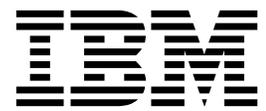


IBM Cúram Social Program Management
Version 7.0.0

*Development Environment Installation
Guide*



Note

Before using this information and the product it supports, read the information in "Notices" on page 39

Edition

This edition applies to IBM Cúram Social Program Management v7.0.0 and to all subsequent releases unless otherwise indicated in new editions.

Licensed Materials - Property of IBM.

© **Copyright IBM Corporation 2011, 2016.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Planning the installation . . . 1

Overview of the installation steps	1
Cúram Development Environment	2
A typical Cúram development environment	3
Disk space requirements	4
DB2 database encoding options	4
Oracle database encoding options	6

Chapter 2. Installing prerequisite products 7

Installing Apache Ant	7
Configuring Apache Ant	7
Installing a DBMS	7
H2 database	8
H2 database installation	8
IBM DB2 database	8
Installing IBM DB2 for Linux, UNIX, and Windows	8
Configuring for circular transaction logging	9
Oracle database	9
Installing the Oracle database	10
Postinstallation configuration for the Oracle database	10
Installing Java SE and Java EE	11
Configuring Java SE and Java EE	11
Installing an enterprise application server	12
WebSphere Application Server	12
Installing WebSphere Application Server.	12
Postinstallation configuration for WebSphere Application Server	12
WebSphere Application Server Network Deployment	12
Installing WebSphere Application Server Network Deployment	12
Setting the WebSphere Application Server environment variable	13
Oracle WebLogic Server	13
Installing Oracle WebLogic Server	13
Postinstallation steps for Oracle WebLogic server	13

Chapter 3. Installing the Cúram software 15

Preparing to install the Cúram software	15
Source control	15
Installing the Cúram platform software	15
Installing the Cúram enterprise modules	17
Installing the Cúram program-based offerings	17
Installing a fix pack.	18
Installing an interim fix	18
Uninstalling the application	18

Chapter 4. Completing postinstallation configuration tasks 21

Setting the Cúram environment variables	21
Configuring the H2 database	21
Encrypting passwords	22
Setting the H2 mode	22
Setting Multi-Version Concurrency Control (MVCC)	22
Starting the H2 Web Console	23
Providing a DB2 License File	23
Creating a database on IBM DB2	24
Creating and configuring a DB2 database with scripts	24
Testing the configuration	24
Running build commands for the server and client applications	25
Starting the XML server	25

Chapter 5. Installing the integrated development environment 27

Eclipse-based IDE	27
Installing an Eclipse-based IDE	27
Postinstallation steps for Eclipse and Apache Tomcat	27
Configuring Eclipse.	27
Configuring Tomcat	28
Configuring the Sysdeo Eclipse Tomcat Launcher plug-in	29
Using Eclipse to validate the tabbed configuration artifacts	29
Setting up the Cúram projects	30
Importing the server projects into Eclipse	30
Importing the client projects into Eclipse	30
Importing the Citizen Portal project into Eclipse	31
Importing the Cúram Business Intelligence and Analytics Viewer project into Eclipse	31
Rational Application Developer.	32
Installing Rational Application Developer	32
Postinstallation steps for the Rational Application Developer IDE	32
Configuring Rational Application Developer	32
Updating the boot class path	32
Importing the server projects into Rational Application Developer.	33
Importing the client projects into Rational Application Developer.	33
Running the application in Rational Application Developer.	34

Chapter 6. Installing the Rational Software Architect modeling tool 35

Installing IBM Installation Manager	35
Installing Rational Software Architect.	35

Postinstallation configuration for Rational Software Architect	36
Installing the Cúram plug-ins	36
Setting up your Rational Software Architect workspace.	36

Chapter 7. Getting started with the Cúram Application Development Environment	37
Starting the server	37

Starting the clients	37
Deploying the Cúram application	37
Logging on to IBM Cúram Social Program Management	38
Logging on to the Citizen Portal application	38

Notices	39
Privacy Policy considerations	40
Trademarks	41

Chapter 1. Planning the installation

Read the following information to understand what you must do to install a Cúram Development Environment for IBM® Cúram Social Program Management.

The Cúram Development Environment is supported on Microsoft Windows only.

Overview of the installation steps

On completion of the following installation steps, you are ready to start development on IBM Cúram Social Program Management.

1. Review the Cúram Application Development Environment (ADE) information and decide which software to install.
2. Review the *IBM Cúram Social Program Management Supported Prerequisites* technote to identify the supported versions of your selected software. For more information, see the related link.
3. Download any of the software that you need from IBM Passport Advantage or from other software vendor websites as appropriate. For information about the Cúram components, see the related link to the download document.
4. Install and configure Apache Ant, a database, Java SE, and Java EE. You can optionally install a supported application server for the Java SE and Java EE requirements.
5. Install the IBM Cúram Social Program Management software in the sequence that is outlined in the next steps, starting with the IBM Cúram Social Program Management Platform. By default, the platform installer also installs the following platform enterprise modules:
 - IBM Cúram Verification Engine
 - IBM Cúram Evidence Broker
 - IBM Cúram Life Event Management

You must have a valid license to use the platform enterprise modules.

6. Review the *IBM Cúram Social Program Management Release Notes* and complete any relevant postinstallation steps.
7. Install the enterprise modules for which you have a valid license:
 - IBM Cúram Universal Access
 - IBM Cúram Outcome Management
 - IBM Cúram Provider Management
 - IBM Cúram Social Enterprise Collaboration
 - IBM Cúram Business Intelligence and Analytics
 - IBM Cúram Appeals
 - IBM Cúram Archiving
 - IBM Cúram Identity Intelligence
 - IBM Cúram Provider Match with Watson™
8. Review the *IBM Cúram Social Program Management Release Notes* for each of the enterprise modules and complete any relevant postinstallation steps.
9. Install any of the program-based offerings. The program-based offerings are the following:

- IBM Cúram Workers Compensation - Do not install any other program-based offerings with this offering.
 - Either IBM Cúram Income Support or IBM Cúram Income Support for Medical Assistance (but not both). You can then install:
 - a. IBM Cúram Income Support Screening followed by
 - b. IBM Cúram Business Intelligence and Analytics Reports for Income Support
 - IBM Cúram Child Welfare. You can then install:
 - a. IBM Cúram Business Intelligence and Analytics Reports for Child Welfare
 - IBM Cúram Youth Services
 - IBM Cúram Child Care
 - Any additional assets such as demonstrations
10. Review the Release Notes for each of the chosen program-based modules, and complete any relevant postinstallation steps.
 11. If applicable, install the latest Fix Pack.
 - Review the Fix Pack Release Notes, and complete any relevant pre-installation steps for the platform and your installed enterprise and program-based modules.
 - Install the Fix Pack.
 - Review the Fix Pack Release Notes, and complete any relevant postinstallation steps for the platform and your installed enterprise modules.
 12. Complete the product postinstallation configuration steps.
 13. Install the integrated development environment (IDE).
 14. Install IBM Rational® Software Architect.
 15. Start the required server and client processes and check that you can log in to the server and the client.

Related information:

-  [IBM Cúram Social Program Management Supported Prerequisites](#)
-  [IBM Cúram Social Program Management V7.0.0 Download Document](#)

Cúram Development Environment

The Cúram Development Environment includes the Cúram Application Development Environment (ADE). It also includes various supporting IBM and third-party Java development tools with which you can customize Cúram applications.

For development only, you can choose to work with the H2 database that is bundled with the Cúram software. However, if you want to deploy and test applications, you must install and configure one of the supported enterprise database and application server combinations.

- Cúram Application Development Environment (ADE)**
 - Cúram Server Development Environment (SDEJ)
 - Cúram Client Development Environment (CDEJ)

Developer tools

The following developer tools can be used to develop Cúram applications:

- A build tool
 - Apache Ant.
- A database management system (DBMS)
 - H2 database (development only).
If you prefer, you can install and configure one of the supported enterprise databases in your environment. For example, you might prefer the stability of an enterprise database for critical product demonstrations.
 - IBM DB2 database.
 - Oracle database.
- An integrated development environment (IDE)
 - Eclipse and Apache Tomcat.
 - IBM Rational Application Developer.

Although technical support is not provided for any particular IDE, the listed IDEs have been tested for use with Cúram software. However, you can choose to use any Java IDE for Cúram development.

- A modeling tool
 - IBM Rational Software Architect.
- The Java Platform
 - Java Platform, Standard Edition.
 - Java Platform, Enterprise Edition.

