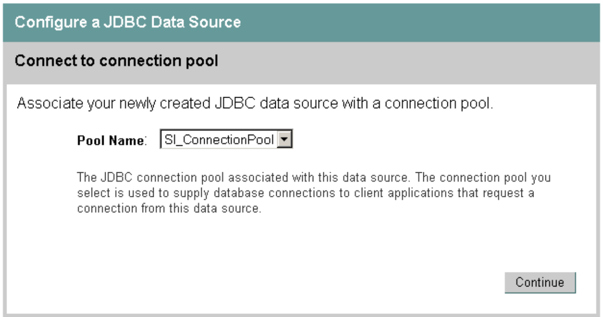
Specifies whether the JDBC resource will emulate participation in a global transaction. This option is only applicable when the associated connection pool uses a non-XA JDBC driver and when global transactions are honored in the data source.

c Enter the following information (leave the default values for the rest of the options on the page):

Field	Value
Name	Enter a name for the new data source.
JNDI Name	Enter **`jdbc/TruAccess`**.

d Click **Continue**.

e Select the connection pool created in Step 4 on page 49 from the **Pool Name** drop-down list.



Configure a JDBC Data Source

Connect to connection pool

Associate your newly created JDBC data source with a connection pool.

Pool Name: SI_ConnectionPool

The JDBC connection pool associated with this data source. The connection pool you select is used to supply database connections to client applications that request a connection from this data source.

f Click **Continue**.

g Ensure your server is selected on the Target data source page. Then, click **Create**.

6 Stop the WebLogic server.

7 Edit the `startweblogic.cmd` file to modify WebLogic's classpath, specify the location of the `TruAccess.properties` file, the location of the `connector.jar` file, and the location of the `logging.properties` file, if necessary.

The `startweblogic.cmd` file resides in the `*WebLogic_home*/user_projects/domains/*domain*/` directory on WebLogic 8.1.

— Add the Select Identity home directory to the classpath. This enables the web server to access the `connector.jar` file. Here is an example of the classpath after it is modified. The new entries are bolded:

```
set CLASSPATH=%WEBLOGIC_CLASSPATH%;%POINTBASE_CLASSPATH%;
%JAVA_HOME%\jre\lib\rt.jar;
```

```
%WL_HOME%\server\lib\webservices.jar;c:\Select_Identity;
c:\Select_Identity\connector.jar;%CLASSPATH%
```

— Add the `TruAccess.properties` file to the line beginning with **%JAVA_HOME%/bin/java**. Add this argument before the **weblogic.Server** entry. If the `TruAccess.properties` file is not located in a path that is relative to the `weblogic81` directory, make sure that you provide a fully-qualified path to the file.

If WebLogic is running on a UNIX server, you will also add `-Djava.awt.headless=true` to the classpath.

Here is an example of this line after it is modified. The new entry is bolded:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
-Dweblogic.Name=%SERVER_NAME%
-Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
-Djava.security.policy="%WL_HOME%/server/lib/weblogic.policy"
-Dcom.trulogica.truaccess.property.file=/export/home/
Select_Identity/TruAccess.properties
-Djava.awt.headless=true weblogic.Server
```

— If `logging.properties` resides in a directory different than WebLogic's installation directory, you must add the location to the line beginning with **%JAVA_HOME%/bin/java**. Add this argument before **weblogic.Server** in the line. If the file is not located in a path that is relative to the `weblogic81` directory, make sure that you provide a fully-qualified path to the file.

Here is an example of this line after it is modified. The new entry is bolded:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
-Dweblogic.Name=%SERVER_NAME%
-Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
-Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
-Dcom.trulogica.truaccess.property.file=c:\Select_Identity\
  TruAccess.properties -Dcom.trulogica.truaccess.util.logging.
misc.config.file=c:\Select_Identity\log\logging.properties
weblogic.Server
```

8 Save your settings.

9 Restart the WebLogic server.

10 Deploy Select Identity on the application server as follows:

   a Log in to the WebLogic Server Console.

   b Select the ***domain_name*** → **Deployments** → **Applications** folder.

   c Click **Deploy a new Application**.

   d Locate and select the `lmz.ear` file, which resides in the Select Identity home directory created in Installing on the Application Server on page 33.

   e Select the radio button next to `lmz.ear` and click **Continue**.

   The Deploy an Application page displays. The fields are populated with default values.

   If multiple servers are configured, click **Target Application** and select the targets of the Select Identity application.

   f Click **Deploy**.

   The deployment may take several seconds to complete.

11 After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebLogic, it resides in the`$BEA_HOME/jdk142_05/jre/lib/` for HP-UX and Solaris systems, or in the `$BEA_HOME/jrockit81sp4_142_05/jre/lib/` directory for Linux systems

   ⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

There are additional configuration steps for WebLogic and WebSphere installations. See Additional Configuration on page 85 to finish installing Select Identity.

5

Installing on WebSphere Servers

Select Identity relies on the web application server to serve the Select Identity interface pages, communicate with the database server to store and retrieve data, and send email based on an action performed through the Select Identity interface. The following sections describe how to install and configure on a WebSphere application server.

Configuring the Application Server

You can install Select Identity on a WebSphere standalone server or on multiple servers in a cluster.

Configuring the WebSphere Standalone Server

To configure WebSphere for use with Select Identity, complete the following steps.

➤ Select Identity supports clustered servers through the application server layer. See the application server documentation for information on clustering these servers.

These steps assume that WebSphere is installed and running.

1 Ensure that the system where WebSphere is installed meets the *minimum* requirements, as documented on System Requirements on page 17.

2 Copy the following high-export security policy files to the `$WAS_ROOT/java/jre/lib/security` directory, where $WAS_ROOT is typically set to the `/opt/Websphere/AppServer` directory:

   `US_export_policy.jar`

   `local_policy.jar`

   Copy the following file that is compatible with the high export security policy files to directory `$WAS_ROOT/java/jre/lib/ext`.

   `sunjce_provider.jar`

   If the WebSphere installation uses the Sun JDK, you can download these JAR files from the Sun web site.

   If you are running WebSphere outside of the United States, the policy files may vary based on the country where the application is hosted.

3 Set the environment variables for the database in WebSphere. Complete the following steps to do so:

   a Log on to the WebSphere Application Server Console.

   b Navigate to **Environment** → **Manage WebSphere Variables**.

   c Verify that the DB2UNIVERSAL_JDBC_DRIVER_PATH variable is set to the path where the DB2 JDBC driver file (`db2java.zip`) resides. Click the name of the variable to modify its settings, if need be. If this variable is not set, click **New** and create it.

   d Verify that the DB2_JDBC_DRIVER_PATH variable is also set to the path where the DB2 JDBC driver file (`db2java.zip`) resides. Click the name of the variable to modify its settings, if need be. If this variable is not set, click **New** and create it.

   e Verify that the LD_LIBRARY_PATH variable contains the path to the DB2 native libraries (such as LD_LIBRARY_PATH=/export/home/db2inst1/sqllib/lib). WebSphere uses this path to connect to DB2. One way to pass this variable to the WebSphere server is by starting the WebSphere Application Server in the same shell where this path is set.

Configuring the WebSphere Cluster

You can install Select Identity in an IBM WebSphere Application Server cluster. Make sure that the following prerequisites are met:

- WebSphere is installed on each node and meets the system requirements as listed in System Requirements on page 17.
- The Network Deployment Manager is configured with appropriate cells, nodes, and clusters.
- The deployment manager node, node agents, and application servers can be started and stopped without any errors.

To configure WebSphere for use with Select Identity in a clustered environment, complete the following steps on each of the servers in the cluster. These steps assume that WebSphere is installed and running.

1. Ensure that the system where WebSphere is installed meets the *minimum* requirements, as documented in System Requirements on page 17.

2. Copy the following high-export security policy files to the `$WAS_ROOT/java/jre/lib/security` directory, where $WAS_ROOT is typically set to the `/opt/Websphere/AppServer` directory:

   `US_export_policy.jar`

   `local_policy.jar`

   Copy the following file that is compatible with the high export security policy files to directory`$WAS_ROOT/java/jre/lib/ext`.

   `sunjce_provider.jar`

   If the WebSphere installation uses the Sun JDK, you can download these JAR files from the Sun web site.

   If you are running WebSphere outside of the United States, the policy files may vary based on the country where the application is hosted.

3. Set the environment variables for the database in WebSphere. Complete the following steps to do so:

   a. Log on to the WebSphere Application Server Network Deployment Manager Console.

   b. Navigate to **Environment** → **Manage WebSphere Variables**.

c From the Server scope, choose a server within the defined cluster.

d Verify that the DB2UNIVERSAL_JDBC_DRIVER_PATH variable is set to the path where the DB2 JDBC driver file (`db2java.zip`) resides, such as `DB2UNIVERSAL_JDBC_DRIVER_PATH=/opt/IBM/db2/V8.1/java`. Click the name of the variable to modify its settings, if need be. If this variable is not set, click **New** and create it.

e Verify that the DB2_JDBC_DRIVER_PATH variable is also set to the path where the DB2 JDBC driver file (`db2java.zip`) resides, such as `DB2_JDBC_DRIVER_PATH=/opt/IBM/db2/V8.1/java`. Click the name of the variable to modify its settings, if need be. If this variable is not set, click **New** and create it.

f Verify that the LD_LIBRARY_PATH variable contains the path to the DB2 native libraries (such as `LD_LIBRARY_PATH=/export/home/db2inst1/sqllib/lib`). WebSphere uses this path to connect to DB2. One way to pass this variable to the WebSphere server is by starting the WebSphere Application Server in the same shell where this path is set.

To set the LD_LIBRARY_PATH, login to the web application host, open a new command shell, and source the DB2 profile as follows:

```
./export/home/db2inst1/.profile
```

where `/export/home/db2inst1` is the DB2 client home directory.

g Restart the WebSphere application server in the same shell.

If the LD_LIBRARY_PATH is passed to the server by sourcing from a command shell, the server cannot be restarted using Node manager. It has to be restarted from a command shell.

➤ It may be possible to push the LD_LIBRARY_PATH from the deployment manager to each of the WebSphere application servers. It may also be possible to perform the previous steps at the Node Manager scope instead of the Server scope. Check with your WebSphere administrator.

Installing on the Application Server

This section provides procedures for installing Select Identity using the installation wizard or by manually configuring the application server. Before completing these steps, be sure to complete the procedures included in the Configuring the WebSphere Standalone Server or Configuring the WebSphere Cluster section; the wizard and manual steps require this.

Installation Wizard Procedure

The following sections describe how to install Select Identity on a WebSphere standalone server. The installation wizard performs the following by default:

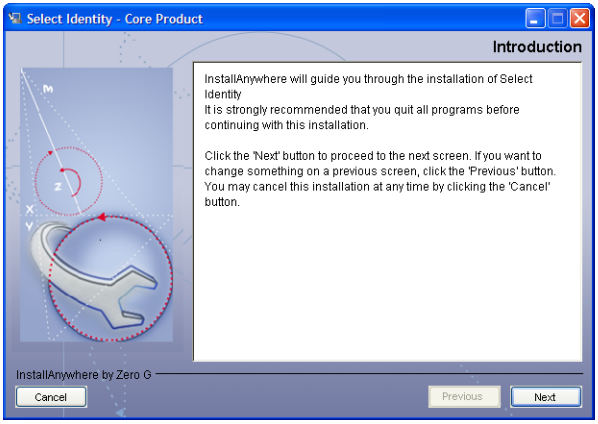
- Copies the server files into the specified home directory
- Creates a JDBC provider called SI_DB_Driver
- Creates a data source called SI_Data_Source
- Creates a mail session called SI_MAIL_Session
- Deploys the EAR file
- Configures the Select Identity server with your specified settings

➤ The Installation Wizard does not support installing Select Identity in a “WebSphere Clustered” environment. For a WebSphere clustered environment, install and configure Select Identity manually as described in Configuring the WebSphere Cluster on page 58.

Complete the following steps to install Select Identity on the WebSphere application server:

1 Log on to the operating system as the same user that was used to install WebSphere.

2 Mount the Select Identity CD, navigate to the `install\Websphere\solaris` directory, and run the `install.bin` executable. The InstallAnywhere wizard displays.



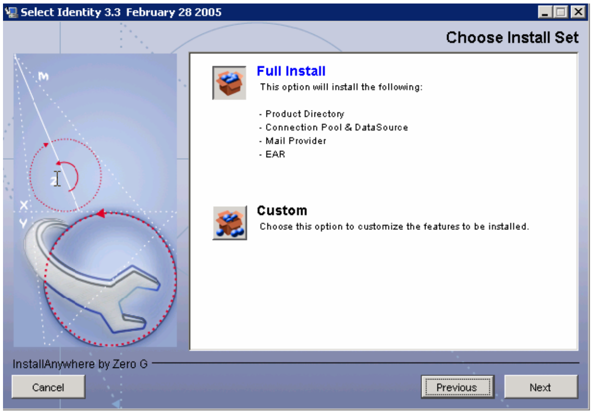
InstallAnywhere will guide you through the installation of Select Identity
It is strongly recommended that you quit all programs before continuing with this installation.

Click the 'Next' button to proceed to the next screen. If you want to change something on a previous screen, click the 'Previous' button. You may cancel this installation at any time by clicking the 'Cancel' button.

3 Click **Next**. The License Agreement page displays.

4 Read the license agreement and select **I accept the terms of the License Agreement** option, if you agree. If you do not select this option, you cannot install the software. Then, click **Next**. The Choose Install Folder page displays.

5 Enter or browse to a path where you would like to install the Select Identity files. This folder will serve as the Select Identity home directory. Then, click **Next**. The Choose Install Set page displays.



Select Identity 3.3 February 28 2005

Choose Install Set

Full Install
This option will install the following:

- Product Directory
- Connection Pool & DataSource
- Mail Provider
- EAR

Custom
Choose this option to customize the features to be installed.

6 Select **Full Install** to accept all of the default configuration or **Custom** to choose your components.

7 Click **Next**. The Pre-Installation Summary page displays.

8 Select the components to deploy in the application server. Then, click **Install**. The Installing Select Identity page displays, and the wizard installs the files according to the chosen settings.

When the wizard completes the installation, the Server Information page displays.

9 On the Server Information page, specify settings that are set in the WebSphere Application Server, as follows:

- WAS Root Directory — The directory where WebSphere is installed (typically `opt/WebSphere/AppServer`).
- Server Host Name — The name of the local host (where WebSphere is installed).
- Server Port — The port used by Select Identity.
- Cell Name — The name of the cell to which the server is limited; this limits the visibility to all servers on the named cell.
- Node Name — The name of the node; this limits the visibility to all servers on the named node. The node scope has precedence over the cell scope.

- Admin Server Login Name — The user name of the WebSphere administrative account.
- Admin Server Password — The password of the Admin account. A value is required even if security is not enabled.
- Server Name — The name of the server created in WebSphere and on which you are installing Select Identity; this limits the visibility to the named server. The server scope has precedence over the node and cell scopes. Note that objects are created for only one scope, though they might be visible at more than one scope.

Then, click **Next**. The Database Information page displays.

10 Specify the settings for the database where Select Identity will store its data. This database must be configured, as described in Configuring the Database Server on page 23. Set the following:

- Database Type — Select **DB2**.
- Database Server Name — The name of the database server
- Database Server Port — The port of the database server
- Database Name — The name of the database created for Select Identity's use
- Database Login — The user name of the database account that can be used by Select Identity to access the database
- Database Password — The user password

Then, click **Next**. The Set WebSphere Variable page displays.

11 As described on the Set WebSphere Variable page, ensure that the environment variable is set. You should have set this before running the installation wizard, as described in Step 3 on page 57. Then, click **Next**. The EMail Information page displays.

12 Specify an email session name and the name of the SMTP host that Select Identity will use when sending email. Then, click **Next**. The Install Complete page displays, and the wizard configures the application server.

13 Click **Done** to close the wizard.

14 After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebSphere, it resides in the `$WAS_HOME/java/jre/lib` directory.

⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

➤ When you begin to deploy connectors for WebSphere (standalone and cluster) configurations, the `connector.jar` file should be copied to the `$WAS_HOME/lib/ext` directory. Make sure that this file resides in this directory when starting the WebSphere application server

Manual Installation Procedures

Installation prerequisites are common to WebSphere standalone, WebSphere cluster, and WebLogic configurations. Platform-specific information is stated where needed. The following procedures require that you perform the following:

- Copy files from the Select Identity CD
- Create a mail session
- Create a JDBC driver (WebSphere only)
- Create a JDBC data source
- Configure the path to the log file (WebSphere only)
- Deploy the Select Identity EAR file

Creating Directories and Copying Files

You must perform the following steps no matter what version of application server you are using. These steps prepare the application server before you configure it and deploy Select Identity.

1. Create a directory on the application server that will serve as the Select Identity home directory, storing its files and subdirectories. The product installation and connector installations will reference this directory. For example, you could create the `C:\Select_Identity` directory on Windows or `/usr/Select_Identity` on UNIX.

2. Copy the following files from the Select Identity product CD to the new directory. Additional directories may be required based on the configuration of the `TruAccess.properties` file. See Configuring TruAccess.properties on page 113 for details.

   - `application/websphere_lmz` for WebSphere
   - `properties/TruAccess.properties`
   - `connector/connector.jar`

   Create a new sub-directory for each connector type that you install and install connector-specific information in its respective directory.

   > For WebSphere (standalone and cluster) configurations, the `connector.jar` file should be copied to the `$WAS_HOME/lib/ext` directory. Make sure that this file resides in this directory when starting the WebSphere application server

3. If you want to designate a log file for Select Identity messages only, copy `logging.properties` from the `properties` directory on the CD to a directory on the application server, such as a subdirectory of the library directory. This is particularly useful if you have multiple application servers configured in a domain.

   By default, a `logging.properties` file is provided by the application server's JVM. On WebSphere, it resides in the `$WAS_HOME/java/jre/lib` directory.

4 If you copied `logging.properties` from the Select Identity CD, ensure that the `C:\temp\log` directory (on Windows) exists on the application server's system. You can use any valid path on UNIX. The `logging.properties` file specifies this directory as the destination of Select Identity log messages by default.

> ⚠ If this directory does not exist, Select Identity will not start. If you wish to log to a different (existing) directory, edit the `logging.properties` file as described in Logging on page 110.
>
> This is not a requirement for WebSphere installations.

5 For easier access to documentation, you can copy the product documentation and help from the `docs` directory on the Select Identity CD to a directory on the application server.

Installing on a Standalone Server

Complete the following steps to configure IBM WebSphere application server for use with Select Identity:

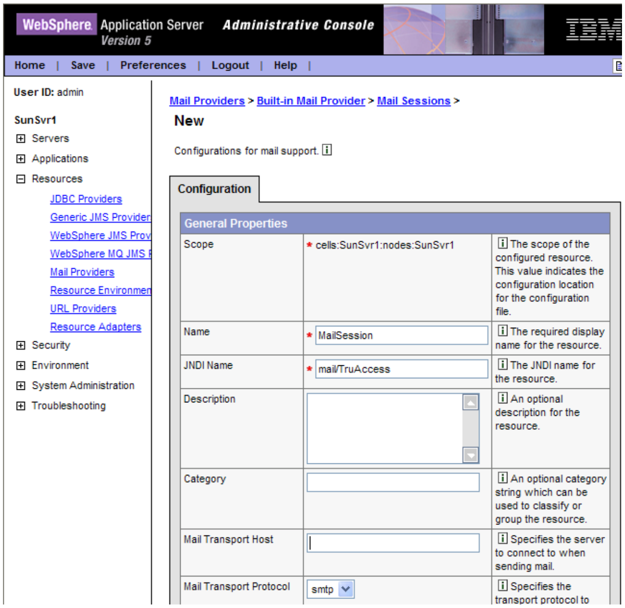
1 Ensure that the system where WebSphere is installed meets the *minimum* requirements, as documented in System Requirements on page 17.

2 Log on to the WebSphere Administrative Console as **admin**.

3 Configure the mail session by completing the following steps:

a From the tree on the left-hand side of the console, select **Resources** → **Mail Providers** → **Built-in Mail Provider** → **Mail Sessions**.

b Click the **New** button on the Mail Sessions page. The following form is displayed.



Mail Providers > Built-in Mail Provider > Mail Sessions >

New

Configurations for mail support.

Configuration

General Properties

Scope	* cells:SunSvr1:nodes:SunSvr1	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Name	* MailSession	The required display name for the resource.
JNDI Name	* mail/TruAccess	The JNDI name for the resource.
Description		An optional description for the resource.
Category		An optional category string which can be used to classify or group the resource.
Mail Transport Host		Specifies the server to connect to when sending mail.
Mail Transport Protocol	smtp	Specifies the transport protocol to

c Provide the following information:

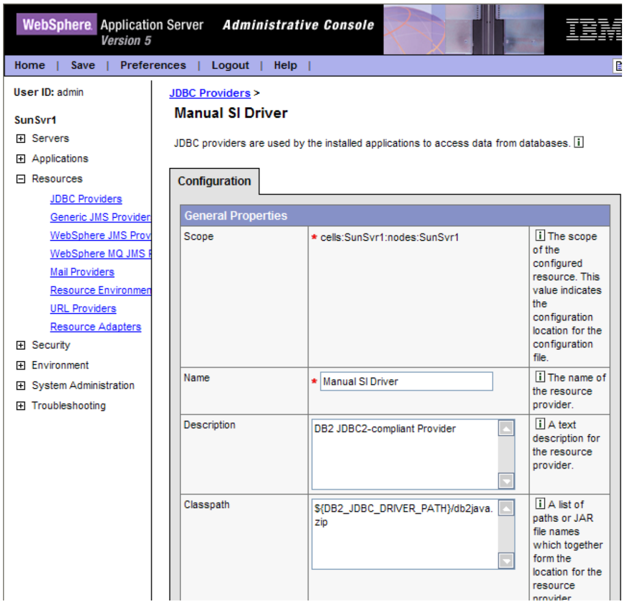
Field	**Value**
Name	Enter **`MailSession`**.
JNDI Name	Enter **`mail/TruAccess`**.
Mail Transport Host	Enter the IP address of the server to which to connect when sending mail.
Mail Transport Protocol	Enter **`smtp`**.

d Click **OK** at the bottom of the page. The Mail Sessions page is displayed again and lists the mail session you just configured.

- **e** Click **Apply** at the bottom of the Mail Sessions page.
- **f** Click **Save** to complete the mail session configuration.
- **g** Click **Save** on the Save Configuration page.

4 Configure a JDBC driver and data source, which will enable WebSphere to communicate with the Select Identity database, by completing these steps:

- **a** Because the database server must be DB2 for use with Select Identity, select **Resources** → **JDBC Providers** → ***Db2JdbcDriver*** from the tree. The driver page is displayed.



JDBC Providers >

Manual SI Driver

JDBC providers are used by the installed applications to access data from databases.

General Properties

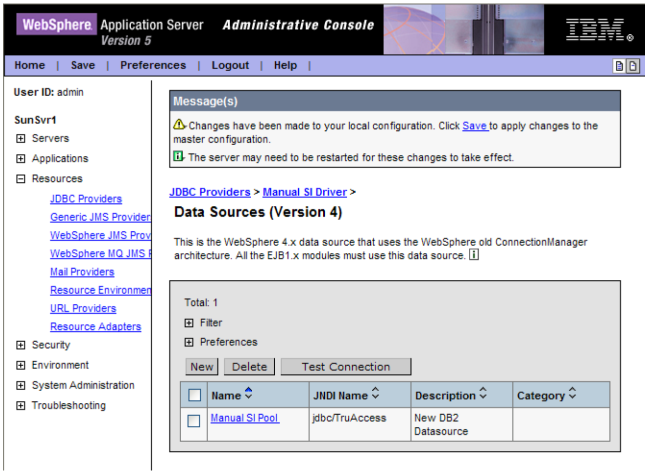
Scope	* cells:SunSvr1:nodes:SunSvr1	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Name	* Manual SI Driver	The name of the resource provider.
Description	DB2 JDBC2-compliant Provider	A text description for the resource provider.
Classpath	${DB2_JDBC_DRIVER_PATH}/db2java.zip	A list of paths or JAR file names which together form the location for the resource provider.

- **b** Enter ***install_dir*/websphere/appserver/classes/db2java.zip** (on Windows) in the Server Class Path field (leave the default values in the remaining fields).
- **c** Click **Apply**.

d Click **Save** at the top of the JDBC Drivers page.

e Click **Save** on the Save Configuration page.

f Select **Resources** → **JDBC Drivers** → **Db2JdbcDriver** → **Data Sources (Version4)**. The Data Sources page is displayed.



Message(s)

⚠ Changes have been made to your local configuration. Click Save to apply changes to the master configuration.

ℹ The server may need to be restarted for these changes to take effect.

JDBC Providers > Manual SI Driver >

Data Sources (Version 4)

This is the WebSphere 4.x data source that uses the WebSphere old ConnectionManager architecture. All the EJB1.x modules must use this data source. ℹ

Total: 1

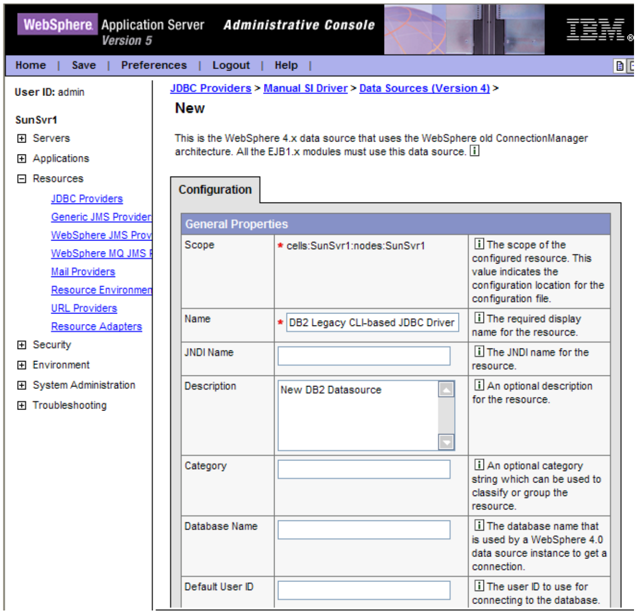
⊞ Filter

⊞ Preferences

New Delete Test Connection

□	Name	JNDI Name	Description	Category
□	Manual SI Pool	jdbc/TruAccess	New DB2 Datasource	

g Click **New** on the Data Sources page. The following form is displayed.



New

This is the WebSphere 4.x data source that uses the WebSphere old ConnectionManager architecture. All the EJB1.x modules must use this data source.

General Properties

Scope	* cells:SunSvr1:nodes:SunSvr1	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Name	* DB2 Legacy CLI-based JDBC Driver	The required display name for the resource.
JNDI Name		The JNDI name for the resource.
Description	New DB2 Datasource	An optional description for the resource.
Category		An optional category string which can be used to classify or group the resource.
Database Name		The database name that is used by a WebSphere 4.0 data source instance to get a connection.
Default User ID		The user ID to use for connecting to the database.

h Provide the following information:

Field	Value
Name	Enter a name for the data source. A data source is a pool of managed database connections.
JNDI Name	Enter **`jdbc/TruAccess`**.
Database Name	Enter the name of the database created for Select Identity use.
Default User ID	Enter the database user ID.
Default Password	Enter the user's password.

Leave the default values in the remaining fields.

i Click **Apply**. The Data Sources page is displayed again.

j Click **Save** at the top of the Data Sources page.

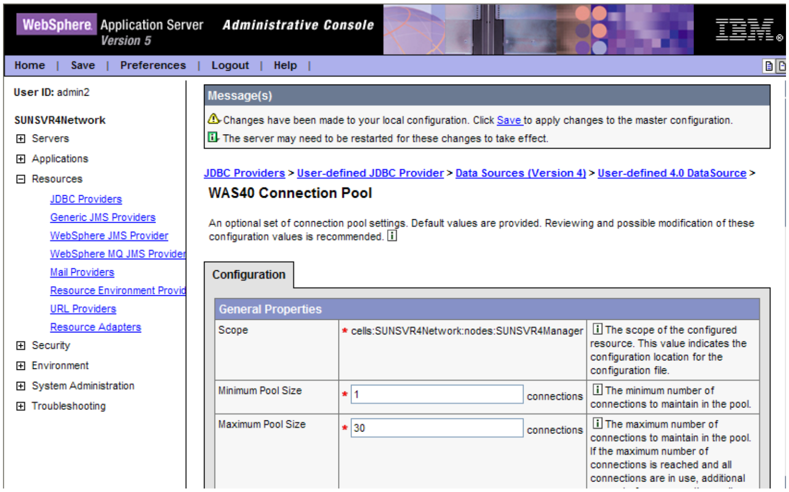
k Click **Save** on the Save Configuration page to complete the JDBC driver configuration.

l Select **Resources** → **JDBC Providers** → **Db2JdbcDriver** → **Data Sources (Version4)**→ ***data_source***. Under Additional Properties, click on the **Custom Properties** link.

m Change **cursorhold=0** to **cursorhold=1**.

n Click **Apply** and **Save**.

o Select **Resources** → **JDBC Providers** → **Db2JdbcDriver** → **Data Sources (Version4)**→ ***data_source***. Under Additional Properties, click on the **Connection Pool** link. The Configuration page is displayed.



Message(s)

⚠ Changes have been made to your local configuration. Click Save to apply changes to the master configuration.

ⓘ The server may need to be restarted for these changes to take effect.

JDBC Providers > User-defined JDBC Provider > Data Sources (Version 4) > User-defined 4.0 DataSource >

WAS40 Connection Pool

An optional set of connection pool settings. Default values are provided. Reviewing and possible modification of these configuration values is recommended. ⓘ

Configuration

General Properties

Scope	* cells:SUNSVR4Network:nodes:SUNSVR4Manager	ⓘ The scope of the configured resource. This value indicates the configuration location for the configuration file.
Minimum Pool Size	* 1 connections	ⓘ The minimum number of connections to maintain in the pool.
Maximum Pool Size	* 30 connections	ⓘ The maximum number of connections to maintain in the pool. If the maximum number of connections is reached and all connections are in use, additional

p Select the Disable Auto Connection Cleanup check box. For a cluster, this has to be done at the server scope.

q Click **Java Virtual Machine** link under Additional Properties.

r Under Generic JVM Arguments, enter the location of the `TruAccess.properties` file, such as:

```
Dcom.trulogica.truaccess.property.file=/selectIdentity/
properties/TruAccess.properties
```

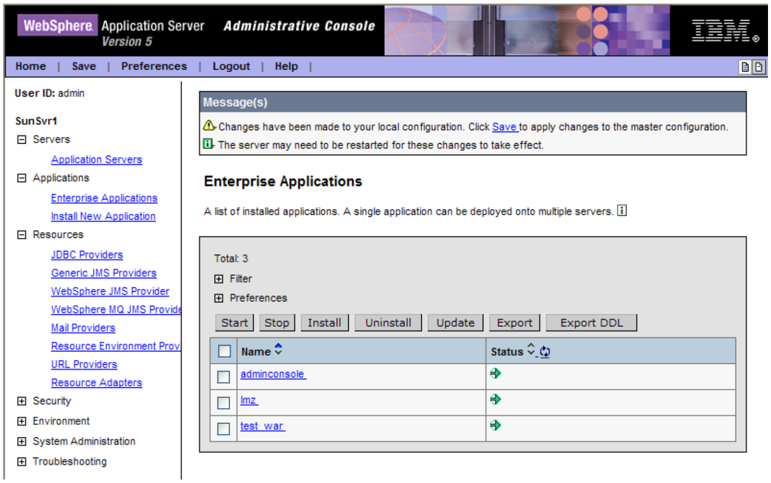
If installing in a UNIX environment add the following to the line above.

```
-Djava.awt.headless=true
```

s Click **Apply** and **Save**.

