Informatica®

Informatica® Multidomain MDM
10.4 HotFix 2

Installation Guide for IBM Db2 with WebSphere

Informatica Multidomain MDM Installation Guide for IBM Db2 with WebSphere
10.4 HotFix 2
December 2020

Publication Date: 2023-02-12

Table of Contents

Preface

Follow the instructions in the Informatica® *Multidomain MDM Installation Guide* to install and set up Multidomain MDM in the supported database and application server environment of your choice. In addition to the installation tasks, the guide includes pre-installation and post-installation tasks.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at http://velocity.informatica.com. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through the Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
https://www.informatica.com/services-and-training/customer-success-services/contact-us.html.

To find online support resources on the Informatica Network, visit https://network.informatica.com and select the eSupport option.

CHAPTER 1

Installation Overview

This chapter includes the following topics:

Multidomain MDM Installation

Multidomain MDM is a master data management solution that enhances data reliability and data maintenance procedures. Multidomain MDM is also referred to as the MDM Hub. You can access the MDM Hub features through the Hub Console.

The MDM Hub consists of multiple components. You can install the MDM Hub in the graphical mode, the console mode, or the silent mode.

Core Components

The following table describes the core installation components:

Component	Description
MDM Hub Master Database	A schema that stores and consolidates business data for the MDM Hub. Contains the MDM Hub environment configuration settings, such as user accounts, security configuration, Operational Reference Store registry, and message queue settings. You can access and manage an Operational Reference Store from an MDM Hub Master Database. The default name of an MDM Hub Master Database is CMX_SYSTEM, but you can use a custom name. You can create multiple MDM Hub Master Databases, each with its own set of Operational Reference Stores, in the same database instance.
Operational Reference Store	A schema that stores and consolidates business data for the MDM Hub. Contains the master data, content metadata, and the rules to process and manage the master data. You can configure separate Operational Reference Store databases for different geographies, different organizational departments, and for the development and production environments. You can distribute Operational Reference Store databases across multiple server machines. The default name of an Operational Reference Store is CMX_ORS.
Hub Server	A J2EE application that you deploy on an application server. The Hub Server processes data stored within the MDM Hub and integrates the MDM Hub with external applications. The Hub Server manages core and common services for the MDM Hub.

Component	Description
Process Server	A J2EE application that you deploy on an application server. The Process Server processes batch jobs such as load, recalculate BVT, and revalidate, and performs data cleansing and match operations. The Process Server interfaces with the cleanse engine that you configure to standardize and optimize data for match and consolidation.
Provisioning tool	A tool to build business entity models, and to configure the Entity 360 framework for Data Director. After you build business entity models, you can publish the configuration to the MDM Hub.
Informatica ActiveVOS ®	A business process management (BPM) tool that is required internally by the MDM Hub for processing data. Informatica ActiveVOS supports automated business processes, including change-approval processes for data. You can also use Informatica ActiveVOS to ensure that changes to master data undergo a review-and-approval process before inclusion in the best version of the truth (BVT) records. When you install ActiveVOS Server as part of the Hub Server installation, you install the ActiveVOS Server, ActiveVOS Console, and Process Central. Also, you install predefined MDM workflows, tasks, and roles.
Data Director (IDD)	A user interface to master and manage the data that is stored in the MDM Hub. In IDD, data is organized by business entities, such as customers, suppliers, and employees. Business entities are data groups that have significance for organizations.

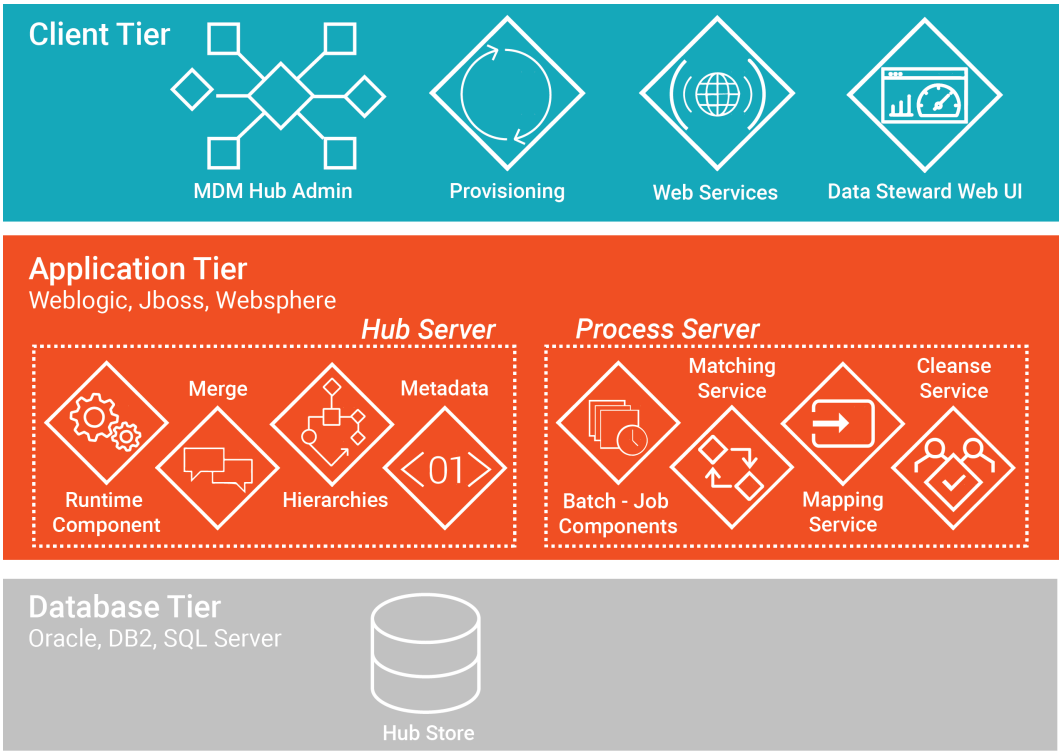
Optional Components

The following table describes the optional installation components:

Component	Description
Resource Kit	Set of samples, applications, and utilities to integrate the MDM Hub into your applications and workflows. You can select the Resource Kit components that you want to install.
Dynamic Data Masking	A data security tool that operates between the MDM Hub and databases to prevent unauthorized access to sensitive information. Dynamic Data Masking intercepts requests sent to databases and applies data masking rules to the request to mask the data before it is sent back to the MDM Hub.
Informatica Data Controls (IDC)	Applicable to Informatica Data Director (IDD) based on the subject area data model only. IDC is a set of user interface controls that expose the MDM Hub data in third-party applications that are used by business users.
Zero Downtime (ZDT) module	A module to ensure that applications have access to data in the MDM Hub during the MDM Hub upgrade. In a ZDT environment, you duplicate the databases: source databases and target databases. During the MDM Hub upgrade, the ZDT module replicates the data changes in the source databases to the target databases. To buy the ZDT module, contact your Informatica sales representative. For information about installing a zero downtime environment, see the *Multidomain MDM Zero Downtime Installation Guide* for the database.

Multidomain MDM architecture

The following image depicts Multidomain MDM architecture:



Installation Topology

Before you install the MDM Hub, decide on the installation topology. Usually, infrastructure planners and Master Data Management solution architects determine on the topology to implement.

You can install the MDM Hub in multiple environments, such as development, test, and production. The requirements and priorities for each type of environment are unique. Therefore, the installation topology differs for each environment.

The following table describes the MDM Hub installation topologies that you can use:

Topology	Description
Standalone application server instance	All the MDM Hub components are installed on a standalone application server instance.
Multiple application server instances	The MDM Hub components are installed on multiple application server instances.
Application server cluster	The MDM Hub components are installed in an application server cluster.

For more information about installation topologies, see the *Multidomain MDM Infrastructure Planning Guide*.

Note: All the components of the MDM Hub implementation must have the same version. If you have multiple versions of the MDM Hub, install each version in a separate environment.

Installation Tasks

Complete the preinstallation tasks before you install the MDM Hub components. After the installation, complete the post-installation tasks.

To install the MDM Hub, perform the following tasks:

1. Complete the preinstallation tasks. To ensure that you can successfully run the installers for the Hub Server and the Process Server and create the Hub Store, complete the pre-installation tasks.
2. Create the MDM Hub Master Database. Create the MDM Hub Master Database before you install the Hub Server and the Process Server.

   Use the setup script provided with the MDM Hub distribution to create the MDM Hub Master Database.
3. Create the Operational Reference Store. Create Operational Reference Stores at any time after you complete the preinstallation tasks.

   Use the setup script provided with the MDM Hub distribution to create the Operational Reference Store.
4. Install the Hub Server. Use the MDM Hub installer to install the Hub Server.
5. Install the Process Server. Use the MDM Hub installer to install the Process Server.
6. Perform the post-installation configuration tasks. Test the database connections. To ensure that you can use the MDM Hub features, configure the Hub Server and the Process Server.
7. Optionally, install the Resource Kit. Use the MDM Hub installer to install the Resource Kit.

CHAPTER 2

Pre-Installation Tasks

This chapter includes the following topics:

Prepare for Installation

Before you install the MDM Hub, prepare for the installation.

The following table describes the preparatory tasks for the installation:

Task	Description
Read the Release Notes	Read the latest *Multidomain MDM Release Notes* for updates to the installation and upgrade process. **Important:** Some versions of application servers and databases have known limitations when running Multidomain MDM. Ensure that you perform all suggested workarounds.
Read the Product Availability Matrix	Read the Product Availability Matrix (PAM) for information about product requirements and supported platforms. You can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.
Understand the MDM infrastructure and architecture plan	Acquire and understand the plan for the MDM infrastructure and architecture from the infrastructure planners or the MDM solution architects in your organization. For more information about infrastructure planning and MDM architecture, see the *Multidomain MDM Infrastructure Planning Guide*.
Download and extract the installer files	Download the installation files from the Informatica Electronic Software Download site to a directory on your machine. To extract the compressed files, use an extraction utility that also extracts empty folders. Download and extract the following installation files: - MDM Hub installer for the operating system - Database files - ActiveVOS Server installer for the operating system

Task	Description
Verify license key	Verify that you have the license key, which is in an email message from Informatica. Copy the license key file to a directory accessible to the user account that installs the product. If you do not have a license key, contact Informatica Global Customer Support.
Create an installation documentation directory	Create a directory to store copies of installation files, such as the validation results, environment reports, database debug logs, and log files. For example, create the directory install_doc. To troubleshoot the installation, you can create an archive file of the installation documentation directory and send it to Informatica Global Customer Support for analysis.

Prepare the Environment

Before you install the MDM Hub, prepare the installation environment.

The following table describes the tasks you perform to prepare the environment for the installation:

Task	Description
Verify minimum system requirements	Verify that the machines meet the hardware and software requirements for the MDM Hub installation. The hardware requirements are dependent on the data, processing volumes, and business rules. To install the MDM Hub, the machines must meet the following minimum requirements: - Disk space. 4.9 GB - RAM for the development environment. 4 GB To verify the run-time physical memory requirements of the MDM Hub components, use the following formula: `Total run-time memory requirement for MDM Hub components = JDK JVM max heap size of the application server + operating system native heap size`
Install Java Development Kit (JDK)	Install a supported version of the JDK on the machine on which you want to install the MDM Hub. The JDK is not bundled with the MDM Hub installers. **Note:** Use the same Java version on the application server machines and on the machines on which you want to launch the Hub Console.
Install Visual C++ Redistributable for Visual Studio 2015 on Windows only	On Windows systems, Multidomain MDM requires Visual C++ Redistributable for Visual Studio 2015 to support the name search feature and the matching feature.
Set environment variables	Set the environment variables for the MDM Hub installation. To use the correct JDK, set the following environment variables to point to the JDK directory: - JAVA_HOME. Required - PATH. Required
Set the operating system locale	Set the same operating system locale for the Hub Server, the MDM Hub Master Database, Operational Reference Store, and the Hub Console.

Task	Description
Set up the X Window System on UNIX	If you want to run the installer in graphical mode on UNIX, set up an X Window System. An X Window System is a graphics display server. For more information about setting up an X Window System, see the documentation for your operating system.
Disable access to the root welcome page for your application server	To improve security, disable access to the root welcome page for your application server. For instructions, see the documentation for your application server.
Disable insecure TLS cipher suites	To improve security, in the Java runtime environment that is used with Multidomain MDM, disable insecure TLS cipher suites. 1. Open the following file: `../jdk<version>/jre/lib//security/java.security` 2. Find the property `jdk.tls.disabledAlgorithms` and update the value to include the following list of insecure cipher suites: `jdk.tls.disabledAlgorithms = SSLv3, RC4, MD5withRSA, DH keySize < 1024, EC keySize < 224, DES40_CBC, RC4_40, 3DES_EDE_CBC, EDH-RSA-DES-CBC3-SHA, ECDHE-RSA-DES-CBC3-SHA, DES-CBC3-SHA` For more information about the property, see the documentation for your JDK.

For more information about product requirements and supported platforms, see the Product Availability Matrix on Informatica Network:
https://network.informatica.com/community/informatica-network/product-availability-matrices

Configure JDBC Drivers

Download and copy the correct version of the JDBC driver file to the WebServer `lib` directory.

1. Download the JDBC driver from the website of the database server vendor.
2. Copy the JDBC driver file to the following MDM Hub distribution directory:

   `<WebSphere installation directory>/AppServer/lib`

Set Up the Database Environment

Before you create an MDM Hub Master Database and Operational Reference Store, set up the database environment.

To set up the database environment, perform the following tasks:

1. Install and configure IBM Db2.
2. Create a database and tablespaces.
3. Bind packages on the database server.
4. Create the ActiveVOS® schema.

Step 1. Install and Configure IBM Db2

You can install and configure IBM Db2 according to the instructions in the IBM Db2 documentation.

The following table describes the IBM Db2 installation and configuration tasks that you must perform on each machine on which you want an IBM Db2 instance:

Tasks	Description
Install IBM Db2	Install the supported version of IBM Db2.
Install clients and utilities	Install the IBM Db2 client and utilities software to communicate with the MDM Hub and run the MDM Hub processes. On each machine where you want to run the Hub Server or Process Server, install the following software: - Db2 client - Db2 Java utilities for the Db2 client Ensure that you catalog the IBM Db2 database from each Db2 client.
Configure IBM Db2 drivers	To configure IBM Db2 drivers, copy the `db2jcc.jar` and `db2jcc_license_cu.jar` driver files from the source to the target directory: Source: `<IBM Db2 installation directory>/java` Target: `<MDM Hub distribution directory>/database/lib`
Create the MDM Hub schema users	Create users to access the following MDM Hub schemas: - MDM Hub Master Databases - Operational Reference Stores

Step 2. Create a Database and Tablespaces

After you install and configure IBM Db2, create and configure databases and tablespaces. You must create a database for each database instance.

Note: If you want to create multiple MDM Hub Master Databases, create unique tablespaces for each MDM Hub Master Database.

The following table describes the tablespaces that you require for the MDM Hub schemas:

Tablespace Name	Description
CMX_DATA	Default tablespace for the Operational Reference Store schema. Contains the metadata and user data of the MDM Hub.
CMX_INDX	Tablespace to contain indexes that the MDM Hub creates and uses.
CMX_TEMP	Tablespace to contain temporary tables that the MDM Hub creates and uses.
CMX_REPOS	Tablespace to contain the Operational Reference Store objects.
CMX_USER_TEMP	Temporary tablespace to contain operational temporary tables.
CMX_SYS_TEMP	Temporary tablespace for SQL operations.

Use one of the following procedures to create a database and tablespaces:

- Manually create the database and tablespaces

- Use a script to create the database and tablespaces

Manually Create a Database and Tablespaces

You can manually create a database and tablespaces. Ensure that you create the database with the compatibility vector turned on and with the `UTF-8 TERRITORY US` locale.

Set the Db2 Environment and Db2 Registry Variables

If you create the database manually, set the Db2 environment and Db2 registry variables that the MDM Hub requires.

Use the following commands to set the Db2 environment and Db2 registry variables:

```
db2set DB2CODEPAGE=1208
db2set DB2_COMPATIBILITY_VECTOR=
db2set DB2_DEFERRED_PREPARE_SEMANTICS=YES
db2set DB2_RESTORE_GRANT_ADMIN_AUTHORITIES=ON
db2set DB2_HASH_JOIN=YES
db2set DB2_ANTIJOIN=YES
db2set DB2_INLIST_TO_NLJN=NO
db2set DB2_SELECTIVITY=ALL
db2set DB2_SKIPINSERTED=YES
db2set DB2_SKIPDELETED=YES
db2set DB2_EXTENDED_OPTIMIZATION=ON,ENHANCED_MULTIPLE_DISTINCT,IXOR,SNHD
db2set DB2NTNOCACHE=ON
db2set DB2_REDUCED_OPTIMIZATION=REDUCE_LOCKING
```

Set the Database Manager Configuration for the Database Instance

You need to optimize the database manager configuration for the database instance.

Use the following commands to optimize the database manager configuration:

```
db2 update dbm cfg using MON_HEAP_SZ AUTOMATIC
db2 update dbm cfg using JAVA_HEAP_SZ 2048
db2 update dbm cfg using AGENT_STACK_SZ 256
db2 update dbm cfg using SHEAPTHRES 0
db2 update dbm cfg using INTRA_PARALLEL YES
```

Note: The values specified in the commands are minimum requirements for the MDM Hub.

Set Database Configuration Parameters

Set the configuration parameters for the database.

Use the following commands to set the database configuration parameters:

```
db2 update db cfg using LOCKLIST AUTOMATIC
db2 update db cfg using MAXLOCKS AUTOMATIC
db2 update db cfg using PCKCACHESZ 128000
db2 update db cfg using DBHEAP AUTOMATIC
db2 update db cfg using CATALOGCACHE_SZ 25000
db2 update db cfg using LOGBUFSZ 4096
db2 update db cfg using UTIL_HEAP_SZ 50000
db2 update db cfg using BUFFPAGE 250
db2 update db cfg using STMTHEAP AUTOMATIC
db2 update db cfg using APPLHEAPSZ AUTOMATIC
db2 update db cfg using APPL_MEMORY AUTOMATIC
db2 update db cfg using STAT_HEAP_SZ AUTOMATIC
db2 update db cfg using LOGFILSIZ 128000
db2 update db cfg using LOGPRIMARY 10
db2 update db cfg using LOGSECOND 200
db2 update db cfg using auto_reval deferred_force
```

```
db2 update db cfg using decflt_rounding round_half_up
db2 update db cfg using SHEAPTHRES_SHR AUTOMATIC
db2 update db cfg using DFT_DEGREE 1
```

Note: The values specified in the commands are minimum requirements for the MDM Hub.

Grant Privileges to SYSIBMADM Modules

You must grant privileges to UTL_DIR, UTL_FILE, and DBMS_SQL SYSIBMADM modules.

Use the following commands to grant privileges to modules:

```
GRANT EXECUTE ON MODULE SYSIBMADM.UTL_DIR TO PUBLIC WITH GRANT OPTION
GRANT EXECUTE ON MODULE SYSIBMADM.UTL_FILE TO PUBLIC WITH GRANT OPTION
GRANT EXECUTE ON MODULE SYSIBMADM.DBMS_SQL TO PUBLIC WITH GRANT OPTION
```

Define Buffer Pools for the Database Manager

Define the REPOS_POOL and CMX_POOL buffer pools.

Use the following commands to define buffer pools:

```
CREATE BUFFERPOOL REPOS_POOL IMMEDIATE SIZE 1500 PAGESIZE 32 K
CREATE BUFFERPOOL CMX_POOL IMMEDIATE SIZE 3000 PAGESIZE 32 K
```

Create Tablespaces

You need to create tablespaces that the MDM Hub schemas require.

Create the tablespaces in the following sequence:

1. CMX_DATA
2. CMX_INDX
3. CMX_REPOS
4. CMX_TEMP
5. CMX_USER_TEMP
6. CMX_SYS_TEMP

Use the following statements to create tablespaces for the MDM Hub schemas:

```
CREATE TABLESPACE CMX_DATA PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_DATA\cmx_data01.dat' 500
M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
    PREFETCHSIZE 16
    BUFFERPOOL CMX_POOL

CREATE TABLESPACE CMX_INDX PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_INDX\cmx_indx01.dat' 500
M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
    PREFETCHSIZE 16
    BUFFERPOOL CMX_POOL

CREATE TABLESPACE CMX_REPOS PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_REPOS\cmx_repos01.dat' 500
M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
```

```
    PREFETCHSIZE 16
    BUFFERPOOL REPOS_POOL

CREATE TABLESPACE CMX_TEMP PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_TEMP\cmx_temp01.dat' 500
M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
    PREFETCHSIZE 16
    BUFFERPOOL CMX_POOL

CREATE USER TEMPORARY TABLESPACE CMX_USER_TEMP PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\USER_TEMP\cmx_user_temp01.dat'
500 M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
    PREFETCHSIZE 16
    BUFFERPOOL CMX_POOL

CREATE SYSTEM TEMPORARY TABLESPACE CMX_SYS_TEMP PAGESIZE 32 K
    MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\SYSTEM_TEMP\cmx_sys_temp01.dat'
500 M )
    EXTENTSIZE 16
    AUTORESIZE YES
    OVERHEAD 10.5
    PREFETCHSIZE 16
    BUFFERPOOL CMX_POOL
```

Optionally, to create tablespaces with the dropped table recovery feature enabled, add the following clause to the `CREATE TABLESPACE` statement:

```
DROPPED TABLE RECOVERY ON
```

Use a Script to Create a Database and Tablespaces

The MDM Hub distribution includes a script to create the database and associated tablespaces. To run the script, you need administrative privileges with write and execute permissions to the Db2 data directory.

On UNIX, before you create the database, update the `db2.storage.path` property in the `database.properties` file with the correct database storage path. The `database.properties` file is in the following directory:

```
<MDM Hub distribution directory>/database/bin/db2
```

1. Open a command prompt, and change to the following directory:

   ```
   <MDM Hub distribution directory>/database/bin
   ```

2. To create the database, run the following command:

   On UNIX. `./sip_ant.sh create_db`

   On Windows. `sip_ant.bat create_db`

3. Answer the prompts described in the following table:

Prompt	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify `DB2`.
Enter the database instance name [db2]	Name of the database instance. Default is `db2`.
Enter the database name [SIP97]	Name of the database. Default is `SIP97`.
Enter the database storage path [C:\DB2DATA]	Path to the directory where the database must be stored. Default is `C:\DB2DATA`. **Note:** On UNIX, accept the default value. The database storage path that you specify in the `database.properties` file will be used.
Enter the DBA user name [DB2ADMIN]	User name of the administrative user. Default is `DB2ADMIN`.
Enter the DBA password	Password of the administrative user.

The script creates the database and the following tablespaces:

- CMX_DATA
- CMX_INDX
- CMX_TEMP
- CMX_REPOS
- CMX_USER_TEMP
- CMX_SYS_TEMP

To verify that the database was created successfully, review the `sip_ant.log` file in the `<MDM Hub distribution directory>/database/bin` directory.

Step 3. Bind Packages on the Database Server

To ensure that the IBM Db2 client can connect to the database server to run DB2 commands, bind packages on the database server.

1. Open an IBM Db2 command window, and change to the following directory:

   `<IBM Db2 installation directory>/SQLLIB/bnd`

2. Connect to the database by running the following command:

   ```
   db2 connect to <database name> user <database user> using <database user password>
   ```

   Note: The database user must have the bind permission.