A typical Cúram development environment

You can set up the Cúram development environment in various ways to suit your environment and product preferences. However, this typical installation configuration provides you with a development environment from which you can start to develop applications. If you prefer to change any of the product choices, install the alternative tools instead.

- A build tool
 - Apache Ant.
- Java Platform
 - Oracle Java SE and Java EE.
- DBMS
 - H2 database (development only).
- IBM Cúram Social Program Management
- Integrated development environment (IDE)
 - Eclipse and Apache Tomcat.
- A modeling tool
 - IBM Rational Software Architect.

Note: Deploying Cúram applications on IBM WebSphere Application Server is out of the scope of this representative configuration. You can develop Cúram applications without using an enterprise application server. However, you can optionally include WebSphere Application Server in your representative configuration. Adding an application server completes the set of tools that are required for developing and deploying Cúram applications.

For the exact versions of these products, see the *IBM Cúram Social Program Management Supported Prerequisites* technote. For more information, see the related link.

Related information:

 [IBM Cúram Social Program Management Supported Prerequisites](#)

Disk space requirements

For an installation that is based on IBM DB2[®], you need the following estimated disk space to install all of the prerequisite products. The representative configuration requires less disk space requirements.

- 1 GB based on IBM WebSphere Application Server
- 1 GB based on WebSphere Application Server Network Deployment.
- 1 GB based on Oracle WebLogic Server.

Approximately 2.5 GB extra is required for an installation that is based on Oracle database.

You can optionally install on a free drive or partition, if available. If you choose this option, you must change the default installation location for any products that install to the C: drive by default:

DB2 database encoding options

If you plan to install DB2, read this important background information about issues with DB2 database encoding and related sizing information. During a DB2 installation, you must identify your requirement for SBCS or MBCS data. Depending on your choice, you might have to complete some extra post-configuration steps before you build the Cúram database.

What is the issue?

For a multi-byte character set (MBCS) or encoding, DB2 processes columns by their byte size, not their character length. Therefore, for multi-byte characters, a CHAR, VARCHAR, or CLOB column might store fewer characters than the column length specification indicates, depending on the actual character length.

Consider the following example:

- A CHAR or VARCHAR column that is modeled with a length of 16.
- The 16-character string, "Marge says hello" that does not have accented character, requires 16 bytes for storage in a single-byte character set (SBCS).
- A similar 16-character string, but with accented characters, "Márge says hélló", requires 18 bytes for storage in UTF-8, a multi-byte character set (MBCS).

For the single-byte data, the string fits and processing is successful. For the multi-byte data, the string does not fit, resulting in overflow errors at run time. The Cúram web client usually captures and reports field size errors in a user-friendly manner. In this case, the user receives an "un-handled server exception" error, which is an underlying SQL Code **-302** error. This is because the client does not capture this size mismatch as it checks the number of characters, and not the byte length.

How Cúram addresses the issue

Cúram provides modeling and build-time capabilities to resize its database columns to address this issue. These capabilities are described further in the *Cúram Modeling Reference Guide* and *Cúram Server Developer's Guide*.

As Cúram provides support for multiple languages, support for MBCS data is enabled by default with the maximum expansion set. These expansion settings are appropriate to ensure that new users, testing environments, and so on, do not encounter any errors because of their language, encoding, and database sizing. Also, users can find they require MBCS data when they import or paste data from other applications into their Cúram system. However, these defaults might not be appropriate for all environments. The following section describes some considerations for changing these expansion settings.

What you must consider

It is important to carefully consider your data encoding requirements regarding DB2 and Cúram to avoid unexpected behavior with how the database stores characters.

The preceding example represents a boundary case in that the data length matches the maximum column width. In many cases it is unlikely that, even with MBCS characters, an overflow situation will occur. Most data does not reach the maximum defined size. However, you must be prepared for the possibility of these error situations.

Use the database character set encoding appropriate to your application and environment. If possible, consider using an SBCS and encoding that supports your requirements. For example, CP1252 supports most Western European characters. However, CP1252 (or other SBCS encodings) might not support characters from different or "broader" character sets or encodings (for example, UTF-8) that users might be used to copying and pasting into their browser for Cúram.

When installing your DB2 database, you must only identify your requirement for SBCS or MBCS data and be prepared to take appropriate action before you build your Cúram database:

- If you require characters that use multiple bytes, then you must consider whether the default Cúram settings are appropriate. The necessary database space is dependent on various factors such as the following factors.
 - The specific character sizes. In DB2 and DB2 for z/OS, MBCS data can range from 1 to 4 bytes.
 - The frequency of MBCS characters, which can depend on the application, language, locale, column usage within the application, and so on.
 - The information density of the language and locale. For example, while some languages can require more bytes per character, each character can represent more information than, for instance, an alphabetic character and might fit into a field without any size adjustment.

For more information about MBCS data sizing considerations, see the *Cúram Server Developer's Guide* section, *Planning for DB2 MBCS Data*.

- If an SBCS is adequate, plan to disable database expansion as described in the *Cúram Server Developer's Guide* section: *Planning for DB2 MBCS Data*.

Related concepts:

“IBM DB2 database” on page 8
IBM DB2 is supported as a database server.

Oracle database encoding options

If you plan to install Oracle, it is important to consider the character set for the data that you plan to store in your database when you configure the database for use with IBM Cúram Social Program Management.

For Oracle, there are two parameters to consider: `NLS_CHARACTERSET` and `NLS_LENGTH_SEMANTICS`.

- The `NLS_CHARACTERSET` parameter details the allowable character set of any data that is loaded to the database, generally `AL32UTF8` is recommended by Oracle.
- The `NLS_LENGTH_SEMANTICS` determines how Oracle interprets length specifiers on `CHAR` and `VARCHAR` columns. To handle supplementary characters, for example, ß in German, where the storage of the character would be 2 bytes and might overrun the length of a defined column; set the `NLS_LENGTH_SEMANTICS` parameter to `CHAR`. This setting directs the database to size columns with a character length rather than byte length.

Chapter 2. Installing prerequisite products

You must install certain prerequisite products before you install the Cúram software.

If you want to use the Java SE and Java EE that are provided with IBM WebSphere Application Server or Oracle WebLogic Server, then you must install those product as prerequisites.

If you want to use an enterprise database for Cúram development, then you must install DB2 or Oracle as a prerequisite.

For the exact versions of these products, see the *IBM Cúram Social Program Management Supported Prerequisites* technote. For more information, see the related link.

Related information:

 [IBM Cúram Social Program Management Supported Prerequisites](#)

Installing Apache Ant

Apache Ant from the Apache Jakarta project is a build tool that is based on Java.

Before you begin

You do not need to extract the Apache Ant compressed file into a directory called ant. The file extracts to the `apache-ant-version` directory.

Procedure

1. Download the Ant compressed file from the Apache website.
2. Extract the file to a directory of your choice on your computer. For example, extract `apache-ant-version-bin.zip` to `C:\apache-ant-version`. The installation is now complete.

Configuring Apache Ant

You must create Microsoft Windows environment variables and update your Microsoft Windows path for Apache Ant.

Procedure

1. Create an `ANT_HOME` system environment variable with the value set to the Apache Ant installation directory.
2. Add `%ANT_HOME%\bin` to the `PATH` environment variable.
3. Create an `ANT_OPTS` system environment variable with the value `-Xmx756m`.

Installing a DBMS

The H2 database is supported as a development database. Both IBM DB2 and Oracle database are supported as database servers.

Note: No particular character set is required for the installation and setup of the DBMS. Configure a character set that is appropriate for the character range that is needed in the application.

H2 database

H2 is an SQL database engine that is written in Java™ that implements the JDBC API. A browser-based console application is included.

Limitations in the support of the H2 database:

- For development use only.
- Not supported at run time.
- EAR files cannot be built for this database.
- You cannot run the **configure** target while this database is in use. This target automatically configures the application server.

For more information about build files and their targets, see the *Cúram Server Developers Guide*.

H2 database installation

The H2 database is provided with the Cúram software and does not require any specific installation steps. The self-contained database is in the %CURAMSDEJ%\drivers\h2.jar file after you install the Cúram platform software.

If you plan to use the H2 database, you can select the DB2 option when you are installing the Cúram software. Enter values for DB2 so you can proceed through the wizard. After you complete the installation, you must edit the database properties in the %CURAM_DIR%\EJBServer\project\properties\Bootstrap.properties file for the H2 database instead. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Cúram\Development.

Related information:



H2 Database Engine User Guide

To see the H2 database user guide, click this link.

IBM DB2 database

IBM DB2 is supported as a database server.

Note: It is possible to use IBM Cúram Social Program Management against a remote database with the DB2 Universal Type 4 Driver. The driver is supplied with the Server Development Environment for Java (SDEJ).