5 Install Select Identity on WebSphere as follows:

a Select **Applications** → **Enterprise Applications** from the tree. The Enterprise Applications page is displayed.



Message(s)

⚠ Changes have been made to your local configuration. Click Save to apply changes to the master configuration.

The server may need to be restarted for these changes to take effect.

Enterprise Applications

A list of installed applications. A single application can be deployed onto multiple servers.

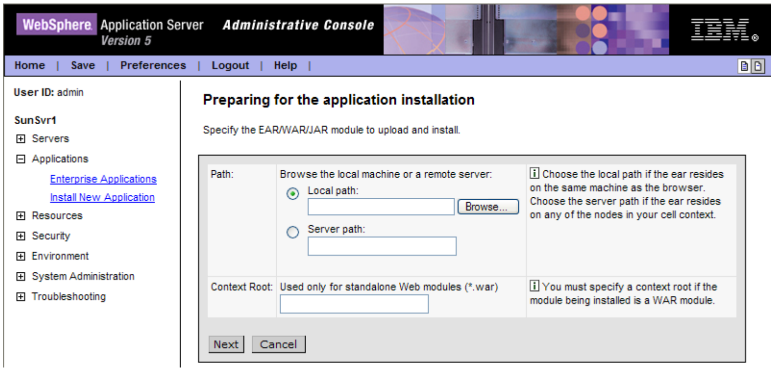
Total: 3

⊞ Filter

⊞ Preferences

□	Name	Status
□	adminconsole	➜
□	lmz	➜
□	test_war	➜

b Click **Install** on the Enterprise Applications page. The following page is displayed.

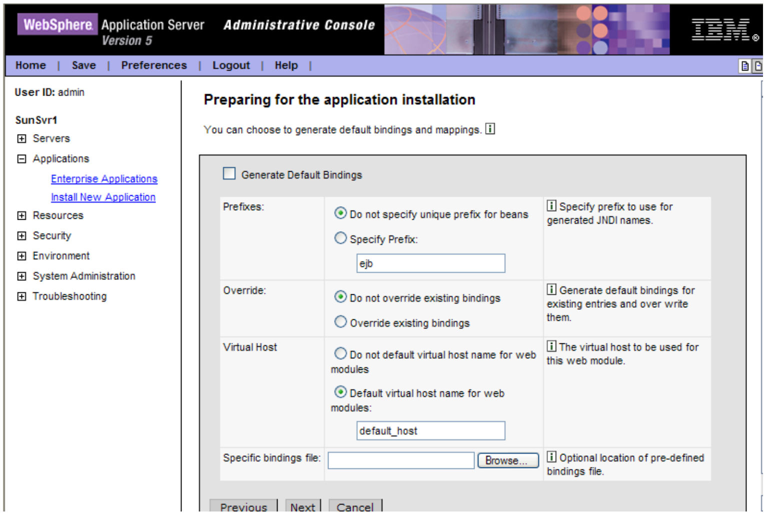


Preparing for the application installation

Specify the EAR/WAR/JAR module to upload and install.

Path:	Browse the local machine or a remote server: ● Local path: ○ Server path:	Choose the local path if the ear resides on the same machine as the browser. Choose the server path if the ear resides on any of the nodes in your cell context.
Context Root:	Used only for standalone Web modules (*.war)	You must specify a context root if the module being installed is a WAR module.

c Using the Browse button next to the Path field, locate and select the `websphere_lmz.ear` file, then click **Next**. The Installation Preparation page is displayed.



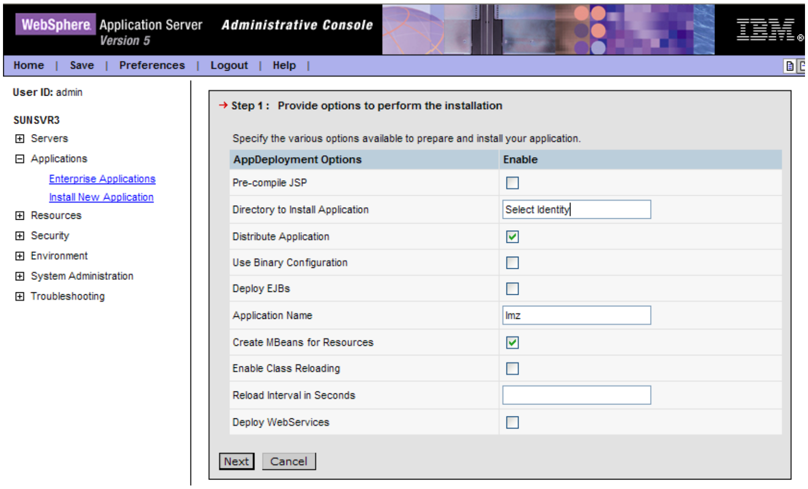
Preparing for the application installation

You can choose to generate default bindings and mappings.

□ Generate Default Bindings

Prefixes:	● Do not specify unique prefix for beans ○ Specify Prefix: ejb	Specify prefix to use for generated JNDI names.
Override:	● Do not override existing bindings ○ Override existing bindings	Generate default bindings for existing entries and over write them.
Virtual Host	○ Do not default virtual host name for web modules ● Default virtual host name for web modules: default_host	The virtual host to be used for this web module.
Specific bindings file:	Browse...	Optional location of pre-defined bindings file.

d Review the settings and click **Next**. The Deployment Options page is displayed.



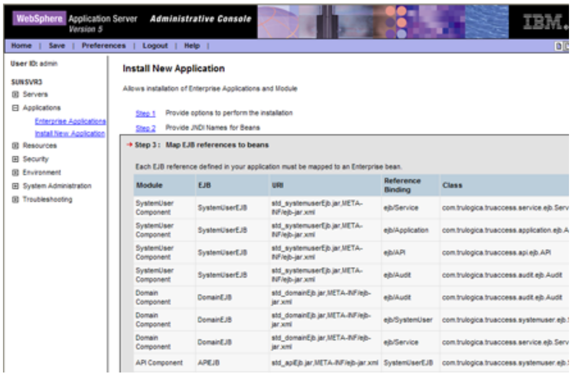
→ Step 1 : Provide options to perform the installation

Specify the various options available to prepare and install your application.

AppDeployment Options	Enable
Pre-compile JSP	□
Directory to Install Application	Select Identity
Distribute Application	☑
Use Binary Configuration	□
Deploy EJBs	□
Application Name	lmz
Create MBeans for Resources	☑
Enable Class Reloading	□
Reload Interval in Seconds	
Deploy WebServices	□

e Enter any desired options and click **Next**. The Enterprise Bean JNDI Names table is displayed.

f Click **Next**. The Mapping Resource References to JNDI Names table is displayed.



Install New Application

Allows installation of Enterprise Applications and Module

Step 1 Provide options to perform the installation

Step 2 Provide JNDI Names for Beans

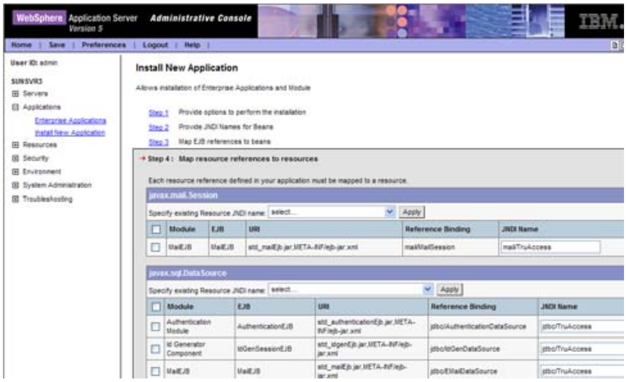
→ Step 3 : Map EJB references to beans

Each EJB reference defined in your application must be mapped to an Enterprise bean.

Module	EJB	URI	Reference Binding	Class
SystemUser Component	SystemUserEJB	std_systemuserEjb.jar,META-INF/ejb-jar.xml	ejb/Service	com.trulogica.truaccess.service.ejb.Serv
SystemUser Component	SystemUserEJB	std_systemuserEjb.jar,META-INF/ejb-jar.xml	ejb/Application	com.trulogica.truaccess.application.ejb.A
SystemUser Component	SystemUserEJB	std_systemuserEjb.jar,META-INF/ejb-jar.xml	ejb/API	com.trulogica.truaccess.api.ejb.API
SystemUser Component	SystemUserEJB	std_systemuserEjb.jar,META-INF/ejb-jar.xml	ejb/Audit	com.trulogica.truaccess.audit.ejb.Audit
Domain Component	DomainEJB	std_domainEjb.jar,META-INF/ejb-jar.xml	ejb/Audit	com.trulogica.truaccess.audit.ejb.Audit
Domain Component	DomainEJB	std_domainEjb.jar,META-INF/ejb-jar.xml	ejb/SystemUser	com.trulogica.truaccess.systemuser.ejb.!
Domain Component	DomainEJB	std_domainEjb.jar,META-INF/ejb-jar.xml	ejb/Service	com.trulogica.truaccess.service.ejb.Serv
API Component	APIEJB	std_apiEjb.jar,META-INF/ejb-jar.xml	SystemUserEJB	com.trulogica.truaccess.systemuser.ejb.!

g Click **Next**.

The Mapping Resource References table is displayed.



Install New Application

Allows installation of Enterprise Applications and Module

Step 1 Provide options to perform the installation

Step 2 Provide JNDI Names for Beans

Step 3 Map EJB references to beans

→ Step 4 : Map resource references to resources

Each resource reference defined in your application must be mapped to a resource.

javax.mail.Session

Specify existing Resource JNDI name: select... Apply

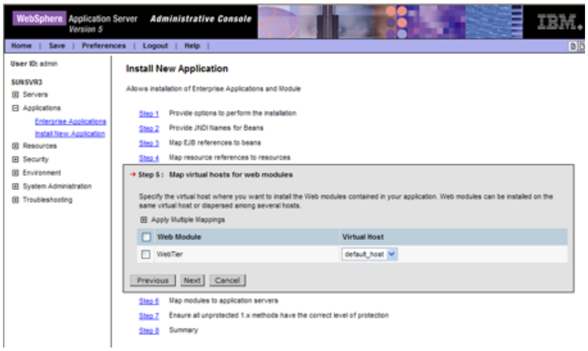
□	Module	EJB	URI	Reference Binding	JNDI Name
□	MailEJB	MailEJB	std_mailEjb.jar,META-INF/ejb-jar.xml	mail/MailSession	mail/TruAccess

javax.sql.DataSource

Specify existing Resource JNDI name: select... Apply

□	Module	EJB	URI	Reference Binding	JNDI Name
□	Authentication Module	AuthenticationEJB	std_authenticationEjb.jar,META-INF/ejb-jar.xml	jdbc/AuthenticationDataSource	jdbc/TruAccess
□	Id Generator Component	IdGenSessionEJB	std_idgenEjb.jar,META-INF/ejb-jar.xml	jdbc/IdGenDataSource	jdbc/TruAccess
□	MailEJB	MailEJB	std_mailEjb.jar,META-INF/ejb-jar.xml	jdbc/EMailDataSource	jdbc/TruAccess

h Click **Next**. The Specifying Virtual Host names and Recompiled JSP options for Web Modules table is displayed.



Install New Application

Allows installation of Enterprise Applications and Module

Step 1 Provide options to perform the installation

Step 2 Provide JNDI Names for Beans

Step 3 Map EJB references to beans

Step 4 Map resource references to resources

→ Step 5 : Map virtual hosts for web modules

Specify the virtual host where you want to install the Web modules contained in your application. Web modules can be installed on the same virtual host or dispersed among several hosts.

⊞ Apply Multiple Mappings

□	Web Module	Virtual Host
□	WebTier	default_host

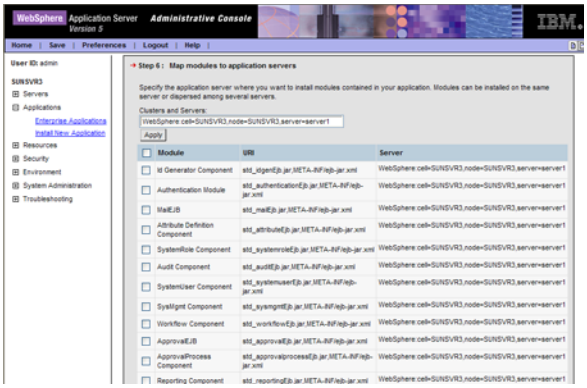
Previous | Next | Cancel

Step 6 Map modules to application servers

Step 7 Ensure all unprotected 1.x methods have the correct level of protection

Step 8 Summary

i Click **Next**. The Map Modules to application servers table is displayed.



→ Step 6 : Map modules to application servers

Specify the application server where you want to install modules contained in your application. Modules can be installed on the same server or dispersed among several servers.

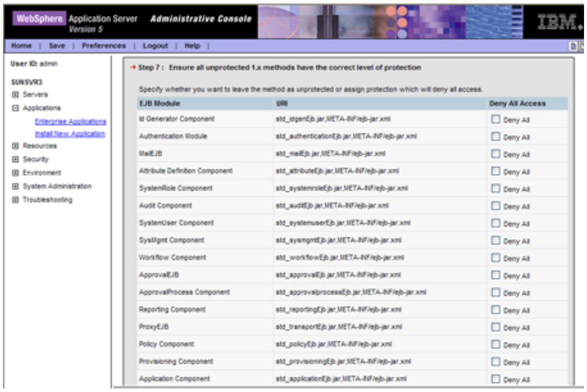
Clusters and Servers:
WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1

Apply

□	Module	URI	Server
□	Id Generator Component	std_idgenEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	Authentication Module	std_authenticationEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	MailEJB	std_mailEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	Attribute Definition Component	std_attributeEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	SystemRole Component	std_systemroleEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	Audit Component	std_auditEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	SystemUser Component	std_systemuserEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	SysMgmt Component	std_sysmgmtEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	Workflow Component	std_workflowEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	ApprovalEJB	std_approvalEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	ApprovalProcess Component	std_approvalprocessEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1
□	Reporting Component	std_reportingEjb.jar,META-INF/ejb-jar.xml	WebSphere:cell=SUNSVR3,node=SUNSVR3,server=server1

j Deselect all of the check boxes in the EJB Deploy table.

k Click **Next**. The Method Protection page is displayed.



→ Step 7 : Ensure all unprotected 1.x methods have the correct level of protection

Specify whether you want to leave the method as unprotected or assign protection which will deny all access.

EJB Module	URI	Deny All Access
Id Generator Component	std_idgenEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Authentication Module	std_authenticationEjb.jar,META-INF/ejb-jar.xml	□ Deny All
MailEJB	std_mailEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Attribute Definition Component	std_attributeEjb.jar,META-INF/ejb-jar.xml	□ Deny All
SystemRole Component	std_systemroleEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Audit Component	std_auditEjb.jar,META-INF/ejb-jar.xml	□ Deny All
SystemUser Component	std_systemuserEjb.jar,META-INF/ejb-jar.xml	□ Deny All
SysMgmt Component	std_sysmgmtEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Workflow Component	std_workflowEjb.jar,META-INF/ejb-jar.xml	□ Deny All
ApprovalEJB	std_approvalEjb.jar,META-INF/ejb-jar.xml	□ Deny All
ApprovalProcess Component	std_approvalprocessEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Reporting Component	std_reportingEjb.jar,META-INF/ejb-jar.xml	□ Deny All
ProxyEJB	std_transportEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Policy Component	std_policyEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Provisioning Component	std_provisioningEjb.jar,META-INF/ejb-jar.xml	□ Deny All
Application Component	std_applicationEjb.jar,META-INF/ejb-jar.xml	□ Deny All

l Review the setting and click **Next**.

m Click **Finish** on the confirmation page. The installation process may take some time.