3. Run the following bind command:

   ```
   db2 bind @db2cli.lst blocking all grant public sqlerror continue CLIPKG 10
   ```

   The required packages are bound to the database server.

Step 4. Create the ActiveVOS Schema

To install ActiveVOS, you need to create the ActiveVOS schema. To create the schema, run the `create_bpm` script.

If you want to create multiple MDM Hub Master Databases, create an ActiveVOS schema for each MDM Hub Master Database.

1. Open a command prompt and change to the following directory:

   `<MDM Hub distribution directory>/database/bin`

2. Run the following command:

   On UNIX. `./sip_ant.sh create_bpm`

   On Windows. `sip_ant.bat create_bpm`

3. Answer the prompts that appear.

   The prompt displays default text in brackets. Press **Enter** to use the default value and go to the next prompt.

   Note: IBM Db2 data sources are case-sensitive. To avoid issues related to case-sensitivity, Informatica recommends that you use uppercase letters to define names, such as those for schemas, columns, and triggers.

Property	Description
Database Type	Type of database to use. For IBM Db2, specify `DB2`. The database type must be the same as the database type selected for the MDM Hub Master Database and the Operational Reference Stores.
ActiveVOS Database Host Name	Name of the machine that hosts the database.
ActiveVOS Database TCP/IP Port	Port number that the database listener uses.
ActiveVOS Database Name	Name of the database.
ActiveVOS Database Schema/User Name	User name of the ActiveVOS Server administrative user.
ActiveVOS User Password	Password of the administrative user.
DBA User Name	User name of the database administrative user.
DBA Password	Password of the administrative user.
ActiveVOS Tablespace Name	The name of the tablespace that contains the records that are involved in MDM workflows.

4. After you create the schema, review the `sip_ant.log` file in the following directory:

   `<MDM Hub distribution directory>/database/bin`

   The `sip_ant.log` file logs any errors that might occur when you run the `sip_ant` script to create the ActiveVOS database schema.

Set Up the Application Server Environment

You can install the MDM Hub in WebSphere cluster environments or standalone WebSphere instances. Install and configure WebSphere according to the instructions in the WebSphere documentation.

Ensure that there are no white spaces in the WebSphere installation directory path.

Note: Install the application server in the same time zone as the database server.

The following table lists the properties and their values to configure before installation, followed by a brief description of where to set the property:

Custom Property	Value	Description
com.ibm.ws.scripting.echoparams	false	Set this property in the `wsadmin.properties` file, which is located the following directory: `<WebSphere installation directory>\WebSphere\AppServer\profiles\<profile name>\properties` Set this value to false to prevent the ActiveVOS database password from appearing in script text in the `patchinstallSetup.log` file. Default is true.

Configure Java Virtual Machines

To configure a Java Virtual Machine (JVM), set Java options by using the JAVA_OPTIONS environment variable. After you edit or add any Java options, restart the JVM.

If you use a WebSphere clustered environment, set the Java options for the following cluster components:

- Server. Set all the required Java options on each server in the cluster.
- Deployment Manager. Set all the required Java options.
- Node agent. Set only heap size by using the `-Xmx` and `-Xms` Java options.

The following table describes the Java options settings:

Java Options	Description
-server	Results in a slower startup but subsequent operations are faster.
-Djava.net.preferIPv4Stack	Specifies whether Java uses Internet Protocol version 4 (IPv4). If the operating system uses Internet Protocol version 6 (IPv6), set to `true`.
- Djava.security.egd	Reduces the startup time of Data Director in Linux environments. Set the value to `file:/dev/./urandom`.
-DUseESLegacyFqSearch	Specifies whether fielded search returns exact matches from within child nodes for a business entity type. Applicable only when you perform a fielded search on multiple fields. Indicates whether a search must return records that contain search values in the same child node if multiple query fields are at the child level. Set to `true` to return records that might match the child level query field from different child nodes. Default is `false`.

Java Options	Description
-Ddb2.jcc.charsetDecoderEncoder	Required to use the MDM Hub Sample Operational Reference Store. Enables the JDBC driver to return the Unicode replacement character (U+FFFD) in place of a sequence of bytes that is not a UTF-8 string. Set to `3`.
-Dcom.ibm.crypto.provider.DoRSATypeChecking	Specifies whether Java allows the RSA type encryption of data with private key and decryption with public key. Required for the MDM Hub installer to read the license certificates and for password hashing to work in the MDM Hub. Set to `false`. If you do not set `-Dcom.ibm.crypto.provider.DoRSATypeChecking` to `false`, the Hub Server might not start and you can encounter license errors.
-Djgroups.bind_addr	Interface on which JGroup must receive and send messages. Required in a multinode or clustered environment. Ensure that each node binds to its own network interface.
-De360.mdm.host -De360.mdm.port -De360.connection.channel	Application server communication protocol, host, and port. To deploy the MDM Hub applications on a Bootstrap port other than 2809, set the following Java options: - `-De360.connection.channel`. Set to the communication protocol that you want to use. Valid values are HTTP and HTTPS. Default is HTTP. - `-De360.mdm.host`. Set to the IP address of the WebSphere host. If the environment uses the HTTPS communication protocol and the security certificate is issued to a Fully Qualified Domain Name (FQDN), set to the FQDN. - `-De360.mdm.port`. Set to the WebSphere Bootstrap port configured in place of 2809. If you do not configure this parameter, Data Director screens that are based on the Entity 360 Framework might not work as expected.
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	Required for Data Director with subject areas. To deploy Data Director with subject areas, set the following Java options: - `-Didd.mdm.host`. Set to the host name or IP address of the application server host. - `-Didd.mdm.port`. Required property, used internally by the Data Director with subject areas application during server initialization. Specifies the HTTP or HTTPS listener port used by the JVM for the applications. Default is `8080`. - `-Didd.protocol`. Required property that is used for deploying the subject area application during server initialization. Specifies whether the communication protocol to use is HTTP or HTTPS. Default is `HTTP`.
-DFrameworksLogConfigurationPath	Path to the `log4j.xml` file.
-Dmdm.node.groupid	Specifies a group ID for Java Virtual Machines in the MDM Hub implementation. Required only if you want logical groupings of Hub Servers and Process Servers.

Java Options	Description
-Dfile.encoding -Dclient.encoding.override	Required if you want to use Informatica Data Director and use REST APIs to search for records. Set both the Java options to `UTF-8` to ensure that you can find and save records that contain UTF-8 characters.
-Dstricttransportsecurity.flag	Specifies whether web browsers must convert all attempts to access Data Director using the HTTP requests to the HTTPS requests instead. Set to `true`.
-XX:codecachetotal	JIT code cache size. To enhance the performance of the MDM Hub environment, set to `512m`.
-Xmx	Maximum JVM heap size. Set to 6 GB or higher. For example, to set the `-Xmx` to `6144m`, use the following JAVA_OPTIONS environment variable setting: `set "JAVA_OPTIONS=-server ... -Xmx6144m"`
-Xms	Initial heap size. Set to `2048m`.
-Xmso	Required for the Process Server JVMs. Initial stack size for operating system threads. Prevents the application server from shutting down unexpectedly due to low system thread stack size. Set to `4096k`.
-Xss	Initial stack size. Set to `2000k`.
XX:+UseCodeCacheFlushing	Specifies whether the JVM disposes of compiled code when the code cache is full.
-Dtask.pageSize=<maximum number of tasks>	Specifies the maximum number of ActiveVOS tasks that are retrieved for each request. Default is `5000`. Increase the number if your environment has a large number of tasks.

Logical Grouping of Java Virtual Machine Example

By grouping Java Virtual Machines (JVMs), you get a logical group of Hub Servers and Process Servers. When you deploy the Hub Server and Process Server applications in a logical JVM group, communication between the Hub Server and Process Server applications stay within the group. To group JVMs, you assign a group ID to each JVM in the MDM Hub environment.

Note: Process Server grouping is applicable to the cleanse and match process only. The logical groups are not applied to the internal server cache of the MDM Hub.

The following table shows an example of logical JVM groups:

JVM Group	JVM	Hub Server	Process Server
Group1	JVM1	Yes	Yes
Group1	JVM4	-	Yes

JVM Group	JVM	Hub Server	Process Server
Group2	JVM2	Yes	Yes
Group3	JVM3	-	Yes

For JVM1, add the following Java option in the startup script:

```
-Dmdm.node.groupid=Group1
```

For JVM2, add the following Java option in the startup script:

```
-Dmdm.node.groupid=Group2
```

For JVM3, add the following Java option in the startup script:

```
-Dmdm.node.groupid=Group3
```

For JVM4, add the following Java option in the startup script:

```
-Dmdm.node.groupid=Group1
```

After you configure the JVMs, and deploy the Hub Servers and Process Servers, the groups have the following characteristics:

- Group1 has two Process Servers, Group2 has one Process Server, and Group3 has one Process Server.
- All cleanse and batch calls stay in their own group with the exception of search. For example, any real-time call on the Hub Server in Group1 affects only the Group1 Process Servers (JVM1 and JVM4).

Configure Transport Layer Security (TLS)

Important: To ensure secure communication, on the application where you deploy Multidomain MDM, disable TLS 1.0 and enable TLS 1.2.

For information about how to configure TLS, see the documentation for your application server.

Secure HTTP Response Headers

To secure your application server environment, use secure HTTP response headers. Change or remove headers, such as X-Powered-By and Server.

For information about how to change or remove HTTP response headers, see the documentation for your application server.

Encrypt Passwords in the MDM Hub Environment

To encrypt sensitive data such as passwords that appear in log files in the MDM Hub environment, configure scripting administration in WebSphere.

1. Open the `wsadmin.properties` file in the following directory:

   ```
   <WebSphere installation directory>/profiles/<Application server profile name>/properties
   ```

2. Set the `com.ibm.ws.scripting.echoparams` Java property to `false`.

Create a Secure Profile in a WebSphere Environment

In WebSphere, configure a secure profile to use with Multidomain MDM and Informatica ActiveVOS.

1. From a command line, create a secure profile as shown in the following sample code:

   On Windows

   ```
   <app_server_root>\bin\manageprofiles.bat -create -profileName AppSrv01
   -profilePath <app_server_root>\profiles\AppSrv01
   -templatePath <app_server_root>\profileTemplates\default
   -adminUserName administrator -adminPassword password1 -enableAdminSecurity true
   ```

   On UNIX

   ```
   <app_server_root>/bin/manageprofiles.sh -create -profileName AppSrv01
   -profilePath <app_server_root>/profiles/AppSrv01
   -templatePath <app_server_root>/profileTemplates/default
   -adminUserName administrator -adminPassword password1 -enableAdminSecurity true
   ```

2. In the WebSphere console, change the security Transport type to SSL-Supported.
   a. Expand **Security** and click **Global Security**.
   b. Under Authentication, expand **RMI/IIOP security** and click **CSIv2 inbound communications**
   c. Under CSIv2 Transport Layer, from the Transport list, select **SSL-Supported**.
   d. Click **Apply**, and then click **Save**.
   e. Click **CSIv2 outbound communications**
   f. Under CSIv2 Transport Layer, from the Transport list, select **SSL-Supported**.
   g. Click **Apply**, and then click **Save**.
3. In the WebSphere console, ensure that application security is set.
   a. Expand **Security** and click **Global Security**.
   b. Under Application Security, select **Enable application security**.
   c. Click **Apply**, and then click **Save**.
4. Set up federated repositories.
   a. Expand **Security** and click **Global Security**.
   b. Under User account repository, from the Available realm definitions list, select **Federated repositories**.
   c. Click **Configure**.
   d. Under Repositories in the realm, click **Use built-in repository**.
   e. Specify a password for the administrative user.
   f. Click **Apply**, and then click **Save**.
5. Restart the WebSphere profile.

Configure JAAS Application Login

If you enabled application security in a WebSphere cluster environment, configure JAAS application login. The JAAS application login setting is required for ActiveVOS to start up.

1. Start the WebSphere console.
2. Configure a JAAS login module.
   a. Expand **Security**, and click **Global security** > **JAAS - Application logins**.
   b. Add an application login for the ActiveVOS provided user.

c. Specify the alias as `ActiveVOSProvidedUser`.

d. In the **JAAS login modules** section, add the following module class:

`com.activee.rt.websphere.trustvalidation.AeBasicLoginModule`

e. Save the changes.

3. In the **General properties** section, set the module properties.
   a. Set the authentication strategy to **Required**.
   b. Enter the user name and password for the module.
   c. Save the changes.
4. Create the ActiveVOS user.
   a. Expand **Users and Groups**, and click **Manage Users**.
   b. Click **Create**, and enter user details, such as name and password.
   c. Click **Create**.

   The ActiveVOS user is created.
5. Add an administrative role for the ActiveVOS user.
   a. Under **Users and Groups**, click **Administrative user roles** > **User**.
   b. Select the **Monitor** role for the user.
   c. Move the user from the **Available** list to the **Mapped to role** list.
   d. Click **OK**.
6. Configure ActiveVOS to use the JAAS application login.
   a. Start the ActiveVOS Configuration Wizard.
   b. In the **Global Security Configuration** page, select **Cluster Communications Use JAAS Login**.
   c. From the **Security JAAS Login Name** list, select **ActiveVOSProvidedUser**.
   d. In the **JAAS Login Configuration** page, select **ActiveVOS Provided User**.
   e. Specify the user name and password for the ActiveVOS provided user.
   f. Save the changes.

Create the ActiveVOS Console Administrative User

If you want to use ActiveVOS, create the ActiveVOS Console administrative user with the `abAdmin` role in the application server container. If you do not create an administrative user, the Hub Server deployment fails. Use the ActiveVOS Console administrative user name and password when the Hub Server installer prompts you to enter the administrative user credentials for the ActiveVOS Console.

- Log in to the WebSphere console, and create the ActiveVOS Console administrative user.

  Note: The ActiveVOS console user is mapped to the `abAdmin` role when you run the `postInstallSetup` or the `patchInstallSetup` script during the post-installation or post-upgrade process.

Configure SOAP Request Timeout for MDM Hub Deployments

To ensure that deployment of the MDM Hub components do not time out, set the SOAP request timeout property. After a successful installation, you can reset the property to its default value.

1. Open the `soap.client.props` file in the following directory:

   `<WebSphere installation directory>/profiles/<Application server profile name>/properties`

2. Set the `com.ibm.SOAP.requestTimeout` property to 1800 or higher.

Additional Application Server Configuration (Optional)

Perform additional WebSphere configuration based on the requirements of the MDM Hub environment.

The following table describes the configurations that you can perform:

Configuration	Description
Configuring WebSphere for standalone Process Server instances	Required to configure WebSphere for standalone Process Server instances in the following scenarios: - You want to install a Process Server instance on a WebSphere instance on which you do not have the Hub Server installed. - You want to install multiple, standalone Process Server instances.
Configuring WebSphere for multiple MDM Hub Master Databases	Required if you want to configure multiple MDM Hub Master Database instances.
Configuring the HTTPS protocol	Required if you want to configure the HTTPS protocol for the MDM Hub communications.
Configuring WebSphere for Informatica Data Director	Required if you want to use Informatica Data Director.

Configuring WebSphere for Standalone Process Server Instances

If you want to install multiple, standalone Process Server instances, configure WebSphere to use the appropriate data source. Also, if you want to install a Process Server instance on a WebSphere instance on which you do not have the Hub Server installed, configure the data source.

Perform the following tasks to configure WebSphere to use the appropriate data source:

1. Install the JDBC driver.
2. Create an MDM Hub Master Database data source.
3. Create an Operational Reference Store data source.

Step 1. Install the JDBC Driver

Before you create data sources for the MDM Hub Master Database and the Operational Reference Store (ORS), install the JDBC driver.

Contact IBM to get the supported version of the JDBC driver.

- Copy the JDBC driver to the following directory:

```
<WebSphere installation directory>/lib
```

Step 2. Create an MDM Hub Master Database Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for the MDM Hub Master Database.

1. Start the WebSphere Application Server Administrative Console.
2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      ```
      <WebSphere installation directory>/lib
      ```
3. Create the security account that the MDM Hub Master Database data source will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.
   d. Click **New**, and specify the following properties:

Property	Description
Alias	Name of the MDM Hub Master Database.
User ID	User name to connect to the MDM Hub Master Database.
Password	Password to access the MDM Hub Master Database.

   e. Click **OK**.
4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.

      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.
   c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

   d. Click **Next**, and then click **Finish**.

5. Create the MDM Hub Master Database data source.
   a. Click the JDBC provider that you created.

      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.

      The **Data Sources** page appears.
   c. Click **New**.
   d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM Master Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-cmx_system-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/cmx_system`.

   e. Click **Next**, and then click **Finish**.

Step 3. Create an Operational Reference Store Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for each Operational Reference Store.

1. Start the WebSphere Application Server Administrative Console.
2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      `<WebSphere installation directory>/lib`
3. Create the security account that the Operational Reference Store will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.

d. Click **New**, and set the following properties:

Property	Description
Alias	Name of the Operational Reference Store.
User ID	User name to connect to the Operational Reference Store.
Password	Password to access the Operational Reference Store.

e. Click **OK**.

4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.
      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.
   c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

   d. Click **Next**, and then click **Finish**.

5. Create the Operational Reference Store data source.
   a. Click the JDBC provider that you created.
      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.
      The **Data Sources** page appears.
   c. Click **New**.

d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM ORS Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-<IBM Db2 host name>-<IBM Db2 database name>-<Operational Reference Store name>-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/<Operational Reference Store name>`.

e. Click **Next**, and then click **Finish**.

Configuring WebSphere for Multiple MDM Hub Master Databases

If you want to configure multiple MDM Hub Master Database instances, configure as many WebSphere profiles as the number of MDM Hub Master Database instances. Each MDM Hub Master Database instance must have its own MDM Hub instance. Therefore, create as many WebSphere profiles to deploy each MDM Hub instance on a separate WebSphere profile.

Configuring the HTTPS Protocol

To use the HTTPS protocol for communication between the MDM Hub components, such as the Hub Server, Process Server, and ActiveVOS, configure the HTTPS protocol in the WebSphere application server.

1. Create an SSL-enabled WebSphere port.
2. Configure WebSphere to allow self-signed certificates.
3. Configure the following custom JVM properties:

Custom JVM Property	Description
javax.net.ssl.keyStore	Location of the keystore.
javax.net.ssl.keyStorePassword	Password of the keystore.
javax.net.ssl.keyStoreType	Type of the keystore.
javax.net.ssl.trustStore	Location of the truststore.
javax.net.ssl.trustStorePassword	Password of the truststore.
javax.net.ssl.trustStoreType	Type of the truststore.

For more information about configuring the HTTPS protocol, see the WebSphere documentation.

Configuring WebSphere for Informatica Data Director

If you want to use Data Director, configure WebSphere and then restart WebSphere for the changes to take effect.

Ensure that you perform the following configurations:

- Set the web container custom property.
  Use the WebSphere Console to set `com.ibm.ws.webcontainer.invokerequestlistenerforfilter` to `true`. For instructions on setting web container custom properties, see the WebSphere documentation.
- To support the management of tasks, increase the value for timeout properties by a factor of 2.
  Perform the following task by using the the WebSphere Console:
  1. navigate to **WebSphere Console Servers** > **Server Types** > **WebSphere application servers** > **<target server name>**.
  2. In the **Container Services** category, click **Transaction service** and increase the values for the timeout properties.

Configure the Properties Files for Silent Installation

If you want to install the Hub Server and the Process Server without user interaction in silent mode, configure the installation properties files. You might want to perform a silent installation if you need multiple installations, or if you need to install on a machine cluster. A silent installation does not show any progress or failure messages.

The installer reads the silent installation properties file to determine the installation options. Ensure that you provide correct settings in the properties file because the silent installation process might complete successfully even if the settings are incorrect.

You can configure the following silent installation properties files:

- Hub Server. Required to install the Hub Server in silent mode.
- Process Server. Required to install the Process Server in silent mode.

Note: If you do not want to manually configure the silent installation properties file, you can use the `-r` command-line option during installation to generate the silent installation properties file.

Configuring the Hub Server Properties File

If you want to install the Hub Server in silent mode, configure the Hub Server properties file. Specify the options for the installation in the properties file, and save the file with a new name.

1. Find the `silentInstallServer_sample.properties` file in the following directory: `/silent_install/mrmserver`
2. Create a backup copy of the `silentInstallServer_sample.properties` file.
3. Open the file in a text editor, and configure the values of the installation parameters.
4. Save the properties file with a new name such as `silentInstallServer.properties`.

Configuring the Process Server Properties File

If you want to install the Process Server in silent mode, configure the Hub Server properties file. Specify the options for the installation in the properties file, and save the file with the new name.

1. Find the `silentInstallCleanse_sample.properties` file in the following directory: `/silent_install/mrmcleanse`
2. Create a backup copy of the `silentInstallCleanse_sample.properties` file.
3. Open the file in a text editor, and configure the values of the installation parameters.
4. Save the properties file with a name such as `silentInstallCleanse.properties`.

CHAPTER 3

Hub Store Installation

This chapter includes the following topics:

Create the MDM Hub Master Database

After you install IBM Db2, create an MDM Hub Master Database. If you want to create multiple MDM Hub Master Databases, ensure that you create each MDM Hub Master Database in a different schema. The default name of the MDM Hub Master Database is `CMX_SYSTEM`, but you can use a custom name.

Note: If you change the folder names in the distribution directory, metadata import fails.

1. Open a command prompt, and navigate to the following directory:

   ```
   <MDM Hub distribution directory>/database/bin
   ```

2. To create the MDM Hub Master Database, run the following command:

   On UNIX. `./sip_ant.sh create_system`

   On Windows. `sip_ant.bat create_system`

3. Answer the prompts described in the following table:

Note: The prompt displays default text in brackets. Press **Enter** to use the default value and go to the next prompt.

Prompts	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify `DB2`. **Note:** Db2 datasources are case-sensitive. To avoid issues related to case-sensitivity, Informatica recommends that you use uppercase letters to define names, such as the schema names, column names, and triggers.
Enter the database host name [localhost]	Name of the host that runs the database. Default is `localhost`. **Important:** In cluster environments, specify the absolute host name or the IP address to avoid caching issues.
Enter the database port number [50000]	Port number that the database listener uses. Default is `50000`. **Note:** You cannot create the Db2 database if the port number is not the default 50000. You must manually change the port number in the Db2 properties file.
Enter the database instance name [SIP97]	Name of the database instance. Default is `SIP97`.
Enter master database name [cmx_system]	Name of the MDM Hub Master Database schema. Default is `cmx_system`.
Enter master user name [cmx_system]	User name to access the MDM Hub Master Database. Default is `cmx_system`.
Enter master database user password	Password to access the MDM Hub Master Database.
Enter locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]	Operating system locale. Default is `en_US`.
Enter the DBA user name [DB2ADMIN]	User name of the administrative user. Default is `DB2ADMIN`.
Enter the DBA password	Password of the administrative user.

4. To verify that the MDM Hub Master Database was created successfully, review `sip_ant.log` in the following directory:

   ```
   <MDM Hub distribution directory>/database/bin
   ```

   The `sip_ant.log` file logs any errors that might occur when you run the `sip_ant` script to create the MDM Hub Master Database.

5. If you intend to configure single sign-on authentication for other applications such as Salesforce, increase the BLOB column length. Run the following command on the MDM Hub Master Database:

   ```
   SET SCHEMA CMX_SYSTEM; ALTER TABLE C_REPOS_SAM_PROVIDER_FILE ALTER COLUMN PROVIDER_FILE
   SET DATA TYPE BLOB (10240000); CALL ADMIN_CMD('REORG TABLE C_REPOS_SAM_PROVIDER_FILE');
   ```

Create an Operational Reference Store

After you complete the preinstallation tasks, create an Operational Reference Store (ORS). The default name of the ORS is `CMX_ORS`.