Related concepts:

“DB2 database encoding options” on page 4

If you plan to install DB2, read this important background information about issues with DB2 database encoding and related sizing information. During a DB2 installation, you must identify your requirement for SBCS or MBCS data.

Depending on your choice, you might have to complete some extra post-configuration steps before you build the Cúram database.

Installing IBM DB2 for Linux, UNIX, and Windows

Ensure that your account has administrative privileges and then follow the DB2 installer instructions to complete a default installation. You do not need to manually create a DB2 database. The platform software provides Ant scripts that you can run as a postinstallation step to create a basic test database.

Note the following options that are presented during a default installation:

- The *Name* and *Password* of the administrator account. Use an account and password as per the standards and requirements of your site and DB2. If it is an existing user, that user must be a member of the Administrator group. The informational message about OLE DB support component can be safely ignored.
- Certain editions of the DB2 installer support federated databases. If the installer presents an option that is defaulted to **This machine will be the instance-owning database partition server**, then change this option to **This machine will be a single-partition database server**.
- You must choose MBCS or SBCS, depending on your requirements. If you are unsure of what database encoding option to select, see the related information about data encoding.

Configuring for circular transaction logging

When you use a database with circular transaction logging enabled, certain transactions can exceed the available log file space and fail. To avoid this issue, either use archive logging or set the available log size and quantity appropriately until it meets the needs of the transaction.

About this task

A common point for this failure is when the `prepare.application.data` Ant target is running, as this target publishes all the CER rule sets on the system. This Ant target is typically run after a clean database build. If the log is too small, it can result in an `SQLCODE -964` error.

You can use the following example to help you to increase the DB2 log file size and quantity. The exact amount of log file storage that is required varies from system to system. For more information about increasing the number and size of the log files available, see the specific documentation for your database.

Procedure

1. Open a command prompt and enter `db2cmd`.
2. Enter the following command:

```
db2 connect to db_name user db_user_name using db_password
```

Where *db_name*, *db_user_name*, *db_password* are the credentials of the database.

3. Enter the following commands:

```
db2 update db cfg for db_name using logfilsiz log_file_size
db2 update db cfg for db_name using logprimary primary_log_files
db2 update db cfg for db_name using logsecond secondary_log_files
```

Where the temporary values are as follows:

- The log file size. Set *log_file_size* to 1024.
 - The number of primary log files. Set *primary_log_files* to 50.
 - The number of secondary log files. Set *secondary_log_files* to 100.
4. Restart the database by entering the following commands:

```
db2stop
db2start
```

Oracle database

Oracle database is supported as a database server.

Note: It is possible to use a remote database by using the Oracle Type 4 Driver that is supplied with the SDEJ.

Installing the Oracle database

Assuming that no previous versions of Oracle are installed, you can complete a typical Oracle database server installation.

Postinstallation configuration for the Oracle database

Complete the following postinstallation task on the Oracle database.

Note: The postinstallation tasks require connecting to Oracle as the privileged 'sys' user. Immediately after you install Oracle, the password for this user is *change_on_install*. Oracle requires that you change this password.

Creating an Oracle role for application servers:

The application needs certain privileges to use the Oracle XA interface. Later, when you configure the application, the user name under which the server connects to Oracle is specified. The appropriate privileges must be assigned to this user name for the server to work successfully.

About this task

An easy way to bundle together the various privileges that are required is to create an Oracle role. Privileges can be granted to this role. Later this role can be granted to your users, granting all the privileges that are associated with that role.

The following commands create a role that is called CURAM_SERVER and give it the necessary privileges. A user named CURAM_USER is then assigned that role and given the password PASSWORD. You run these commands inside an Oracle SQLPlus window.

Procedure

1. To run the commands from SQLPlus, type the following at a command prompt:

```
sqlplus ?/? as SYSDBA
```

2. Enter the following commands:

```
CREATE ROLE "CURAM_SERVER";
GRANT RESOURCE TO "CURAM_SERVER";
@%ORACLE_HOME%\RDBMS\ADMIN\xaview.sql
GRANT SELECT ON V$XATRANS$ TO PUBLIC;
GRANT SELECT ON PENDING_TRANS$ TO PUBLIC;
GRANT SELECT ON DBA_2PC_PENDING TO PUBLIC;
GRANT SELECT ON DBA_PENDING_TRANSACTIONS TO PUBLIC;
GRANT EXECUTE ON DBMS_SYSTEM TO CURAM_SERVER;
CREATE USER CURAM_USER IDENTIFIED BY
PASSWORD DEFAULT TABLESPACE "USERS" TEMPORARY TABLESPACE "TEMP";
GRANT "CONNECT", "CURAM_SERVER", UNLIMITED TABLESPACE TO <CURAM_USER>;
```

Where CURAM_USER and PASSWORD are the database user credentials.

Configuring for circular transaction logging:

When you use a database with circular transaction logging enabled, certain transactions can exceed the available log file space and fail. To avoid this issue, either use archive logging or set the available log size and quantity appropriately until it meets the needs of the transaction.

A common point for this failure is when the `prepare.application.data` Ant target is running, as this target publishes all the CER rule sets on the system. This Ant target is typically run after a clean database build.

For information about increasing the number and size of the log files available, see the specific documentation for your database. The exact amount of log file storage that is required varies from system to system.

Configuring redo log space:

Certain Cúram transactions that have significant insert activity are affected by the available redo log space. To avoid this issue, allocate the appropriate redo log space for your system.

A common point for this failure is when the `prepare.application.data` Ant target is running, as this target publishes all the CER rule sets on the system. This Ant target is typically run after a clean database build.

For information about allocating the appropriate size for the redo logs, see the Oracle documentation. The exact amount of activity and the required redo log space varies from system to system.

Installing Java SE and Java EE

You can install a stand-alone Java SE and Java EE, or use the Java SE and Java EE that are included with each supported application server. Follow the Oracle documentation to install the Oracle Java SE and Java EE. No further installation steps are required if you intend to use the versions that are included with the application server.

Configuring Java SE and Java EE

Regardless of which Java SE and Java EE you use, you must complete the following configuration steps.

About this task

You might need multiple versions of Java SE and Java EE installed on a single computer. For this reason, you can choose the scope for these Microsoft Windows environment variables. For example, system wide, or through a script file or symbolic links.

Procedure

1. Create a `JAVA_HOME` environment variable that points to the installed Java SE.
2. Place `%JAVA_HOME%\bin` at the beginning of the `PATH` environment variable.
3. Create a `J2EE_JAR` environment variable that points to the installed Java EE JAR file.
 - For WebSphere Application Server, point to `%WAS_HOME%\lib\j2ee.jar`.
 - For Oracle WebLogic Server, point to `%WLS_HOME%\lib\weblogic.jar`.
 - For Oracle Java Platform EE SDK 5, point to `installation_directory\lib\j2ee.jar`. Where *installation_directory* is the directory where you installed the software. By default `C:\Sun\SDK`.

Installing an enterprise application server

Deploying onto an application server is outside the scope of this information. However, you can decide to install an enterprise application server to give you a complete set of development tools. You can also install the application server if you want to use the Java SE and Java EE that are bundled with WebSphere Application Server or WebLogic Server.

An application server must be installed on both a computer that builds applications for deployment, and a computer that hosts the deployed application.

Note: You must not install an application server on a computer that has an underscore in the computer name.

WebSphere Application Server

WebSphere Application Server is supported as an enterprise application server.

Installing WebSphere Application Server

Install WebSphere Application Server from the installation media.

If you are optionally installing WebSphere Application Server as a service, you must create a user account in advance. This user is then used as the credentials for the service. This user account must have administrator privileges.

Important:

- Do not install WebSphere Application Server in a directory that contains spaces in the name, such as the default Program Files directory.
- Do not install the WebSphere Application Server sample applications. The sample application Apache Derby data source results in a class path conflict with the application web client use of Derby.

Postinstallation configuration for WebSphere Application Server

Complete the following tasks to configure WebSphere Application Server.

Setting the WebSphere Application Server environment variable:

Set the required Microsoft Windows environment variable for WebSphere Application Server.

Procedure

Set the WAS_HOME environment variable to the server directory of the WebSphere Application Server installation. For example, C:\WebSphere\AppServer

WebSphere Application Server Network Deployment

IBM WebSphere Application Server Network Deployment is an optional component that manages one or many IBM WebSphere Application Server editions and offers advanced deployment services, including clustering, edge services and high availability for distributed configurations.

Installing WebSphere Application Server Network Deployment

Install WebSphere Application Server Network Deployment from the installation media. The default installation is used, without any configuration changes.