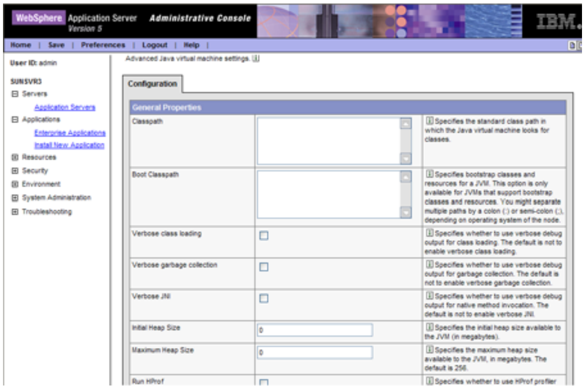
n Click **Save to Master Configuration** when the installation completes and click **Save** to confirm the action.

6 Log out of the WebSphere Administrative Console.

7 Stop and restart WebSphere.

8 Configure the path for `logging.properities` on WebSphere, as follows:

a Locate **WebSphere Administrative Domain** → **Servers** → **Application Servers** → ***application_server_name*** → **Process Definition** → **Java Virtual Machine**.



Advanced Java virtual machine settings.

General Properties

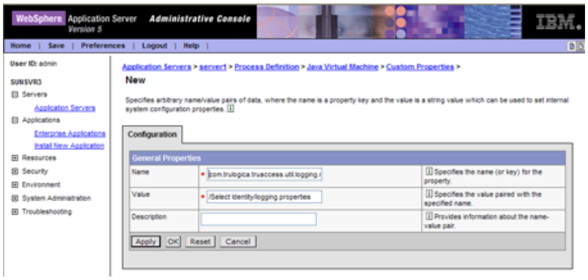
Classpath		Specifies the standard class path in which the Java virtual machine looks for classes.
Boot Classpath		Specifies bootstrap classes and resources for a JVM. This option is only available for JVMs that support bootstrap classes and resources. You might separate multiple paths by a colon (:) or semi-colon (;), depending on operating system of the node.
Verbose class loading	□	Specifies whether to use verbose debug output for class loading. The default is not to enable verbose class loading.
Verbose garbage collection	□	Specifies whether to use verbose debug output for garbage collection. The default is not to enable verbose garbage collection.
Verbose JNI	□	Specifies whether to use verbose debug output for native method invocation. The default is not to enable verbose JNI.
Initial Heap Size	0	Specifies the initial heap size available to the JVM (in megabytes).
Maximum Heap Size	0	Specifies the maximum heap size available to the JVM, in megabytes. The default is 256.
Run HProf	□	Specifies whether to use HProf profiler

b Click **Custom Properties** under the Advanced Settings section on the JVM Settings page.

c Add a new system property and enter the following information:

Name: **`com.trulogica.truaccess.util.logging.misc.config.file`**

Value: The path to `logging.properties`



New

Specifies arbitrary name/value pairs of data, where the name is a property key and the value is a string value which can be used to set internal system configuration properties.

Name	* com.trulogica.truaccess.util.logging.r	Specifies the name (or key) for the property.
Value	* /Select Identity/logging.properties	Specifies the value paired with the specified name.
Description		Provides information about the name-value pair.

d Click **OK** to save your settings.

9 Compile the JSPs on the server. The file `connector.jar` should be in WebSphere's classpath. Make sure that this file is copied to the `$WAS_HOME/lib/ext` directory.

Run the following from a command shell:

```
cd $WAS_HOME/bin
```

```
JspBatchCompiler.sh -enterpriseapp.name lmz -verbose false -
keepgenerated true -server.name <servername>
```

Make sure that compilation goes through without any errors.

10 After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebSphere, it resides in the `$WAS_HOME/java/jre/lib` directory.

⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

Installing in a Clustered Environment

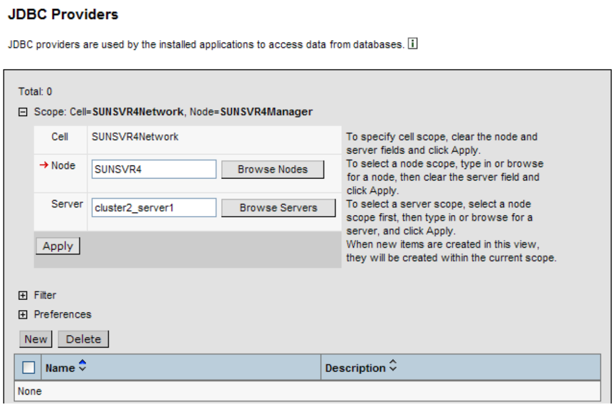
Complete the following steps to configure an IBM WebSphere application clustered environment for use with Select Identity:

1 Ensure that all WebSphere servers in the cluster meet the *minimum* requirements, as documented in System Requirements on page 17.

2 Log on to the WebSphere Network Deployment Manger's Administrative Console as **admin**.

3 Deploy the JDBC Provider.

a Click **Resources->JDBC Providers**.



JDBC Providers

JDBC providers are used by the installed applications to access data from databases. i

Total: 0

⊟ Scope: Cell=**SUNSVR4Network**, Node=**SUNSVR4Manager**

Cell	SUNSVR4Network	To specify cell scope, clear the node and server fields and click Apply.
→ Node	SUNSVR4 [Browse Nodes]	To select a node scope, type in or browse for a node, then clear the server field and click Apply.
Server	cluster2_server1 [Browse Servers]	To select a server scope, select a node scope first, then type in or browse for a server, and click Apply.
[Apply]		When new items are created in this view, they will be created within the current scope.

⊞ Filter

⊞ Preferences

[New] [Delete]

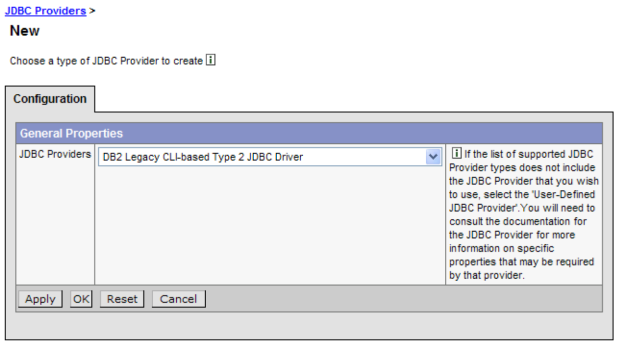
□	Name ⇕	Description ⇕
None		

b Navigate to and choose the Node and Server on which you are installing.

c When finished click **Apply**.

➤ These steps must be performed on each application server in the cluster. You may be able to deploy the JDBC Provider a at the cell or node scope. Check with your WebSphere administrator.

d Click **New** on the same page.



JDBC Providers >

New

Choose a type of JDBC Provider to create

Configuration

General Properties

JDBC Providers	DB2 Legacy CLI-based Type 2 JDBC Driver	If the list of supported JDBC Provider types does not include the JDBC Provider that you wish to use, select the 'User-Defined JDBC Provider'.You will need to consult the documentation for the JDBC Provider for more information on specific properties that may be required by that provider.

e From the drop-down list, select the **DB2 Legacy CLI-Based Type2 JDBC Driver**.

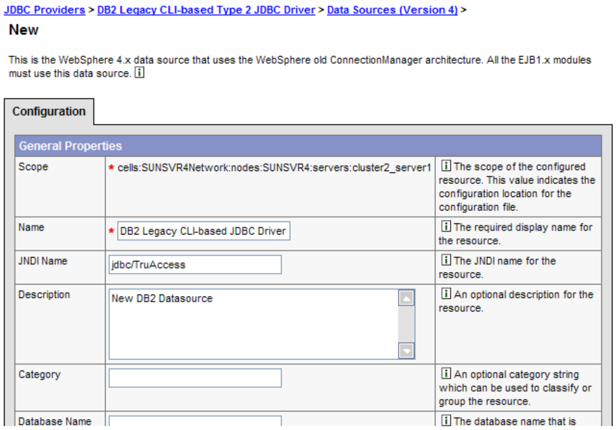
f Click **Apply**. The Configuration page displays.

g Enter a name for the resource provider in the Name field, such as SI DB2 JDBC Driver.

h To save your settings, click **Apply**.

i At the bottom of the page under Additional Properties, click **Data Sources (Version4)**.

j Click **New**. The Configuration page displays.



JDBC Providers > DB2 Legacy CLI-based Type 2 JDBC Driver > Data Sources (Version 4) >

New

This is the WebSphere 4.x data source that uses the WebSphere old ConnectionManager architecture. All the EJB1.x modules must use this data source.

Configuration

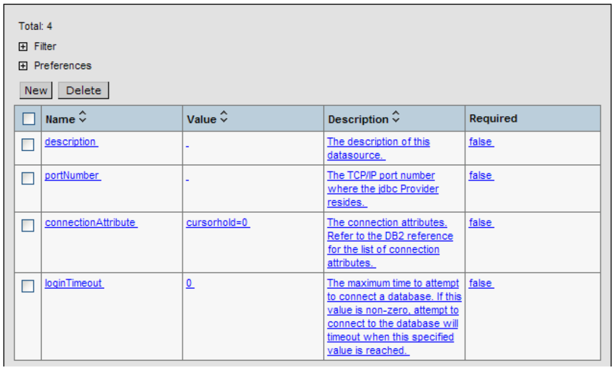
General Properties

Scope	* cells:SUNSVR4Network:nodes:SUNSVR4:servers:cluster2_server1	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Name	* DB2 Legacy CLI-based JDBC Driver	The required display name for the resource.
JNDI Name	jdbc/TruAccess	The JNDI name for the resource.
Description	New DB2 Datasource	An optional description for the resource.
Category		An optional category string which can be used to classify or group the resource.
Database Name		The database name that is

k Enter the path to the JNDI name for the resource in the JNDI Name field.

l Enter your database connection information in the fields provided.

m Click **Apply**. Additional Properties are available at the bottom of the page. Click the **Custom Properties** link. The following displays.



Total: 4

⊞ Filter

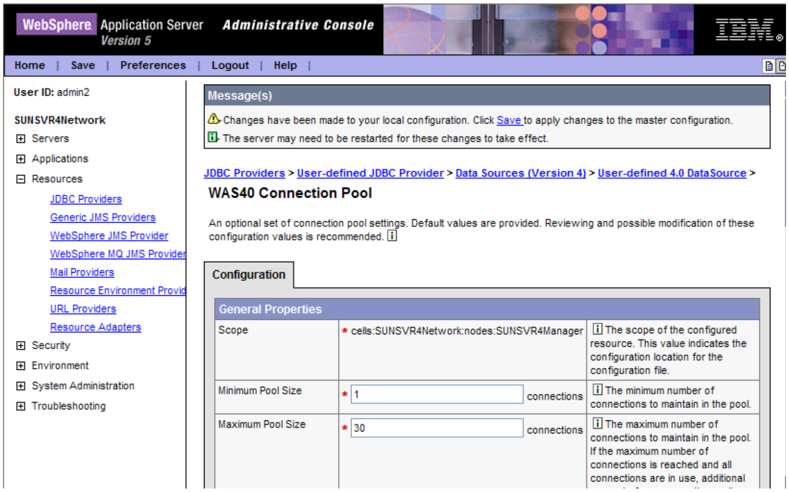
⊞ Preferences

□	Name	Value	Description	Required
□	description	_	The description of this datasource.	false
□	portNumber	_	The TCP/IP port number where the jdbc Provider resides.	false
□	connectionAttribute	cursorhold=0	The connection attributes. Refer to the DB2 reference for the list of connection attributes.	false
□	loginTimeout	0	The maximum time to attempt to connect a database. If this value is non-zero, attempt to connect to the database will timeout when this specified value is reached.	false

n For the Connection Attribute, set the **cursorhold** value to **1**.

o Click **Apply**.The General Properties page displays.

p Select **Resources** → **JDBC Providers** → **Db2JdbcDriver** → **Data Sources (Version4)**→ ***data_source***. Under Additional Properties, click on the **Connection Pool** link. The Configuration page is displayed.



Message(s)

Changes have been made to your local configuration. Click Save to apply changes to the master configuration.

The server may need to be restarted for these changes to take effect.

JDBC Providers > User-defined JDBC Provider > Data Sources (Version 4) > User-defined 4.0 DataSource >

WAS40 Connection Pool

An optional set of connection pool settings. Default values are provided. Reviewing and possible modification of these configuration values is recommended.

Configuration

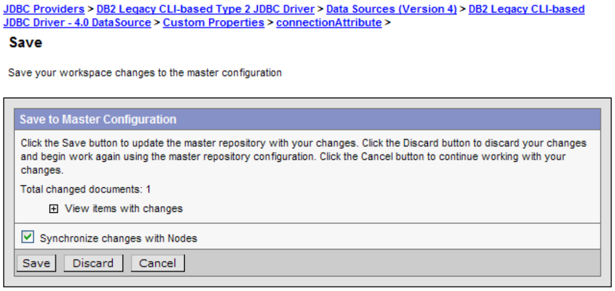
General Properties

Scope	* cells:SUNSVR4Network:nodes:SUNSVR4Manager	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Minimum Pool Size	* 1 connections	The minimum number of connections to maintain in the pool.
Maximum Pool Size	* 30 connections	The maximum number of connections to maintain in the pool. If the maximum number of connections is reached and all connections are in use, additional

q Select the Disable Auto Connection Cleanup check box. For a cluster, this has to be done at the server scope.

r Click **Apply** and **Save**.

s Click **Save** at the top of the page.



JDBC Providers > DB2 Legacy CLI-based Type 2 JDBC Driver > Data Sources (Version 4) > DB2 Legacy CLI-based JDBC Driver - 4.0 DataSource > Custom Properties > connectionAttribute >

Save

Save your workspace changes to the master configuration

Save to Master Configuration

Click the Save button to update the master repository with your changes. Click the Discard button to discard your changes and begin work again using the master repository configuration. Click the Cancel button to continue working with your changes.

Total changed documents: 1

⊞ View items with changes

☑ Synchronize changes with Nodes

t Check the **Synchronize changes** check box and click **Save**.

Test the connection and make sure that the connectivity is established.

4 Configure the Generic JVM arguments. This step must be performed for each server in the cluster.

a Click **Servers->Application Servers**. Select the first server in the cluster. The Server Configuration page displays.

b Click **Process Definition.**

c Click **Java Virtual Machine** link under Additional Properties.

d Under Generic JVM Arguments, enter the location of the `TruAccess.properties` file, such as:

```
Dcom.trulogica.truaccess.property.file=/selectIdentity/
properties/TruAccess.properties
```

If installing in a UNIX environment add the following to the line above.

```
-Djava.awt.headless=true
```

e Click **Apply**.

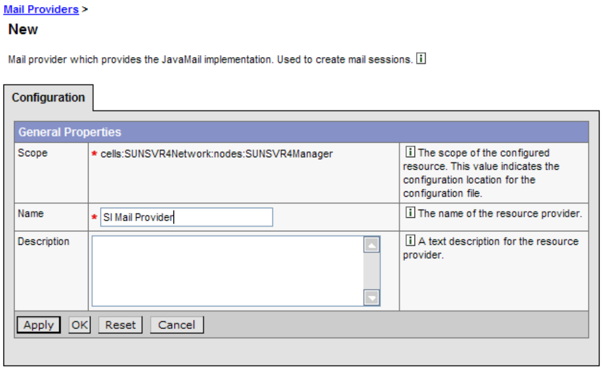
f Click Save at the top of the page.

g Select the **Synchronize changes** check box and click **Save**.

5 Configure the mail session by completing the following steps:

a From the tree on the left-hand side of the console, select **Resources** → **Mail Providers**.

b Click the **New** button on the Mail Sessions page. The following form is displayed.



Mail Providers >
New

Mail provider which provides the JavaMail implementation. Used to create mail sessions.

Configuration

General Properties

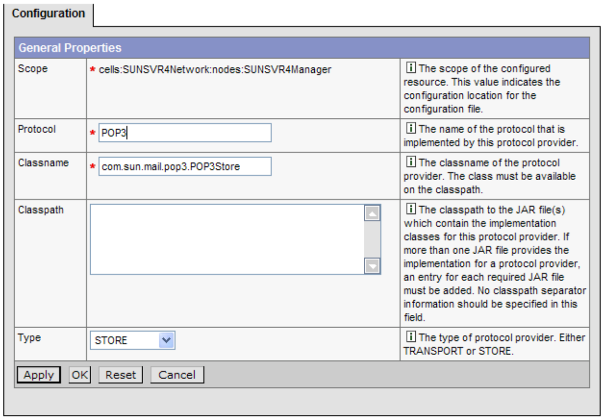
Scope	* cells:SUNSVR4Network:nodes:SUNSVR4Manager	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Name	* SI Mail Provider	The name of the resource provider.
Description		A text description for the resource provider.

c Enter an name for the mail provider, such as SI Mail Provider.

d Click **Apply**. Additional Properties Display at the bottom of the page.

e Click **Protocol Providers**.

f Click **New**.