Note: If you change the folder names in the distribution directory, metadata import fails.

1. Open a command prompt, and navigate to the following directory:

   `<MDM Hub distribution directory>/database/bin`

2. To create an ORS, run the following command:

   On UNIX. `./sip_ant.sh create_ors`

   On Windows. `sip_ant.bat create_ors`

3. Answer the prompts described in the following table:

   Note: The prompts display default text in brackets. Press **Enter** to use the default value and go to the next prompt.

Prompts	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify `DB2`. **Note:** Db2 datasources are case-sensitive. To avoid issues related to case-sensitivity, Informatica recommends that you use uppercase letters to define names, such as the schema names, column names, and triggers.
Enter the Operational Reference Store schema host name [localhost]	Name of the host that is running the database. Default is `localhost`.
Enter the Operational Reference Store schema port number [50000]	Port number that the database listener uses. Default is `50000`.
Enter the Operational Reference Store database instance name [SIP97]	Name of the database instance. Default is `SIP97`.
Enter the Operational Reference Store schema name [cmx_ors]	Name of the Operational Reference Store database. Default is `cmx_ors`.
Enter the Operational Reference Store database user name [cmx_ors]	User name to access the Operational Reference Store. Default is `cmx_ors`. **Note:** If you need to specify a user name that is different from the schema name, you must configure the user as a proxy user.
Enter the Operational Reference Store database user password	Password to access the Operational Reference Store.
Enter locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]	Operating system locale.
Enter the DBA user name [DB2ADMIN]	User name of the administrative user. Default is `DB2ADMIN`.
Enter the DBA password	Password of the administrative user.

4. To verify that the ORS was created successfully, review `sip_ant.log` in the following directory:

   `<MDM Hub distribution directory>/database/bin`

   The `sip_ant.log` file logs any errors that might occur when you run the `sip_ant` script to create the ORS.

Import the Metadata into the MDM Hub Master Database

After you create the MDM Hub Master Database, import the initial metadata into the MDM Hub Master Database. The initial metadata includes repository tables and other objects that the MDM Hub requires in the Hub Store.

Note: If you change the folder names in the distribution directory, metadata import fails.

1. Open a command prompt, and navigate to the following directory:

   `<MDM Hub distribution directory>/database/bin`

2. To import the initial metadata, run the following command:

   On UNIX. `./sip_ant.sh import_system`

   On Windows. `sip_ant.bat import_system`

3. Answer the prompts described in the following table:

   Note: The prompts display default text in brackets. Press **Enter** to use the default value and go to the next prompt.

Prompts	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify `DB2`.
Enter the database host name [localhost]	Name of the host that is running the database.
Enter the database port number [50000]	Port number that the database listener uses. Default is `50000`.
Enter the database instance name [SIP97]	Name of the database. Default is `SIP97`.
Enter master database name [cmx_system]	Name of the MDM Hub Master Database schema. Default is `cmx_system`.
Enter master user name [cmx_system]	User name to access the MDM Hub Master Database. Default is `cmx_system`. **Note:** On UNIX, ensure that you use a user name with 8 characters or less.
Enter master database user password	Password to access the MDM Hub Master Database.

Prompts	Description
Enter locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]	Operating system locale. Default is `en_US`.
Connect URL [jdbc:db2://localhost:50000/SIP97:currentSchema=CMX_SYSTEM;]	Connect URL for IBM Db2. Default is `jdbc:db2://localhost:50000/SIP97:currentSchema=CMX_SYSTEM`.

4. After you import the initial metadata, review the following log files for errors:
   - `seed.log`. Contains database errors.
     The `seed.log` file is in the following directory: `<MDM Hub installation directory>/database/bin/db2`
   - `sip_ant.log`. Contains user input errors.
     The `sip_ant.log` file is in the following directory: `<distribution directory>/database/bin`

Import the Metadata into the Operational Reference Store

After you create the Operational Reference Store, import the initial metadata into the Operational Reference Store. The initial metadata includes repository tables and other objects that the MDM Hub requires in the Hub Store.

Note: If you change the folder names in the distribution directory, metadata import fails.

1. Open a command prompt, and navigate to the following directory:

   `<MDM Hub distribution directory>/database/bin`

2. To import the initial metadata, run the following command:

   On UNIX. `./sip_ant.sh import_ors`

   On Windows. `sip_ant.bat import_ors`

3. Answer the prompts described in the following table:

   Note: The prompts display default text in brackets. Press **Enter** to use the default value and go to the next prompt.

Prompts	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify `DB2`.
Enter the Operational Reference Store database host name [localhost]	Name of the host that is running the database.
Enter the Operational Reference Store database port number [50000]	Port number that the database listener uses. Default is `50000`.
Enter the database name [SIP97]	Name of the database. Default is `SIP97`.

Prompts	Description
Enter the Operational Reference Store database name [cmx_ors]	Name of the Operational Reference Store database. Default is `cmx_ors`.
Connect URL. [jdbc:db2://<host name>:<port>/<database name>]	Connect URL for the master database. Default is `jdbc:db2://<host name>:<port>/<database name>`.
Enter the Operational Reference Store database user name [cmx_ors]	User name to access the Operational Reference Store. Default is `cmx_ors`.
Enter the Operational Reference Store database user password	Password to access the Operational Reference Store.
Enter locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]	Operating system locale. Default is `en_US`.
Enter the integer code of Operational Reference Store Timeline Granularity: Year 5, Month 4, Day 3, Hour 2, Minute 1, Second 0 [3]	Specify timeline units to use. Default is days. **Note:** The timeline granularity that you configure cannot be modified later. For more information about timeline, see the *Multidomain MDM Configuration Guide*.

4. After you import the initial metadata, review the following log files for errors:
   - `seed.log`. Contains database errors.
     The `seed.log` file is in the following directory: `<MDM Hub installation directory>/database/bin/db2`
   - `sip_ant.log`. Contains user input errors.
     The `sip_ant.log` file is in the following directory: `<MDM Hub distribution directory>/database/bin`

CHAPTER 4

Hub Server Installation

This chapter includes the following topics:

Installing the Hub Server

You can install the Hub Server using an installation wizard, a silent installation script, or, on UNIX systems, a command line script. If you complete the pre-installation tasks and collect the information you need before you start the installer, the installation process takes about 15 minutes.

	**STOP! Did you complete the pre-installation tasks?** The installation will fail if you do not complete the pre-installation tasks before you run the installer.
	**Installation Readiness Checklist** ❑Created an MDM implementation plan. ❑Verified that your servers meet the system requirements. ❑Verified that your operating system and software versions are supported. ❑Reviewed the known limitations for your operating system and software versions. ❑Installed and configured a supported version of an application server. ❑Installed and configured a supported version of a database management system. ❑Performed the pre-installation configuration tasks for your environment. ❑Saved the MDM license file in an accessible location.

If you missed a task, go back to the preceding chapters for help in completing the task.

When you are ready to proceed, perform the following steps:

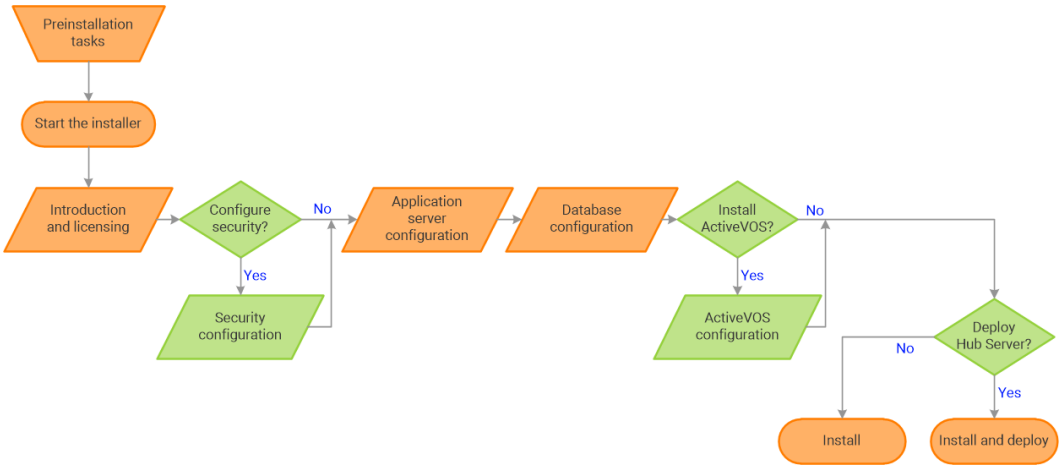
1. Review the installation workflow.

2. Collect all the values to enter during the installation.
3. Begin the installation by using the installation wizard or the command line, or in the silent mode.

Review the Installer Workflow

Whether you use the installation wizard, the command line prompts, or the silent installation script, the installer goes through the steps in the same order. You must follow the installation workflow keeping the decision points in mind.

The following diagram walks you through the steps in the Hub Server installer workflow:



Take a moment to identify the decision points in the installation process. Consult your implementation plan to understand which paths to take at each of the following decision points:

1. **Configure security?** Use the default security or configure your own security by specifying a password hashing algorithm and certificate provider.
2. **Install ActiveVOS?** Administrators use Informatica ActiveVOS to set up review workflows. Authorized users can validate proposed changes to records before the master data is updated.

   Note: If you already have an installation of Informatica ActiveVOS, you do not need to install it again during the Hub Server installation.
3. **Deploy the Hub Server?** Allow the installer to run the `postInstallSetup` script. Among other important tasks, the script deploys the Hub Server to the application server. Alternatively, you can run the `postInstallSetup` script after you exit the installer.

Collect the Installation Values

Before you run the installer, collect the installation values. The installer will prompt you for information about your application server, database management system, and other components. The best practice is to print out these tables and add the values for your environment.

Application Server: IBM WebSphere

Use the following table to collect the WebSphere details that are required for the installation:

Property	Description	Default value	Server instance 1 value
WebSphere Installation Directory	The location where WebSphere is installed.	-	
Is WebSphere security enabled?	If WebSphere security is enabled, select **Yes** and provide the user name and password.	No	
Server Name	Name of the WebSphere application server on which you want to deploy the Hub Server. In a clustered environment, enter one of the cluster server names and its corresponding Bootstrap port and SOAP connector port information.	-	
Bootstrap Port	Bootstrap port number that is used by the server that you specify. **Tip:** To find the port information, go to the WebSphere administrative console, and then click **Application servers** > **<server name>** > **Ports**.	2809	
SOAP Connector Port	SOAP connector port number that is used by the server that you specify. **Tip:** To find the port information, go to the WebSphere administrative console, and then click **Application servers** > **<server name>** > **Ports**.	8880	
Profile Name	Name of the WebSphere profile that contains the WebSphere application server on which you want to deploy the Hub Server.	-	
User Name	Name of a WebSphere user that has administrative privileges.	admin	
Password	Password of the WebSphere administrative user.	-	

In a clustered environment, add details for the other WebSphere instances to the following table:

Property	Server instance 2 value	Server instance 3 value	Server instance 4 value
WebSphere Installation Directory			
Is WebSphere security enabled?			
Server Name			
Bootstrap Port			

Property	Server instance 2 value	Server instance 3 value	Server instance 4 value
SOAP Connector Port			
Profile Name			
User Name			
Password			

Database: IBM Db2

Use the following table to collect the IBM Db2 details that are required for the installation:

Property	Description	Default value	Installation value
JDBC Driver Directory	Db2 directory that contains the following JDBC JAR files: - `db2jcc.jar` - `db2jcc_license_cu.jar`	-	
Server	Host name or IP address of the machine hosting IBM Db2 that contains the Master Database.	localhost	
Port	Port number that you want the Hub Server to use to communicate with IBM Db2.	50000	
Database Name	Name of the database that you created for Multidomain MDM.	-	
MDM Master Database Name	Name of the MDM Master Database. You create the MDM Master Database during the Hub Store installation by running the `sip_ant create_system` script.	cmx_system	
MDM Master Database User Name	Name of the IBM Db2 user that was used to import metadata into the MDM Master Database.	cmx_system	
MDM Master Database Password	Password of the IBM Db2 user.	-	

Security - Optional

You can choose to configure the hash algorithm, the certificate provider, or both.

Hash Key and Hash Algorithm

To configure a hash algorithm, you must implement the Java abstract class `com.siperian.sam.security.hashing.algorithms.HashAlgorithm` in the `siperian-server-hash.jar` file. The JAR file is located in the following directory: `<MDM installation directory>/hub/server/lib/hashing`

Use the following table to collect the hash algorithm details that were used in the Java abstract class implementation in the `siperian-server-hash.jar` file:

Property	Description	Default value	Installation value
Hash Key	Optional. A sequence of up to 32 hexadecimal characters with no delimiters. The key size can be up to 128 bits. Store the key securely. **Caution:** If your custom hash key is exposed or lost, you must reset all user passwords.	-	
Hash Algorithm	List of configured algorithms. To configure a custom algorithm, select **Other**.	SHA-3	
Name	Name of the custom hash algorithm.	-	
Implementation File	Location of the custom hash algorithm archive. **Note:** The compressed file must contain all the necessary JAR files and supporting files.	-	
Canonical Class Name	Canonical class name for the hash algorithm implementation. For example: $HASHING_CLASS_NAME$	-	

Certificate Provider

To implement a custom certificate provider, you must implement the `PKIUtil.java` interface in the `siperian-server-pkiutil.jar` file. The JAR file is located in the following directory: `<MDM installation directory>/hub/server/lib/pkiutils`

Use the following table to collect the details of the custom certificate provider that were used in the Java abstract class implementation in the `siperian-server-pkiutil.jar` file:

Property	Description	Default value	Installation value
Certificate Provider	By default, MDM authenticates trusted applications. To configure a custom provider, select **Other**.	Default	
Implementation File	Name of the custom certificate provider.	-	
Class Name	Class name for the certificate provider implementation.	-	

Hub Console

The Hub Console requires the server details to connect to the Hub Server machine.

Use the following table to collect the server details that the Hub Console requires:

Property	Description	Default value	Installation value
Publicly Accessible Host Name	IP address or publicly accessible host name (FQDN) of the server to which the application server binds.	-	
HTTP Port	HTTP port of the server that the Hub Console must use.	-	

Informatica ActiveVOS - Optional

For more information about Informatica ActiveVOS, ActiveVOS Central, or ActiveVOS Server, see the Informatica ActiveVOS documentation.

Use the following table to collect the details that are required for ActiveVOS installation:

Property	Description	Default value	Installation value
ActiveVOS Server Installation Directory	An empty directory where you want to install the ActiveVOS server.	`<MDM Hub installation directory>/avos/server`	
Server	Host name or IP address of the machine hosting IBM Db2 that contains the ActiveVOS database.	localhost	
Port	Port number that you want the Hub Server to use to communicate with IBM Db2.	50000	
Database Name	Name of the database that you created for ActiveVOS.	-	
ActiveVOS Schema	Name of the ActiveVOS schema. You create the ActiveVOS schema when you perform the pre-installation task by running the `sip_ant create_bpm` script.	avos	
ActiveVOS Schema User Name	Name of the IBM Db2 user that was used to create the ActiveVOS schema.	avos	
ActiveVOS Schema Password	Password of the IBM Db2 user.	-	
Web Services URL	Host and port where the ActiveVOS Server runs. You choose to use either the http or https protocol. The Hub Server uses the same URL that ActiveVOS Central (also called Process Central) uses to call the ActiveVOS Server. The URL is called the Process Central AeTaskService URL in the Informatica ActiveVOS documentation and has the following format: http://[hostname.domainname]:[port]/active-bpel/services/AeB4PTaskClient-taskOperations.	http://localhost:2809	

Property	Description	Default value	Installation value
ActiveVOS Server Installer File	The Multidomain MDM distribution package contains the installer for the ActiveVOS Server.	ActiveVOS_Server_<operating system>_<version>	
Process Console User Name	An authorized user for the ActiveVOS Process Console.	-	
Process Console Password	Password for the ActiveVOS Process Console user.	-	

Product Usage Toolkit

The product usage toolkit sends information about your MDM environment to Informatica. The information is used by Informatica Global Customer Support to help troubleshoot and provide recommendations for your environment. If you do not want the toolkit to send any information to Informatica, you can disable the toolkit after you install MDM.

Use the following table to collect the details that are required for installing the product usage toolkit:

Property	Description	Default value	Installation value
Industry	Type of industry that best matches your organization's business.	-	
Environment	Type of environment that you are installing in. If you install from the command line, enter one of the following numbers: - 1. Production environment - 2. Test or QA environment - 3. Development environment	-	
Does your network have a proxy server?	If yes, provide details about the proxy server.	No	
Host	Name or IP address of the proxy server.	-	
Port	Port number used by the proxy server.	-	
Domain Name	If your proxy server is part of a domain, the name of the domain.	-	
User Name	If you use a secure proxy server, the name of a user that can access MDM.	-	
Password	Password of the user.	-	

Install the Hub Server from the Installation Wizard

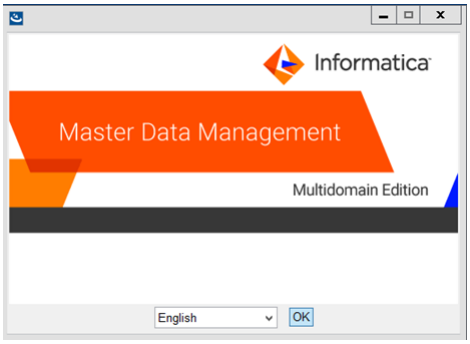
Use the installation wizard when you want to install the Hub Server in graphical mode. The installation wizard guides you through the installation.

You must use the same user name to install the Hub Server and the Process Server.

1. Start the application server.
2. Navigate to the following directory:

   `<MDM Hub distribution directory>/<operating system name>/mrmserver`
3. Start the installer by performing the task for your operating system:
   - **UNIX**. From the command line, run `./hub_install.bin`.
   - **Windows**. From the File Explorer, double-click `hub_install.exe`.

   The Hub Server installation wizard starts.

   Tip: If the installation wizard does not start, verify that a supported version of Java is installed and included in your classpath or environment variable.
4. Choose a language and click **OK**.

   The **Introduction** window appears.
5. Follow the online instructions. When prompted, enter the installation values that you collected.
6. At the end of the installation, in the **Configuration Summary** window, review the options that you selected.
7. If you need to make changes, go back to the appropriate window by clicking **Previous**. When you are done, click **Next** to return to the final window.
8. Click **Install**.
9. **Next step**: The next step depends on whether you chose to deploy the Hub Server from the installer.
   - If you chose to deploy the Hub Server from the installer, you do not need to deploy the Hub Server as part of the post-installation tasks.
   - If you chose to deploy the Hub Server later, you must deploy the Hub Server as part of the post-installation tasks.

Install the Hub Server from the Command Line (UNIX Only)

On UNIX, you can install the Hub Server from the command line. Run the script to start the command line installation.

1. Start the application server.
2. From the command line, navigate to the following directory:

   `<MDM Hub distribution directory>/<operating system name>/mrmserver`
3. Run the following command:

   `./hub_install.bin -i console`

   The Hub Server installation prompts appear.
4. Enter the installation values that you collected.

   To use the default value shown in brackets, press **Enter**.
5. **Next step**: After the installation completes, the next step depends on whether you chose to deploy the Hub Server.
   - If you chose to deploy the Hub Server from the installer, you do not need to deploy the Hub Server as part of the post-installation tasks.
   - If you chose to deploy the Hub Server later, you must deploy the Hub Server as part of the post-installation tasks.

Install the Hub Server Silently

You can install the Hub Server in silent mode. Before you start the silent installation, ensure that you configured the silent installation properties file.

1. Start the application server.
2. Copy the silent installation properties file to the target environment.
3. In the target environment, run the command for your operating system:
   - **UNIX**. `./hub_install.bin -f <absolute path to edited installer properties file>`
   - **Windows**. `hub_install.exe -f <absolute path to edited installer properties file>`

   The silent installer runs in the background. The process can take a while.
4. If you chose to have the installer deploy the Hub Server, check `postinstallSetup.log` to verify that the installation was successful.

   The log file is in the following directory:

   `<MDM Hub installation directory>/hub/server/logs`
5. **Next step**: After the installation completes, the next step depends on whether you chose to deploy the Hub Server.
   - If you chose to deploy the Hub Server from the installer, you do not need to deploy the Hub Server as part of the post-installation tasks.
   - If you chose to deploy the Hub Server later, you must deploy the Hub Server as part of the post-installation tasks.

Install the Hub Server on Nodes in the Cluster

In application server cluster environments, install the Hub Server on all the nodes of the cluster to which you must deploy the Hub Server application. You must complete the installation on one node of a cluster before you start the installation on another node of the cluster.

For example, a WebSphere cluster has four servers that run in host1, host2, host3, host4, and the servers use RMI ports 2812, 2813, 2814, and 2815. Each server has a node. You need to install the Hub Server on node1, node2, node3, and node4. You can install the Hub Server on the nodes in any order. Complete the Hub Server installation on any one node, such as node2, before you start the installation on another node such as node1 or node4.

Ensure that the directory structure of the Hub Server installation is the same on all the nodes.

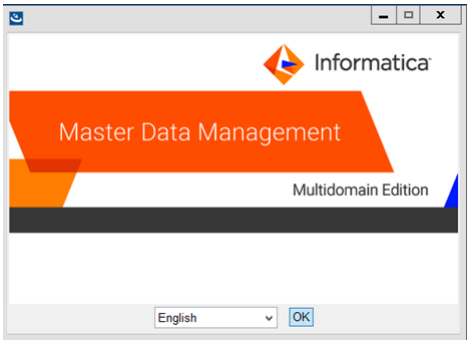
1. To start the WebSphere cluster, perform the following steps:
   a. Start the WebSphere deployment manager.
   b. Start the nodes of the WebSphere cluster on which you want to install the Hub Server.
   c. Start the WebSphere cluster.
2. Navigate to the following directory:

   ```
   <MDM Hub distribution directory>/<operating system name>/mrmserver
   ```
3. To start the Hub Server installer on a cluster node, run the command for your operating system:

   UNIX. `./hub_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

   Windows. `hub_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

   The Hub Server installation wizard starts.



4. Follow the online instructions. When prompted, enter the installation values that you collected.
5. **Next step**: After the installation completes, deploy the Hub Server manually on all the nodes that have the installation.

   If you performed an automatic deployment for the primary node, you do not need to manually deploy on the primary node.

CHAPTER 5

Hub Server Post-Installation Tasks

This chapter includes the following topics:

Copy the Installation Log Files

The installation log files are useful for troubleshooting the Hub Server installation process. Copy the log files to the installation documentation directory. Informatica Global Customer Support might request copies of the log files if you contact them regarding installation issues.