If you want to install the Deployment Manager as a service, you must create user account in advance so that it can be used as the credentials for the service. This user account must have administrator privileges.

Note:

- Do not install WebSphere Application Server Network Deployment to a directory that contains spaces in the name, such as the default Program Files directory.
- Do not install the WebSphere Application Server sample applications. The sample application Apache Derby data source results in a classpath conflict with the application web client's use of Derby.

Setting the WebSphere Application Server environment variable

Set the required Microsoft Windows environment variable for WebSphere Application Server.

Procedure

Set the WAS_HOME environment variable to the server directory of the WebSphere Application Server installation. For example, C:\WebSphere\AppServer

Oracle WebLogic Server

Oracle WebLogic Server is supported as an enterprise application server.

Installing Oracle WebLogic Server

Complete the following steps to install Oracle WebLogic Server.

Procedure

Run the Oracle installer. When prompted in the installation wizard, choose the following options:

- For the installation type, choose a custom installation.
- For products and components, clear all options except the **WebLogic Server** branch.
- Accept the default not to install as a Windows service.
- Do not run the Quickstart on exit.

Postinstallation steps for Oracle WebLogic server

After you install Oracle WebLogic server, you must set up the WLS_HOME environment variable.

Procedure

Set the WLS_HOME environment variable to the server directory of the Oracle WebLogic Server installation. For example, *home_directory*\wlserver_version\server or *home_directory*\wlserver_10.3\server. Where *home_directory* is the home directory that is specified during the WebLogic Server installation.

Configuring the Weblogic application transaction timeout settings:

Set minimum application transaction timeout settings for Weblogic.

About this task

Note: These settings are for testing/development purposes and are not advised for production systems where they represent minimum values. Tuning should be used to find the correct settings for your environment. These settings are recommended for the following Cúram components:

- IBM Cúram Child Care
- IBM Cúram Child Welfare
- IBM Cúram Income Support
- IBM Cúram Income Support for Medical Assistance
- IBM Cúram Workers Compensation
- IBM Cúram Youth Services

In order to test these products on WebLogic during development cycles, these steps to increase the JTA timeout settings have been used successfully.

Procedure

1. Log onto the localhost console via `https://localhost:7002/console`.
2. Go to JTA and change the **Timeout Seconds** to 600 seconds.

Setting the default permanent generation size:

Set the default permanent generation size for Weblogic.

About this task

Note: These settings are for testing/development purposes and are not advised for production systems where they represent minimum values. The default permanent generation size configured for Weblogic is not compatible with testing the following Cúram components on Weblogic:

- IBM Cúram Income Support or
- IBM Cúram Income Support for Medical Assistance

Procedure

1. Add the following entry to `EJBServer\project\properties\Bootstrap.properties`:
`curam.server.jvm.permgen.size=-XX:PermSize=128m -XX:MaxPermSize=384m`
2. Restart the server.

Chapter 3. Installing the Cúram software

An IBM Cúram Social Program Management installation is required in the Cúram Application Development Environment. IBM Cúram Social Program Management is distributed as a series of installations. The IBM Cúram Social Program Management Platform is the required starting point, followed by a series of optional components. Ensure that you have the required installers and all of the required information before you start the installation.

About this task

For installation on Microsoft Windows operating systems, each installer is provided as a .exe file.

You need the Cúram platform installer, plus individual installers for each component you plan to install. Typically, you install enterprise and solution modules, and optional associated add-ons.

During the installation, all installation process and the installation history are saved to the following log files:

- /Installer/CuramInstaller.log
- /Installer/Installhistory.txt

Preparing to install the Cúram software

Gather the required information and complete any required preparation steps before you start the installation.

Before you start, you must have the following information:

- A list of the components of the application for which you are licensed.
- The organization name and address.
- The database server name, port number, database name, database user name, and database password.

Source control

If your project is stored in a source-controlled environment, you might take the following approach to the installation, depending on your requirements:

- Install IBM Cúram Social Program Management and any optional components.
- Place the installed code base under source control.
- To support future installations, the files in the /Installer folder must also be maintained under source control.
- Use your source-control procedures to distribute the environment to other developers.

Installing the Cúram platform software

Run the platform installer to install the base platform upon which all the other modules are installed.

Before you begin

Review the release notes for the software that you are about to install.

Note: If you are installing a modification release, install it in a new installation location. Do not install it in a previous installation location. For example, do not install version 6.0.5 in the same installation location as version 6.0.4.

Procedure

1. Copy all of the installation files from the media to a temporary directory.
2. Navigate to the directory that contains the installer you want to run.
3. Run the installer by double-clicking the IBM Curam SPM Platform Development.exe file.
4. On the welcome page, click **Next**.
5. Do not alter the default installation path. Accept the default installation path by clicking **Next > OK** to create the directory. If the target installation directory exists, you are prompted to overwrite the existing files.
6. Choose the installation language from the Language dropdown. If you leave the Language dropdown blank, it defaults to English - US
7. If you are licensed for Universal Access and require additional languages, select the check box beside the supported languages.
8. Click **Next** to accept the license type.
9. The components that you are licensed for are displayed. Confirm the components that you require are selected and click **Next**.
10. Enter the **Organization Name** and **Organization Address**, and click **Next**.
11. Select the **Cúram Database Platform** that you plan to use with the application. For example, **DB2/UDB**. If you intend to use the H2 database, select **DB2/UDB** during the installation. Enter values for DB2 so you can proceed through the wizard. After you complete the installation, you must edit the database properties in the %CURAM_DIR%\EJBServer\project\properties\Bootstrap.properties file for the H2 database instead. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
12. In the **Database Account Logon** and **Database Account Password** fields, enter the values as defined during the database installation and click **Next**.
13. In the **Database Server Name** field, enter the fully qualified host name of the computer on which you installed the database. Enter a value for the **Database Server Port** field, for example, 50000 for DB2. Enter the database name in the **Curam Database Name** and click **Next**. The installation files are extracted. This step can take several minutes.
14. When the extraction completes, click **Next**. The installation files are configured based on the inputs that are provided in the previous steps.
15. Click **Next > Done** to complete the platform installation.
16. Review the release notes for the platform software and complete any postinstallation steps that are relevant to your configuration.

Related information:

 [IBM Cúram Social Program Management release notes](#)

To see the IBM Cúram Social Program Management Platform Release Notes, click this link.

Installing the Cúram enterprise modules

Install each of the Cúram enterprise modules that you are licenced for.

Before you begin

IBM Cúram Income Support has a dependency on IBM Cúram Outcome Management. Install IBM Cúram Outcome Management if you plan to install IBM Cúram Income Support.

IBM Cúram Income Support for Medical Assistance does not have a dependency on IBM Cúram Outcome Management.

Review the release notes for the enterprise modules you are about to install.

Procedure

1. Copy all of the installation files from the media to a temporary directory.
2. Change to the directory that contains the installer you want to run.
3. Run the installers for each of the required enterprise modules.:
4. Verify your installation by checking the installation history to determine what was installed. A text file for each installer and the `InstallHistory.txt` file that lists all of the installers that ran are found in the installation folder. For example, `C:\IBM\Curam\Development\Installer`.
5. Review the release notes for each of the enterprise modules and complete any postinstallation steps that are relevant to your configuration.

Related information:

 [IBM Cúram Social Program Management release notes](#)
To see the IBM Cúram Release Notes for your enterprise modules, click this link.

Installing the Cúram program-based offerings

Install the Cúram program-based offerings to which you are entitled.

Before you begin

Ensure that you have identified the correct offerings and installation sequence as described in the planning overview.

Review the release notes for the program-based offerings that you are about to install.

Procedure

1. Copy all of the installation files from the media to a temporary directory.
2. Change to the directory that contains the installer you want to run.
3. Run the installer for the solution module.
4. Verify your installation by checking the installation history to determine what was installed. A text file for each installer and the `InstallHistory.txt` file that lists all of the installers that ran are found in the installation folder. For example, `C:\IBM\Curam\Development\Installer`.
5. Review the release notes for the solution module and complete any postinstallation steps that are relevant to your configuration.

Related information:

 IBM Cúram Social Program Management release notes
To see the IBM Cúram Release Notes for your solution module, click this link.

Installing a fix pack

After you install IBM Cúram Social Program Management, install any required fix packs. A fix pack involves moving from one minor release to another within a point version, for example, from 6.0.5.0 to 6.0.5.1.