Configuration

General Properties

Scope	* cells:SUNSVR4Network:nodes:SUNSVR4Manager	The scope of the configured resource. This value indicates the configuration location for the configuration file.
Protocol	* POP3	The name of the protocol that is implemented by this protocol provider.
Classname	* com.sun.mail.pop3.POP3Store	The classname of the protocol provider. The class must be available on the classpath.
Classpath		The classpath to the JAR file(s) which contain the implementation classes for this protocol provider. If more than one JAR file provides the implementation for a protocol provider, an entry for each required JAR file must be added. No classpath separator information should be specified in this field.
Type	STORE	The type of protocol provider. Either TRANSPORT or STORE.

g Enter **POP3** as the Protocol.

h Enter **com.sun.mail.pop3.POP3Store** in the classpath field.

i Click **Apply** and **Save**. Synchronize changes with all nodes.

j Repeat the previous steps to add another Mail provider for SMTP transport. The classpath for SMTP is **com.sun.mail.smtp.SMTPTransport**.

k After creating the two protocol providers, create a new mail session.

a From the tree on the left-hand side of the console, select **Resources** → **Mail Providers**.

b Click the **New** button on the Mail Sessions page.

c Enter an name for the mail provider, such as SI Mail Provider.

d Click **Apply**. Additional Properties Display at the bottom of the page.

e Click **Mail Sessions**.

f Click **New**.

g Enter a name such as SI Mail Session.

h Enter a JNDI name such as `mail/TruAccess`.

i Enter the host name of the mail transport host in the Mail Transport Host field.

j Enter all of the connection information in the fields provided. The transport protocol is **SMTP** and the storage protocol is **POP3**.

k Click **Apply**, save your settings, and synchronize with all nodes.

6 Deploy Select Identity on the web servers.

a Navigate to Applications->Enterprise Applications.

b Click **Install**.

c Click **Browse** to navigate to `websphere_lmz.ear` file.

d Click **Next**.

e Choose all default options and deploy to all servers in the cluster.

> Do not pre-compile the JSPs during the deployment. This is done in the following step.

7 Compile the JSPs on each server in the cluster. The file `connector.jar` should be in WebSphere’s classpath. Make sure that this file is copied to the `$WAS_HOME/lib/ext` directory.

Run the following from a command shell:

```
cd $WAS_HOME/bin
```

```
JspBatchCompiler.sh -enterpriseapp.name lmz -verbose false -
keepgenerated true -server.name <servername>
```

Make sure that compilation goes through without any errors. All connectors that you install must be installed at the Node scope level.

8. Configure Virtual hosts from the Network Deployment Manager Console by selecting **Environment->Virtual Hosts**. Refer to WebSphere Network Deployment Edition manual for virtual host configuration. Host Aliases must be defined for each HTTP transport port in the web container within a cluster. If the virtual host uses the default port (80), an entry for port 80 should be specified in the host alias.
9. Update the Web Server Plugin from the Network Deployment Manager Console by selecting **Environment->Update Web Server Plugin**.
10. Log out of the WebSphere Administrative Console.
11. Stop and restart the web server.
12. After installing Select Identity, refer to Logging on page 110 for instructions on configuring the `logging.properties` file. By default, a `logging.properties` file is provided by the application server's JVM. On WebSphere, it resides in the `$WAS_HOME/java/jre/lib` directory.

    ⚠ Configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

There are additional configuration steps for WebLogic and Websphere installations see Additional Configuration on page 85 to finish installing Select Identity.

6

Additional Configuration

After you install HP OpenView Select Identity on the application server, you configure the properties and keystore for the application. You will perform the following for WebLogic and WebSphere installations.

➤ If you are installing in a clustered environment, these configuration steps must be performed on all nodes in the cluster.

Customizing TruAccess.properties

Several configuration settings are provided in the `TruAccess.properties` file. Most are optional settings that determine defaults for the Select Identity client, but some are required by the installation process.

Required Settings for Installation

- `truaccess.sender.email`

  Specify a general email address that will be used as the sender's address for any email that is sent by Select Identity. For example, you may want to specify **`info@`*`your_company`*`.com`** or **`select_identity_admin@`*`your_company`*`.com`**. This address must exist on the SMTP server configured for use by Select Identity's application server.

  You can also specify a value for `truaccess.sender.name` to coincide with this setting.

- `truaccess.method`, `truaccess.host`, `truaccess.port`

  Provide values that make up the URL of the Console interface. Specify the protocol, host name or IP address, and port, such as **http://localhost:7001/**.

- `truaccess.loginURL`, `truaccess.logoutPage`

  Specify the logon page and the page that is displayed when the user logs out of Select Identity. Synchronize these settings with the values of `truaccess.method`, `truaccess.host`, and `truaccess.port`.

  If you are using external authentication, make sure that the `truaccess.authentication` setting is set to **`off`**.

- `truaccess.repository.type`

  This setting defines the type of database server you are using. Specify **`mssql`** for Microsoft SQL Server, **`oracle`** for Oracle, or **`db2`** for DB2. Oracle is the default setting.

  This setting is case sensitive.

- `truaccess.repository.oracle.driver.bea`

  If you are running Select Identity on WebLogic, connecting to an Oracle database, and using the WebLogic JDBC driver for Oracle (which provides internationalization support), you must set this property to **`yes`**.

- `truaccess.upload.filedir`

  Specify a valid location on the Select Identity server that can be used as temporary storage while Select Identity uploads files to the database.

- `contact_helpdesk=Please contact the helpdesk at 972-309-6408`

  Provide the error message that is displayed if the user cannot log on to the Select Identity client.

Optional Settings

You can replace the default HP logo with your client logo. Uncomment the setting and add your relative or fully qualified path name. For example:

Relative path name:
`hp.si.logo.file=/lmz/images/<your_logo>.gif`

Fully qualified path name:

`hp.si.logo.file=http://welcome.<yourcompany>.com/img/<your_logo>.gif`

Specify a valid location, either a path name or URL link. The replacement of the HP logo file is a simple replacement only. Following are restrictions that apply, but are not limited to:

- Cropped Image Size: 755 x 71, while the image can be larger than this, it will be cropped at 755 x 71. In the event that the image is smaller than the cropped size, the image will be top-left justified and will not repeat like tiles.
- Image Maps: Image maps are not usable on the HP logo.
- URL Links: URL links cannot be applied to the HP logo.

You can configure the remaining settings in the file, though default values are provided and you can set them at your convenience. See Configuring TruAccess.properties on page 113 for information about additional file settings.

Generating the Keystore

If you wish to enable the Select Identity server to encrypt and decrypt data it stores in the database using your keystore (rather than the default provided by Select Identity), you must generate the keystore and integrate it into Select Identity. A keystore is a database of keys. The private keys are associated with a certificate chain, which authenticates the corresponding public key. The keystore also contains certificates from trusted entities. By generating the keystore, you add another layer of security to the data that is exchanged in the Select Identity system.

You must perform this procedure before using Select Identity. You cannot use your keystore to decrypt data after Select Identity stores (and encrypts) data in the database using the default keystore.

Complete the following steps to generate a keystore:

1 Copy the contents of the `SI302\keystore\` directory on the Select Identity CD to a keystore directory on the Select Identity server. The files include the following:

   — `ks_gen.bat` — The DOS batch file that generates a keystore containing the secret key to encrypt and decrypt data

   — `ks_gen.sh` — The UNIX shell script that generates a keystore containing the secret key to encrypt and decrypt data

   — `SIPubKey` — The binary file that contains the public key to encrypt the password of the keystore and alias

   — `SIKeyStoreUtil.jar` — The executables of they Keystore utility

   — `bcprov-jdk13-124.jar` — The executables of the Bouncy Castle Java Cryptography RSA implementation

   — `sunjce_provider.jar` — The executables of the Sun Java Cryptography implementation

2 Ensure that the JRE (version 1.3.1+) is installed and included in the PATH environment variable on the system.

3 Run the `ks_gen.bat` (on Windows) or `ks_gen.sh` (on UNIX) script and following the instructions, using the distributed public key file during the process.

4 Save the secret string you use in case Support needs to analyze encrypted data for data recovery.

The resulting keystore properties file should look similar to this:

```
#Select Identity Keystore Parameters
#Fri. Aug 20 10:02:42 CDT 2005
si.keystore.alias=test_alias
si.keystore.storepass=<encoded string>
si.keystore.keypass=<encoded string>
si.keystore.filepath=c:/temp/SI/test.keystore
```

Next, you must configure the Select Identity server to use the keystore. Complete the following steps to do so:

1. Shutdown the Select Identity server.
2. Edit the keystore properties file and change the location where the keystore is saved (specified by the si.keystore.filepath parameter).
3. Edit the `TruAccess.properties` file, and add the following lines:

   ```
   si.rsa.provider=
   org.bouncycastle.jce.provider.BouncyCastleProvider
   ```

   ```
   si.keystore.paramfile=location_of_the_keystore_properties_file
   ```
4. Restart the Select Identity server.

Internationalization and Localization

Select Identity is internationalized and is able to operate with languages that are supported by the Java Unicode specification. Internationalization support in Select Identity includes the following capabilities:

- The user can enter the local language characters as input data. The display text provided by Select Identity, such as labels, help text, and other static display strings are shown in English.

  XML files used for Select Identity web service, Auto Discovery, and rules can take foreign characters as tag or attribute values. The exported XML files through Configuration pages allow foreign characters as well. You can enter foreign characters directly into the XML files as long as they are entered in an editor with UTF-8 encoding enabled. In general, any UTF-8 supported editors can be used for this purpose. However, some editors could store additional hidden characters while saving the file. To ensure that the XML files containing foreign characters are stored correctly, Select Identity recommends using XML editors such as XMLSpy.
- The date and time are displayed in the local format.
- Linguistic sorting is not supported.

Internationalization is supported for Select Identity on the following platform:

- Application server – WebLogic 8.1
- Database – Oracle 9i/UTF-8

- Connectors – LDAP/UTF-8

➤ Make sure that your database supports the language characters that you want to use.

The following sections provide platform and environment-specific configurations.

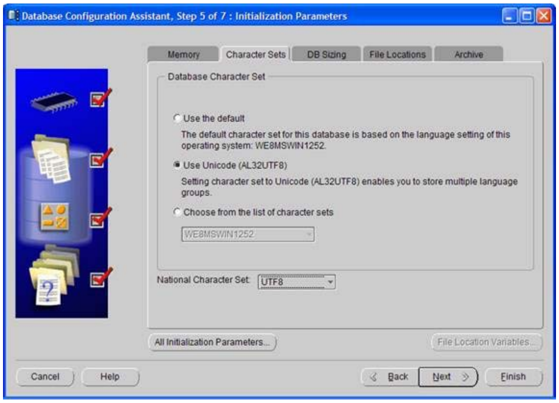
UTF-8 Encoding on Oracle9i

➤ If you use the BEA Oracle driver to connect to an internationalized Oracle database, you must:

- Edit the connection pool created by the installation, as documented in Step 4 in Installing Manually on WebLogic on page 48.
- Set the `truaccess.repository.oracle.driver.bea` property in the `TruAcccess.properties` file to `yes`, as documented in Customizing TruAccess.properties on page 85.

Perform the following to set UTF-8 encoding for Oracle at database creation:

1. For Oracle 9i, open the Initialization Parameters window and select the **Character Set** tab.
2. Select the **Use Uicode (AL32UTF8)** radio button as shown.



Database Configuration Assistant, Step 5 of 7 : Initialization Parameters

Database Character Set

○ Use the default

The default character set for this database is based on the language setting of this operating system: WE8MSWIN1252.

● Use Unicode (AL32UTF8)

Setting character set to Unicode (AL32UTF8) enables you to store multiple language groups.

○ Choose from the list of character sets

WE8MSWIN1252

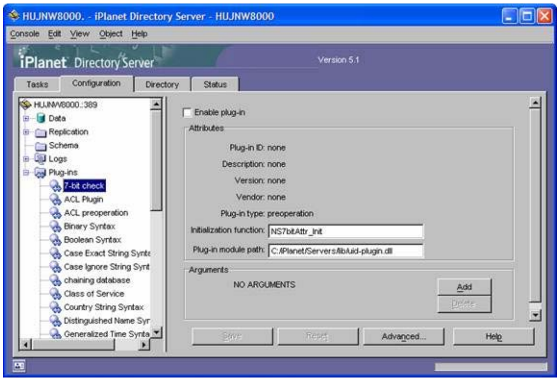
National Character Set: UTF8

iPlanet LDAP Configuration

Perform the following to disable 7-bit ASCII:

1 In iPlanet’s Configuration window, expand the plug-ins node and select the **7-bit** check box.

2 Deselect **Enable plug-in**, which is selected by default.



□ Enable plug-in

Attributes

Plug-in ID: none
Description: none
Version: none
Vendor: none
Plug-in type: preoperation
Initialization function: NS7bitAttr_Init
Plug-in module path: C:/iPlanet/Servers/lib/uid-plugin.dll

Arguments

NO ARGUMENTS

Set Encoding in Internet Explorer

Perform the following to set encoding in Internet Explorer to UTF-8 and define a language:

1 From the main menu, select **View->Endoding->UTF-8**.

2 Select **Tools->Internet Options**.

3 Click the **Languages** button.

4 Click **Add**.

5 Select the desired locale from the Language list and click OK.

6 Select the language and move it to the top of the list.

Add Supported Language Fonts

The JDK font properties file ships with most languages. Perform the following to add language fonts that do not exist in the file:

In `$JAVA_HOME/jre/lib/font.properties`, add font entries for supported languages.

For example, to add Chinese GB2312 for normal and bold face fonts, add the following lines near font definition lines with similar names:

```
dialog.3=\u5b8b\u4f53,GB2312_CHARSET
dialog.bold.3=\u5b8b\u4f53,GB2312_CHARSET
```

Logging on to Select Identity

After Select Identity is installed, an administrative account is created:

Login: `sisa`

Password: `abc123`

Log on to the system with this account information and create a new Select Identity system administrator based on your company's security policies, and delete the `sisa` account.

Log on to Select Identity by entering the following URL in the web browser (ports cited below are the defaults):

http://*app_svr_host*:*port*/lmz/control/home

On WebLogic, the default port is 7001. On WebSphere, the default port is 9080.

7

Upgrading From Version 3.0.2

If you are upgrading your 3.0.2 Select Identity WebLogic installation to this version, you will perform the following migration steps.

Migrating Procedures

1. In the 3.0.2 Select Identity client, approve any "pending" workflow tasks before starting the migration process.
2. Backup your existing 3.0.2 Select Identity database, folders, and files.
3. Using an SQL Client, execute the database migration scripts located in the Upgrade folder on the CD. These scripts will update your existing database.
   a. For SQL Server:
      `\3.3\SI33WebLogic\database\Upgrade\MSQL`

      See Microsoft SQL Server on page 23 for installation procedures.
   b. For Oracle:
      `\3.3\SI33WebLogic\database\Upgrade\Oracle`

      See Oracle on page 26 for installation procedures.

c. For DB2:
   `\3.3\SI33WebLogic\database\Upgrade\DB2`

   See DB2 on page 28 for installation procedures.

4. Uninstall the previous release of Select Identity (3.0.2) using the manual steps for uninstall specified in Uninstalling Select Identity on page 98.

   Save your existing `TruAccess.properties` file. You may need to reference it when configuring the new file.

5. Install the new release of Select Identity (3.3) using the manual steps specified in Using the Manual Installation Process on page 46. You do not need to create a new database.