The following table describes the different types of installation log files:

Log File Type	Description
Installation log	- File name. `Informatica_MDM_Hub_Server_Install_<timestamp>.xml` - Location. `<MDM Hub installation directory>/hub/server/logs` - Contents. Directories and registry entries that are created, names of the files installed and commands run, and status for each installed file.
Installation prerequisites log	- File name. `installPrereq.log` - Location. `<MDM Hub installation directory>/hub/server/logs` - Contents. Logs of prerequisite checks performed by the installer.
Debug log	- File name. `infamdm_installer_debug.txt` - Location. `<MDM Hub installation directory>/hub/server` - Contents. Detailed information about the choices that are made during the installation, and the actions performed by the installer.
Post-installation setup log	- File name. `postInstallSetup.log` - Location. `<MDM Hub installation directory>/hub/server/logs` - Contents. Summary of actions performed by the installer during the post-installation process and the errors in the post-installation process.
Hub Server log	- File name. `cmxserver.log` - Location. `<MDM Hub installation directory>/hub/server/logs` - Contents. Summary of the Hub Server operations.
WebSphere logs	- File names. `startServer.log`, `stopServer.log`, `SystemErr.log`, and `SystemOut.log` - Location. `<WebSphere installation directory>/profiles/<Application server profile name>/logs/<server name>` - Contents. WebSphere server status, and performance information.

Verify the Version and Build Number

Ensure that the correct version and build number of the Hub Server is installed.

1. Open a command prompt, and navigate to the following directory: `<MDM Hub installation directory>/hub/server/bin`
2. To verify the Hub Server version and build number, run the following command:

   On UNIX. `versionInfo.sh`

   On Windows. `versionInfo.bat`

   Note: For AIX systems, change the `versionInfo.sh` script to run Java from the `<Java home>/jre/bin` directory.

Install and Configure Elasticsearch

To use search, install and setup Elasticsearch for the MDM Hub.

For more information about installing and configuring search, see the "Search with Elasticsearch" chapter in the *Multidomain MDM Configuration Guide*.

Configure the Hub Console Client (Conditional)

The Hub Console requires the host name and port properties in the `build.properties` file to connect to the Hub Server machine. You can override the host name and port number when you launch the Hub Console.

Edit the properties file in the following scenarios:

- If HTTPS is enabled for the application server and you need to use an HTTPS port.
- In a multi-node or a cluster environment, if you copied the Hub Server installation directory from one node to the other nodes, edit on each node.

1. Open the `build.properties` file that is in the following directory:

   `<MDM Hub installation directory>/hub/server/bin`
2. Edit the following properties:
   - `console.hostname`. Specify the IP address or publicly accessible host name (FQDN) of the server to which the application server binds.
   - `console.webport`. Specify the HTTP or HTTPS port of the current node that the Hub Console must use.
3. Save the file.
4. Set relevant values for the `cmx.appserver.console.mode` property in the `cmxserver.properties` file in the following directory:

   `<MDM Hub installation directory>/hub/server/resources`
   Set the value to the communication protocol that you use, either HTTP or HTTPS.

After you edit the `build.properties` file, ensure that you run the `postInstallSetup` script to deploy the Hub Server applications.

Configure the MDM Hub Master Database Name

If the MDM Hub Master Database has a name other than `cmx_system`, configure the MDM Hub Master Database name in the `cmxserver.properties` file.

1. Open the `cmxserver.properties` file in the following directory:

   `<MDM Hub installation directory>/hub/server/resources`
2. Set the `cmx.server.masterdatabase.schemaname` property to the name that you specified for the MDM Hub Master Database.

   You specified the name for the MDM Hub Master Database when you created it.

Redeploy the Hub Server EAR File

After you run the `postInstallSetup` script either manually or as part of the Hub Server installation, use the WebSphere Server Administration Console to undeploy and deploy the Hub Server EAR file `siperian-mrm.ear`. You must deploy the EAR file from the Hub Server installation directory.

1. Log in to the WebSphere Server Administration Console.
2. Undeploy `siperian-mrm.ear`.
3. Deploy `siperian-mrm.ear`.

   The EAR file is in the following directory:

   `<MDM Hub installation directory>/hub/server`

   Configure the following deployment options:

   - In the **Preparing for the application installation** panel, enable the option for deployments to generate default bindings.
   - In the **Metadata for modules** panel, disable the `metadata-complete` attribute for the `siperian-ejb.jar` module to scan annotation-based metadata each time the module is read.

For more information, see the WebSphere Server documentation.

Configure Class Loaders

To configure class loaders for the Hub Server applications, use the WebSphere deployment manager.

1. Ensure that the class loaders for the Hub Server applications, `siperian-mrm.ear`, `provisioning-ear.ear`, and `entity360view-ear.ear` are configured to load classes with parent class loader last.

   If the class loaders are configured to load classes with parent class loader first, configure the class loaders for the application.

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.

   b. On the **Enterprise Applications** page, click one of the applications.

   c. On the page for configuring applications, click the **Class loading and update detection** link.

   d. On the **Class loader** configuration page, select the **Classes loaded with local class loader first (parent last)** class loader order option.

   e. Click **Apply**, and then click **OK**.
2. Configure class loaders for the web modules of the following application EAR files:

Application EAR File	Web Module	Class Loader Order
`siperian-mrm.ear`	`zds-gui.war`	Classes loaded with local class loader first (parent last)
`provisioning-ear.ear`	`provisioning.war`	Classes loaded with local class loader first (parent last)

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.

   b. On the **Enterprise Applications** page, click the name of the application EAR file.

c. On the page for configuring the application, click the **Manage Modues** link.

d. From the list of modules, click the link for the web module.

e. On the web module configuration page, select the class loader order.

f. Click **Apply**, and then click **OK**.

3. Restart WebSphere, and then start the Hub Server applications.

Verify and Configure Application Server Settings (Conditional)

Verify and configure application server settings based on the requirements of the MDM Hub environment.

The following table describes the configuration tasks that you can perform:

Configuration Task	Description
Editing the application server settings	Required if you run the postInstallSetup script during the installation and the script fails because of incorrect application server settings.
Configuring the Hub Server for a WebSphere cluster	Required if you installed the Hub Server in a WebSphere cluster.

Editing the Application Server Settings

If you run the `postInstallSetup` script during the installation and the script fails because of incorrect application server settings, edit the `build.properties` file. Also, if you want to change any application server settings, edit the file.

1. Open `build.properties` file is in the following directory:

   `<MDM Hub installation directory>/hub/server/bin`

2. Edit the application server settings and save the file.

After you edit the `build.properties` file, ensure that you run the `postInstallSetup` script to deploy the Hub Server applications.

Configuring the Hub Server for a WebSphere Multi-node or Cluster Environment

If you installed the Hub Server in a WebSphere multi-node or a cluster environment, configure the Hub Server for the WebSphere environment. To configure the Hub Server for a WebSphere environment, configure the Hub Server properties in the `cmxserver.properties` file.

For example, a WebSphere cluster or a multi-node environment has two servers that run on host1 and host2, and use RMI ports 2812 and 2813. You need to configure the WebSphere properties on both the servers.

1. On each server, open the `cmxserver.properties` file in the following directory:

   `<MDM Hub installation directory>/hub/server/resources`

2. Configure the following properties:

Property	Description
cluster.flag	Must be added manually. Specifies whether clustering is enabled. To enable clustering, set to `true`. Default is `false`.
cmx.appserver.hostname	Specifies the machine names of all the servers separated by a comma. For example, if the WebSphere cluster or the multi-node environment has two servers that run on host1 and host2, set the property to `cmx.appserver.hostname=host1,host2`.
cmx.appserver.rmi.port	Specifies the RMI port numbers that the servers use separated by a comma. For example, if the servers in the WebSphere cluster or the multi-node environment use RMI ports 2812 and 2813, set the property to `cmx.appserver.rmi.port=2812,2813`.

In the properties description, the host name and the port number of the first server are host1 and 2812. The host name and the port number of the second server are host2 and 2813.

Deploy the Hub Server Applications (Conditional)

You must deploy the Hub Server applications on the same machine on which you install the Hub Server.

The Hub Server applications must be able to find the Hub Server installation to which they belong. Therefore, do not transfer the EAR files for deployment on another machine. For example, you install the Hub Server on a test machine, and then deploy the applications on the production machine. The applications that you deploy on the production machine cannot find the installation on the test machine for information such as logging configuration.

You need to deploy the Hub Server applications in any of the following scenarios:

- The installation is in an application server multi-node environment or cluster environment.
- The installation completes, but the `postInstallSetup` script that you run during the installation fails.
- You skipped the `postInstallSetup` script during the installation.

If the installation is in an application server multi-node or a cluster environment, perform the following steps to complete the deployment:

1. Deploy the Hub Server applications on one node.
2. Copy all the files from the `certificates` directory on the node with the deployment to the `certificates` directory on all the other nodes. The `certificates` directory is in the following location:

   `<MDM Hub installation directory>/hub/server/resources`
3. Repackage the Hub Server EAR files on all the nodes that the certificates are copied to. The repackaging process updates the `hubConsole.jar` file with the correct certificates.
   a. Navigate to the following directory:
      `<MDM Hub installation directory>/hub/server/bin`
   b. Run the following command:

On UNIX.

```
./sip_ant.sh repackage
```

On Windows.

```
sip_ant.bat repackage
```

4. Deploy the Hub Server applications on the other nodes.

To deploy the Hub Server applications, use one of the following procedures described in the following table:

Procedure	Description
Use a script for deployment	You run the `postInstallSetup` script to deploy the Hub Server applications. Also, the script creates data sources and configures JMS message queues.
Manual deployment	You manually deploy the Hub Server applications. Also, you must manually create data sources and configure JMS message queues.

Use a Script to Deploy the Hub Server Applications (Conditional)

If you skipped the `postInstallSetup` script during the installation, run the script. The post-installation process deploys the Hub Server applications, creates data sources, and configures JMS message queues.

1. Open a command prompt, and change to the following directory:

   ```
   <MDM Hub installation directory>/hub/server
   ```

2. Run the `postInstallSetup` script.

   Note: If you did not install the ActiveVOS version that is bundled with the MDM Hub installer, do not include the ActiveVOS user names and passwords in the command.

   If security is not enabled on WebSphere, run the following command:

   On UNIX.

   ```
   ./postInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password>
   -Davos.username=<ActiveVOS Console username>
   -Davos.password=<ActiveVOS Console password>
   -Davos.jdbc.database.username=<ActiveVOS database username>
   -Davos.jdbc.database.password=<ActiveVOS database password>
   ```

   Note: If you include the exclamation mark (!) in your password, you must include a backslash before the exclamation mark. For example, if your password is !!cmx!!, enter the following password: \!\!cmx\!\!

   On Windows.

   ```
   postInstallSetup.bat
   -Ddatabase.password=<MDM Hub Master database password>
   -Davos.username=<ActiveVOS Console username>
   -Davos.password=<ActiveVOS Console password>
   -Davos.jdbc.database.username=<ActiveVOS database username>
   -Davos.jdbc.database.password=<ActiveVOS database password>
   ```

   If you enabled security on WebSphere, run the following command:

On UNIX.

```
postInstallSetup.sh
-Dwebsphere.password=<WebSphere password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

On Windows.

```
postInstallSetup.bat
-Dwebsphere.password=<WebSphere password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

The ActiveVOS Console credentials are the same credentials as the administrative user in the application server.

The ActiveVOS database credentials are the same credentials that were used to run the `create_bpm` script.

3. To enable scanning for annotation-based metadata of the `siperian-ejb.jar` module, use the WebSphere Server Administration Console to undeploy and deploy the EAR file siperian-mrm.ear.

   For more information, see "Redeploy the Hub Server EAR File" on page 55.

Manually Deploy the Hub Server Applications (Conditional)

If you skipped the `postInstallSetup` script during the installation or the `postInstallSetup` script fails, you can manually deploy the Hub Server applications. Ensure that you deploy the Hub Server applications from the Hub Server installation directory.

To deploy the Hub Server applications, perform the following tasks:

1. Creating data sources
2. Configuring JMS message queues
3. Repackaging the Hub Server applications
4. Deploying the Hub Server applications
5. Configuring class loaders
6. Configuring JMS message queues in the Hub Server
7. Configuring server resources for Informatica Data Director (IDD)

Step 1. Creating Data Sources

Before you manually deploy the Hub Server applications, create data sources. Also, if you want to configure multiple Process Servers or troubleshoot installation issues, create data sources.

1. Install the JDBC driver.
2. Create an MDM Hub Master Database data source.
3. Create an Operational Reference Store data source.

Step 1. Install the JDBC Driver

Before you create data sources for the MDM Hub Master Database and the Operational Reference Store (ORS), install the JDBC driver.

Contact IBM to get the supported version of the JDBC driver.

- Copy the JDBC driver to the following directory:

  ```
  <WebSphere installation directory>/lib
  ```

Step 2. Create an MDM Hub Master Database Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for the MDM Hub Master Database.

1. Start the WebSphere Application Server Administrative Console.
2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      ```
      <WebSphere installation directory>/lib
      ```
3. Create the security account that the MDM Hub Master Database data source will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.
   d. Click **New**, and specify the following properties:

Property	Description
Alias	Name of the MDM Hub Master Database.
User ID	User name to connect to the MDM Hub Master Database.
Password	Password to access the MDM Hub Master Database.

   e. Click **OK**.
4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.

      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.

c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

d. Click **Next**, and then click **Finish**.

5. Create the MDM Hub Master Database data source.
   a. Click the JDBC provider that you created.

      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.

      The **Data Sources** page appears.
   c. Click **New**.
   d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM Master Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-cmx_system-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/cmx_system`.

   e. Click **Next**, and then click **Finish**.

Step 3. Create an Operational Reference Store Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for each Operational Reference Store.

1. Start the WebSphere Application Server Administrative Console.
2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      `<WebSphere installation directory>/lib`

3. Create the security account that the Operational Reference Store will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.
   d. Click **New**, and set the following properties:

Property	Description
Alias	Name of the Operational Reference Store.
User ID	User name to connect to the Operational Reference Store.
Password	Password to access the Operational Reference Store.

   e. Click **OK**.

4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.

      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.
   c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

   d. Click **Next**, and then click **Finish**.

5. Create the Operational Reference Store data source.
   a. Click the JDBC provider that you created.

      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.

      The **Data Sources** page appears.
   c. Click **New**.

d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM ORS Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-<IBM Db2 host name>-<IBM Db2 database name>-<Operational Reference Store name>-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/<Operational Reference Store name>`.

e. Click **Next**, and then click **Finish**.

Step 2. Configuring JMS Message Queues

Before you manually deploy the Hub Server applications, configure JMS message queues. Also, to troubleshoot issues, you might need to manually configure JMS message queues. For example, if the automated queue creation process fails or the queues accidentally drop after installation, you need to manually configure message queues.

The Services Integration Framework (SIF) uses a message-driven bean on the JMS message queue to process incoming asynchronous SIF requests. Configure the message queue and the connection factory for the application server that you use for the MDM Hub implementation. When you configure a JMS message queue, you also create a connection factory.

To configure the JMS message queue, perform the following tasks:

1. Create a bus in the WebSphere Server.
2. Configure a queue connection factory.
3. Configure a JMS message queue for SIF requests.
4. Configure a JMS message queue for search. (Conditional)

Step 1. Create a Bus in the WebSphere Server

To create a JMS message queue, create a bus by using the WebSphere Console.

1. In the WebSphere Console, navigate to **Service integration** > **Buses**.
2. Create a bus with the name `SiperianBus` and save the change.
3. Navigate to SiperianBus that you created, and click **Bus Members**.
4. Add a new member to the **Server** list.
5. Choose the server on which the application is running, click **Next**, and select **Data Store**.
6. Click **Next** and ensure that the **Create default data source with generated JNDI** name is checked.
7. Click **Next**, and then click **Finish**.
8. Click **Save**.
9. Navigate to SiperianBus that you created, and click **Destination**.
10. Click **New** and choose Queue as destination type, and click **Next**.
11. Use the name `SiperianQueue` as the Identifier, and click **Next**.

12. Choose the bus member that you created, and click **Next**.
13. Click **Finish**, and then click **Save**.

Step 2. Configure a Queue Connection Factory

Create and configure a queue connection factory for the bus that you created.

1. In the WebSphere Console, navigate to **Resources** > **JMS** > **JMS Providers**.
2. Select a node from the list, such as **Node=<servername>Node01**.
3. Select the JMS provider that you want to use.
4. Click **Queue connection factories**.
5. Specify `siperian.mrm.jms.xaconnectionfactory` as the name and the JNDI name.
6. Select **SiperianBus** as the bus name for the queue connection factory.
7. Click **Save**.
8. Click **Queues**, and select the same node for the scope, such as **Node=<servername>Node01**.

Step 3. Configure JMS Queues

After you configure the bus and the connection factory, configure a system JMS message queue and a custom JMS message queue. Then, create an activation specification for the queues.

1. In the **Queues** page, click **New**.
2. Select a JMS provider and click **OK**.
3. In a clustered environment, you can set the **Scope** to one or more nodes.

   Tip: You might want to start by configuring the queue on a single node. Later, you can expand to multiple nodes.
4. Specify the following options for the JMS queues:

Option	System Queue Value	Custom Queue Value
Name	`siperian.sif.jms.queue`	`siperian.sif.test.jms.queue`
JNDI Name	`queue/siperian.sif.jms.queue`	`queue/siperian.sif.test.jms.queue`
Description	MDM JMS system queue	MDM JMS custom queue
Bus Name	SiperianBus	SiperianBus
Queue Name	SiperianQueue	SiperianQueue

5. Click **OK**.
6. Click **JMS activation specification** and select the scope from the list.
7. To configure an activation, click **New**.

8. Specify the following options:

Option	Value
Name	`SiperianActivation`
JNDI Name	`SiperianActivation`
Destination Type	Queue
Destination JNDI Name	`queue/siperian.sif.jms.queue`
Bus Name	SiperianBus

9. Click **OK**.

Step 4. Configure a JMS Queue for Search

The search feature requires a JMS queue to enable full-text search within Data Director. You do not need to activate the search JMS queue nor configure the queue in the Hub Console.

You must configure Elasticsearch before you configure a JMS queue for search.

1. In the **Queues** page, click **New**.
2. Select a JMS provider and click **OK**.
3. In a clustered environment, set the **Scope** to one or more nodes.

   Tip: You might want to configure the queue on a single node. Later, you can expand to multiple nodes.
4. Specify the following options:

Option	Value
Name	`informatica.mdm.sss.jms.queue`
JNDI Name	`queue/informatica.mdm.sss.jms.queue`
Description	Siperian JMS Queue for Search
Bus Name	SiperianBus
Queue Name	SiperianQueue
Delivery Mode	application

5. Click **OK**.

Step 3. Repackaging the Hub Server EAR Files

If you edit the `cmx.home` property in the `cmxserver.properties` file or if you installed in an application server cluster, repackage the Hub Server EAR files.

1. Create a directory named `EAR`.
   a. Navigate to the following directory:

      ```
      <MDM Hub installation directory>/hub/server/lib
      ```

   b. Run the following command:

      ```
      mkdir ear
      ```

2. If you have custom JAR files, copy each custom JAR file to the EAR directory that you created in the preceding step.

   To copy a custom JAR file to the EAR directory, run the following command:

   ```
   copy <location of custom JAR file>/< custom JAR file name>.jar ear
   ```

   You might require custom JAR files for custom user exits.

3. Repackage the EAR files.
   a. Navigate to the following directory:

      ```
      <MDM Hub installation directory>/hub/server/bin
      ```

   b. Run the following command:

      On UNIX. `./sip_ant.sh repackage`

      On Windows. `sip_ant.bat repackage`

Step 4. Deploying the Hub Server Application

You can manually deploy the Hub Server applications. Ensure that you deploy the Hub Server applications from the Hub Server installation directory.

1. If you have any existing deployments, use the WebSphere Server Administration Console to undeploy the following deployment files:

Deployment File Name	Description
siperian-mrm.ear	Required. The Hub Server application.
provisioning-ear.ear	Required. The Provisioning tool application.
entity360view-ear.ear	Optional. The Entity 360 framework.

2. Use the WebSphere administration console to deploy the files listed in the preceding step.

   The deployment files are in the following directory:

   ```
   <MDM Hub installation directory>/hub/server
   ```

   Configure the following deployment options:

   - In the **Preparing for the application installation** panel, enable the option for deployments to generate default bindings.
   - In the **Metadata for modules** panel, disable the `metadata-complete` attribute for the `siperian-ejb.jar` module to scan annotation-based metadata each time the module is read.

- If you deploy on cluster nodes, in the **Select installation options** panel, enable the option for deployments to distribute application.

3. If you deployed the Hub Server applications on cluster nodes, perform the following steps:
   a. Stop the cluster, deployment manager, and the node.
   b. Start the node, then the deployment manager, and then the cluster.

For more information, see the WebSphere Server documentation.

Step 5. Configuring Class Loaders

After you manually deploy the Hub Server applications, configure class loaders for each Hub Server application.

1. Configure class loaders for the following Hub Server applications: `siperian-mrm.ear`, `provisioning-ear.ear`, and `entity360view-ear.ear`
   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click one of the applications.
   c. On the page for configuring applications, click the **Class loading and update detection** link.
   d. On the **Class loader** configuration page, select the **Classes loaded with local class loader first (parent last)** class loader order option.
   e. Click **Apply**, and then click **OK**.
2. Configure class loaders for the web modules of the following application EAR files:

Application EAR File	Web Module	Class Loader Order
`siperian-mrm.ear`	`zds-gui.war`	Classes loaded with local class loader first (parent last)
`provisioning-ear.ear`	`provisioning.war`	Classes loaded with local class loader first (parent last)

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click the name of the application EAR file.
   c. On the page for configuring the application, click the **Manage Modues** link.
   d. From the list of modules, click the link for the web module.
   e. On the web module configuration page, select the class loader order.
   f. Click **Apply**, and then click **OK**.
3. Restart WebSphere, and then start the Hub Server applications.

Step 6. Configuring JMS Message Queues on the Hub Server

After you manually deploy the Hub Server applications, configure JMS message queues on the Hub Server.

To configure the JMS message queue on the Hub Server, perform the following tasks:

1. Start the Hub Console.
2. Add a message queue server.
3. Add a message queue.

Step 1. Start the Hub Console

To access the MDM Hub, start the Hub Console.

1. Open a browser window and enter the following URL:

   ```
   http://<MDM Hub host>:<port number>/cmx/
   ```

   The Hub Console launch page appears.

2. Enter your user name and password, and then click **Download**.

   The MDM Hub application JAR file that is required to launch the Hub Console downloads.

   Note: If you cannot download the MDM Hub application JAR file, contact your MDM administrator. The administrator can distribute the JAR file from the following directory: `<MDM Hub installation directory>/hub/server/resources/hub`

3. Run the application JAR file.

   Note: If you do not have an SSL certificate on the client machine, and want to access the Hub Console through an HTTPS connection, you must install it. To do so, you can use the following procedure:

   - Import the certificate to the Java keystore of your local client machine, by running the following command:

     ```
     keytool -import -trustcacerts -alias <certificate alias name> -file <certificate alias file> -keystore <local java cacerts keystore location>
     ```

   - Pass the location and password of the truststore file that contains the certificate, by running the following command:

     ```
     java -Djavax.net.ssl.trustStore=<truststore file location> -Djavax.net.ssl.trustStorePassword=<truststore_password> -jar hubConsole.jar
     ```

     Use a separate truststore that contains all the custom trust certificates, rather than the default cacert file. Obtain the certificates from the team that maintains the application server. The server might use either a self signed certificate or security certificate. Download the .jar file only if there is a version change on the server. Each time you download the .jar file, launch it using the same command.

4. To specify the maximum memory allocation pool for the application, run the following command:

   ```
   java -Xmx<n>G -jar hubConsole.jar
   ```

   Where <n> is the maximum memory allocation in GB.

   The **Informatica MDM Hub Login** dialog box appears.