Procedure

1. Go to the IBM Fix Central website and search the site for your product and version to locate the fix pack for your installation.
2. Download and extract the fix pack installation image.
3. Read the latest version of the IBM Cúram Social Program Management fix pack release notes. Take note of any preinstallation steps, requirements, restrictions, installation steps, and postinstallation steps that might apply to the fix pack.
4. Read the readme file that is downloaded with the fix pack, taking note of any preinstallation steps, requirements, restrictions, installation steps, and postinstallation steps.
5. Run the fix pack installer, following the instructions in the fix pack documentation.
6. When prompted to move obsolete files, select **Yes**. Moving the files can take up to 30 minutes during which no progress indicator is displayed.
7. Click **Finish** to complete the installation.

Installing an interim fix

After you install IBM Cúram Social Program Management and the relevant fix pack, install the latest interim fix. Interim fixes contain important bug fixes and optimizations. An interim fix is a cumulative release that includes all previous interim fixes for the related fix pack release.

Before you begin

You must install the related fix pack before you install the latest interim fix.

Procedure

1. Go to the IBM Fix Central website and search the site for your product and version to locate the interim fix for your installation.
2. Download and extract the interim fix installation image.
3. Read the IBM Cúram Social Program Management interim fix release notes. Take note of any preinstallation steps, requirements, restrictions, installation steps, and postinstallation steps that might apply to the interim fix.
4. Complete pre-installation steps.
5. Run the Cúram installer.
6. Complete postinstallation steps.

Uninstalling the application

During installation of the Cúram software, an uninstallation file is created in the <install>\Uninstaller\uninstaller.jar directory. You can use this file to uninstall the application.

About this task

JAR files might be recognized as executable by being associated with a suitable launcher, such as javaw. If this is the case for your operating system, start the Cúram Uninstaller with the standard method that is supported by your operating system. For example, double-clicking the Cúram Uninstaller file.

Note: The uninstaller does not reset any system variables that are set by a previous installation.

Procedure

1. Navigate to the <install>\Uninstaller\ directory.
2. Double-click the uninstaller.jar file to uninstall the Cúram software.

Chapter 4. Completing postinstallation configuration tasks

Before you proceed, complete the required postinstallation configuration tasks to ensure that the Cúram software is configured and working correctly with the prerequisite software.

Setting the Cúram environment variables

Before you proceed, you must run a script to set required Cúram environment variables.

Procedure

1. Change to the %CURAM_DIR% directory. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
2. Run the following command:
SetEnvironment.bat

Configuring the H2 database

To use the H2 database, you must update the `Bootstrap.properties` file with the correct credentials to connect to the H2 database. Ensure that you encrypt the password.

About this task

For example, here is typical H2 database content from a `Bootstrap.properties` file.

```
curam.db.type=h2
curam.db.name=curamdb
curam.db.username=curam
curam.db.password=qqnscP4c4+s=
# H2 directory.
# Default is home directory
# (i.e. C:/Documents and Settings/<username>). (Optional)
curam.db.h2.directory=C:/H2
# Mode remote|embedded
curam.db.h2.mode=embedded
# For remote mode also specify:
curam.db.serverport=9092
curam.db.servername=localhost
# Lock Time Out in ms. Default is 1000, i.e. 1 second. (Optional)
curam.db.h2.locktimeout=20000
# Property to disable MVCC. Default: true. (Optional)
curam.db.h2.mvcc=true
```

After you update the `Bootstrap.properties` file and rebuild the server and database, you can develop in the same way as you would with Oracle or DB2.

For more information about the `Bootstrap.properties` file, see the *Cúram Server Developers Guide*.

Procedure

1. Edit the `Bootstrap.properties` file.
2. Ensure that each of the database properties has the correct values for the H2 database.

Encrypting passwords

You must encrypt passwords before you put them in the `Bootstrap.properties` file.

Procedure

1. Open a command prompt and change to the `%CURAM_DIR%\EJBServer` directory. `%CURAM_DIR%` is the Cúram installation directory, which by default is `C:\IBM\Curam\Development`.
2. Issue the following command:

```
build encrypt -Dpassword=password
```

where *password* is the password you want to encrypt.
3. Copy the encrypted string in the output to the correct location in the `Bootstrap.properties` file. For example, the `curam.db.password` parameter.

Setting the H2 mode

Set your preferred mode for developing applications.

About this task

The following H2 modes are supported for application development:

Embedded mode

In embedded mode, an application opens the database from within the same JVM by using JDBC. This mode is the fastest and easiest connection mode. The disadvantage is that a database can be open in only one virtual machine (and class loader) at any time.

Remote mode

In remote mode, sometimes called client/server mode, an application opens the database remotely by using the JDBC or the ODBC API. Many applications can connect to the same database at the same time. The remote mode is slower than the embedded mode because all data is transferred over TCP/IP.

Procedure

1. Edit the `%CURAM_DIR%\EJBServer\project\properties\Bootstrap.properties` file. `%CURAM_DIR%` is the Cúram installation directory, which by default is `C:\IBM\Curam\Development`.
2. Specify the mode in the `curam.db.h2.mode` property. For example:

```
# Mode remote|embedded  
curam.db.h2.mode=embedded
```

Setting Multi-Version Concurrency Control (MVCC)

You can enable or disable Multi-Version Concurrency Control (MVCC). MVCC is enabled by default.

About this task

The MVCC feature allows higher concurrency than using exclusive table level or row level locks. When using MVCC in this database, delete, insert, and update operations only issue a shared lock on the table. An exclusive lock is still used

when adding or removing columns, when dropping the table, and when using SELECT... FOR UPDATE. Connections only see committed data, and their own changes.

That means, if connection A updates a row but has not committed the change, connection B sees the old value. Only when the change from connection A is committed, the new value is visible to other connections (read committed). If multiple connections concurrently try to update the same row, the database waits until it can apply the change, but at most until the lock timeout expires.

Procedure

1. Edit the %CURAM_DIR%\EJBServer\project\properties\Bootstrap.properties file. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
2. Specify true or false in the **curam.db.h2.mvcc** property. For example:
Property to disable MVCC. Default: true. (Optional)
curam.db.h2.mvcc=false

Starting the H2 Web Console

Start the H2 Web Console by running the org.h2.tools.Server class in h2.jar as follows:

```
java -cp %CuramSDEJ%\drivers\h2.jar org.h2.tools.Server -tcp -web
```

Use this updated command for 6.0.5.6 Fix Pack and later versions:

```
java -cp %CuramSDEJ%\drivers\h2-1.3.176.jar org.h2.tools.Server -tcp -web
```

You can access the H2 Web Console at the following URL:

```
http://localhost:8082/
```

The JDBC connection URL that you specify in the login screen is based on the curam.db.name, curam.db.username, and curam.db.h2.directory values in Bootstrap.properties. These values define the database name, SCHEMA name, and the database location in the file system. So, if your database name is curamdb, your user name is curam and curam.db.h2.directory defaults to your home directory, then your JDBC string would look like this example:

```
jdbc:h2:tcp://localhost/~ /curamdb;schema=curam;FILE_LOCK=SOCKET
```

For example, if the curam.db.h2.directory is C:/H2, then your JDBC string would look like this example:

```
jdbc:h2:tcp://localhost/file:C:/H2/curamdb;schema=curam;FILE_LOCK=SOCKET
```

Specify the values for **User Name** and **Password** as in your Bootstrap.properties file and then click the **Connect button** (or **Test Connect button**). When connected, a SQL text control is available.

Providing a DB2 License File

This postinstallation step is required for all users of IBM DB2 for Linux, UNIX, and Windows.

An empty `db2jcc_license_cu.jar` file exists in the `%CURAMSDEJ%\drivers` directory to allow for Eclipse class path dependencies in the CuramSDEJ project. Overwrite this empty JAR file with a real license for accessing IBM DB2 for Linux, UNIX, and Windows.

Copy the IBM DB2 `db2jcc_license_cu.jar` file from `DB2_directory\java\` `db2jcc_license_cu.jar` to `%CURAMSDEJ%\drivers`.

where

- `DB2_directory` is the DB2 installation path. For example, `C:\IBM\SQLLIB`.
- `%CURAMSDEJ%` points to the root CuramSDEJ location.

Creating a database on IBM DB2

If you are using IBM DB2, complete the following task to create the required database.

Creating and configuring a DB2 database with scripts

Ant scripts are provided for creating and configuring a basic test database. They use the database properties from your `Bootstrap.properties` file.

To create a database, issue the following commands:

```
ant -f %CURAMSDEJ%\util\db2_createdb.xml
ant -f %CURAMSDEJ%\util\db2_postconfig.xml -Ddb2.dir=db2_directory
ant -f %CURAMSDEJ%\util\db2_createdb.xml restart.db2
ant -f %CURAMSDEJ%\util\db2_optimizedbrecreation.xml
```

where `db2_directory` is the DB2 installation path. By default, `c:\IBM\SQLLIB`.