6. Configure the new `TruAccess.properties` file. See Customizing TruAccess.properties on page 85 for required settings.

   You may also want to refer to your 3.0.2 file to assist you when updating the values for the new properties file.

7. Check the values for `truaccess.fixedtemplate.bulk_default` and `truaccess.fixedtemplate.bulk_move` properties in the `TruAccess.properties` file. The ReconciliationDefaultProcess cannot be designated as the workflow for either of these properties.

   Change the properties value to use either the SIBulkOneStageApproval or the SI Provisioning Only Bulk template.

8. Using the Configurations capability in the Select Identity client, import the following:

   a. Select the Workflow Application Definition configuration type and import the `Workflow Application Definition.xml` file located in the Migration folder on the CD.

   b. Select the Request Instance Report configuration type and import the `Request Instance Report.xml` file located in the Migration folder on the CD.

9. Modify all resources. In some cases, the passwords for administrators may need to be re-entered. Verify all resource connections on all resources by clicking **Test** and **Submit**.

10. Many of the default workflow templates have changed in version 3.3. If you modified the default 3.0.2 workflow templates, you may want to save a copy of these templates before performing the next step (which replaces the existing templates with a newer template).

The Copy Workflow feature can be used to copy the 3.0.2 workflow templates to a new name. Template Names starting with "SI," "Reconciliation," or "UserAccountExpiration" are all default templates you may want to copy.

11 It is recommended that you import the new 3.3 workflow templates located in the Migration\Workflow Templates folder on the CD.

If you choose to use your existing templates, perform the required and optional changes in the next section.

12 If you are updating a cluster, restart the managed servers to ensure that changes are propagated to all servers in the cluster.

Using Existing Templates

Following is a list of required and optional changes that should be made to existing templates.

Required Changes

Bulk Post Provisioning

For any workflow template used for bulk processing, do the following:

1 Use the Application Invocation **Post Provision in Bulk Processing to save data** for its **post provisioning** activity block.

2 Delete the old application invocation called **Post Provision in Reconciliation to save data**

3 Replace it with the new Application Invocation, **Post Provision in Bulk Processing to save data**. For default templates, this would be in the PostProvisioningSB activity.

The following default templates are used by bulk processing and should be changed:

SIBulkOneStageApproval
SI Provisioning Only Bulk
ReconciliationDefaultProcessMove

4 Remember to change any additional templates you have created that are used for bulk processing.

Optional Changes

Single Block Activities

A new feature of 3.3 is the Workflow Studio's new set of activities that represent a single block. Use of these in the workflow help simplify the look of the workflow and provide some performance improvements. You may change your existing workflow templates to incorporate the single block feature.

For examples of using single block activities, view the default workflow templates, SI OneStageApproval and SI ThreeStageApproval provided in the Migration\Workflow Templates folder. For information on using single block activities, see "Activities and Blocks" in Chapter 3 of the *HP OpenView Select Identity Workflow Studio Guide*.

Reset Password Enhancements

To improve the error handling within a Workflow, Select Identity 3.3 has introduced the following new features you may want to use:

- Modify Provisioning Transitions: Provisioning has added a new return code of 105. The 105 return code occurs when provisioning a "reset password" for a user, and the request fails for one of the resources. This allows workflows to distinguish "password reset failures" from other types of failures.

  View the workflow templates provided in the Migration\Workflow Templates folder for examples of how to use this new return code. Existing workflows, which are not changed to reflect the new return code, will continue to work as they did in 3.0.2.

- The default notification template Reset Password has been enhanced for 3.3. The 3.0.2 notification template used the variable [RQT:Password]. For 3.3, it is recommended that the variable [USERDEF:ResetStatus] be used instead. The new variable includes a breakdown of each resource that was provisioned for the reset password action and the success or failure associated with the operation.

- In 3.3, the email notification template has been improved to reference the new [USERDEF:ResetStatus] variable. To use this new variable do the following:

  a. Add a new activity to any workflow that could be used in "resetting a password" for a user.

b. Use the Application Invocation **email notification** for the new activity to send a notification using the Reset Password template.

c. Modify the transitions in the workflow to use this activity when it is a reset password action.

8

Uninstalling Select Identity

There are a number of places where Select Identity stores information. To completely uninstall the product, perform the steps in each of the following sections.

Using the Wizard to Uninstall from the Application Server

To use the uninstall wizard to remove Select Identity from the application server, run the `Uninstall Select Identity.exe` (on Windows) or `Uninstall Select Identity.bin` (on UNIX) to launch the wizard. These files reside in the Select Identity home directory on the application server. Follow the prompts. When complete, the wizard removes the LMZ file, data source, connection pool, and mail session.

Manually Uninstalling from the Application Server

The following sections describe how to manually remove Select Identity from each type of supported application server.

On WebLogic

The following sections provide steps for a complete uninstall from WebLogic.

Deleting the EAR File

To uninstall Select Identity on WebLogic, you delete the `lmz.ear` file from the WebLogic server.

Make sure that all dependencies on the system are removed.

Complete the following steps:

1. Log in to the WebLogic Server Console.
2. Select the ***domain_name*** → **Deployments** → **Applications** folder.
3. Click the **Delete** button (🗑) next to the lmz application.
4. When prompted to confirm the deletion, click **Yes**.

Deleting the Connectors

You may have any number of connectors installed to support system resources. If you are completely uninstalling the Select Identity product you will want to uninstall the connectors.

Perform the following steps:

1. Log in to the WebLogic Server Console.
2. Select the ***domain_name*** → **Deployments** → **Connector Module** folder.
3. Click the **Delete** button (🗑) next to the connectors that you have installed.

4. When prompted to confirm the deletion, click **Yes**.
5. Click **Continue**.

Deleting the Data Source

Perform the following steps to delete the Select Identity data source:

1. Log in to the WebLogic Server Console.
2. Select the ***domain_name*** → **Services** → **JDBC**→ **Data Sources** folder.
3. Click the **Delete** button (🗑) next to the **jdbc/TruAccess** connection.
4. When prompted to confirm the deletion, click **Yes**.
5. Click **Continue**.

Deleting the Connection Pool

Perform the following steps to delete the Select Identity connection pool:

1. Log in to the WebLogic Server Console.
2. Select the ***domain_name*** → **Services** → **JDBC**→ **Connection Pools** folder.
3. Click the **Delete** button (🗑) next to the connection pool that was used by the data source.
4. When prompted to confirm the deletion, click **Yes**.
5. Click **Continue**.

Deleting the Mail Session

Perform the following steps to delete the Select Identity mail session:

1. Log in to the WebLogic Server Console.
2. Select the ***domain_name*** → **Services** → **JDBC**→ **Mail Session** folder.
3. Click the **Delete** button (🗑) next to the **mail/TruAccess** connection.
4. When prompted to confirm the deletion, click **Yes**.
5. Click **Continue**.

On WebSphere

Perform the following to uninstall Select Identity from the WebSphere server:

1. Open the Application Server Administrative Console.
2. Click the server node on which Select Identity is installed from the tree to the left.
3. In the tree view for this server, select **Enterprise Applications**. The following is displayed.
4. Select **lmz** from the list and click **Stop** to stop the service.
5. Select **lmz** again and click **Uninstall**.
6. Ensure that the Delete all related files setting is **Yes**.
7. Click **Save** to save the configuration settings.
8. Exit the Console.

Uninstalling the Select Identity Database

After you uninstall the product from the web server, you can uninstall the data and tables from the database.

On Microsoft SQL Server

Perform the following steps to uninstall the Select Identity from Microsoft SQL Server:

1. Log in to the Microsoft SQL Server Enterprise Manager.
2. Under Microsoft SQL Server, locate the `Select_Identity` database instance.
3. Right-click on the database.
4. Select **Delete**.
5. To confirm the action, click **Yes**.
6. Under the **Security** heading, click **Logins**.

7 Right-click on the Select Identity database user name, such as `SI` in previous procedures.

8 Select **Delete**.

9 To confirm the action, click **Yes**.

On Oracle

Perform the following steps to uninstall the Select Identity database from Oracle:

1 From a SQL Plus command prompt, log in to Oracle as a user with system permissions.

2 Enter the following command:

```
drop user Select_Identity_database_username cascade
```

On DB2

To remove the Select Identity database and schema from DB2, launch the DB2 Control Center, select the Select Identity database, and remove it. Refer to the DB2 documentation for details.

A

Troubleshooting

This chapter provides error messages that you may encounter when configuring the web application server for use with HP OpenView Select Identity. A suggested solution is also provided for each message.

General Installation Errors

If you encounter errors during the installation or cannot open the Select Identity client, please check the following:

- The installer fails to deploy the connection pool. Check each of the following:
  - — Make sure that the correct port number is configured during the installation. The default port option is for Oracle.
  - — The database schema may not have been populated before the installer started. Make sure that the installation prerequsites are met before installing.
- The Select Identity client does not start after installation.

  Ensure that the `connector.jar` file is correctly added to the WebLogic class path in the `startweblogic.cmd` file.

System Errors on WebLogic

By default, trace information is displayed in the window from which the WebLogic Application Server was started.

- The WebLogic Server does not start.

  Possible Cause: The `logging.properties` file is not configured properly.

  Possible Solution: For more information, see Logging on page 110 for details. In particular, make sure that the directory specified for the FileHandler log file (the **pattern** attribute in the message format) exists.

- The WebLogic Server does not recognize the lmz application.

  Possible Cause: An anomaly in the installation.

  Possible Solution: Add the EJBs to the WebLogic server using the WebLogic Server Console.

- When the WebLogic Server starts, the following error is displayed:

  ```
  <Error> <JDBC> <Cannot startup connection pool
  "ConceroConnectionPool" weblogic.common.ResourceException:
  Could not create pool connection. The DBMS driver exception was:
  java.sql.SQLException: SQL Server has been paused.
  ```

  Possible Cause: SQL Server is not running.

  Possible Solution: Start SQL Server.

- When the WebLogic Server starts, the following error is displayed:

  ```
  <Error> <JDBC> <Cannot startup connection pool
  "ConceroConnectionPool" weblogic.common.ResourceException:
  Could not create pool connection. The DBMS driver exception was:
  java.sql.SQLException:
  Login failed for user 'sa'. Severity 14, State 1, Procedure
  'null null', Line 0 Unable to connect, please check your
  server's version and availability.
     at weblogic.jdbc.mssqlserver4.TdsStatement.
     microsoftLogin(TdsStatement.java:2872)
  ```

  Possible Cause: The user ID or password is configured incorrectly for SQL Server.

- When attempting to sign in to Select Identity (through the web browser), an Error 500 -Internal Server Error is displayed on the page and the following error message is displayed in the server’s window:

  ```
  <Error> <JDBC> <Error during Data Source creation:
  weblogic.common.ResourceException: DataSource(jdbc.AccessUsDB)
  can't be created with non-existent Pool (connection or multi)
  (ConceroConnectionPool)>
  ```

  Possible Cause: The targets for the JDBC connection pool may not be configured correctly.

- When attempting to create an administrator, this error is displayed:

  ```
  createAndSendMail exception : javax.mail.SendFailedException:
  Sending failed;
  nested exception is:
  javax.mail.MessagingException: Could not connect to SMTP host:
  65.70.174.236, port: 25;
  ```

  Possible Cause: The mail server is not available or the mail server configuration is not correct.

System Errors on WebSphere

By default, WebSphere logs informational messages to `default_server_stderr.log` and error messages to `default_server_stdout.log`. These files reside in the *`WebSphere_home_dir`*`\AppServer\logs\` directory.

The following system errors may be logged in the files:

- When the WebSphere Application Server started, the following error was logged to `default_server_stdout.log`:

  ```
  ExceptionUtil X CNTR0020E: Non-application exception occurred
  while processing method getDomainList on bean
  BeanId(lmz#sysmgmtEjb.jar#SysMgmtEJB, null):
  java.lang.NullPointerException
  at COM.ibm.db2.jdbc.DB2PooledConnection.close
   (DB2PooledConnection.java:235)
  at com.ibm.ejs.cm.portability.PortabilityLayerImpl.
   getInstance(PortabilityLayerImpl.java:1019)
  at com.ibm.ejs.cm.portability.PortabilityLayerFactory.
  ```

```
 getPortabilityLayer(PortabilityLayerFactory.java:62)
at com.ibm.ejs.cm.JDBC1PhaseRF.createConnectionFactory
 (JDBC1PhaseRF.java:57)
```

Possible Cause: The database server is not running.

- When the WebSphere Application Server started, the following error was logged to `default_server_stdout.log`:

```
ExceptionUtil X CNTR0020E: Non-application exception occurred
while processing method getDomainList on bean
BeanId(lmz#sysmgmtEjb.jar#SysMgmtEJB, null):
javax.ejb.EJBException: com.Concero.truaccess.sysmgmt.
exceptions.SysMgmtDAOSysException: Unable to get domain list
com.ibm.ejs.cm.portability.TableDoesNotExistException:
[IBM][CLI Driver][DB2/SUN] SQL0204N  "DB2INST2.DOMAIN" is an
undefined name.
```

Possible Cause: There is a JDBC driver configuration problem; the default user ID is invalid. Refer to Installing on WebSphere Servers on page 56 for installation instructions on WebSphere.

- When attempting to sign in to Select Identity (from a web browser), and "HTTP 500 Internal Server Error - The page cannot be displayed" was displayed on the page and the following error was logged to `default_server_stdout.log`:

```
WebGroup X Servlet Error: : java.lang.NullPointerException
at com.Concero.truaccess.control.web.helper.
 ServiceHelper.getPolicyID(ServiceHelper.java:1028)
at com.Concero.truaccess.control.web.
 AccessControlManager.process(AccessControlManager.
 java:183)"
```

Possible Cause: The database server is not running.

- When the WebSphere Application Server started, the following error was logged to `default_server_stdout.log`:

```
Server X WSVR0017E: Error encountered binding J2EE resource,
ConceroMail, as mail\Session
javax.naming.InvalidNameException: Escape character (\) was
followed by a character other than "/" or "\" in the name
```

Possible Cause: The Mail Session configuration may have some problem. Refer to Installing on WebSphere Servers on page 56 for installation instructions on WebSphere.

- When the WebSphere Application Server started, the following error was logged to `default_server_stdout.log`:

  ```
  ExceptionUtil X CNTR0020E: Non-application exception occurred
  while processing method getDomainList on bean
  BeanId(lmz#sysmgmtEjb.jar#SysMgmtEJB, null):
  java.lang.NullPointerException
  at COM.ibm.db2.jdbc.DB2PooledConnection.close
   (DB2PooledConnection.java:235)
  at com.ibm.ejs.cm.portability.PortabilityLayerImpl.
   getInstance(PortabilityLayerImpl.java:1019)
  at com.ibm.ejs.cm.portability.PortabilityLayerFactory.
   getPortabilityLayer(PortabilityLayerFactory.java:62)
  at com.ibm.ejs.cm.JDBC1PhaseRF.createConnectionFactory
   (JDBC1PhaseRF.java:57)
  at com.ibm.ejs.cm.DataSourceImpl$3.run(DataSourceImpl.
   java:219)
  at java.security.AccessController.doPrivileged
   (Native Method)
  ```

  Possible Cause: The default password for the database configuration may be wrong. To reconfigure, refer to Installing on WebSphere Servers on page 56 for installation instructions on WebSphere.