5. Enter your user name and password.

6. If you want to connect to a specific Hub Server node or if you use a load balancer or a reverse proxy server, override the pre-configured connection parameters in the Connection Property field.

   Enter the parameters in the following format:

   ```
   <host name>:<port name>
   ```

   Where, host name and port name are either the host name and port name of the Hub Server or that of the load balancer or the reverse proxy server that you use.

7. Click **OK**.

   The **Change database** dialog box appears.

8. Select the target database.

   The target database is the MDM Hub Master Database.

9. Select a language from the list, and click **Connect.**

   The Hub Console user interface appears in the language that you select. If you need to change the language in which the Hub Console user interface appears, restart the Hub Console with the language of your choice.

Step 2. Add a Message Queue Server

Before you add a message queue, you must add a message queue server to the MDM Hub implementation.

1. In the Hub Console, click **Message Queues** under the Configuration workbench.
2. Click **Write Lock** > **Acquire Lock**.
3. Right-click the middle pane of the Message Queues tool, and then click **Add Message Queue Server**.

   The **Add Message Queue Server** dialog box appears.
4. Enter the message queue server details.

   The following table describes the fields that you use to configure the JMS message queue server:

Field Name	Value
Connection Factory Name	Name of the connection factory. Specify `siperian.mrm.jms.xaconnectionfactory`.
Display Name	Name of the message queue server that must appear in the Hub Console. Specify `siperian.mrm.jms.xaconnectionfactory`.

5. Click **OK**.

   The message queue server is added.

Step 3. Add a Message Queue

You can add the custom message queue to a message queue server.

1. In the Hub Console, click **Message Queues** under the Configuration workbench.
2. Click **Write Lock** > **Acquire Lock**.
3. Right-click the message queue server in the middle pane of the Message Queues tool, and then click **Add Message Queue**.

   The **Add Message Queue** dialog box appears.
4. Enter JMS message queue details.

   The following table describes the JMS message queue fields:

Field Name	Value
Queue Name	Specify the name of the message queue.
Display Name	Specify the name of the message queue that must appear in the Hub Console.

5. Click **OK**.

   The message queue is added to the message queue server.
6. In the right pane, select the **Use with message triggers** option.
7. Click **Test**.

   The result of the message queue test appears.

Step 7. Configuring Server Resources for Informatica Data Director

If you want to use Informatica Data Director (IDD), configure the JNDI URL resource.

1. On the WebSphere Server Administration Console, click **Resources** > **URLs**.
2. To configure the JNDI URL resource, set the following properties:

Property	Value
Scope	Specify the scope of the Hub Server.
Name	Hub server home dir
JNDI name	url/hubserver/home
Specification	file:///<Hub Server installation directory>

Configure Metadata Caching (Optional)

Metadata caches manage items such as data objects, repository objects, and search tokens. The MDM Hub uses Infinispan for metadata caching. Infinispan is installed with the Hub Server. For the caches that the Hub Server uses, the Infinispan configuration file contains default attribute values.

Run the MDM Hub with the default attribute values for the caches. If you experience performance issues, you can fine-tune the attribute values to better suit your environment.

The following table summarizes the default attribute values:

Infinispan Element and Attribute	Default Value	Description
`locking acquire-timeout`	`60000`	Maximum time during which the Hub Server can try to acquire a lock.
`transaction stop-timeout`	`30000`	When a cache stops, this attribute sets the maximum time that Infinispan waits while the Hub Server finishes remote and local transactions.
`transport cluster`	`infinispan-cluster`	Name for the underlying group communication cluster.
`transport stack`	`UDP`	Type of configuration: UDP or TCP. The configurations are defined in the `jgroups-udp.xml` file and the `jgroups-tcp.xml` file.
`transport node-name`	`$node$`	Name of the current node. The Hub Server sets this attribute. The node-name defaults to a combination of the host name and a random number. The number differentiates multiple nodes on the same host.
`transport machine`	`$machine$`	ID of the machine where the node runs. The Hub Server sets this attribute.

Infinispan Element and Attribute	Default Value	Description
`expiration lifespan`	--	Maximum lifespan of a cache entry in milliseconds. When a cache entry exceeds its lifespan, the entry expires within the cluster. If you need to optimize performance, increase the lifespan for the following caches: DISABLE_WHEN_LOCK, DATA_OBJECTS, and REPOS_OBJECTS. For example, you can increase the lifespan from one hour (3600000) to one day (86400000). Each cache has its own default value for this attribute. To find the default values, open the `inifinspanConfig.xml` file.
`expiration interval`	--	Maximum interval for checking the lifespan. If you need to optimize performance, increase the interval for the following caches: DISABLE_WHEN_LOCK, DATA_OBJECTS, and REPOS_OBJECTS. For example, you can increase the interval from five seconds (5000) to five minutes (300000). Each cache has its own default value for this attribute. To find the default values, open the `inifinspanConfig.xml` file.

Editing Infinispan Attributes

To configure metadata caching attributes, edit the `infinispanConfig.xml` file for the Hub Server. For help with the Infinispan configuration, see the Infinispan documentation.

Note: The Process Server also has an Infinispan configuration file. The default attribute values should be sufficient, however if you notice issues with the performance of the Process Server, you can fine-tune the attribute values.

1. Navigate to the following directory: `<MDM Hub installation directory>/hub/server/resources`
2. Make a backup copy of the following file: `infinispanConfig.xml`
3. Open the `infinispanConfig.xml` file and find the Infinispan version number, which appears in the `xsi:schemaLocation` attribute.
4. Review the documentation for the Infinispan version.

   Note: In the following URLs, substitute the version number wherever the path contains #.#.

   - To view the configuration schema, go to the URL that is contained in the `xsi:schemaLocation` attribute in the file.
   - To learn about the attributes, go to `https://docs.jboss.org/infinispan/<#.#.x>/configdocs/`
   - To learn about Infinispan, go to `http://infinispan.org/docs/<#.#.x>/` and select the "Frequently Asked Questions" link.
5. Edit the file and save it.

Start the Hub Console

To access the MDM Hub, start the Hub Console by using an HTTP or HTTPS connection.

Before you start the Hub Console, ensure that you have the following information:

- Host name and port number for the URL
- User name and password
- An SSL certificate on the client machine, if you want to access the Hub Console through an HTTPS connection

1. Open a browser window and enter the following URL:

   ```
   http://<MDM Hub host>:<port number>/cmx/
   ```

   The Hub Console launch page appears.

2. Enter your user name and password, and then click **Download**.

   The MDM Hub application JAR file that is required to launch the Hub Console downloads.

   Note: If you cannot download the MDM Hub application JAR file, contact your MDM administrator. The administrator can distribute the JAR file from the following directory: `<MDM Hub installation directory>/hub/server/resources/hub`

3. Run the application JAR file.

   Note: If you do not have an SSL certificate on the client machine, and want to access the Hub Console through an HTTPS connection, you must install it. To do so, you can use the following procedure:

   - Import the certificate to the Java keystore of your local client machine, by running the following command:

     ```
     keytool -import -trustcacerts -alias <certificate alias name> -file <certificate alias file> -keystore <local java cacerts keystore location>
     ```

   - Pass the location and password of the truststore file that contains the certificate, by running the following command:

     ```
     java -Djavax.net.ssl.trustStore=<truststore file location> -
     Djavax.net.ssl.trustStorePassword=<truststore_password> -jar hubConsole.jar
     ```

     Use a separate truststore that contains all the custom trust certificates, rather than the default cacert file. Obtain the certificates from the team that maintains the application server. The server might use either a self signed certificate or security certificate. Download the .jar file only if there is a version change on the server. Each time you download the .jar file, launch it using the same command.

4. To specify the maximum memory allocation pool for the application, run the following command:

   ```
   java -Xmx<n>G -jar hubConsole.jar
   ```

   Where <n> is the maximum memory allocation in GB.

   The **Informatica MDM Hub Login** dialog box appears.

5. Enter your user name and password.

6. If you want to connect to a specific Hub Server node or if you use a load balancer or a reverse proxy server, override the pre-configured connection parameters in the Connection Property field.

   Enter the parameters in the following format:

   ```
   <host name>:<port name>
   ```

   Where, host name and port name are either the host name and port name of the Hub Server or that of the load balancer or the reverse proxy server that you use.

7. Click **OK**.

   The **Change database** dialog box appears.

8. Select the target database.

   The target database is the MDM Hub Master Database.

9. Select a language from the list, and click **Connect.**

   The Hub Console user interface appears in the language that you select. If you need to change the language in which the Hub Console user interface appears, restart the Hub Console with the language of your choice.

Register an Operational Reference Store

After you create an Operational Reference Store, you must register it through the Hub Console. Register an Operational Reference Store with a single MDM Hub Master Database.

1. Start the Hub Console.

   The **Change database** dialog box appears.

2. Select **MDM Hub Master Database**, and click **Connect**.
3. Under the **Configuration** workbench, click the **Databases** tool.
4. From the **Write Lock** menu, click **Acquire Lock**.
5. In the Databases pane, click the **Register database** button.

   The **Informatica MDM Hub Connection Wizard** appears.

6. Select the IBM Db2 database type option, and click **Next**.
7. Configure connection properties for the database.
   a. Specify the connection properties, and click **Next**.

      The following table lists and describes the connection properties:

Property	Description
Database Display Name	Name for the Operational Reference Store that must appear in the Hub Console.
Machine Identifier	Prefix given to keys to uniquely identify records from the Hub Store instance.
Database server name	IP address or name of the server that hosts the IBM Db2 database.
Database name	Name of the database that you create.
Port	Port of the IBM Db2 database. The default is 50000.
Schema Name	Name of the Operational Reference Store.

Property	Description
User name	User name for the Operational Reference Store. By default, this is the user name that you specify in the script that you use to create the Operational Reference Store. This user owns all the Operational Reference Store database objects in the Hub Store. **Note:** If you created a proxy user, use the proxy user name instead of the Operational Reference Store user name.
Password	Password associated with the user name for the Operational Reference Store. For IBM Db2, the password is case sensitive. By default, this is the password that you specify when you create the Operational Reference Store.
DDM connection URL	Optional. URL to connect to the Dynamic Data Masking application. The URL is similar to the URL that you use to connect to the database, except that the Dynamic Data Masking application URL uses the Dynamic Data Masking host name and port number.

Note: The **Schema Name** and the **User Name** are both the names of the Operational Reference Store that you specified when you created the Operational Reference Store. If you need this information, consult your database administrator.

The **Summary** page appears.

b. Review the summary, and specify additional connection properties.

The following table lists additional connection properties that you can configure:

Property	Description
Connection URL	Connect URL. The Connection Wizard generates the connect URL by default. The following example shows the format of the connect URL: `jdbc:db2://database_host:port/db_name`
Create data source after registration	Select to create the data source on the application server after registration. Select to create the data source on the application server after registration. If you do not select the option, you must manually configure the data source. **Note:** In application server cluster environments, manually create the data sources and connection pools for the Operational Reference Stores.

8. Click **Finish**.

The **Registering Database** dialog box appears.

9. Click **OK**.

The MDM Hub registers the Operational Reference Store.

10. Select the Operational Reference Store that you registered, and click the **Test database connection** button.

You must restart the application server before you test the database connection.

The Test Database dialog box displays the result of the database connection test.

11. Click **OK**.

The Operational Reference Store is registered, and the connection to the database is tested.

CHAPTER 6

Process Server Installation

This chapter includes the following topics:

Installing the Process Server

You can install the Process Server using an installation wizard, a silent installation script, or, on UNIX systems, a command line script. If you complete the pre-installation tasks and collect the information you need before you start the installer, the installation process takes about 15 minutes.

STOP! Did you complete the pre-installation tasks?
The installation will fail if you do not complete the pre-installation tasks before you run the installer.

Installation Readiness Checklist

- ❑ Created an MDM implementation plan.
- ❑ Verified that your servers meet the system requirements.
- ❑ Verified that your operating system and software versions are supported.
- ❑ Reviewed the known limitations for your operating system and software versions.
- ❑ Installed and configured a supported version of an application server.
- ❑ Installed and configured a supported version of a database management system.
- ❑ Performed the pre-installation configuration tasks for your environment.
- ❑ Saved the MDM license file in an accessible location.

If you missed a task, go back to the preceding chapters for help in completing the task.

When you are ready to proceed, perform the following steps:

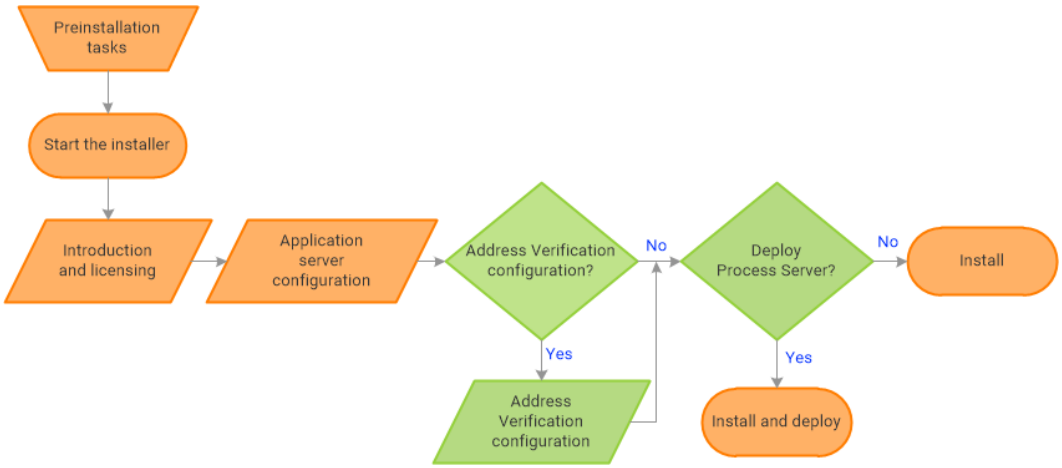
1. Review the installation workflow.

2. Collect all the values to enter during the installation.
3. Begin the installation by using the installation wizard or the command line, or in the silent mode.

Review the Installer Workflow

Whether you use the installation wizard, the command line prompts, or the silent installation script, the installer goes through the steps in the same order. You must follow the installation workflow keeping the decision points in mind.

The following diagram walks you through the steps in the Process Server installer workflow:



Take a moment to identify the decision points in the installation process. Consult your implementation plan to understand which paths to take at each of the following decision points:

1. **Configure Address Verification?** Administrators use Informatica Address Verification to interpret, process, and format the addresses included in records. Authorized users can validate, correct and certify addresses that are included in records before the master data is updated.

   Note: If you already installed a supported version of Informatica Address Verification, configure the Configuration and Parameters file locations and the type of address processing that you want to perform. Otherwise, use the default file locations and specify the type of address verification that you want to perform.
2. **Deploy the Process Server?** Allow the installer to run the `postInstallSetup` script. Among other important tasks, the script deploys the Process Server to the application server. Alternatively, you can run the `postInstallSetup` script after you exit the installer.

Collect the Installation Values

Before you run the installer, collect the installation values. The installer will prompt you for information about your application server, database management system, and other components. The best practice is to print out these tables and add the values for your environment.

Application Server: IBM WebSphere

Use the following table to collect the WebSphere details that are required for the installation:

Property	Description	Default value	Server instance 1 value
WebSphere Installation Directory	The location where WebSphere is installed.	-	
Is WebSphere security enabled?	If WebSphere security is enabled, select **Yes** and provide the user name and password.	No	
Server Name	Name of the WebSphere application server on which you want to deploy the Process Server. In a clustered environment, enter one of the cluster server names and its corresponding Bootstrap port and SOAP connector port information.	-	
Bootstrap Port	Bootstrap port number that is used by the server that you specify. **Tip:** To find the port information, go to the WebSphere administrative console, and then click **Application servers** > **<server name>** > **Ports**.	2809	
SOAP Connector Port	SOAP connector port number that is used by the server that you specify. **Tip:** To find the port information, go to the WebSphere administrative console, and then click **Application servers** > **<server name>** > **Ports**.	8880	
Profile Name	Name of the WebSphere profile that contains the WebSphere application server on which you want to deploy the Process Server.	-	
User Name	Name of a WebSphere user that has administrative privileges.	admin	
Password	Password of the WebSphere administrative user.	-	

In a clustered environment, add details for the other WebSphere instances to the following table:

Property	Server instance 2 value	Server instance 3 value	Server instance 4 value
WebSphere Installation Directory			
Is WebSphere security enabled?			
Server Name			
Bootstrap Port			

Property	Server instance 2 value	Server instance 3 value	Server instance 4 value
SOAP Connector Port			
Profile Name			
User Name			
Password			

Informatica Address Verification

Note: Informatica Address Verification was previously called AddressDoctor.

If you plan to install Informatica Address Verification, record the following properties:

Property	Description	Default values	Server instance 1 value
Configuration File	The location of the Informatica Address Verification configuration file `SetConfig.xml`. You use the file for general configurations, such as reference address database, unlock code for Informatica Address Verification, and memory settings. If you already installed Address Verification, use the location of your configuration file. Otherwise, use the default file location	<MDM Hub installation directory>\hub\cleanse\resources\AddressDoctor\5\SetConfig.xml	
Parameters File	The location of the Informatica Address Verification parameters file `Parameters.xml`. You use the file to configure how Informatica Address Verification interprets, processes, and formats the addresses. If you already installed Address Verification, use the location of your parameters file. Otherwise, use the default file location	<MDM Hub installation directory>\hub\cleanse\resources\AddressDoctor\5\Parameters.xml	
Correction Type	The type of address processing that you want to perform. Use one of the following correction types: - PARAMETERS_DEFAULT. Default correction type. Indicates the use of the correction type defined in the `Parameters.xml` file. - PARSE_ONLY. Parses and assigns address elements to the appropriate fields. - CORRECT_ONLY. Validates addresses against the postal data and corrects the addresses. - CERTIFY_ONLY. Verifies addresses in accordance with the postal certifications to meet country-specific postal authority requirements. - CORRECT_THEN_CERTIFY. Validates addresses against the postal data and corrects the addresses. Then verifies addresses in accordance with postal certifications to meet country-specific postal authority requirements. - TRY_CERTIFY_THEN_CORRECT. Verifies addresses in accordance with postal certifications to meet country-specific postal authority requirements. If the address verifications fail, the process validates the addresses against the postal data and corrects the addresses.	PARAMETERS_DEFAULT	

Product Usage Toolkit

The product usage toolkit sends information about your MDM environment to Informatica. The information is used by Informatica Global Customer Support to help troubleshoot and provide recommendations for your environment. If you do not want the toolkit to send any information to Informatica, you can disable the toolkit after you install MDM.

Use the following table to collect the details that are required for installing the product usage toolkit:

Property	Description	Default value	Installation value
Industry	Type of industry that best matches your organization's business.	-	
Environment	Type of environment that you are installing in. If you install from the command line, enter one of the following numbers: - 1. Production environment - 2. Test or QA environment - 3. Development environment	-	
Does your network have a proxy server?	If yes, provide details about the proxy server.	No	
Host	Name or IP address of the proxy server.	-	
Port	Port number used by the proxy server.	-	
Domain Name	If your proxy server is part of a domain, the name of the domain.	-	
User Name	If you use a secure proxy server, the name of a user that can access MDM.	-	
Password	Password of the user.	-	

Install the Process Server from the Installation Wizard

Use the installation wizard when you want to install the Process Server in graphical mode. The installation wizard guides you through the installation.

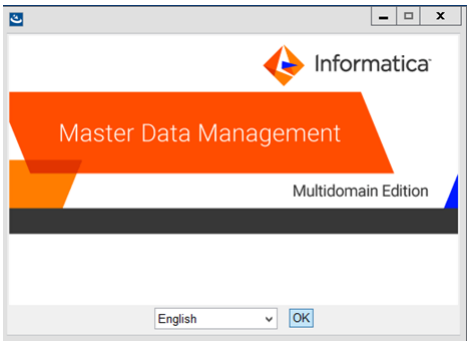
You must use the same user name to install the Hub Server and the Process Server.

1. Start the application server.
2. Navigate to the following directory:

   ```
   <MDM Hub distribution directory>/<operating system name>/mrmcleanse
   ```

3. Start the installer by performing the task for your operating system:
   - **UNIX**. From the command line, run `./hub_cleanse_install.bin`.
   - **Windows**. From the File Explorer, double-click `hub_cleanse_install.exe`.

   The Process Server installation wizard starts.



4. Choose a language and click **OK**.

   The **Introduction** window appears.
5. Follow the online instructions. When prompted, enter the installation values that you collected.
6. At the end of the installation, in the **Configuration Summary** window, review the options that you selected.
7. If you need to make changes, go back to the appropriate window by clicking **Previous**. When you are done, click **Next** to return to the final window.
8. Click **Install**.
9. **Next step**: The next step depends on whether you chose to deploy the Process Server from the installer.
   - If you chose to deploy the Process Server from the installer, you do not need to deploy the Process Server as part of the post-installation tasks.
   - If you chose to deploy the Process Server later, you must deploy the Process Server as part of the post-installation tasks.

Install the Process Server from the Command Line (UNIX Only)

On UNIX, you can install the Process Server from the command line. Run the script to start the command line installation.

1. Start the application server.
2. From the command line, navigate to the following directory:

   `<MDM Hub distribution directory>/<operating system name>/mrmcleanse`
3. Run the following command:

   `./hub_cleanse_install.bin -i console`

   The Process Server installation prompts appear.
4. Enter the installation values that you collected.

To use the default value shown in brackets, press **Enter**.

5. **Next step**: After the installation completes, the next step depends on whether you chose to deploy the Process Server.
   - If you chose to deploy the Process Server from the installer, you do not need to deploy the Process Server as part of the post-installation tasks.
   - If you chose to deploy the Process Server later, you must deploy the Process Server as part of the post-installation tasks.

Install the Process Server Silently

You can install the Process Server in silent mode. Before you start the silent installation, ensure that you configured the silent installation properties file.

1. Start the application server.
2. Copy the silent installation properties file to the target environment.
3. In the target environment, run the command for your operating system:
   - **UNIX**. `./hub_install.bin -f <absolute path to edited installer properties file>`
   - **Windows**. `hub_install.exe -f <absolute path to edited installer properties file>`

   The silent installer runs in the background. The process can take a while.
4. If you chose to have the installer deploy the Process Server, check `postinstallSetup.log` to verify whether the installation was successful.

   The log file is in the following directory:

   `<MDM Hub installation directory>/hub/server/logs`
5. **Next step**: After the installation completes, the next step depends on whether you chose to deploy the Process Server.
   - If you chose to deploy the Process Server from the installer, you do not need to deploy the Process Server as part of the post-installation tasks.
   - If you chose to deploy the Process Server later, you must deploy the Process Server as part of the post-installation tasks.

Install the Process Server on Nodes in the Cluster

In application server cluster environments, install the Process Server on all nodes of the cluster to which you must deploy the Process Server application. Complete the installation on one node of a cluster before you proceed to install on another node of a cluster.

Ensure that the directory structure of the Process Server installation is the same on all the nodes.

1. To start the WebServer cluster, perform the following steps:
   a. Start the WebSphere deployment manager.
   b. Start the nodes of the WebServer cluster on which you want to install the Process Server.

c. Start the WebServer cluster.