If you have any problems with creating the database, you can run the following script to drop the database and try again:

```
ant -f %CURAMSDEJ%\util\db2_createdb.xml dropdb
```

Note: The `db2_createdb.xml restart.db2` script restarts your DB2 system.

Testing the configuration

The Cúram Application Development Environment (ADE) includes a configuration test tool, which helps to confirm that the installation and third-party tools are set up correctly. You can run this tool to detect problems with your installation.

Before you begin

If you are using the H2 database, ensure that you complete these steps before you start this task.

- Build the server and the database.
- If you are using H2 in remote mode, ensure that the H2 Web Console is started.

Procedure

1. Open a command prompt.
2. Navigate to the `%Curam%/EJBServer` directory and issue the following commands:

```
build configtest
```
3. Ensure that the build is successful before proceeding.

Running build commands for the server and client applications

Before you can log on to a Cúram application or to the Universal Access home page, you must run a number of build commands.

Before you begin

Ensure that you are in the Cúram installation directory, denoted as %CURAM_DIR%, before you run each of these commands. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.

About this task

Important: Ensure that you are in the correct directory before you run each of these commands.

Procedure

1. Open a command prompt.
2. Navigate to the %Curam%/EJBServer directory and issue the following commands:

```
build clean server
build database
build prepare.application.data
build runExtractor
build createClasspaths
```
3. Navigate to the %Curam%/webclient directory and issue the following command:

```
build clean client
build external-client -Dapp=CitizenPortal
```
4. Ensure that the builds are successful before proceeding.

Starting the XML server

Before you start the Cúram application, you must start the XML server in your ADE.

Procedure

1. Change to the %CURAM_DIR%\CuramSDEJ\xmlserver directory. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
2. Run the following build target to start the XML server:

```
ant -f xmlserver.xml
```

Chapter 5. Installing the integrated development environment

An integrated development environment (IDE) is required to develop Cúram applications. Use this information to install an IDE for Cúram development.

Eclipse-based IDE

Use this information to understand how to install and configure Eclipse and Apache Tomcat.

Installing an Eclipse-based IDE

The following software is required for an Eclipse-based IDE. Download the software from the relevant third-party websites, and install the software by following the instructions that are provided on the website.

Eclipse IDE

An IDE that you can use to develop an application. If you are unsure about which Eclipse package to download, you can download and install the Eclipse IDE for Java EE Developers.

Tomcat

A servlet container that you can use to run the client web application.

Sysdeo Eclipse Tomcat Launcher

An open source Eclipse plug-in that integrates with a Tomcat installation to start Tomcat from within Eclipse.

Java SE and Java EE are also needed. You can use the Java SE and Java EE that were installed as prerequisites for the Cúram software for the Eclipse IDE.

Postinstallation steps for Eclipse and Apache Tomcat

Complete the following postinstallation steps to configure Eclipse and Apache Tomcat.

Configuring Eclipse

Complete the following tasks to configure the Eclipse integrated development environment.

Configuring the default Java SE for Eclipse:

You must ensure that Eclipse always starts with the correct Java SE. Multiple Java SE installations can be present on your computer from other products that are based on Java.

Before you begin

When starting Eclipse for this task, you must ensure that Eclipse starts with the correct Java SE by using one of the following methods: .

- Put the correct Java SE first on the Windows system path.
- Use the `-vm` command-line argument to the `eclipse.exe` command. For more information about Eclipse commands, see the Eclipse documentation.

Procedure

1. Start Eclipse by double-clicking the `eclipse.exe` executable file.
2. After you start Eclipse, select **Window > Preferences > Java > Installed JREs**. On the Installed JREs page, ensure that the check box for the correct Java SE is selected as the default.
3. To add default VM arguments, select the Java SE and click **Edit**.
4. In the **Default VM Arguments** field, enter `-Xmx1024M -XX:MaxPermSize=256m -Xms512m`.

Supported Eclipse text file encoding: In Eclipse, you can set the default text file encoding at the project level. Changing the text file encoding from the default is unsupported for IBM Cúram Social Program Management projects within Eclipse. This restriction does not affect your ability to save files in various encodings on a file-by-file basis.

For more information about file encoding, see the *Cúram Server Developer's Guide* and the *Cúram Web Client Reference Manual* .

Configuring Tomcat

After installation, you must update the default Tomcat configuration files with the appropriate settings.

About this task

UTF-8

By default, Tomcat assumes that requests are encoded with ISO-8859-1 instead of UTF-8. This default setting can break string handling if request parameters contained UTF-8 extended characters. For correct string handling, you must add the `useBodyEncodingForURI="true"` parameter to the `<Connector>` element of the `server.xml` configuration file.

POST Data limit

By default Tomcat limits POST data to 2 MB. This limit can cause an issue when you use rule sets, which can post data greater than this limit. To disable the POST limit in Tomcat, you can add the `maxPostSize="-1"` attribute to the `<Connector>` element of the `server.xml` configuration file.

Non-ASCII character in Java source files

Tomcat converts JSPs into servlets that are contained in UTF-8 encoded Java sources files by default (for multi-byte character support). These files are generated into the work folder of the project. The Sysdeo plug-in marks the work folder as an Eclipse source folder. If you use the Eclipse build command, the Java compiler expects system encoding sources files by default. If any source file in the work folder contains non-ASCII characters, such as `ú`, an `Invalid Character` compiler error is generated and you cannot access the page in a web browser.

The `keepgenerated` attribute prevents Tomcat from saving the source files in the work folder and avoids this problem. You can prevent this occurring by updating the Tomcat `web.xml` configuration file with a new `init-param` element.

The Eclipse compiler cannot be changed to compile UTF-8 source files because of a second source folder that is called `JavaSource` that contains files that are not in UTF-8 encoding. Changing this setting does not affect the use of the application in any way. The `keepgenerated` parameter can be

set to true if you want to view and debug through source files that are generated by Tomcat, but the error and browser access problem then occurs.

Procedure

1. Edit the `tomcat_install_dir\conf\server.xml` configuration file and update the `<Connector>` element as follows.
Where `tomcat_install_dir\conf\server.xml` is the directory where you installed Tomcat.
 - a. Change the default port number to `port="9080"`.
 - b. Add the `useBodyEncodingForURI="true"` attribute.
 - c. Optional: If you intend to use rule sets, add the `maxPostSize="-1"` attribute.

```
<Connector port="9080" maxThreads="150" minSpareThreads="25"
maxSpareThreads="75" enableLookups="false" redirectPort="8443"
acceptCount="100" connectionTimeout="20000" disableUploadTimeout="true"
useBodyEncodingForURI="true" maxPostSize="-1" />
```
2. Edit the `tomcat_install_dir\conf\context.xml` configuration file. Update the `<Context>` element to include a `reloadable="true"` attribute. For example,
`<Context reloadable="true">`
3. Edit the `tomcat_install_dir\conf\web.xml` configuration file. Update the `org.apache.jasper.servlet.JspServlet` servlet with a new `init-param` element with the value `false`. For example,

```
<init-param>
<param-name>keepgenerated</param-name>
<param-value>>false</param-value>
</init-param>
```

Configuring the Sysdeo Eclipse Tomcat Launcher plug-in

Configure the Sysdeo Eclipse Tomcat Launcher to add a toolbar to Eclipse for starting and stopping Apache Tomcat.

Procedure

1. In Eclipse, select **Window > Preferences > Tomcat**.
2. Set the value of **Tomcat version** to **Version 6.x**.
3. Set the value of **Tomcat home** to the location of the Tomcat installation.
4. Expand **Tomcat** and select **JVM Settings**.
5. In **Append to JVM Parameters**, click **Add**, enter `-Xmx512m`, and click **OK**.

Using Eclipse to validate the tabbed configuration artifacts

You can set up Eclipse to validate the tabbed configuration files with the correct schema.

Open the Eclipse Preferences dialog by selecting **Window > Preferences** and complete the following steps:

- Select **XML > XML Catalog**.
- Click **Add...** to add an entry.
- For the **Location**, point at the schema file (for example, `tab.xsd`) in the `%CURAMSDEJ%\lib` directory.
- Leave the rest as defaults and click **OK**.
- Repeat for each of the schema files for the tabbed configuration artifacts. For the full list of schema files, see the *Cúram Client Developer's Guide*. Click **OK** to exit the XML Catalog window.
- Select **General > Editors > File Associations**.