- When the WebSphere Application Server starts, the following error is displayed:

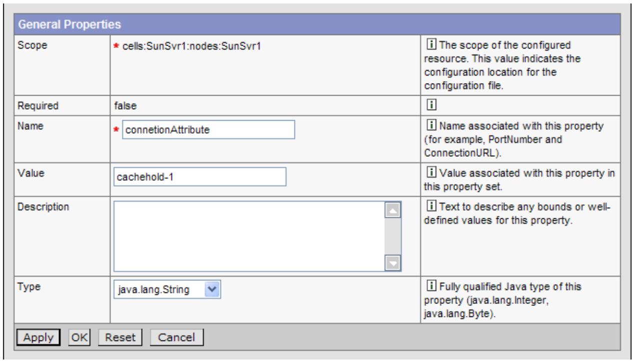
  ```
  [Error] COM.ibm.db2.jdbc.DB2Exception: [IBM][CLI Driver]
  CLI0125E  Function sequence error. SQLSTATE=HY010

  65bdd424 ExceptionUtil X CNTR0020E: Non-application exception
  occurred while processing method getDomainList on bean
  BeanId(lmz#domainEjb.jar#DomainEJB, null):
  javax.ejb.EJBException:
  com.trulogica.truaccess.domain.exceptions.DomainDAOSysException
  : Unable to get domain list COM.ibm.db2.jdbc.DB2Exception:
  [IBM][CLI Driver] CLI0125E  Function sequence error.
  SQLSTATE=HY010
  ```

  Probable Cause: The following property needs to be added under Driver Specific Settings.

  a. Access **WebSphere Administrative Domain → Resources → JDBC Providers → *Db2JdbcDriver* → Data Sources → Data Sources (Version4) → *application_datasource***.

b Click **Custom Properties** in the Driver Specific Settings section.

c Click **New** to add a new property.



General Properties		
Scope	* cells:SunSvr1:nodes:SunSvr1	ⓘ The scope of the configured resource. This value indicates the configuration location for the configuration file.
Required	false	ⓘ
Name	* connetionAttribute	ⓘ Name associated with this property (for example, PortNumber and ConnectionURL).
Value	cachehold-1	ⓘ Value associated with this property in this property set.
Description		ⓘ Text to describe any bounds or well-defined values for this property.
Type	java.lang.String	ⓘ Fully qualified Java type of this property (java.lang.Integer, java.lang.Byte).

d Enter the following values.

Name: **connectionAttribute**

Value: **cachehold=1**

Type: **java.lang.String**

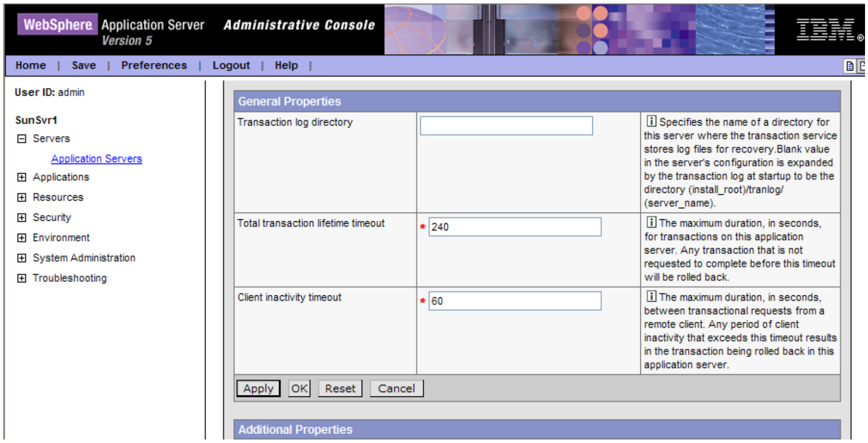
e Click **OK** to save your settings.

- When the WebSphere Application Server starts, the following error is displayed:

```
[4/21/03 16:20:34:465 EDT]  216605 CoordinatorIm I WTRN0066W:
Transaction com.ibm.ejs.jts.jts.CoordinatorImpl@3371145#tid=3
has timed out after 120 seconds.

[4/21/03 16:20:34:577 EDT]  71f38e SystemOut     U [Error]
com.ibm.websphere.ce.cm.StaleConnectionException: class
com.ibm.ejs.cm.proxy.ResultSetProxy is closed

[4/21/03 16:20:34:596 EDT]  71f38e ExceptionUtil X CNTR0020E:
Non-application exception occurred while processing method
getDomainList on bean
BeanId(Concero-TruAccess#domainEjb.jar#DomainEJB, null):
javax.ejb.EJBException:
com.trulogica.truaccess.domain.exceptions.DomainDAOSysException
: Unable to get domain list
com.ibm.websphere.ce.cm.StaleConnectionException: class
com.ibm.ejs.cm.proxy.ResultSetProxy is closed at
com.trulogica.truaccess.domain.ejb.DomainEJB.getDomainList(Doma
inEJB.java:58)
```

Probable Cause: The transaction timeout needs to be increased. Access **WebSphere Administrative Domain** → **Servers** → ***server_name*** → **Application Servers** → ***application_server_name*** → **Transaction Service**.

Increase Total Transaction Lifetime Timeout to **`240`**.



General Properties

Transaction log directory		Specifies the name of a directory for this server where the transaction service stores log files for recovery.Blank value in the server's configuration is expanded by the transaction log at startup to be the directory (install_root)/tranlog/(server_name).
Total transaction lifetime timeout	* 240	The maximum duration, in seconds, for transactions on this application server. Any transaction that is not requested to complete before this timeout will be rolled back.
Client inactivity timeout	* 60	The maximum duration, in seconds, between transactional requests from a remote client. Any period of client inactivity that exceeds this timeout results in the transaction being rolled back in this application server.

B

Logging

HP OpenView Select Identity implements the `java.util.logging.Logger` class, as defined by the Java 2, Standard Edition, v 1.4.1 API Specification. During installation, the `logging.properties` file is copied from the Select Identity CD to a subdirectory on the application server. This file defines how Select Identity logs messages and exceptions, according to the specification.

The following options are available for you to configure. For more detail about each option, refer to the `Logger` class in the API specification.

- **Handlers**

  Defines where messages are logged. You *must* configure the following handlers in `logging.properties`: ConsoleHandler and FileHandler. In addition, the following handlers are available: MemoryHandler and StreamHandler. In the example on page 110, a FileHandler and ConsoleHandler are configured (you must also configure the handler's format, as shown in the following example):

  ```
  # List of global handlers
  handlers = java.util.logging.FileHandler, java.util.logging.ConsoleHandler

  # Properties for the FileHandler
  java.util.logging.FileHandler.limit = 500000
  ...
  ```

- **Message format**

  Defines the format of logged messages based on the handler type. For example:

  ```
  # Properties for the FileHandler
  java.util.logging.FileHandler.pattern = c:/temp/log/java.log
  java.util.logging.FileHandler.limit = 5000000
  java.util.logging.FileHandler.count = 20
  java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
  ```

  Note the **pattern** attribute for FileHandler, which defines the location of the log file. The file location is relative to the user's root directory (the user under which the application server is running). This directory must exist. If it does not, Select Identity will not start.

  For example, if you specify **`log/log.txt`** and the application server is running under the administrative user whose home directory is `/user/admin`, the file is written to the `/user/admin/log/log.txt` file. You can also specify an absolute path, such as `/temp/log/log.txt`.

  Refer to the Logger class in the API specification for a list of format parameters required for each handler type.

- **Log level**

  Defines the logging output. Specify a level using the level entry and set the level to SEVERE (the highest value - no logging output is logged), WARNING, INFO, CONFIG, FINE, FINER, or FINEST (the lowest value - all logging output is logged). You can specify a level for all messages or only those written by a specific component.

  You can selectively modify the logging levels of the different components by specifying different levels for each. For example:

```
com.trulogica.truaccess.util.persistence.PersistenceManager.level=FINEST

com.trulogica.truaccess.util.scheduler.dao.BatchDAOImpl.level=FINE

com.trulogica.truaccess.reconciliation.util.ReconciliationTimerTask.level=
    WARNING

com.trulogica.truaccess.util.SMTPTimerTask.level=WARNING
```

Hibernate provides a log of information when the logging level is set to FINEST. If you do not want the Hibernate log messages, add the following line to the JRE `logging.properties` file:

```
net.sf.hibernate.level=WARNING
```

In the following example, the default logging level is set to WARNING but a log level is also specified for the LDAP connector component (you must also specify a handler for component-specific log levels):

```
# Set the logging level for the root of the namespace.
# This becomes the default logging level for all Loggers.
.level=WARNING

# List of global handlers
...

# Properties for the FileHandler
...

# Default level for ConsoleHandler. This can be used to
# limit the levels that are displayed on the console even
# when the global default has been set to a trace level
java.util.logging.misc.ConsoleHandler.level = FINEST
com.trulogica.truaccess.connector.ldap.ldapv3.LDAPConnector.level = FINE
```

C

Configuring TruAccess.properties

You can configure general settings for the Select Identity server and interface by editing the `TruAccess.properties` file. This file provides settings for the following:

- Email that is sent as part of the provisioning process
- Provisioning retry attempts
- Clustering of servers
- The Select Identity interface URL and logon page
- Authentication and security settings
- Temporary storage needed during uploads to the database
- Auditing
- LDAP
- Account reconciliation with system resources
- Facilitating user searches

➤ Make sure that there are no ^M characters in the `TruAccess.properties` file if Select Identity is running on a UNIX system. These characters are generated when files are copied from a Windows system to a UNIX system.

TruAccess.properties Settings

Each property in the file is described below. Properties that should not be edited are specified.

- `truaccess.email.new.timeinterval=120`

  Specifies the time interval (in seconds) that the email daemon uses to send new email.

- `truaccess.email.retry.timeinterval=900`

  Specifies the time interval (in seconds) that the email daemon uses for sending new email if initial attempts were unsuccessful.

- `truaccess.email.retry.maximum=3`

  Specifies the maximum number of retry attempts for sending email. Setting this to **`0`** causes Select Identity to retry indefinitely.

- `truaccess.email.to.empty=off`

  Specifies whether to send email if the “to” email address cannot be determined. Specify **`on`** if you want to send email to the administrator in this event. Specify **`off`** if you do not want email sent.

- `truaccess.email.userinfochange=off`

  Do not change the value of `truaccess.email.userinfochange`.

-
  ```
  truaccess.email.redirect=off
  truaccess.email.redirect.dir=C:/temp/email
  ```

  Specifies if and where email should be written if a mail server is not available. In general, this is for testing purposes only.

-
  ```
  truaccess.email=on
  truaccess.email.inprogresstimeout=600000
  truaccess.email.batchcount=50
  truaccess.email.authetication=smtp
  ```

  Determines whether Select Identity sends email. If `truaccess.email` is set to **`off`**, no email is sent.

-
  ```
  truaccess.sender.name=Select Identity
  truaccess.sender.email=support@hp.com
  ```

  Specifies a default name and email address to use if the sender’s information cannot be determined.

-
  ```
  truaccess.job.retry.timeinterval=600
  truaccess.job.retry.maximum=3
  ```

  Specifies the time interval (in seconds) that Select Identity will wait between attempts and the maximum number of retries when trying to execute a function, such as deleting a user.

-
  ```
  truaccess.postprovision.retry.timeinterval=5000
  truaccess.postprovision.retry.maximum=20
  ```

  Specifies the time (in milliseconds) to sleep before retrying a post-provisioning attempt (to add an account to the Select Identity database) and number of retry times.

-
  ```
  truaccess.method=http
  truaccess.host=localhost
  truaccess.port=7001
  ```

  Constructs the URL for the Select Identity system within email notifications.

- `truaccess.pageredirect.timeout=10`

  Specifies the time-out (in seconds) for page redirects.

- `truaccess.dataSource=jdbc/TruAccess`

  Specifies the JNDI name of the data source. You should not have to modify this setting.

- `truaccess.mailSession=mail/TruAccess`

  Specifies the JNDI name for the mail session ID. You should not have to modify this setting.

- `truaccess.repository.type=mssql`

  Specifies the Select Identity database type. The default value is **`mssql`** **for** Microsoft SQL Server. If using Oracle, specify **`oracle`**. If using DB2, specify **`db2`**. Values are in lower case.

- `truaccess.repository.oracle.driver.bea=no`

  If you are running Select Identity on WebLogic, connecting to an Oracle database, and using the WebLogic JDBC driver for Oracle (which provides internationalization support), you must add this property to the `TruAccess.properties` file. Set the property to **`yes`**.

-
  ```
  truaccess.authentication=on
  truaccess.sso.token.name=ct_remote_user
  truaccess.loginURL=https://localhost:7001/lmz/control/signin
  truaccess.logoutPage=https://localhost:7001/lmz/control/logout
  ```

  Specifies authentication settings. If `truaccess.authentication` is set to **`on`**, the next three attributes are ignored. If set to **`off`**, you must specify the single sign-on token name, the logon URL, and the logout URL for cleaning up the session.

-
  ```
  truaccess.upload.filedir=c:/temp
  truaccess.upload.maxfilesize=10485760
  ```

  Specifies a temporary directory that the Auto Discovery process uses and the maximum file size (in bytes).

- `truaccess.audit.detail=off`

  Specifies whether to increase the level of detail stored for audit history reports. If set to **`on`**, performance may be affected.

- `truaccess.provisioning.delay=2`

  Specifies the delay (in seconds) for asynchronous provisioning.

-
  ```
  truaccess.userdiscovery.mapping.file=C:/temp/
  AttributeMapping.xml
  ```

  Specifies the location of the attribute mapping file for user discovery.

- `truaccess.resource.record.max=1000`

  Specifies the maximum number of users allowed during reconciliation.

- `truaccess.dateformat=yyyy-MM-dd`

  Specifies the date format.

- `truaccess.timestampformat=yyyy-MM-dd hh:mm:ss a`

  Specifies the timestamp format.

- `truaccess.version=3.3`

  Specifies the version number of Select Identity; do not change this value.

-
  ```
  #truaccess.hibernate.config=/com/trulogica/truaccess/util/
  persistence/mssqlserver.hibernate.cfg.xml
  ```

  Specifies the hibernate property file. DO NOT UNCOMMENT.

-
  ```
  truaccess.fixedtemplate.passwordreset=SI\ Provisioning\ Only
  truaccess.fixedtemplate.terminate=SI\ Provisioning\ Only
  truaccess.fixedtemplate.disable=SI\ Provisioning\ Only
  truaccess.fixedtemplate.enable=SI\ Provisioning\ Only
  truaccess.fixedtemplate.expiration=UserAccountExpirationWF
  truaccess.fixedtemplate.securityviolation=SI\ Email\ Only
  truaccess.fixedtemplate.modifyprofile=SI Provisioning Only
  truaccess.fixedtemplate.passwordexpirenot=SI\ PasswordExpire\
  Email
  truaccess.fixedtemplate.passwordexpire=SI\ Provisioning\ Only
  truaccess.fixedtemplate.disable.terminate=SI\ Provisioning\
  Only
  truaccess.fixedtemplate.reconciliation=ReconciliationDefault
    Process
  truaccess.fixedtemplate.recon_enable=ReconciliationDefault
    Process
  truaccess.fixedtemplate.recon_terminate=ReconciliationDefault
    Process
  truaccess.fixedtemplate.recon_disable=ReconciliationDefault
    Process
  truaccess.fixedtemplate.recon_disable_terminate=ReconciliationD
  efault
    Process
  truaccess.fixedtemplate.bulk_default=SIBulkOneStageApproval
  truaccess.fixedtemplate.bulk_move=SIBulkOneStageApproval
  ```

  Specifies the template for all Select Identity operations.

-
  ```
  truaccess.expirationProcessPeriod=30
  ```

  Specifies when manager notification is sent prior to automatic account expiration (in days). The default is 30 days.

-
  ```
  #truaccess.expire.administrator.userId=sisa
  truaccess.expire.administrator.adminFunc=Concero Sys Admin
  ```

  Specifies the Select Identity system administrator user ID and administrative role.

-
  ```
  truaccess.disable=true
  truaccess.disabledays=1
  #truaccess.system.terminate.administrator.userId=sisa
  #truaccess.system.expire_notification.administrator.userId=sisa
  ```

  Specifies the account disable period before the account is terminated. Set the `truaccess.disable` property to **`true`** if the user needs to be disabled before a terminate.

-
  ```
  personId.attributes=FirstName,LastName
  standardId.attributes=personId,Email
  __managerEmailLookup.attributes=Email
  ```

  Specifies the attributes for external calls.

- `truaccess.policy.id=1`

  Specifies the default Select Identity policy identifier.

-
  ```
  truaccess.recon.rootdir=c:/temp/reconroot
  truaccess.recon.stagingdir=c:/temp/reconstaging
  truaccess.recon.backupdir=c:/temp/reconbackup
  truaccess.recon.filename.timeformat=yyyy_MM_dd_H_mm
  truaccess.recontimer.startdelay=30
  truaccess.recontimer.timeinterval=30
  truaccess.recon.task.check.threshold=3
  truaccess.recon.check_serviceassignment_authadd=false
  ```

  Specifies the attributes for account reconciliation.