2. Navigate to the following directory:

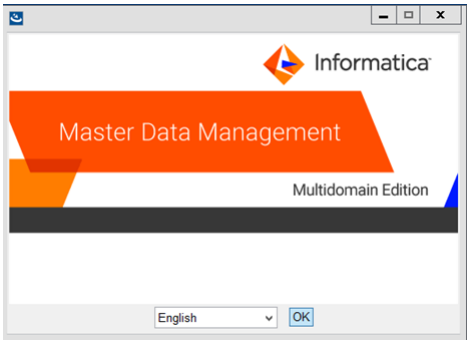
   `<MDM Hub distribution directory>/<operating system name>/mrmcleanse`

3. To start the Process Server installer on a cluster node, run the command for your operating system:

   UNIX. `./hub_cleanse_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

   Windows. `hub_cleanse_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

   The Process Server installation wizard starts.



4. Follow the online instructions. When prompted, enter the installation values that you collected.
5. **Next step**: After the installation completes, deploy the Process Server manually on all the nodes that have the installation.

   Note: If you performed an automatic deployment for the primary node, you do not need to manually deploy on the primary node.

CHAPTER 7

Process Server Post-Installation Tasks

This chapter includes the following topics:

Copy the Installation Log Files

The installation log files are useful for troubleshooting the Process Server installation process. Copy the log files to the installation documentation directory. Informatica Global Customer Support might request copies of the log files if you contact them regarding installation issues.

The following table describes the different types of installation log files:

Log File Type	Description
Installation log	- File name. `Informatica_MDM_Cleanse_Match_Server_Install_<timestamp>.xml` - Location. `<MDM Hub installation directory>/hub/cleanse/UninstallerData/Logs` - Contents. Directories created, names of the files installed and commands run, and status for each installed file.
Installation prerequisites log	- File name. `installPrereq.log` - Location. `<MDM Hub installation directory>/hub/cleanse/Logs` - Contents. Logs of prerequisite checks performed by the installer.

Log File Type	Description
Debug log	- File name. `infamdm_installer_debug.txt` - Location. `<MDM Hub installation directory>/hub/cleanse/` - Contents. Detailed information about the choices that are made during installation and the actions performed by the installer.
Post-installation setup log	- File name. `postInstallSetup.log` - Location. `<MDM Hub installation directory>/hub/cleanse/logs` - Contents. Summary of actions performed by the installer during the post-installation process and the errors in the post-installation process.
Process Server log	- File name. `cmxserver.log` - Location. `<MDM Hub installation directory>/hub/cleanse/logs` - Contents. Summary of the Process Server operations.
WebSphere logs	- File names. `startServer.log`, `stopServer.log`, `SystemErr.log`, and `SystemOut.log` - Location. `<WebSphere installation directory>/profiles/AppSrv01/logs/<server name>` - Contents. Contains information about server start and stop, and performance.

Verify the Version and Build Number

Ensure that the correct version and build number of the Process Server is installed.

1. Open a command prompt, and navigate to the following directory: `<MDM Hub installation directory>/hub/cleanse/bin`
2. To verify the Process Server version and build number, run the following command:

   On UNIX. `versionInfo.sh`

   On Windows. `versionInfo.bat`

   Note: For AIX systems, change the `versionInfo.sh` script to run Java from the `<Java home>/jre/bin` directory.

Configuring the Process Server for a WebSphere Multi-node or Cluster Environment

If you installed the Process Server in a WebSphere multi-node or a cluster environment, configure the Process Server for the WebSphere environment. To configure the Hub Server for a WebSphere environment, add the `cluster.flag` property in the `cmxcleanse.properties` file.

1. Open the `cmxcleanse.properties` file in the following directory:

   `<MDM Hub installation directory>/hub/cleanse/resources`
2. Add the `cluster.flag` property.

   The property specifies whether clustering is enabled. To enable clustering, set to `true`. Default is `false`.

Redeploy the Process Server EAR File

After you run the `postInstallSetup` script either manually or as part of the Process Server installation, use the WebSphere Server Administration Console to undeploy and deploy the Process Server EAR file `siperian-mrm-cleanse.ear`. You must deploy the EAR file from the Process Server installation directory.

1. Log in to the WebSphere Server Administration Console.
2. Undeploy `siperian-mrm-cleanse.ear`.
3. Deploy the `siperian-mrm-cleanse.ear` file.

   The EAR file is in the following directory:

   `<MDM Hub installation directory>/hub/cleanse`

   Configure the following deployment options:

   - In the **Preparing for the application installation** panel, enable the option for deployments to generate default bindings.
   - In the **Metadata for modules** panel, disable the `metadata-complete` attribute for the `siperian-cleanse-ejb.jar` module to scan annotation-based metadata each time the module is read.

For more information about deploying applications, see the WebSphere Server documentation.

Configure Class Loaders

To configure class loaders for the Process Server application, use the WebSphere deployment manager.

1. Ensure that the class loaders for the Process Server application `siperian-mrm-cleanse.ear` are configured to load classes with parent class loader last.

   If the class loaders are configured to load classes with parent class loader first, configure the class loaders for the application.

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click one of the applications.
   c. On the page for configuring applications, click the **Class loading and update detection** link.
   d. On the **Class loader** configuration page, select the **Classes loaded with local class loader first (parent last)** class loader order option.
   e. Click **Apply**, and then click **OK**.
2. Configure class loaders for the web modules of the following application EAR file:

Application EAR File	Web Module	Class Loader Order
`siperian-mrm-cleanse.ear`	`siperian-mrm-cleanse.war`	Classes loaded with local class loader first (parent last)

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click the name of the application EAR file.
   c. On the page for configuring the application, click the **Manage Modues** link.

d. From the list of modules, click the link for the web module.

e. On the web module configuration page, select the class loader order.

f. Click **Apply**, and then click **OK**.

3. Restart WebSphere, and then start the Process Server application.

Deploy the Process Server Application (Conditional)

If you have a scenario that requires deployment of the Process Server application, deploy the Process Server application.

You need to deploy the Process Server application in any of the following scenarios:

- The installation is in an application server multi-node environment or cluster environment.
- The installation completes, but the `postInstallSetup` script that you run during the installation fails.
- You skipped the `postInstallSetup` script during the installation.

Perform the following steps to deploy the Process Server application:

1. If the Process Server is not installed on the same application server instance as the Hub Server, create data sources.
2. Deploy the Process Server application `siperian-mrm-cleanse.ear`.
3. Configure class loaders.

Step 1. Creating Data Sources (Conditional)

If the Process Server is not deployed on the same application server instance as the Hub Server, configure the data sources for the application server.

1. Install the JDBC driver.
2. Create an MDM Hub Master Database data source.
3. Create an Operational Reference Store data source.

Step 1. Install the JDBC Driver

Before you create data sources for the MDM Hub Master Database and the Operational Reference Store (ORS), install the JDBC driver.

Contact IBM to get the supported version of the JDBC driver.

▶ Copy the JDBC driver to the following directory:

```
<WebSphere installation directory>/lib
```

Step 2. Create an MDM Hub Master Database Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for the MDM Hub Master Database.

1. Start the WebSphere Application Server Administrative Console.

2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      ```
      <WebSphere installation directory>/lib
      ```
3. Create the security account that the MDM Hub Master Database data source will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.
   d. Click **New**, and specify the following properties:

Property	Description
Alias	Name of the MDM Hub Master Database.
User ID	User name to connect to the MDM Hub Master Database.
Password	Password to access the MDM Hub Master Database.

   e. Click **OK**.
4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.

      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.
   c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

   d. Click **Next**, and then click **Finish**.
5. Create the MDM Hub Master Database data source.
   a. Click the JDBC provider that you created.

      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.

      The **Data Sources** page appears.

c. Click **New**.

d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM Master Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-cmx_system-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/cmx_system`.

e. Click **Next**, and then click **Finish**.

Step 3. Create an Operational Reference Store Data Source

After you install the JDBC driver, on the Process Server machine, create a data source for each Operational Reference Store.

1. Start the WebSphere Application Server Administrative Console.
2. Specify the location of the driver libraries.
   a. Expand **Environment** in the console navigation tree.
   b. Click the **WebSphere Variables** link.
   c. Update the JDBC variable to point to the following JDBC driver directory:

      `<WebSphere installation directory>/lib`
3. Create the security account that the Operational Reference Store will use.
   a. Expand **Security** in the console navigation tree.
   b. Click the **Secure administration, applications, and infrastructure** link.
   c. Under **Authentication**, expand **Java Authentication and Authorization Service**, and click **J2C Authentication Data**.
   d. Click **New**, and set the following properties:

Property	Description
Alias	Name of the Operational Reference Store.
User ID	User name to connect to the Operational Reference Store.
Password	Password to access the Operational Reference Store.

   e. Click **OK**.

4. Create the JDBC Provider.
   a. Expand **Resources** > **JDBC**, and then click **JDBC Providers**.

      The **JDBC Provider** page appears.
   b. Select the scope for applications to use the data source.
   c. Click **New**, and specify the following properties:

Property	Description
Database type	Type of database.
Provider type	Type of JDBC provider.
Implementation type	Data source implementation type.
Name	Name of the JDBC provider.

   d. Click **Next**, and then click **Finish**.

5. Create the Operational Reference Store data source.
   a. Click the JDBC provider that you created.

      The **Configuration** page appears.
   b. Under **Additional Properties**, click **Data sources**.

      The **Data Sources** page appears.
   c. Click **New**.
   d. Specify the following data source properties:

Property	Description
Name	Data source name. Specify `MDM ORS Data Source`.
JNDI Name	JNDI path to where the JDBC data source will be bound. Specify `jdbc/siperian-<IBM Db2 host name>-<IBM Db2 database name>-<Operational Reference Store name>-ds`. **Note:** The JNDI name must be in lower case.
Component-managed Authentication Alias	Authentication alias of the Master Database data source. Select `<host name>/<Operational Reference Store name>`.

   e. Click **Next**, and then click **Finish**.

Step 2. Deploying the Process Server Application (Conditional)

If the installation is in an application server multi-node environment or cluster environment, or the `postInstallSetup` script was skipped or fails, deploy the Process Server application.

Deploy the Process Server application on the same machine on which you installed the Process Server. The Process Server application must be able to find the Process Server installation associated with it. Therefore, do not copy the application EAR file for deployment on another machine. For example, you install the Process

Server on a test machine, and then deploy the application on the production machine. The application that you deploy on the production machine cannot find the installation on the test machine.

Deploy the Process Server application by using one of the following procedures:

Using a script for deployment

You run the `postInstallSetup` script to deploy the Process Server application.

Manual deployment

You manually deploy the Process Server application.

Using a Script for Deploying the Process Server Application (Conditional)

You can run the `PostInstallSetup` script to deploy the Process Server application.

Important: If the installation is in an application server multi-node or a cluster environment, first deploy the Process Server application on one node, and then deploy the Process Server application on the other nodes. Ensure that you deploy the Process Server application on the same machine on which you installed the Process Server.

1. Open a command prompt, and change to the following directory:

   `<MDM Hub installation directory>/hub/cleanse`

2. Run the `PostInstallSetup` script.

   On UNIX. `./postInstallSetup.sh`

   Note: If you enabled security on WebSphere, run `postInstallSetup.sh -Dwebsphere.password=<WebSpherePassword>`

   On Windows. `postInstallSetup.bat`

   Note: If you enabled security on WebSphere, run `postInstallSetup.bat -Dwebsphere.password=<WebSphere Password>`

3. To enable scanning for annotation-based metadata of the `siperian-cleanse-ejb.jar` module, use the WebSphere Server Administration Console to undeploy and deploy the EAR file `siperian-mrm-cleanse.ear`.

   For more information, see "Redeploy the Process Server EAR File" on page 86.

Manually Deploying the Process Server Application (Conditional)

You can manually deploy the Process Server application. You must deploy the Process Server application from the Process Server installation directory.

1. If you have any existing deployment, use the WebSphere Server Administration Console to undeploy `siperian-mrm-cleanse.ear`.
2. Use the WebSphere Server Administration Console to deploy the `siperian-mrm-cleanse.ear` file.

   The deployment file is in the following directory:

   `<MDM Hub installation directory>/hub/cleanse`

   Configure the following deployment options:

   - In the **Preparing for the application installation** panel, enable the option for deployments to generate default bindings.

- In the **Metadata for modules** panel, disable the `metadata-complete` attribute for the `siperian-cleanse-ejb.jar` module to scan annotation-based metadata each time the module is read.
- If you deploy on cluster nodes, in the **Select installation options** panel, enable the option for deployments to distribute application.

3. If you deployed on cluster nodes, perform the following steps:
   a. Stop the cluster, deployment manager, and the node.
   b. Start the node, then the deployment manager, and then the cluster.

For more information about deploying applications, see the WebSphere Server documentation.

Step 3. Configuring Class Loaders

After you manually deploy the Process Server application, configure class loaders for the application.

1. Configure class loaders for the Process Server application `siperian-mrm-cleanse.ear`.

   If the class loaders are configured to load classes with parent class loader first, configure the class loaders for the application.

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click one of the applications.
   c. On the page for configuring applications, click the **Class loading and update detection** link.
   d. On the **Class loader** configuration page, select the **Classes loaded with local class loader first (parent last)** class loader order option.
   e. Click **Apply**, and then click **OK**.

2. Configure class loaders for the web modules of the following application EAR file:

Application EAR File	Web Module	Class Loader Order
`siperian-mrm-cleanse.ear`	`siperian-mrm-cleanse.war`	Classes loaded with local class loader first (parent last)

   a. Select **Applications** > **Application Types** > **WebSphere enterprise applications**.
   b. On the **Enterprise Applications** page, click the name of the application EAR file.
   c. On the page for configuring the application, click the **Manage Modues** link.
   d. From the list of modules, click the link for the web module.
   e. On the web module configuration page, select the class loader order.
   f. Click **Apply**, and then click **OK**.

3. Restart WebSphere, and then start the Process Server application.

Enabling Secured Communications for Process Servers

Each Process Server requires a signed certificate. Use the Hub Console to enable the HTTPS protocol and specify a secure port for each Process Server.

1. Create signed certificates for the Process Servers in the certificate store.
2. Ensure that the application server can access the certificate store.
3. Log in to the Hub Console.
4. Select an Operational Reference Store database.
5. Acquire a write lock.
6. In the **Utilities** workbench, select **Process Server**.
7. Select a Process Server and click the **Edit Process Server** icon.

   The Add/Edit Process Server dialog box opens.
8. Verify that the **Port** is a secure port.
9. Select the **Secured Connection (HTTPS)** check box.
10. Click **OK**.
11. Verify other Process Servers that appear in the list.

Install and Configure Elasticsearch

To use search, install and setup Elasticsearch for the MDM Hub.

For more information about installing and configuring search, see the "Search with Elasticsearch" chapter in the *Multidomain MDM Configuration Guide*.

Configure Match Population

The match population contains the standard population set to use for the match process. Each supported country, language, or population has a standard population set. You must enable the match population to use for the match rules.

The match population is available as a *population*.ysp file with the Informatica MDM Hub installation. The population name is the same as the ysp file name. If you add a Japanese population, and want to use the Person_Name_Kanji match field, add _Kanji to the population name. For example, Japan_Kanji or Japan_i_Kanji. If you do this, the standard Person_Name match field is not available.

The population that you use must be compatible with the SSA-Name3 version of the MDM Hub. If you need additional population files or if you need an updated population file to upgrade to a later version, contact Informatica Global Customer Support. The first population file that you request with the product is free. You might need population files for other countries or you might need an updated population file to upgrade to a later version of the MDM Hub.

Enabling Match Population

You must enable the match population to use for the match rules.

1. Copy the `<population>.ysp` files to the following location:

   On UNIX. `<infamdm_install_directory>/hub/cleanse/resources/match`

   On Windows. `<infamdm_install_directory>\hub\cleanse\resources\match`
2. In the C_REPOS_SSA_POPULATION metadata table, verify that the population is registered.

   The seed database for the MDM Hub installation has some populations registered in the C_REPOS_SSA_POPULATION table, but not enabled.
3. Restart the Process Server after you enable populations.
4. Log in to the Hub Console to verify that the population is enabled.

   The population appears in the **Match/Merge Setup** user interface for base objects.

Configuring the Process Server with Cleanse Engines

After you install the Process Server, you can configure a cleanse engine with the Process Server.

For more information about cleanse engine configuration, see the *Multidomain MDM Cleanse Adapter Guide*.

CHAPTER 8

ActiveVOS Post-Installation Tasks for the Application Server

This chapter includes the following topics:

Install and Deploy ActiveVOS in WebSphere Cluster Environments

In a WebSphere cluster environment, to connect to the MDM identity service provider using the ActiveVOS Console, perform additional steps after the MDM Hub installation.

After you install the Hub Server, embedded ActiveVOS, and the Process Server, complete the following steps:

1. Configure the WebSphere work managers.
2. Configure a WebSphere time manager.
3. Configure JAAS application logins.
4. Install ActiveVOS Server and ActiveVOS Central.
5. Edit the ActiveVOS installation files.
6. Deploy ActiveVOS and Identity Resolution.

Note: The installation and deployment procedures may vary for different versions of WebSphere. For information specific to your version of WebSphere, see the WebSphere documentation for your version.

Configure the WebSphere Work Managers

Use the WebSphere administration console to create enterprise and system work managers for ActiveVOS. The work managers act as thread pools for the ActiveVOS application components that use asynchronous beans.

1. From the WebSphere administration console, go to **Resources** > **Asynchronous beans** > **Work managers**.

2. Add work managers with the following properties:

Property	Work Manager 1	Work Manager 2
Name	ActiveVOS Enterprise Work Manager	ActiveVOS System Work Manager
JNDI name	wm/ActiveVOS	wm/ActiveVOSSystem
Minimum number of threads	10	5
Maximum number of threads	150	50
Scope	Cluster	Cluster
Security and work area for the service	Enabled	Enabled

Configure a WebSphere Time Manager

Use the WebSphere administration console to create a time manager for ActiveVOS. A timer manager acts as a thread pool for the ActiveVOS application components that use asynchronous beans.

1. From the WebSphere administration console, go to **Resources** > **Asynchronous beans** > **Time managers**.
2. Add a time manager with the following properties:

Property	Value
Name	ActiveVOS Enterprise Time Manager
JNDI name	tm/ActiveVOS
Number of alarm threads	10
Scope	Cluster
Security and work area for the service	Enabled

Configure JAAS Application Logins

Use the the WebSphere administration console to configure application logins for the ActiveBPELIdentityAssertion and ActiveBPELProvidedUser applications.

1. Configure a JAAS login module for ActiveBPEL identity assertion.
   a. From the WebSphere administration console, go to **Security** > **Global security** > **JAAS configurations** > **Application login configuration**.
   b. Add an application login and specify the alias as `ActiveBPELIdentityAssertion`.
   c. In the **JAAS login modules** section, add the following login module classes in the specified order:
      1. `com.activee.rt.websphere.trustvalidation.AeIdentityAssertionLoginModule`

2. `com.ibm.wsspi.security.common.auth.module.IdentityAssertionLoginModule`

d. Save the changes.

2. Configure a JAAS login module for ActiveBPEL provided user.
   a. Add an application login and specify the alias as `ActiveBPELProvidedUser`.
   b. In the **JAAS login modules** section, add the following login module class:

      `com.activee.rt.websphere.trustvalidation.AeBasicLoginModule`

   c. Add the following custom properties for the module:

Property	Description
username	User name of the ActiveVOS administrative user.
password	Password of the ActiveVOS administrative user.

   d. Save the changes.

3. Add an administrative role for the ActiveVOS administrative user.
   a. Under **Users and Groups**, click **Administrative user roles** > **User**.
   b. Select the **Monitor** role for the user.
   c. Move the user from the **Available** list to the **Mapped to role** list.
   d. Click **OK**.

Complete the ActiveVOS Server and ActiveVOS Central Installation

After you configure the WebSphere environment, to complete the ActiveVOS Server and ActiveVOS Central installation, run the installer utility.

1. Open a command prompt, and navigate to the following directory:

   ```
   <MDM Hub installation directory>/<ActiveVOS directory>/server-enterprise/
   websphere_config/bin
   ```

2. To complete the ActiveVOS Server and ActiveVOS Central installation, run the following installer utility:

   On UNIX. `./config_deploy.sh`

   On Windows. `config_deploy.bat`

3. Follow the onscreen instructions.

Edit the ActiveVOS Installation Files

Before you deploy ActiveVOS, edit the ActiveVOS installation files.

1. Navigate to the following directory:

   ```
   <MDM Hub installation directory>/hub/server/conf
   ```

2. Edit the `avos.install.properties` file to set the `install.web.application` property to `false`.
3. Navigate to the following directory:

   ```
   <MDM Hub installation directory>/avos/server/server-enterprise/websphere_config/bin
   ```

4. Edit the `deployer.xml` file by commenting out the following block of code:

```
<!-- <target name="deploy.resources"
depends="deploy.timer.manager,deploy.work.manager"/>
<target name="deploy.timer.manager">
<echo message="${basedir}"/>

<run.wsadmin script="${basedir}
/scripts/timermanager.jacl"/>
</target>
<target name="deploy.work.manager">
<run.wsadmin script="${basedir}/scripts/workmanager.jacl"/>

<run.wsadmin script="${basedir}
/scripts/systemworkmanager.jacl"/>
</target>
<target name="deploy.jaas">
<run.wsadmin script="${basedir}/scripts/jaaslogin.jacl"/>

</target>

<target name="deploy.apps">

<run.wsadmin script="${basedir}
/scripts/installapp.jacl"/>
</target> -->
```

Deploy ActiveVOS and Identity Resolution

After you edit the ActiveVOS installation files, deploy ActiveVOS Server, ActiveVOS Central, and MDM Identity Resolution.

For more information, see the WebSphere Server documentation.

1. Navigate to the following directory:

   `<MDM Hub installation directory>/hub/server/bin`

2. To deploy ActiveVOS Server and MDM Identity Resolution, run the following commands:

   ```
   sip_ant.bat deploy_mdm_identity_resolution
   sip_ant.bat deploy_avos_server
   ```

3. From the WebSphere administration console, deploy the following ActiveVOS applications:

   - `ave_websphere.ear`
   - `activevos-central.war`

   The ActiveVOS applications are in the following directory: `<MDM Hub installation directory>/hub/server`

   Note: Ensure that you select the option to allow the deployments to generate default bindings.

Create a Trusted User in a WebSphere Environment

To use the ActiveVOS workflow engine, create a trusted user and map the trusted user to the abTrust, abServiceConsumer, and abTaskClient roles.

The trusted user is the same user as the ActiveVOS workflow adapter user in the Hub Console. The name of the trusted user cannot be the same name as the application server administrative user.

1. In the WebSphere console, stop the ave_websphere EAR application.

2. Create the trusted user.
3. Open the `ave_websphere.ear` file.
4. In the `ave_websphere.ear` file, map the trusted user to the abTrust, abServiceConsumer, and abTaskClient roles.
5. Restart the WebSphere profile.

Adding Users and Groups to the Secure Profile

Create users and groups for MDM Hub administrators and users. For more information about how to create users and groups, see the WebSphere documentation.

1. In the WebSphere console, create a user for each MDM Hub administrator and user that you want to authenticate with the ActiveVOS Server.
2. Create a group for the MDM Hub administrators.
3. Create a group for the MDM Hub users.
4. Add the administrators to the MDM Hub administrators group.
5. Add the users to the MDM Hub users group.

CHAPTER 9

ActiveVOS Post-Installation Tasks for the Business Entity Adapter

This chapter includes the following topics:

ActiveVOS Web Applications

When you install the bundled, licensed version of the ActiveVOS Server, you are also licensed to use two ActiveVOS web applications. After you add users to the application server container, you can use these applications.