- Click **Add...** to add an entry: *.tab.
- Select the new *.tab entry and click **Add** to add the XML Editor as the **Associated Editor**.
- Repeat for all the tabbed configuration artifact file extensions. For the full list of extensions, see the *Cúram Client Developer's Guide*.
- Select **General > Content Types**.
- Expand **Text** and select **XML**.
- Click **Add** to enter a file association for XML content and click **OK**. Do this step for each of the file extensions.
- Click **OK** to save the preference changes.

Setting up the Cúram projects

Complete the following tasks to import and configure the Cúram Eclipse projects. When importing a project, do not import the files into the workspace.

Importing the server projects into Eclipse

In Eclipse, import and configure the server project.

Procedure

1. Import the project (.project file) in the CuramSDEJ folder in the development installation.
2. Import the project (.project file) in the EJBServer folder in the development installation.
3. After you import the projects, you must resolve a number of errors that are caused by missing class path variables. To resolve these errors, click **Window > Preferences > Java > Build Path > Classpath Variables**, and create the following class path variables.
 - a. J2EE_JAR - The JAR file of your Java EE implementation.
 - b. JAVAMAIL_HOME - The folder that contains mail.jar and activation.jar files for your Java EE implementation.

Note: If your version of Java EE does not contain these files, you can copy these files from another Java installation to any directory. Then, configure JAVAMAIL_HOME to point to that directory.

Importing the client projects into Eclipse

In Eclipse, import and configure the client projects.

Procedure

1. Import the project (.project file) in the CuramCDEJ folder in the development installation.
2. Import the project (.project file) in the webclient folder in the development installation.
3. After you import the projects, right-click the **Curam** project and select **Properties > Tomcat**. Set the following properties:
 - Context Name: Set to /Curam.
 - Can update server.xml file: Selected
 - Mark this context as reloadable: Selected
 - Redirect context logger to Eclipse console: Selected
 - Subdirectory to set as application root: Set to WebContent.

4. Update the Tomcat server.xml file with an entry for the client application.
Right-click the project and select **Tomcat Project > Update Context Definition**.

Importing the Citizen Portal project into Eclipse

Building the Citizen Portal client project creates a Tomcat project that you must import into Eclipse.

Procedure

1. In Eclipse, select **File > Import**.
2. In the Import wizard, select **General > Existing Projects into Workspace > Next**.
3. Select the %CURAM_DIR%\webclient\build\apps\CitizenPortal root directory and click **Finish**. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.

Importing the Cúram Business Intelligence and Analytics Viewer project into Eclipse

In Eclipse, import the Cúram Business Intelligence and Analytics Viewer project, and configure the Eclipse environment to view Cúram Business Intelligence and Analytics content.

For more information about Business Intelligence reporting, see the *BIRT Developers Guide* for more details.

1. Import the project into Eclipse by going to **File > Import...** Select **General > Existing Projects into Workspace** and click **Next**. Select the root directory to be %CURAM_DIR%\BIApp\CuramBIRTViewer and click **Finish**. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
2. Select the CuramBIRTViewer project in Eclipse, then open the preferences dialog by selecting **Window > Preferences** and expand **Java > Compiler** and set **Compiler compliance level:** to 1.5.
3. Select the CuramBIRTViewer project in Eclipse, right-click and select **Tomcat Project > Update context definition**.
4. Open the preferences dialog by selecting **Window > Preferences** and expand **Tomcat > JVM Settings** and for the **Classpath** text box select the **Directory** button and add in the full path to the %CURAM_DIR%\EJBServer\project\properties.
5. You must now publish all Cúram Business Intelligence and Analytics content to the BIRT Viewer. From the directory %CURAM_DIR%\BIContent execute the command **build client.birt**.
6. To verify that the content is available, start Tomcat and check `http://localhost:9080/CuramBIRTViewer`.
7. From the directory %CURAM_DIR%\BIContent execute the command **appbuild biapp.configureBIRTviewer.resources**.

The following shows a sample output after executing the command:

```
biapp.configureBIRTviewer.resources:
[echo] -----
[echo] start : initializing BIRT Viewer plugins - oda plugins
[copy] Copying 5 files to C:\CC\afarrell\BIAppMAIN-LINE\BIApp-VOB\BIApp\CuramBIRTViewer\We
[copy] Warning: Could not find file C:\CC\anonymous\BIAppMAIN-LINE\BIApp-VOB\BIContent\${s
[echo] end   : initializing BIRT Viewer plugins - oda plugins
[echo] -----
```

```
BUILD SUCCESSFUL
Total time: 1 second
```

If you are developing new Cúram Business Intelligence and Analytics content, please see the *BIRT Developers Guide* for more details on how to set up a development environment.

Rational Application Developer

Use this information to install and configure IBM Rational Application Developer.

Installing Rational Application Developer

Complete a standard installation of Rational Application Developer.

Postinstallation steps for the Rational Application Developer IDE

You must configure Rational Application Developer and import the Cúram server and client projects.

Configuring Rational Application Developer

Set the following Rational Application Developer preference to ensure that Rational Application Developer works with the application.

Procedure

1. Switch off build automatically by clicking **Project -> Build Automatically** and ensuring that the option is not selected.
2. Switch off validation by going to **Windows -> Preferences -> Validation**, clicking the **Disable All** and clicking **OK**.
3. Select **Windows > Preferences > Java > Installed JREs** and set the installed JRE to the correct value.
4. Set the following file associations by going to **Windows > Preferences > Workbench > File Associations**:
 - Add *.uim to the file types.
 - Add the XML Editor as the associated editor for .uim files.
5. Add the schema to the XML catalog. Select **Windows > Preferences > Web and Xml > Xml Catalog**, select **User Specified Entries**. Click **Add** and set the following values:
 - Location: <CuramCDEJ>/lib/curam/xml/schema/uim.xsd
 - Key Type: Schema Location
 - Key: file://Curam/UIMSchema.xsd

Updating the boot class path

If you plan to use the IBM WebSphere Java SDK, you must add some XML API JAR files to the boot class path. The files are provided with the SDEJ.

About this task

The following XML API JAR files are required:

- %CURAMSDEJ%\lib\xalan-2.7.1.jar
- %CURAMSDEJ%\lib\serializer-2.7.1.jar
- %CURAMSDEJ%\lib\xerces-2.9.1.jar

Procedure

You can set the boot class path as a JVM parameter. For example:

```
-Xbootclasspath/p: %CURAMSDEJ%\lib\xalan-2.7.1.jar;%CURAMSDEJ%\lib  
\serializer-2.7.1.jar;%CURAMSDEJ%\lib\xerces-2.9.1.jar
```

Importing the server projects into Rational Application Developer

Import the EJBServer server project and CuramSDEJ project and configure them for use in Rational Application Developer.

1. Import the project (.project) located in the CuramSDEJ folder in the development installation.
2. Import the project (.project) located in the EJBServer folder in the development installation.

Importing the client projects into Rational Application Developer

The webclient client project needs to be imported and configured to run the embedded WebSphere server.

1. Extract the %CURAM_DIR%\CURAMCDEJ\doc\RAD\RAD.zip file into the %CURAM_DIR% directory. That is, the project base directory. Overwrite the existing files if requested. These default files are for use with Rational Application Developer only. %CURAM_DIR% is the Curam installation directory, which by default is C:\IBM\Curam\Development.

2. Add the following to the %CURAM_DIR%\webclient\classpath file.

```
<classpathentry kind="con"  
path="org.eclipse.jst.server.core.container/  
com.ibm.ws.ast.st.runtime.runtimeTarget.v70/was.base.v7">  
<attributes>  
<attribute name="owner.project.facets" value="jst.web"/>  
</attributes>  
</classpathentry>  
<classpathentry kind="con"  
path="org.eclipse.jst.j2ee.internal.web.container"/>  
<classpathentry kind="con"  
path="org.eclipse.jst.j2ee.internal.module.container"/>
```

3. Add the following to the %CURAM_DIR%\webclient\project file to replace the current buildSpec and natures nodes:

```
<buildSpec>  
<buildCommand>  
<name>org.eclipse.wst.jsdt.core.javascriptValidator</name>  
<arguments>  
</arguments>  
</buildCommand>  
<buildCommand>  
<name>org.eclipse.jdt.core.javabuilder</name>  
<arguments>  
</arguments>  
</buildCommand>  
<buildCommand>  
<name>org.eclipse.wst.common.project.facet.core.builder</name>  
<arguments>  
</arguments>  
</buildCommand>  
<buildCommand>  
<name>org.eclipse.wst.validation.validationbuilder</name>  
<arguments>  
</arguments>  
</buildCommand>  
</buildSpec>  
<natures>  
<nature>
```

```

    org.eclipse.jem.workbench.JavaEMFNature
</nature>
<nature>
    org.eclipse.wst.common.modulecore.ModuleCoreNature
</nature>
<nature>
    org.eclipse.wst.common.project.facet.core.nature
</nature>
<nature>org.eclipse.jdt.core.javanature</nature>
<nature>org.eclipse.wst.jsdt.core.jsNature</nature>
</natures>

```

4. Import the project (.project) located in the CuramCDEJ folder in the development installation.
5. Import the project (.project) located in the webclient folder in the development installation.
 - Create a webclient\work directory to match the required source directory in the project classpath, if it is not present already.
6. Import the project (.project) located in the CuramEAR folder in the development installation.