- `truaccess.reconcliation.postprovpolicy=1`

  Specifies when Select Identity performs post-provisioning reconciliation. Specify one of the following values:

  1 — if all provisioning activities were successful
  2 — if the corresponding provisioning activity was successful
  3 — always

- `truaccess.bulk.postprovpolicy=2`

  Specifies when Select Identity performs post-provisioning after a bulk upload. Specify one of the following values:

  1 — if all provisioning activities were successful
  2 — if the corresponding provisioning activity was successful
  3 — always

-
  ```
  truaccess.batch.inprogresstimeout=18000000
  truaccess.batch.ownerkey=0
  ```

  Specifies the time-out and owner for batch processing for the Auto Discovery facility. To specify common batch processing, set `truaccess.batch.ownerkey` to **0**, or you can specify a specific application server.

-
  ```
  truaccess.batch.pickuppolicy=1
  truaccess.batch.reportdir=c:/temp/reports
  ```

  Specifies the policy to pick up the batch files for the Auto Discovery facility and the directory to which reports are written. Specify one of the following values for `truaccess.batch.pickuppolicy`:

  1 — Common batch only (`truaccess.batch.ownerkey` property is set to **0**)
  2 — own batch only (`truaccess.batch.ownerkey` must specify an application server)
  3 — common and own batch

-
  ```
  si.serviceassign.evaluation=1
  ```

  Specifies whether to evaluate user attributes or service assignments. Specify one of the following values:

  0— Evaluate all (attributes and service assignments)
  1— Skip services previously assigned to users
  2— Skip users that have at least one service assignment

-
  ```
  truaccess.singlevalue.attribute.delete=false
  ```

  Specifies whether a user's single value attributes should be deleted.

-
  ```
  #com.hp.si.webservice.auth.resource=ldap
  #com.hp.si.webservice.auth.ldap.accessurl=ldap://localhost:389
  #com.hp.si.webservice.auth.ldap.uidattr=uid
  #com.hp.si.webservice.auth.ldap.suffix=ou=People,dc=trulogica,
    dc=com
  #com.hp.si.webservice.auth.ldap.needssl=false
  ```

  Specifies external authentication for Web Service requests.

-
  ```
  com.hp.si.user.attributes.dropdown.constraint.count=50
  ```

  This property determines if a drop-down list is displayed or a search is used when a user selects an attribute which contains a constraint list. If the number of constraint values for the attribute is below the property value (such as 50 in the example), a drop-down will appear on the registration or approval form. If the number of contraint values is equal to or greater than the property value, a Search will be required for selecting values from the list.

-
  ```
  truaccess.homepage=http://www.hp.com
  #com.hp.si.clientName=HP
  ```

  Specifies your home page and name.

- `#com.hp.si.clientHeadTag=/ClientPages/SI/headTags.jsp`
  `#com.hp.si.clientSecureHeadTag=/ClientPages/SI/secureHeadTags.jsp`

  Provides the JSP paths for company-specific headers. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

- `#com.hp.si.clientHeader=/ClientPages/SI/header.jsp`
  `#com.hp.si.clientSecureHeader=/ClientPages/SI/secureHeader.jsp`

  Provides the JSP paths for company-specific headers. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

- `#com.hp.si.clientFooter=`

  Provides the JSP paths for company-specific footers. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

- `truaccess.sqlQueryInListSize=200`

  Specifies the maximum number of positional parameters to be used in a SQL query “in” list or array.

- `truaccess.batchQuerySize=500`

  Specifies the maximum number of queries to be executed in a single batch insert or update statement.

- `truaccess.generatedFileSizeLimit=2000000`

  Indicates the size of the files (in bytes) that are generated by the reporting subsystem. This is a soft limit; the actual file size may exceed this by a small amount.

- `truaccess.userdetailconfigrpt.sortattributes=UserName, FirstName,LastName,Email,Company,Department,CostCenter`

  Indicates the column on which sorting should take place in the user detail configuration report.

- `si.rsa.provider=org.bouncycastle.jce.provider.BouncyCastle Provider`

  Specifies the provider of the key store parameters. *Do not modify this setting.*

- `truaccess.AZN.schema.owner=db2inst1.`

  Specifies the schema owner for AZN DB Stored Procedures. This value should end with a period (.).

- `truaccess.NEWCO.schema.owner=db2inst1.`

  Specifies the schema owner for NEWCO DB Stored Procedures. This value should end with a period (.).

- `com.hp.si.selfreg.schedule = true`

  Specifies whether the “schedule field” in the self-registration form will be visible.

- `contact_helpdesk=Please contact the helpdesk at 972-309-6408`

  Indicates an error message that is displayed if the user cannot log on to the Select Identity client.

-
  ```
  #hp.si.logo.file=http://welcome.hp-ww.com/img/
  hpweb_1-2_topnav_hp_logo.gif
  ```

  Specifies the relative or fully qualified path name for your logo file. See Customizing TruAccess.properties on page 85 for more details about specifying the logo.

-
  ```
  com.hp.si.usersearch.criteria.names.default =
  UserName,Email,FirstName,LastName
  ```

  Specifies the user search criteria names. The names are separated by commas. Use “_Status” to search for the user state status.

-
  ```
  com.hp.si.usersearch.result.columns =
  UserName,FirstName,LastName,Email
  ```

  Specifies the order in which the attribute columns will display in the search results page. The names are separated by commas. The UserName is required.

- `com.hp.si.usersearch.result.max = 300`

  Specifies the maximum number of users that can display in a user search.

-
  ```
  com.hp.si.selfreg.instruct = Welcome and thank you for accessing
  Self-Registration. After completing this page, press ''{0}''.
  You will then be asked for additional information. Once you have
  completed all pages, your request will be submitted for
  processing.
  ```

  Self-registration instructions.

- `com.hp.ovsi.connector.schema.dir=C:/SI33/schema`

  This is the folder where the connector schema files are extracted and this needs to be in the CLASSPATH of the application server. This is used by connectors that have Attribute Mapper integration.

Clustering Environments

Select Identity can leverage the clustering capabilities of the application servers to support high throughput and fault tolerance. Multiple copies of Select Identity can also be installed and can work together when they are connected to the same back-end database.

The following configuration is recommended for heavy use:

- Multiple application servers in a cluster to handle all user provisioning tasks
- One or more reconciliation servers to handle reconciliation tasks
- One report server for generating reports

To partition the tasks, you must modify the `TruAccess.properties` file for different servers:

- For the servers in a cluster, set the following properties:

  ```
  truaccess.batch.ownerkey=0
  ```

  ```
  truaccess.batch.pickuppolicy=1
  ```

- For a reconciliation or report servers, set the following properties:

  ```
  truaccess.batch.ownerkey=100
  ```

  ```
  truaccess.batch.pickuppolicy=2
  ```

  Also, use a unique owner key for each reconciliation or report server.

This will ensure that the appropriate servers pick up the appropriate tasks.

It is also possible to have a collection of reconciliation servers as a server farm. In that case, they must share a common file system and batch owner (as specified by `truaccess.batch.ownerkey`).

All connectors need to be installed on all resource servers.

Attribute Mapping for Search Efficiency

User accounts can consist of many attributes. Typically, users are searched based on certain key attributes (email, SSN, employee ID). Certain user profile attributes can be added to the `TruAccess.properties` file and used to expedite search functions. If these attributes are set, the `TAUser` database table must be extended by adding extra columns that reflect these values. The extra attributes must then be mapped to those columns.

➤ Having a key search attribute in place will expedite Auto Discovery and Reconciliation functions. See the *HP OpenView Select Identity Administrator Guide* for details.

To specify certain attributes on which you want to search, you can perform the following:

- Identify the key attributes, such as SSN, employeeId, or email. Make sure that these are defined within Select Identity and within the mapping file used for each system resource in which data is stored.
- Add corresponding columns to the `TAUser` table in the database.
- Add entries in the `TruAccess.properties` file.

For example, you may want to use the SSN and employeeId attributes to simplify searches. Perform the following:

1 Add the following columns to the `TAUser` table and create the corresponding indices:

   a Add to the TAUser table:

   ```
   SSN VARCHAR(11) default 'XXX-XX-XXXX';
   EMPID VARCHAR(20) default 'XXXXXXXXXX';
   ```

   b Create the following indices:

   ```
   TAUSER_SSNIDX on TAUser(SSN);
   TAUSER_EMPIDIDX on TAUser(EMPID);
   ```

2 Update the following properties in the `TruAccess.properties` file:

   ```
   truaccess.user.extra=SSN,EmpId
   truaccess.user.extra.SSN.column=SSN
   truaccess.user.extra.EmpId.column=EMPID
   ```

If there is no corresponding column mapping (`truaccess.user.extra.<Attribute Name>.column=<Column Name>`), the attribute name is assumed to be the column name.

glossary

A

Access Control List (ACL)

An abstraction that organizes entitlements and controls authorization. An ACL is list of entitlements and users that is associated with a secured object, such as a file, an operation, or an application. In an ACL-based security system, protected objects carry their protection settings in the form of an ACL.

Access Management

The process of authentication and authorization.

Action

An action represents a task that can be performed within each Select Identity capability.

See also: ***capability***

Admin Role

A template that defines the administrative actions that can be performed by a user. An Administrative Service is created to provide access to roles. Users are then given access to the Service. Users with administrative roles can also grant their set of roles to another administrator within their Service context.

Approval Process

The process of approving the association, modification, or revocation of entitlements for an identity. This process is automated of these through workflow templates.

Approver

A Select Identity administrator who has been given approval actions through an Admin Role.

Attribute

An attribute is an individual field that helps define an identity profile. For each identity, an attribute has a corresponding value. For example, an attribute could be “department” with possible values of “IT,” “sales,” or “support.”

Audit Report

A report that provides regular account interaction information within the Select Identity system.

Authentication

Verification of an identiy’s credentials.

Authoritative Source

A resource that has been designated as the “authority” for identity information. Select Identity accounts can be reconciled against accounts in an authoritative source.

Authorization

Real-time enforcement of an identity’s entitlements. Authentication is a prerequisite for authorization.

Auto Discovery

The process of adding user accounts to the Select Identity system for a specified Service through the use of a data file.

B

Business Service

A business service is a product or facility offered by, or a core process used by, a business in support of its day-to-day operations. Example business services could include an online banking service, the customer support process, and IT infrastructure services such as email, calendaring, and network access.

See also: ***service***

C

Capability

Actions that can be performed within the Select Identity client are grouped by capability, or link, in the interface.

See also: ***action***

Challenge and Response

A method of supplying alternate authentication credentials, typically used when a password is forgotten. Select Identity challenges the end user with a question and the user must provide a correct response. If the user answers the question correctly, Select Identity resets the password to a random value and sends email to the user. The challenge question can be configured by the administrator. The valid response is stored for each user with the user’s profile and can be updated by an authenticated user through the Self Service pages.

Configurations

The Configurations capability enables you to import and export Select Identity settings and configurations. This is useful when moving from a test to a production environment.

Configuration Reports

Configuration reports provide current system information for user, administrator, and Service management activities.

Connector

A J2EE connector that communicates with the system resources that contain your identity profile information.

Context

A Select Identity concept that defines a logical grouping of users that can access a Service.

Contextual Identity Management (CIM)

An organizational model that introduces new abstractions that simplify and provide scale to the business processes associated with identity management. These abstractions are modeled after elements that exist in businesses today and include Select Identity Services and Service Roles.

Credentials

A mechanism or device used to verify the authenticity of an identity. For example, a user ID and password, biometrics, and digital certificates are considered credentials.

D

Data File

An SPML file that enables you to define user accounts to be added to Select Identity through Auto Discovery or Reconciliation.

Delegated Administration

The ability to securely assign a subset of administrative roles to one or more users for administrative management and distribution of workload. Select Identity enables role delegation through the Self Service pages from one administrator to another user within the same Service context.

Delegated Registration

Registration performed by an administrator on behalf of an end user.

E

End User

A role associated to every user in the Select Identity system that enables access to the Self Service pages.

Entitlement

An abstraction of the resource privileges granted to an identity. Entitlements are resource-specific and can be resource account IDs, resource role memberships, resource group memberships, and resource access rights and privileges. Entitlements are also considered privileges, permissions, or access rights.

External Call

A programatic call to a third-party application or system for the purpose of validating accounts or constraining attribute values.

F

Form

An electronic document used to capture information from end users. Forms are used by Select Identity in many business processes for information capture and system operation.

I

Identity

The set of authentication credentials, profile information, and entitlements for a single user or system entity. Identity is often used as a synonym for “user,” although an identity can represent a system and not necessarily a person.

Identity Management

The set of processes and technologies involved in creating, modifying, deleting, organizing, and auditing identities.

M

Management

The ongoing maintenance of an object or set of objects, including creating, modifying, deleting, organizing, auditing, and reporting.

N

Notifications

The capability that enables you to create and manage templates that define the messages that are sent when a system event occurs.

P

Password Reset

The ability to set a password to a system-generated value. Select Identity uses a challenge and response method to authenticate the user and then allow the user to reset or change a password.

Policy

A set of regulations set by an organization to assist in managing some aspect of its business. For example, policy may determine the type of internal and external information resources that employees can access.

Process

A repeatable procedure used to perform a set of tasks or achieve some objective. Whether manual or automated, all processes require input and generate output. A process can be as simple as a single task or as complicated a multi-step, conditional procedure.

See also: ***approval process***

Profile

Descriptive attributes associated with an identity, such as name, address, title, company, or cost center.

Provisioning

The process of assigning authentication credentials to identities.

R

Reconciliation

The process by which Select Identity accounts are synchronized with a system resource. Accounts can be added to the Select Identity system through the use of an SPML data file.

Registration

The process of requesting access to one or more resources. Registration is generally performed by an end user seeking resource access, or by an administrator registering a user on a user's behalf.

See also: ***delegated registration***, ***self registration***

Request

An event within the Select Identity system for the addition, modification, or removal of a user account. Requests are monitored through the Request Status capability.

Resource

Any single application or information repository. Resources typically include applications, directories, and databases that store identity information.

Role

A simple abstraction that associates entitlements with identities. A role is an aggregation of entitlements and users, typically organized by job function.

See also: ***administrative role***

Rule

A programmatic control over system behavior. Rules in Select Identity are typically used for programmatic assignment of Services. Rules can also be used to detect changes in system resources.

S

Self Registration

Registration performed by an end user seeking access to one or more resources.

Self Service

The ability to securely allow end-users to manage aspects of a system on their own behalf. Select Identity provides the following self-service capabilities: registration, profile management, and password management (including password change, reset, and synchronization).

Service

A business-centric abstraction representing resources, entitlements, and other identity-related entities. Services represent the products and services that you offer to customers and partners.

Service Attribute

A set of attributes and values that are available for or required by a Service. Attributes are created and managed through the Attributes pages.

See also: ***Attributes***

Service Role

A Select Identity abstraction that defines how a logical grouping of users will access a Select Identity Service. The Select Identity Service is a superset of all the identity management elements of a business service.

Service View

A restricted view of a Service that is valid for a group of users. Views enable you to define a subset of Service registration fields, change field names, reorder fields, and mask field values for specific users.

Single Sign-On (SSO)

A session/authentication process that permits a user to enter one set of credentials (name and password) in order to access multiple applications. A Web SSO is a specialized SSO system for web applications.

SPML Data File

A file that is used to add and provision accounts within Select Identity.

See also: ***Data File***

U

Users

The Select Identity capability that provides consistent account creation and management across Services.

W

Workflow

The tasks, procedural steps, organizations or people involved, and required input and output information needed for each step in a business process. In identity management, the most common workflows are for provisioning and approval processes.

Workflow Engine

A system component that executes workflows and advances them through their flow steps.

Workflow Studio

The Select Identity capability that enables you to create and manage workflow templates.

index