You use the web applications for different purposes:

ActiveVOS Console

Administrators use the ActiveVOS Console to manage deployed processes, the alerting system, and endpoint locations. You can also configure the engine for performance monitoring and management.

ActiveVOS Central

Business users can use ActiveVOS Central to manage tasks, requests, and reports. However, in general, business users use an Data Director (IDD) application to manage tasks because they can open the entities to review from the Task Manager.

To use ActiveVOS Central, you must add the MDM Hub users to the application server container.

For more information about the web applications, see the Informatica ActiveVOS documentation.

Configuring the ActiveVOS URNs for the Business Entity Workflow Adapter

The ActiveVOS Server has two predefined uniform resource names (URNs) that it uses internally. You need to update the URL in the URN mappings to use the host name and the port number where the ActiveVOS Server runs.

1. Launch the ActiveVOS Console. In a browser, type the following URL, substituting the correct host name and port number:

   Encrypted connections.`https://[host]:[port]/activevos`

   Non-encrypted connections.`http://[host]:[port]/activevos`
2. In the ActiveVOS Console, on the Home page, click **Administration > Configure Server > URN Mappings**.
3. For the following URNs, update the paths to reflect the host name and port number of the ActiveVOS Server:

URN	URL Path
ae:internal-reporting	Encrypted connections. `https://[host]:[port]/activevos/internalreports` Non-encrypted connections. `http://[host]:[port]/activevos/internalreports`
ae:task-inbox	Encrypted connections. `https://[host]:[port]/activevos-central/avc` Non-encrypted connections. `http://[host]:[port]/activevos-central/avc`

4. Verify that **urn:mdm:service** is mapped to the host name and port number of the MDM Hub Server:

   Encrypted connections.`https://[host]:[port]/cmx/services/BEServices`

   Non-encrypted connections.`http://[host]:[port]/cmx/services/BEServices`

Configure the Protocol of the ActiveVOS URL

You can configure the protocol of the ActiveVOS URL in the `build.properties` file.

1. Find the `build.properties` file in the following directory:
   - On UNIX. `<MDM Hub installation directory>/hub/bin`
   - On Windows. `<MDM Hub installation directory>\hub\bin`
2. Change the protocols of the following parameters from `http` to `https`.
   - `activevos.mdm.sif.url`
   - `activevos.mdm.cs.url`
3. Save the `build.properties` file.
4. Navigate to the following directory:
   - On UNIX. `<MDM Hub installation directory>/hub/server`
   - On Windows. `<MDM Hub installation directory>\hub\server`
5. Run the following command to deploy the Hub Server application and apply changes to the security configuration:

On UNIX

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic password> -
Ddatabase.password=<your database password>
```

WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<your database password>
```

JBoss

```
patchInstallsetup.sh -Ddatabase.password=<your database password>
```

On Windows

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic password> -
Ddatabase.password=<your database password>
```

WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<your database password>
```

JBoss

```
patchInstallsetup.bat -Ddatabase.password=<your database password>
```

Note: On UNIX, if you include an exclamation mark (!) character in the password, you must include a backslash (\) before the exclamation mark (!) character. For example, if the password is `!!cmx!!`, enter `\!\!cmx\!\!`.

Set the ActiveVOS Protocol to HTTPS

To enable secure communication between ActiveVOS and the MDM Hub, set the protocol to HTTPS in the Hub Console Workflow Manager.

You must first configure the application server for HTTPS communications.

1. Start the Hub Console.
2. Acquire a write lock.
3. Click **Workflow Manager** under the Configuration workbench.
4. In the Workflow Manager, click the **Workflow Engines** tab.
5. Select the ActiveVOS workflow engine, and then click the **Edit** button.
6. In the Edit Workflow dialog box, set the protocol to HTTPS.
7. In a WebLogic environment, in the Edit Workflow dialog box, enter the user name and password of the user that belongs to the abAdmin role.

Configure the Primary Workflow Engine

To configure the primary workflow engine, add a workflow engine for ActiveVOS workflows based on business entities. The secondary workflow engine is for existing customers who want to process existing tasks with a deprecated workflow engine.

1. In the Hub Console, click **Workflow Manager** in the Configuration workbench.

2. Acquire a write lock.
3. Select the **Workflow Engines** tab and click the **Add** button.
4. In the **Add Workflow** dialog box, enter the workflow engine properties.

   The following table describes the workflow engine properties:

Field	Description
Workflow Engine	The display name of the workflow engine
Adapter Name	Select **BE ActiveVOS** for the ActiveVOS workflow adapter based on business entities.
Host	The host name of the Informatica ActiveVOS instance.
Port	The port name of the Informatica ActiveVOS instance.
Username	The user name of the trusted user.
Password	The password of the trusted user.
Protocol	The protocol for communication between the MDM Hub and ActiveVOS. The protocol can be http or https.

5. Click **OK**.

Configure the MDM Identity Services for ActiveVOS

If you use embedded ActiveVOS, ensure that you configure ActiveVOS to use MDM Identity Services. To configure the MDM Identity Services for ActiveVOS, use the ActiveVOS Console to set the Identity Services password to the password of the MDM Hub workflow engine user.

1. In the ActiveVOS console, select **Admin** > **Configure Services** > **Identity Services**.
2. In the Provider Configuration section, enable the **Enable** check box and select **MDM** from the **Provider Type** list.
3. In the Connection tab, enter the password of the MDM Hub user with the user name `admin`.

   Note: If you later change the password for the admin user, you must enter the new password in the ActiveVOS identity services settings.
4. Click **Update**.
5. Test that ActiveVOS can log in to the MDM Hub as the `admin` user, and that ActiveVOS can retrieve a list of roles for the user you specify as the **User for test**.
   a. Select the **Test** tab.
   b. In the **User for test** field, enter an MDM Hub user that is assigned to a role.
   c. Click **Test Settings**.

      Note: The test fails if an Operational Reference Store is not configured, the user for test does not belong to a role, or the role name contains spaces.

Configure Tasks

Before you begin using task workflows in Informatica Data Director, configure task templates, task triggers, and task types in the Provisioning tool.

For more information, see the *Multidomain MDM Provisioning Tool Guide*.

CHAPTER 10

Customize ActiveVOS

You can customize the ActiveVOS installation by configuring the ActiveVOS properties in the `build.properties` file. The properties have a default value. You do not need to set them unless the default value does not work in your environment.

The following table describes the ActiveVOS properties that you can configure:

Property	Description
activevos.install.console	Controls if the ActiveVOS console is deployed in the server. Default is true. If set to false, ActiveVOS Console is not installed.
activevos.secure.https.only	Forces and redirects all ActiveVOS HTTP traffic to HTTPS. Default is false

Adding ActiveVOS Properties

To enable the ActiveVOS properties, you must add the properties to the `build.properties` file in hub console and run the post-install script run again for the changes to take place.

1. Open the following `build.properties` directory in a text editor:

   ```
   <MDM Hub installation directory>\hub\server\bin\build.properties
   ```

2. Add the ActiveVOS properties to the `build.properties`directory.
3. Open a command prompt.
4. Navigate to the `PostInstallSetup` script in the following directory:
   - On Unix. `<MDM Hub installation directory>/hub/cleanse`
   - On Windows. `<MDM Hub installation directory>\hub\cleanse`
5. Run the `PostInstallSetup` script:
   - On Unix. `postinstallsetup.sh`
   - On Windows. `postinstallsetup.bat`
6. Restart the application server.

CHAPTER 11

Resource Kit Installation

This chapter includes the following topics:

Setting Up the MDM Hub Sample Operational Reference Store

Before you can use the MDM Hub sample Operational Reference Store, you must set it up. Before you install the Resource Kit, set up the MDM Hub sample Operational Reference Store. To set up the MDM Hub sample Operational Reference Store, create an Operational Reference Store and import `mdm_sample` into it.

1. Create an MDM Hub sample Operational Reference Store user on the machine on which the database is installed.

   On UNIX, ensure that you create the user name with 8 characters or less.
2. Add the MDM Hub sample Operational Reference Store user to the DB2ADMNS and DB2USERS user groups.
3. Navigate to the following location in the distribution directory:

   On UNIX. `<distribution directory>/database/bin`

   On Windows. `<distribution directory>\database\bin`
4. Run the following command:

   On UNIX. `./sip_ant.sh create_ors`

   On Windows. `sip_ant.bat create_ors`
5. Answer the prompts that appear.

   Note: The prompt displays default text in brackets. Press **Enter** to use the default value and go to the next prompt.

   Enter database type. (ORACLE, MSSQL, DB2)

   Database type. Specify `DB2`.

Enter the Operational Reference Store database host name. [localhost]

Name of the machine that hosts the database. Default is `localhost`.

Enter the Operational Reference Store database port number. [50000]

Port number that the database uses.Default is `50000`.

Enter the database name. [SIP97]

Name of the database. Default is `SIP97`.

Connect URL. [jdbc:db2://<*host name*>:<*port*>/<database name>]

Connect URL for the database connection.

Enter the Operational Reference Store database user name. [cmx_ors]

User name of the MDM Hub sample Operational Reference Store database. Default is `cmx_ors`.

Enter the Operational Reference Store database user password.

Password of the MDM Hub sample Operational Reference Store user.

Enter a locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]

Operating system locale. Default is en_US.

Enter the DBA user name. [DB2ADMIN]

User name of the administrative user. Default is `DB2ADMIN`.

Enter the DBA password.

Password of the administrative user.

6. After you create the sample Operational Reference Store, review `sip_ant.log` in the following directory:

   On UNIX. <*distribution directory*>/database/bin

   On Windows. <*distribution directory*>\database\bin

   The `sip_ant.log` file logs any errors that might occur when you run the `sip_ant` script to create the Operational Reference Store.

7. To import `mdm_sample`, run the following command:

   On UNIX. `./sip_ant.sh import_schema`

   On Windows. `sip_ant.bat import_schema`

8. Answer the prompts that appear.

   Note: The prompt displays default text in brackets. Press **Enter** to use the default value and go to the next prompt.

   Enter database type. (ORACLE, MSSQL, DB2)

   Database type. Specify `DB2`.

   Enter the Operational Reference Store database host name. [localhost]

   Name of the machine that hosts the database. Default is `localhost`.

   Enter the Operational Reference Store database port number. [50000]

   Port number that the database uses.Default is `50000`.

   Enter the database name. [SIP97]

   Name of the database. Default is `SIP97`.

   Connect URL. [jdbc:db2://<*host name*>:<*port*>/<database name>]

   Connect URL for the database connection.

   Enter the Operational Reference Store database user name. [cmx_ors]

   Name of the MDM Hub sample Operational Reference Store database. Default is `cmx_ors`.

   Enter the Operational Reference Store database user password.

   Name of the MDM Hub sample Operational Reference Store database user.

   Enter a locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]

   Operating system locale. Default is en_US.

   Enter the path to the ZIP dump file. [<*distribution directory*>\resources\database]

   Path to the `mdm_sample.zip` file.

   Enter the name of the ZIP dump file. [mdm_sample.zip]

   Name of the ZIP dump file. Default is `mdm_sample.zip`.

Registering the Informatica MDM Hub Sample Operational Reference Store

After you set up the MDM Hub sample Operational Reference Store, you must register it. Register the MDM Hub sample Operational Reference Store through the Hub Console.

1. Start the Hub Console.

   The **Change database** dialog box appears.
2. Select the MDM Hub Master Database, and click **Connect**.
3. Start the **Databases** tool under the Configuration workbench.
4. Click **Write Lock** > **Acquire Lock**.
5. Click the **Register database** button.

   The **Informatica MDM Hub Connection Wizard** appears and prompts you to select the database type.
6. Select the type of database, and click **Next**.

7. Configure connection properties for the database.
   a. Specify the connection properties, and click **Next**.

      Specify the connection properties, and click **Next**.

      The **Summary** page appears.
   b. Review the summary, and specify additional connection properties.

      The following table lists additional connection properties that you can configure:

Property	Description
Connection URL	Connect URL. The Connection Wizard generates the connect URL by default. The following example shows the format of the connect URL: `jdbc:db2:@//database_host:port/service_name`
Create data source after registration	Select to create the data source on the application server after registration. **Note:** If you do not select the option, you must manually configure the data source.

8. Click **Finish**.

   The **Registering Database** dialog box appears.
9. Click **OK**.

   The MDM Hub registers the MDM Hub sample Operational Reference Store.
10. Select the MDM Hub sample Operational Reference Store that you registered, and click the **Test database connection** button to test the database settings.

    The Test Database dialog box displays the result of the database connection test.
11. Click **OK**.

    The Operational Reference Store is registered, and the connection to the database is tested.

Installing the Resource Kit in Graphical Mode

You can install the Resource Kit in graphical mode.

Before you install the Resource Kit, you must have installed and configured the MDM Hub.

1. Start the application server.
2. Open a command prompt and navigate to the Resource Kit installer. By default the installer is in the following directory:

   On UNIX. `<distribution directory>/<operating system name>/mrmresourcekit`

   On Windows. `<distribution directory>\windows\mrmresourcekit`
3. Run the following command:

   On UNIX. `hub_resourcekit_install.bin`

   On Windows. `hub_resourcekit_install.exe`
4. Select the language for the installation, and then click **OK**.

The **Introduction** window appears.

5. Click **Next**.

   The **License Agreement** window appears.

6. Select the **I accept the terms of the License Agreement** option, and then click **Next**.

   The **Installation Feature** window appears.

7. Select the Resource Kit features that you want to install and click **Next**.

   You can select the following options:

   Sample Schema

   Installs the MDM Hub sample schema resources. You must create a sample schema and register it with the Hub Server before you install the sample applications.

   Samples and Utilities

   Installs the sample applications and utilities.
   The list of sample applications that are deployed is stored in the `build.properties` file in the following directory:

   `<Resourcekit_Home>\samples`

   SIF SDK and Javadocs

   Installs the javadocs, libraries, and resources associated with the SIF SDK.

   BPM SDK

   Installs the resources associated with the BPM SDK.

   Jaspersoft

   Copies the Jaspersoft installer to the Resource Kit home.

   SSA-NAME3

   Copies the SSA-NAME3 installer to the Resource Kit home.

   A message about the requirement of having created and registered a sample schema with the MDM Hub appears.

8. Click **OK**.

   The **Choose Install Folder** window appears.

9. Select the location of the Resource Kit installation.

   - To choose the default location, click **Next**.
   - To enter a path, type the path to the installation folder, and click **Next**.

     Note: The installation fails if you specify a path that has spaces in the directory or folder names.
   - To return to the default installation location, click **Restore Default Folder**.
   - To choose another location, click **Choose**, and then click **Next**.

   On UNIX, the **Choose Link Folder** window appears.

   On Windows, the **Choose Shortcut Folder** window appears.

10. On UNIX, choose a link folder or select the option to not create links, and click Next. On Windows, select the location to create a product icon or select the option not to create a product icon.

    The **Configuration Selection** window appears.

11. Select a configuration option, and click **Next**.

    You can select one of the following options:

Configure Samples

Installs and configures the samples.

Source Only

Installs the sources of samples but does not configure the samples.

If you select **Configure samples**, the **Resource Kit App Server** window appears. If you select **Source only**, the **Pre-Installation Summary** window appears.

12. From the **Resource Kit App Server** window, select the application server on which you want to install the Resource Kit, and click **Next**.

    The **Application Server Home** window for the application server that you select appears.

13. Configure the application server settings.
    a. Choose a path to the WebSphere application server, and click **Next**.

       The **Reminder** window appears.

    b. Ensure that you have fulfilled the prerequisites, and click **OK**.

       The **WebSphere Security Selection** window appears.

    c. Select whether WebSphere is security enabled or not, and click **Next**.
       - If you select **No**, and then click **Next**, the **WebSphere Application Server Port** window appears. Default is **No**.

         Set the server name, and the RMI and SOAP ports for the WebSphere application server.
       - If you select **Yes**, and then click **Next**, the **WebSphere Application Server Port and User Credentials** window appears. Specify the WebSphere user name and the WebSphere password.

    The **Informatica MDM Hub Server** window appears.

14. Enter the information for the Hub Server installation, and click **Next**.

    Enter values in the following fields:
    Server Name

    Name of the server that hosts the Hub Server.

    Server HTTP Port

    Port number of the Hub Server.

    Informatica MDM Administrative Password

    Password to access the MDM Hub.

    MDM Hub Home Directory

    Directory for the Hub Server installation.

    The **Resource Kit ORS ID** window appears.

15. Select a Resource Kit ORS ID from the list, and then click **Next**.

    The list contains the Operational Reference Store IDs that you created. Select an Operational Reference Store ID related to the sample schema.

    If you have not registered the sample schema, you will not see the Operational Reference Store ID for the sample schema. Register the sample Operational Reference Store, and then restart the installation.

    The **Deployment Selection** window appears.

16. Select one of the following options and click **Next**:

Yes, run it during this installation.

Deploys and configures the Resource Kit during the installation.

No, it can be deployed later.

Select this option to deploy and configure manually at a later time.
If you chose to install the Samples and Utilities feature, you must deploy and configure the Resource Kit in this installation step. If you do not deploy the Resource Kit in this step, you cannot make changes and redeploy the samples by using the postInstallSetup script provided in the Resource Kit.

If you choose to run the post-installation setup manually, you cannot deploy the EAR file by using the postInstallSetup script at a later time. You must manually edit the EAR file and deploy it to make any changes to your installation.

The **Pre-Installation Summary** window appears.

17. Review the Pre-Installation Summary to confirm your installation choices, and then click **Install**.

    When the installation completes, the **Install Complete** window appears.

18. Click **Done** to exit the Resource Kit installer.

Installing the Resource Kit in Console Mode

You can install the Resource Kit in console mode.

Ensure that you register the MDM_SAMPLE schema before you install the Resource Kit.

1. Start the application server.
2. Navigate to the following directory in the MDM Hub distribution:

   On UNIX. `<MDM Hub distribution directory>/<operating system name>/resourcekit`

   On Windows. `<MDM Hub distribution directory>/windows/resourcekit`
3. Run the following command from the command prompt:

   On UNIX. `./hub_resourcekit_install.bin -i console`

   On Windows. `hub_resourcekit_install.exe -i console`
4. Enter the number of the locale you want to choose for the installation, and then press **Enter**.

   The introduction information about the installation appears.
5. Press **Enter**.

   The license agreement appears.
6. Read the License Agreement. Type **Y** to accept the terms of the license agreement, or type **N** if you do not want to accept the license agreement and want to exit the installation program.
7. Press **Enter**.

   If you entered **Y** in the preceding step, information about the installation folder appears.
8. Enter the numbers of the Resource Kit features that you want to install separated by commas, and press **Enter**.

   The prompt for the sample schema installation appears.
9. Choose a folder for the Resource Kit installation.
   - To choose the default folder, press **Enter**.

- To change the path, type the absolute path of the installation folder, and press **Enter**.

10. Confirm the location of the installation folder. Type **OK** to confirm the installation folder or type **Cancel** to change the installation folder.
11. Press **Enter**.

    A list of link location options appears.
12. Enter the number of a link location option.

    The prompt for the link file location appears.
13. Enter the absolute path of the link file, and press **Enter**.

    The source sample configuration options appears.
14. Enter a configuration option, and press **Enter**.

Option	Description
1	Installs and configures the samples
2	Installs the sources of samples but does not configure the samples

    If you enter **1**, a list of application server options appears. If you enter **2**, the pre-Installation summary appears.
15. If you entered **1**, enter the number for the application server that you want to select, and press **Enter**.

    The application server information prompts appear.
16. Configure the WebSphere settings.
    a. Specify the application server installation directory, and press **Enter**.

       The installer displays the WebSphere pre-requisites for JDBC drivers.
    b. Check the database JDBC driver files location, and press **Enter**.

       The database JDBC driver files are copied to the `<WebSphere_install_dir>/AppServer/lib` directory. The WebSphere Security selection information appears.
    c. If you select **No**, the WebSphere application server port information appears. If you select **Yes**, the WebSphere application server port and user credentials information appears.
       - If you select **No**, enter the server name, RMI port, SOAP port, and profile name, or accept the default values, and press **Enter**.
       - If you select **Yes**, enter the server name, RMI port, SOAP port, profile name, user name, and password, or accept the default values, and press **Enter**.

       The Hub Server information prompts appear.
17. Enter the information for the Hub Server installation, and press **Enter**.

    The following table describes the prompts for the Hub Server installation information:

Prompt	Description
Server Name	Name of the server that hosts the Hub Server.
Server HTTP Port	Port number of the Hub Server.

Prompt	Description
Informatica MDM Administrative password	Password to access the MDM Hub.
MDM Hub Home Directory	Directory for the Hub Server installation.

A list of MDM Hub ORS IDs appears.

18. Enter the Operational Reference Store ID of the MDM sample schema, and press **Enter**.

    If you did not register the sample schema, you will not see the Operational Reference Store ID for the sample schema. Register the sample Operational Reference Store, and then restart the installation.

    The deployment selection prompt appears.
19. Choose whether you want to run the `postInstallSetup` script as part of the installation, or run it manually later.
20. Press **Enter**.

    The summary of the installation choices appears.
21. Verify the information in the pre-installation summary. If the information is correct, press **Enter** to start the installation.

    The Resource Kit is installed according to the configuration information you provide. When the process is complete, the installation complete information appears.
22. Press **Enter** to exit the installer.

Installing the Resource Kit in Silent Mode

You can install the Resource Kit without user interaction in silent mode. You might want to perform a silent installation if you need multiple installations, or if you need to install on a machine cluster. A silent installation does not show any progress or failure messages.

Before you run the silent installation for the Resource Kit, you must configure the properties file for the silent installation. The installer reads the file to determine the installation options. The silent installation process might complete successfully even if you provide incorrect settings, such as an incorrect application server path or port. You must ensure that you provide correct settings in the properties file.

Copy the Resource Kit installation files to the hard disk on the machine where you plan to install the Resource Kit. To install in silent mode, complete the following tasks:

1. Configure the installation properties file and specify the installation options in the properties file.
2. Run the installer with the installation properties file.

Configuring the Properties File

Informatica provides a sample properties file that includes the parameters that the installer requires. You can customize the sample properties file to specify the options for the installation. Then run the silent installation.

The silent installer does not validate the settings in the properties files. You must ensure that you specify correct settings and verify them before you run the silent installer.

1. Find the `silentInstallResourceKit_sample.properties` file in the following directory:

   On UNIX. `/silent_install/mrmresourcekit`

   On Windows. `\silent_install\mrmresourcekit`

   After you customize the file, save it. You can rename the file and put it anywhere on the machine.

   Note: In the silent properties file, slash and backslash are special characters. You must enter two of each of these characters when you enter information in the file, such as when you enter an installation path. For example, to enter the path to the server directory, you must enter `\\u1\\infamdm\\hub\ \resourcekit`.

2. Create a backup copy of the `silentInstallResourceKit_sample.properties` file.
3. Use a text editor to open the file and change the values of the installation parameters.
4. Save the properties file with a new name such as `silentInstallresourcekit.properties`.