Running the application in Rational Application Developer

To run IBM Curam Social Program Management on Rational Application Developer, the relevant servers must be started.

Complete the following steps in Rational Application Developer.

1. The server is started as an RMI Application by running the StartServer class. From the EJBServer/ReferencedLibraries/core.jar (default package), select the StartServer class and select the **Run as Application** option from the **Run** button.
2. Enable the server view, by clicking **Window -> Show View -> Other -> Server -> Servers**.
3. Add CuramEAR to the **WebSphere Application Server** and start the server. To do this step, right-click the **WebSphere Server** and click **Add and Remove Projects**. Move **CuramEAR** from **Available Projects** to **Configured Projects** and click **Finish**. Check that both **WebSphere Server** and **CuramEAR** have a status of started and that **WebSphere Server** has a state of synchronized.
4. From the EJBServer/ReferencedLibraries/core.jar (default package) in your project, select the RMILoginClient class and then select the **Run as Application** option from the **Run** button. The application is now ready for use.
5. Log in from the URL: <http://localhost:9081/Curam/AppController.do>

Chapter 6. Installing the Rational Software Architect modeling tool

IBM Rational Software Architect is an Eclipse-based UML modeling tool that is required to do server development.

Installing IBM Installation Manager

IBM Installation Manager is a tool that you can use to install and maintain your software packages.

Procedure

Install IBM Installation Manager with the default options.

Installing Rational Software Architect

The exact installation steps for installing Rational Software Architect can vary depending on the edition and version of your software.

Procedure

1. Start the IBM Installation Manager by clicking **Start > All Programs > IBM Installation Manager > IBM Installation Manager**.
2. From the **File** menu, select **Preferences**.
3. From the Repositories window, select **Add Repository**.
4. From the Add Repository window, select **Browse** to add an entry that points to your Rational Software Architect installation file and click **OK**.
5. From the Repositories window, ensure that this entry is the only selected repository and click **OK**.
6. From the IBM Installation Manager window, select **Install**.
7. From the Install Packages window, select appropriate version check box and click **Next**.
8. From the Prerequisite window, read the information that is displayed and click **Next**.
9. From the Licenses window, read the information and select **I accept the terms in the license agreements** check box and click **Next**.
10. From the Location window, select **Browse** to select the installation directory or use the default value. For Architecture Selection, update the default values if required. Click **Next**.
11. From the Eclipse window, click **Next**. Choose not to extend an Eclipse version. If you want to use Rational Software Architect in an existing Eclipse instance, see the product documentation:
<https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/Rational+Installation+Wiki/page/Install+into+an+Existing+Eclipse+instance>
12. From the Features window, select the translations to install and click **Next**.
13. From the Features to install window, select the **Rational Rose model import** check box to install the IBM Rational Rose profile that is required by migrated models in Eclipse. Click **Next**.

14. From the Common Configurations window, click **Next**.
15. From the Summary window, click **Install**.

Postinstallation configuration for Rational Software Architect

Complete the following steps to configure Rational Software Architect.

Installing the Cúram plug-ins

You must install the Cúram plug-ins to enable modeling support.

Procedure

1. From the Rational Software Architect installation directory, for example, the C:\Program Files\IBM\SDP directory, create a dropins directory.
2. From the dropins directory, create a new file called `rsa_plugin.link` that contains the path to the plug-ins for Rational Software Architect. Ensure that you use forward slashes. For example,
path=C:/Curam/CuramSDEJ/rsa
3. Edit the Microsoft Windows shortcut that starts Rational Software Architect to pass the **-clean** option so that the plug-ins are picked up. For example:
...\\eclipse.exe -clean -product com.ibm.rational....
4. Start or restart Rational Software Architect.

Setting up your Rational Software Architect workspace

Complete the following steps to set up your Eclipse workspace in Rational Software Architect.

Procedure

1. Start IBM Rational Software Architect by clicking **Start > All Programs > IBM Software Delivery Platform > IBM Rational Software Architect**.
2. From the Workspace Launcher, select **Browse** to navigate to the location where you want your Rational Software Architect workspace to be stored and click **OK**.
3. From the Overview window, hover over the top right icon to display Workbench. Select the **Workbench** icon.
4. Select **File > Import**.
5. From the Import window, expand the General folder and select **Existing Project into Workspace** and click **Next**.
6. From the Import Projects window, select **Browse** and select %CURAM_DIR%\EJBServer for the Select root directory and click **Finish**. %CURAM_DIR% is the Cúram installation directory, which by default is C:\IBM\Curam\Development.
7. Repeat steps 4-6 to import the %CURAMSDEJ% root directory. You are now ready to start using Rational Software Architect.

Chapter 7. Getting started with the Cúram Application Development Environment

The installation is now complete. Use this information to help you to get started with the Cúram Application Development Environment (ADE).

Starting the server

Start the server so that you can log in and test the installation.

About this task

The server is started as a Java process that starts three threads:

tnameserv

The Transient Name Server to facilitate a JNDI lookup service for finding resources such as Java classes.

RMI Server

Server application process.

JMSLite

JMS Message Engine. For more information about JMSLite, see the *Workflow Reference Guide*.

Procedure

1. Check that the database is running.
2. In Package Explorer, expand **EJBServer > components > core > lib**, right-click on **core.jar**, and select **Run As > Java Application**.
3. In the Select Java Application window, select **StartServer** and click **OK**.

Starting the clients

Start Tomcat and the RMILoginClient so that you can log in and test the installation.

Procedure

1. On the Eclipse menu, click the Start Tomcat button on the Sysdeo Tomcat Launcher toolbar.
2. In Package Explorer, expand the **EJBServer > components > core > lib**, right-click on **core.jar**, select **RMILoginClient** and select **Run as application**.
3. In the Select Java Application window, select **RMILoginClient** and click **OK**.
4. In the RMILoginClient window, enter your credentials.

Deploying the Cúram application

If you want to test your applications with an enterprise application server, you can deploy the application and web services application to an application server.

If you want to deploy Cúram applications as part of your development process, then you need one of the following supported application server and DBMS combinations:

- IBM WebSphere Application Server and IBM DB2.
- IBM WebSphere Application Server and Oracle database.
- Oracle WebLogic Server and Oracle database.

For full information about deployment, see the *IBM Cúram Deployment Guide for WebSphere Application Server* or *IBM Cúram Deployment Guide for WebLogic Server*

Logging on to IBM Cúram Social Program Management

You can access the IBM Cúram Social Program Management application from any supported browser. From here, you can access Cúram features that are based on your role. For example, you can log on to administer the system.

Procedure

1. Enter the following URL:

`https://server_name:port/Curam/AppController.do`

where:

- *server_name* is the name of the server where you installed the application.
- *port* is the port for the application. By default, the port number is 9080 for Apache Tomcat, 7002 for WebLogic, or 9044 for WebSphere Application Server.

2. Log on with the appropriate role.

Option	Description
sysadmin	The System Administrator user has access to technical administration features.
admin	The Administrator user has access to administration features.

Logging on to the Citizen Portal application

You can access the Citizen Portal from any supported browser.

Procedure

1. Enter the following URL:

`https://server_name:port/CitizenPortal/application.do`

where:

- *server_name* is the name of the server where you installed the application.
- *port* is the port for the application. By default, the port number is 9080 for Apache Tomcat, 7002 for WebLogic, or 9044 for WebSphere Application Server.

2. Log on as the required user.

Notices

This information was developed for products and services offered in the United States.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-17

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd. 19-21, Nihombashi

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Privacy Policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings

can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies or other similar technologies that collect each user's name, user name, password, and/or other personally identifiable information for purposes of session management, authentication, enhanced user usability, single sign-on configuration and/or other usage tracking and/or functional purposes. These cookies or other similar technologies cannot be disabled.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at " Copyright and trademark information " at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other names may be trademarks of their respective owners. Other company, product, and service names may be trademarks or service marks of others.



Printed in the Republic of Ireland