The following table describes the installation parameters that you can change:

Property Name	Description
INSTALLER_UI	Specifies the mode of installation. Set to `silent`.
SIP.INSTALL.TYPE	Specifies the type of installation. Set to `SIPERIAN_SAMPLE_INSTALL`.
SIP.INSTALL.SAMPLE.SCHEMA	Specifies if you want to install the sample schema. Specify one of the following values: - 0. Does not install the sample schema - 1. Installs the sample schema
SIP.INSTALL.SAMPLES	Specifies if you want to install samples and utilities. Specify one of the following values: - 0. Does not install samples and utilities - 1. Installs samples and utilities
SIP.INSTALL.SIF.SDK	Specifies if you want to install the Services Integration Framework (SIF) SDK. Specify one of the following values: - 0. Does not install the SIF SDK - 1. Installs SIF SDK
SIP.INSTALL.BPM.SDK	Specifies if you want to install the BPM SDK. Specify one of the following values: - 0. Does not install the BPM SDK - 1. Installs the SIF SDK
SIP.INSTALL.JASPERSOFT	Specifies if you want to install the Jaspersoft reporting tool. Specify one of the following values: - 0. Does not install the BPM SDK - 1. Installs the SIF SDK

Property Name	Description
SIP.INSTALL.SSANAME3	Specifies if you want to install SSA-NAME3. Specify one of the following values: - 0. Does not install SSA-NAME3 - 1. Installs SSA-NAME3
USER_INSTALL_DIR	Directory where you want to install the Resource Kit, such as `C\:\\<infamdm_install_directory>\\hub\\resourcekit`.
RUN_CONFIGURE_FLAG	Specifies if you want to configure samples. - 0. Does not configure samples - 1. Installs and configures samples Default is `1`. If you set the RUN_CONFIGURE_FLAG property to `1`, comment out or set the RUN_CONFIGURE_SETUP property to `0`. To configure samples, ensure that the application server and the Hub Server are started and the sample schema is registered in the Hub Console.
RUN_CONFIGURE_SETUP	Specifies if you want to configure only source samples. - 0. Does not install sources of samples - 1. Installs sources of samples If you set the RUN_CONFIGURE_SETUP property to `1`, comment out or set the RUN_CONFIGURE_FLAG property to `0`. If you set the RUN_CONFIGURE_SETUP property to `1`, you cannot configure and deploy samples later.
SIP.AS.CHOICE	Name of the application server. Specify `WebSphere`.
SIP.AS.HOME	The path to the WebSphere installation directory.
SIP.AS.SERVER	Name of the server.
SIP.AS.PROFILENAME	Application server profile name.
SIP.AS.PORT_2	Specify the RMI port number.
SIP.AS.PORT_3	Specify the SOAP port number.
SIP.WEBSPHERE.SECURITY.ENABLED.Yes=1	Set this property if WebSphere security is enabled. If SIP.WEBSPHERE.SECURITY.ENABLED.Yes=1, set the following properties: - SIP.APPSERVER.USERNAME - SIP.APPSERVER.PASSWORD
SIP.APPSERVER.USERNAME	User name required to access WebSphere.
SIP.APPSERVER.PASSWORD	Password required to access WebSphere.
SIP.SERVER.NAME	Name of the server on which the Hub Server is deployed.
SIP.SERVER.HTTP.PORT	Port on which the Hub Server is listening.
SIP.ADMIN.PASSWORD	Password to access the Hub Server.
HUB_SERVER_HOME	Directory for the Hub Server installation.

Property Name	Description
SIP.ORS.ID	Operational Reference Store ID of the MDM Hub sample schema.
RUN_DEPLOYMENT_FLAG	Runs the postInstallSetup script as part of the silent installation. - 0. Does not run the postInstallSetup script - 1. Runs the postInstallSetup script

Running the Silent Installer

After you configure the properties file, you can start the silent installation.

1. Ensure that the application server is running.
2. Open a command window.
3. Run the following command:

   On UNIX. `./hub_resourcekit_install.bin -f <location_of_silent_properties_file_for_resourcekit>`

   On Windows. `.\hub_resourcekit_install.exe -f <location_of_silent_properties_file_for_resourcekit>`

The silent installer runs in the background. The process can take a while. Check the `postinstallSetup.log` files to verify that the installation was successful.

The log file is available in the following directory:

On UNIX. `<infamdm_install_directory>/logs/postInstall.log`

On Windows. `<infamdm_install_directory>\logs\postInstall.log`

CHAPTER 12

Resource Kit Post-Installation Tasks

This chapter includes the following topics:

Edit the sip_ant Script

After you perform the installation tasks, edit the sip_ant script.

1. Open the `sip_ant` script in a text editor.

   The `sip_ant` script is in the following directory:

   On UNIX. `<Resource Kit install directory>/deploy/bin`

   On Windows. `<Resource Kit install directory>\deploy\bin`

2. Find the line similar to the following:

   On UNIX. `"$JAVA_HOME/bin/java" $USER_INSTALL_PROP -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%;`

   On Windows. `"%JAVA_HOME%\bin\java" %USER_INSTALL_PROP% -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%;`

3. Replace with the code similar to the following to set the JAVA_HOME:

   On UNIX. `"$JAVA_HOME/bin/java" -Djava.endorsed.dirs="<WebSphere installation directory>\endorsed_apis" $USER_INSTALL_PROP -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%;`

   On Windows. `"%JAVA_HOME%\bin\java" -Djava.endorsed.dirs="<WebSphere installation directory>\endorsed_apis" %USER_INSTALL_PROP% -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%;`

4. Save the changes and close the `sip_ant` script.

Running the postInstall Script Manually

To ensure that all the required deployment files are deployed on the application server, run the `postInstall` script manually.

1. Open a command prompt.
2. Navigate to the `postInstallSetup` script in the following directory:

   On UNIX. `<MDM Hub installation directory>/hub/resourcekit/deploy`

   On Windows. `<MDM Hub installation directory>\hub\resourcekit\deploy`
3. Run the following command:

   On UNIX. `postInstall.sh -Ddatabase.password=<MDM Hub Master Database Password>`

   Note: If you include the exclamation mark (!) in your password, you must include a backslash before the exclamation mark. For example, if your password is !!cmx!!, enter the password as follows: \!\!cmx\!\!

   On Windows. `postInstall.bat -Ddatabase.password=<MDM Hub Master Database Password>`

   Note: If you enabled security on WebSphere, run the `postInstallSetup` script with the `-Dwebsphere.password=<Secure WebSphere Password>` option.

Validate the MDM Hub Sample Operational Store

After you set up and register the MDM Hub sample Operational Reference Store, validate the metadata in the MDM Hub sample Operational Reference Store. Validation verifies the completeness and integrity of the metadata that describes the MDM Hub sample Operational Reference Store.

1. In the Hub Console, start the Repository Manager.
2. From the **Select the repository to validate** list, select the MDM Hub sample Operational Reference Store that you registered.
3. Select the **Validate** button.

   The **Select Validation Checks** dialog box appears.
4. Enable all the validation checks, and click **OK**.
5. If validation errors are generated, regenerate MTIP views.
   a. Start the Enterprise Manager and acquire a write lock.
   b. On the ORS databases tab, select the MDM Hub sample Operational Reference Store name.

      The Properties tab of the MDM Hub sample Operational Reference Store opens.
   c. Click the **Regenerate MTIP's** button.

      MTIP views are regenerated and the errors are fixed.

CHAPTER 13

Troubleshooting the MDM Hub

This chapter includes the following topic:

Troubleshooting the Installation Process

If the installation fails, use the following information to troubleshoot the failure.

Cannot launch the Hub Console

After installation, the Hub Console fails to launch and generates the following error in the log file:

```
SIP-09131: General Decryption failure and [ERROR] com.delos.util.StringUtil: Unable to decrypt
```

Encrypt and update the MDM Hub Master Database password or Operational Reference Store password.

1. To encrypt a database schema password, run the following command from a command prompt:

   ```
   java -classpath siperian-api.jar;siperian-common.jar;siperian-server.jar com.delos.util.PublicKeyBasedEncryptionHelper <plain text password> <Hub Server installation directory>
   ```

   The results are echoed to the terminal window.

2. To update your Master Database password or Operational Reference Store password, connect as the `cmx_system` user and run the following statement:

   ```
   UPDATE C_REPOS_DATABASE SET PASSWORD = '<new_password>' WHERE USER_NAME = <user_name>;
   COMMIT;
   ```

3. Run the `postInstallSetup` script.

You did not install the application server profile in the default directory

If you did not install the application server profile in the default directory, the postInstallSetup script fails to deploy the Hub Server and Process Server applications to the application server.

Use the following default directory:

On UNIX. *<Websphere_install_home>*`/profiles`

On Windows. *<Websphere_install_home>*`\profiles`

To resolve the issue, repackage the Hub Server and Process Server EAR files and then manually deploy the Hub Server and Process Server applications to the custom directory where you installed the application server.

PostInstallSetUp script fails as the directory contains the siperian-mrm.ear file

If you try to deploy the Hub Server application to a directory that contains a file with the name siperian-mrm.ear, the following error appears:

```
[wsadmin] ADMA5016I: Installation of siperian-mrm.ear started.

 [wsadmin] A composition unit with name siperian-mrm.ear already exists. Select a
different application name.
```

To resolve the issue, remove all the directories that contain a `siperian-mrm.ear` file, and then run postInstallSetup again to deploy the EAR file.

Note: If you undeploy the Hub Server application, a `siperian-mrm.ear` file might still exist in an application server directory.

PostInstallSetUp script fails as the process times out

When you install the Hub Server in a WebSphere environment, the post-installation setup process fails, and the following error appears:

```
[wsadmin] Starting siperian-mrm.ear ...
  [wsadmin] WASX7017E: Exception received while running file "wsinstall.jacl"; exception
information: com.ibm.websphere.management.exception.ConnectorException
  [wsadmin] org.apache.soap.SOAPException: [SOAPException: faultCode=SOAP-ENV:Client;
msg=Read timed out; targetException=java.net.SocketTimeoutException: Read timed out]
```

The issue occurs when the SOAP request times out.

To resolve the issue, perform the following steps:

1. Navigate to the following directory:
   `<Websphere profile root directory>/properties`
2. In the `soap.client.props` file, increase the value of the `com.ibm.SOAP.requestTimeout` property.
3. Restart WebSphere and run the `postInstallSetup` script again.

PostInstallSetUp script fails with `javax.management.MBeanException`

On Linux, when you install or upgrade the Process Server, the `postInstallSetup` script fails with the `javax.management.MBeanException` error.

To resolve the issue, stop and start WebSphere. The Process Server starts up.

MDM Hub users cannot login

If you re-create the CMX_SYSTEM schema after installation of the Hub Server, the MDM Hub cannot recognize the hashed passwords. As a result, users cannot log in to the MDM Hub.

To resolve the issue, run the `postInstallSetup` script again manually. This script ensures that the passwords of the MDM Hub users are hashed again and users can log in.

Alternatively, if you do not want to run the `postInstallSetup` script again, run the following commands to migrate the user passwords to hashed passwords and create application users.

On UNIX.

```
cd <MDM Hub installation directory>/server/bin
./sip_ant.sh hash_users_passwords
./sip_ant.sh add_application_users
```

On Windows.

```
cd <MDM Hub installation directory>\server\bin
sip_ant.bat hash_users_passwords
sip_ant.bat add_application_users
```

Note: In a WebSphere environment, the MDM Hub user must have access and write permissions for the following directory:

```
<MDM Hub installation directory>/server/bin/resources/certificates
```

PostInstallSetUp script fails because the ActiveVOS server deployment times out

When you install the Hub Server, the post-installation setup process might fail after trying to deploy the ActiveVOS server.

To resolve the issue, increase the value of the `deploy.wait.time` property in the `build.properties` file in the following directory:

On UNIX. `<infamdm installation directory>/hub/server/bin`

On Windows. `<infamdm installation directory>\hub\server\bin`

The Hub Server cannot connect to the cmx_system schema

To verify that the Hub Server cannot connect to the cmx_system schema, review the application server log.

To resolve the issue, resolve the database connection issue. Use the application server console to test the connection to the database. If you cannot resolve the connection to the cmx_system schema, re-create the cmx_system schema.

Failed to verify the need to tokenize records

When you run the match process, you might receive the following error:

```
SIP-16062: Failed to verify the need to tokenize records.
```

Verify the following environment variable settings:

- The library path environment variable must contain the following path:

  On UNIX. `<`*infamdm_install_directory*`>/hub/cleanse/lib`

  On Windows. `<`*infamdm_install_directory*`>\hub\cleanse\lib`

  The library path environment variable depends on the operating system:

  - AIX. `LIBPATH`
  - Suse or RedHat Linux. `LD_LIBRARY_PATH`
  - Windows. `PATH`

- The `SSAPR` environment variable must include the following path for all users:

  On UNIX. `<`*infamdm_install_directory*`>/server_install_dir>/cleanse/resources`

  On Windows. `<`*infamdm_install_directory*`>\server_install_dir>\cleanse\resources`

major.minor version errors when loading the Process Server

If you see multiple `major.minor` console errors when you try to load the Process Server, check that the system has the correct Java version installed.

CORBA TRANSACTION_ROLLEDBACK exception

When you use the Data Manager or the Merge Manager, the MDM Hub might generate the `CORBA TRANSACTION_ROLLEDBACK` exception. If the exception is generated, open the WebSphere Administrative console to manually set the `-Djava.vendor=IBM` Java option in the WebSphere process definitions.

Informatica Address Verification exception during certification

Informatica Address Verification generates an exception during certification. Ensure that the stack size for the JVM is sufficient.

1. Open the WebSphere console.
2. Go to **Servers** > **Application Server** > **<Your Server>** > **Process Definition** > **Java Virtual Machine**.
3. Add the following arguments to the generic JVM arguments:

```
-Xss2000k - Initializes the stack size to 2000k
-Xms128m - Initializes the heap with at least 128 MB
-Xmx1024m - Initializes the heap with a maximum of 1024 MB
```

4. Save the configuration.
5. Restart the WebSphere server.

Operational Reference Store does not have a workflow engine configured

If you install the MDM Hub and then import an Operational Reference Store (ORS) from a previous version, you see a fatal error that indicates that the ORS does not have a workflow engine configured. This error occurs because the Siperian BPM workflow engine is not registered by default. Use the Workflow Manager to register the Siperian BPM workflow engine with the name that the ORS expects to find.

After you deploy the Process Server .ear files, an error occurs

On a Linux environment with WebSphere 8.5.5.9, after you deploy the Process Server, the following error occurs:

```
Too many open files. Unable to start cleanse ear.
```

Increase the value of the `ulimit` parameter in Linux, and deploy the Process Server.

A bulk edit job fails with the database connection error

When you perform a bulk edit, the job times out and fails. This issue might occur when you do not have a sufficient number of connections.

Ensure that the adequate number of connections is configured. The maximum number of connections is 250.

1. Open the WebSphere console.
2. Go to **Resources** > **JDBC** > **Data Sources**.
3. Open the proper data source.
4. Go to **Connection Pool Properties**.
5. In the **Maximum connections** field, in the **General Properties** section, enter the required number of connections.
6. Click **Apply**.

Cannot add the application users using the sip_ant script

The following applications are affected: Business Process Manager (formerly known as ActiveVOS), Data Director, MDM Hub Console and Provisioning Tool.

To add the application users, run the `sip_ant` command, using the `add_app_users` parameter.

You can fetch the connect URL from the following location:

`hub/server/bin/build.properties (masterdatabase.jdbc.url))`

Run the following command:

- On Windows.

```
sip_ant.bat add_app_users -Ddatabase.password=<cmx system password> -Dmaster.
connecturl="jdbc:oracle:thin:@<Database Host name>:<DB Port>:<SID>" -
Dmaster.username=cmx_system
```

- On Unix.

```
./sip_ant.sh add_app_users -Ddatabase.password=<cmx system password> -Dmaster.
connecturl="jdbc:oracle:thin:@//<Database Host name>:<DB Port>:<SID>" -
Dmaster.username=cmx_system
```

Run the following command on the SQL Server:

```
./sip_ant.sh add_app_users -Ddatabase.password=<cmx system password> -
Dmaster.connecturl="jdbc:sqlserver://<Database Host Name>:<DB
Port>;DatabaseName=cmx_system" -Dmaster.username=cmx_system
```

The Certificates folder is not created

The Certificates folder is not created as expected, after you back up the existing Certificates folder from `\infamdm\hub\server\resources\`, and run the `sip_ant.sh hash_users_passwords` script. This issue occurs when the `sip_ant.sh hash_users_passwords` script updates the C_REPOS_USER table in `cmx_system`.

To resolve this issue, restart the application server. As a result, the Certificates folder is created in the expected location: `\infamdm\hub\server\resources\`.

The folder will contain only the `MDM_KEYSTORE_FILE_JKS` keystore.

To generate other certificates, run the following command:

```
./sip_ant.sh add_app_users
```

CHAPTER 14

Uninstallation

This chapter includes the following topics:

Uninstallation Overview

To uninstall the MDM Hub, you need to remove the Process Server, the Hub Server, and the Hub Store from the MDM Hub implementation.

Use the following steps to uninstall the MDM Hub:

1. Uninstall the Hub Store.
2. Uninstall the Process Server.
3. Uninstall the Hub Server.

Uninstalling the Hub Store

You can uninstall the Hub Store by dropping the Hub Store schemas and removing the user logins for the Hub Store schemas. Before you drop the Hub Store schemas, use the Hub Console to unregister the Hub Store schemas.

You must have administrator privileges to drop the Hub Store schemas.

1. Start the Hub Console.

2. Click the **Databases** tool under the **Configuration** workbench.

   The **Database Information** page appears.
3. Click **Write Lock** > **Acquire Lock**.
4. From the list of databases, select the Operational Reference Store to unregister.
5. Click the **Unregister database** button.

   The Databases tool prompts you to confirm unregistering the Operational Reference Store.
6. Click **Yes**.
7. Use the command line processor to connect to the IBM Db2 instance.
8. Use the following command for each Hub Store schema to drop the schema:

```
DROP TABLE ERRORSCHEMA.ERRORTABLE
CALL SYSPROC.ADMIN_DROP_SCHEMA('<Schema Name>', NULL, 'ERRORSCHEMA', 'ERRORTABLE')
```

Uninstalling the Process Server in Graphical Mode

You can uninstall the Process Server in graphical mode.

Uninstalling the Process Server in Graphical Mode On UNIX

To uninstall the MDM Hub, you must remove the Process Server. You must perform the steps to uninstall the Process Server for each Process Server in the MDM Hub implementation.

1. Stop the application server.
2. Navigate to the following directory:

   <infamdm_install_directory>/hub/cleanse/UninstallerData
3. Run the uninstaller.

```
./"Uninstall Informatica MDM Hub Cleanse Match Server"
```
4. Click **Uninstall**.

   When the uninstallation process is complete, the Uninstall Complete window appears.
5. Click **Done**.

Uninstalling the Process Server in Graphical Mode On Windows

To uninstall the MDM Hub, you must remove the Process Server. You must perform the steps to uninstall the Process Server for each Process Server in the MDM Hub implementation.

1. Stop the application server.
2. Click **Start** and then click **Programs** > **Infamdm** > **Hub** > **Cleanse** > **Uninstaller Data** > **Uninstall Informatica MDM Hub Cleanse Match Server**.

   The Uninstall introduction window appears.
3. Click **Uninstall**.

   When the uninstallation process is complete, the Uninstall Complete window appears.
4. Click **Done**.

Uninstalling the Hub Server in Graphical Mode

You can uninstall the Hub Server in graphical mode.

Uninstalling the Hub Server in Graphical Mode on UNIX

To uninstall the MDM Hub, you must remove the Hub Server from the MDM Hub implementation.

1. Ensure that you stop the application server.
2. Navigate to the following directory:

   `<infamdm_install_directory>/hub/server/UninstallerData`
3. Run the uninstaller.

   `./"Uninstall Informatica MDM Hub Server"`

   The Uninstall introduction window appears.
4. Click **Uninstall**.

   When the uninstallation process is complete, the Uninstall Complete window appears.
5. Click **Done**.

Uninstalling the Hub Server in Graphical Mode on Windows

To uninstall the MDM Hub, you must remove the Hub Server from the MDM Hub implementation.

1. Ensure that you stop the application server.
2. Click **Start** and then click **Programs** > **Infamdm** > **Hub** > **Server** > **UninstallerData** > **Uninstall Informatica MDM Hub Server**.

   The Uninstall introduction window appears.
3. Click **Uninstall**.

   When the uninstallation process is complete, the Uninstall Complete window appears.
4. Click **Done**.

Uninstalling the Resource Kit in Graphical Mode

You can uninstall the Resource Kit in graphical mode.

Uninstalling the Resource Kit in Graphical Mode on UNIX

To uninstall the Resource Kit, you must remove the Resource Kit from the MDM Hub implementation.

1. Stop the application server.
2. Navigate to the following directory:

   `<infamdm_install_directory>/hub/resourcekit/UninstallerData`
3. Run the following command:

   `./"Uninstall Informatica MDM Hub Resource Kit"`

   The **Uninstall Informatica MDM Hub Resource Kit** window appears.

4. Click **Uninstall**.

   The **Uninstall Complete** window appears with a list of items that could not be removed.
5. Click **Done**.
6. Manually remove the following directory:

   `<infamdm_install_dir>/hub/resourcekit`

Uninstalling the Resource Kit in Graphical Mode on Windows

To uninstall the Resource Kit, you must remove the Resource Kit from the MDM Hub implementation.

1. Stop the application server.
2. Navigate to the following directory:

   `<ResourceKit_install_dir>\deploy\UninstallerData`
3. Double-click `Uninstall Informatica MDM Hub Resource Kit.exe`

   The **Uninstall Informatica MDM Hub Resource Kit** window appears.
4. Click **Uninstall**.

   The **Uninstall Complete** window appears with a list of items that could not be removed.
5. Click **Done**.
6. Manually remove the following directory:

   `<infamdm_install_dir>\hub\resourcekit`

Uninstalling the Process Server in Console Mode

You can uninstall the Process Server in console mode on UNIX. If you installed the Process Server in console mode, uninstall the Process Server in console mode.

1. Go to the following directory:

   `<infamdm_install_dir>/hub/cleanse/UninstallerData`
2. Type the following command to run the uninstaller:

   `./"Uninstall Informatica MDM Hub Cleanse Match Server"`

Uninstalling the Hub Server in Console Mode

You can uninstall the Hub Server in console mode on UNIX. If you installed the Hub Server in console mode, uninstall the Hub Server in console mode.

1. Go to the following directory:

   `<infamdm_install_dir>/hub/server/UninstallerData`
2. Type the following command to run the uninstaller:

   `./"Uninstall Informatica MDM Hub Server"`

Uninstalling the Resource Kit in Console Mode

You can uninstall the Resource Kit in console mode. If you installed the Resource Kit in console mode, uninstall the Resource Kit in console mode.

1. Go to the following directory:

   On UNIX. `<infamdm_install_dir>/hub/resourcekit/UninstallerData`

   On Windows. `<infamdm_install_dir>\hub\resourcekit\UninstallerData`

2. Run the following command from the command prompt:

   On UNIX. `"Uninstall Informatica MDM Hub Resource Kit.bin" -i console`

   On Windows. `"Uninstall Informatica MDM Hub Resource Kit.exe" -i console`

Manually Undeploying the Process Server

You might need to manually undeploy the Process Server from the WebSphere environment.

- Use the WebSphere Application Server Administrative Console to manually undeploy the `siperian-mrmcleanse.ear` file.

  For more information, see the WebSphere documentation.

Manually Undeploying the Hub Server

You might need to manually undeploy the Hub Server from the WebSphere environment.

- Use theWebSphere Application Server Administrative Console to undeploy the following deployment files:

Deployment File Name	Description
siperian-mrm.ear	Required. The Hub Server application.
entity360view-ear.ear	Optional. The Entity 360 framework.

For more information, see the WebSphere documentation.

INDEX