MISYS Open Source Solutions | Code for Healthcare Innovation

openxds

IHE Xds.b Document
Registry & Repository
and XCA Gateways

Project Created: 05/08/2009

Project Categorization

charter
(projects/categorization.root.cat1002)

Project Members

Total Project Members: 4

Project Administrators:
Prashanth Chandra M
(/sf/global/do/viewUser/kondamullapudi)

OpenXDS 1.0.1 Installation

Software Requirements

Make sure to have these requirements:

- JDK: Java SDK 1.6 or above
- Databases: PostgreSQL, or MySQL, or SQL Server
- Application server (optional): Any (Tested with Jboss-4.2.3 GA)
- Operating System: all major operating systems are supported

To build from source code, need to have these additional requirements:

- Maven: Maven 2.0.6 or above
- Subversion Client: any svn client such as CollabNet svn client, or Elicpse IDE

Installation from Binary Code

The binary build can be downloaded from OpenXDS 1.0.1 file release (/sf/frs/do/listReleases/projects.openxds/frs.openxds_releases).

- Unzip the build to any folder, referred to as OpenXDS_HOME
- Go to the Set Up Databases (page1120##_setupdb) section to configure the databases
- If you wish, go to the OpenXDS_HOME/conf/actors directory to modify the actor configuration. See more information in the Actor Configuration (page1121) section
- Proceed to the Run OpenXDS servers (page1120##_runserver) section for execution

Installation from Source Code

A subversion client such as Collabnet Subversion Client, Eclipse IDE, or any other type of subversion client is required to download the OpenXDS source code. The following section illustrates how to use the Collabnet Subversion Client a download and build the source code, respectively.

1. Build Source Code Using Collabnet Subversion Client

- Download and install Maven (http://maven.apache.org/download.html) 2.0.6 or above
- Download and install Collabnet Subversion Client (http://www.collab.net/downloads/subversion)
- Add your OHT login credentials to the maven setting, so that Maven can download the required libraries from the OpenXDS internal repository automatically using your user and password
  - Open or create a new *settings.xml* file under the .m2 directory of your local maven repository, which is by default under your operating system user home directory
  - In settings.xml, add the following *servers* tag to the parent *settings* tag

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
 http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <servers>
   <server>
    <id>openxds</id>
    <username>username</username>
    <password>password</password>
   </server>
  </servers>
</settings>
```

  - Replace the username and password with your OHT user name and password
  - Save the change and close the file
- Create a work directory. At a command line, go to that work directory, and execute the following command to get OpenXDS 1.0.1 source code:

```
svn checkout https://openxds.projects.openhealthtools.org/svn/openxds/tags/openxds-1.0.1-20100721 (//openxds.projects.openhealthtools.org/svn/openxds/tags/openxds-1.0-2010
```

- At the same command line, go to the openxds directory (the one containing a pom.xml file), and execute *mvn install -DskipTests* to build the source code. This usually takes a few minutes to complete, so just be patient
- When the above step is done, a build directory (referred to as OpenXDS_HOME in the following sections) is generated at the root of the work directory
- Go to the Set Up Databases (page1120##_setupdb) section to configure the databases
- If you wish, go to the OpenXDS_HOME/conf/actors directory to modify the actor configuration. See more information in the Actor Configuration (page1121) section
- Proceed to the Run OpenXDS servers (page1120##_runserver) section for execution

2. Build Source Code Using Eclipse IDE

- Download and install Maven (http://maven.apache.org/download.html) 2.0.6 or above
- Download and install Eclipse (http://www.eclipse.org/downloads/packages/release/ganymede/sr2) 3.4 or 3.5 JEE version
- Download and install Maven Eclipse plug-in m2eclipse (http://m2eclipse.codehaus.org/)
- Check out the OpenXDS source code from the svn repository
  - Create a new work space
  - Install a JDK 1.6 by selecting Window -> Preferences -> Java -> Installed JREs -> Add -> Standard VM. Make sure to select this JDK as the default Standard VM in the workspace
  - Use a Maven Checkout Wizard by selecting New -> Checkout Maven Projects from SCM
  - On the *Checkout as Maven project from SCM* screen and in the SCM URL field, select svn and enter
    `https://openxds.projects.openhealthtools.org/svn/openxds/tags/openxds-1.0.1-20100721` (//openxds.projects.openhealthtools.org/svn/openxds/tags/openxds-1.0-20100309)
    to download the openxds-1.0.1 source code.
  - Click the Finish button to download the source code and the maven libraries. You may be prompted to enter your OHT login and password. An OpenXDS Eclipse project with all its child modules will be created. It will take a while
  - If you see a validation error from the openxds-web module, simply disable the XML schema validator and the XML Validator from Window -> Preferences -> Validation
- At a command line, go to the openxds directory (the one containing a pom.xml file), and execute *mvn install -DskipTests* to build the source code. This usually takes a few minutes to complete
- When the above step is done, a build directory (referred to as OpenXDS_HOME in the following sections) is generated at the root of the work space
- Go to the Set Up Databases (page1120##_setupdb) section to configure the databases
- If you wish, go to the OpenXDS_HOME/conf/actors directory to modify the actor configuration. See more information in the Actor Configuration (page1121) section
- Proceed to the Run OpenXDS servers (page1120##_runserver) section for execution

Set Up Databases

The out-of-the-box binary build supports PostgreSQL (the tested version is PostsgresSQL 8.3).
MySQL and SQL Server are also supported, but their drivers are not included in the binary distribution. If you choose MySQL or SQL Server, you need to provide its driver, and add it to the manifest classpath of openxds-1.0.1.jar.

- Download and install your database. For PostgreSQL, you can download from here (http://www.postgresql.org/)
- Create openxds database
  - Create a new openxds database and a database user named openxds, and assign the user to the database. The default user and password are both openxds. If different, go to the***openxds.properties***, ***omar.properties*** and ***repository.jdbc.cfg.xml*** files under the OpenXDS_HOME directory (If running in an application server, these files are located in the WEB-INF/classes folder of the openxds web ap change the relevant database connection properties
  - Run the corresponding database schema script based on the database of your choice. The schema file is available in the **OpenXDS_HOME/misc** folder. Executing the script will create tables and pre-load some data
- Create log2 database
  - Create a new log2 database and a new database user/password: logs/xdslogs, and assign this user to the log2 database. If the database URL, user or password is different from the default, make sure to update the logs.db prope openxds.properties file

Run OpenXDS Servers

The OpenXDS Servers including Document Repository, Document Registry, Initiating Gateway, and Responding Gateway can be run in either a stand-alone mode or in an application server. They also can be run separately in different ma different JVMs

Run in a Stand Alone Mode

- To execute the OpenXDS servers, at a command line, go to the OpenXDS_HOME directory, and execute the following command
  java -jar openxds-1.0.1.jar
  The console message will indicate whether each server is started or not
- If you did not change the actor configuration files, the default XDS connection information is as follows (replace localhost with your server ip):
  - Document Submission (ITI-41): **http://localhost:8020/axis2/services/xdsrepositoryb**
  - Document Stored Query (ITI-18): **http://localhost:8010/axis2/services/xdsregistryb**
  - Document Retrieval (ITI-43): **http://localhost:8020/axis2/services/xdsrepositoryb**
- Proceed to the Testing (page1120##_testing) section to verify the installation

Deployment and Run in an Application Server

The OpenXDS servers can also run in an application server

1. Deployment

- To deploy the OpenXDS web application, go to the OpenXDS_HOME/misc directory, copy and unwrap the openxds-web.war file to your application server deployment folder
- Go to the openxds.properties file under the WEB-INF/classes folder, and uncomment the *ihe.actors.dir*, *axis2.repo.dir* and *xds.schema.dir* properties to use the ones for war, and comment out the other corresponding values
- If you have changed the database default connection parameters such as database URL, user and password, make sure you change them in the *openxds.properties*, *omar.properties* and*repository.jdbc.cfg.xml* files under the folder
- If you wish, create an actors directory by copying the OpenXDS_HOME/conf/actors directory, and modify the actor configuration based on your needs. For more information see Actor Configuration (page1121)

2. Start up the OpenXDS servers

- Start up your application server
- Open a browser (currently IE is supported) on the machine where your application server runs, go to this URL **http://<host>:<port>/openxds-web/ (http://<host>:<port>/openxds-web/)** (replace <host> with system hostname

the application server port), and click the Configuration tab.

- In the Load configuration file text box, browse or enter the path of the actors/IheActors.xml file created in the above deployment section, and click the **LoadActors** button
- Select the check boxes before the actors that you would like to run, and click the Start button. The messages on the console will indicate whether each actor is started successfully or not
- If you did not change the actor configuration files, the default XDS connection information is as follows (replace localhost with your server ip):
  - Document Submission (ITI-41): **http://localhost:8020/axis2/services/xdsrepositoryb**
  - Document Stored Query (ITI-18): **http://localhost:8010/axis2/services/xdsregistryb**
  - Document Retrieval (ITI-43): **http://localhost:8020/axis2/services/xdsrepositoryb**
- Proceed to the Testing (page1120##_testing)section to verify the installation

Testing

There are two approaches to test the OpenXDS servers:
1. Integration test clients
2. NIST xdstest tool.

Using the Integration Test Clients

You will need to get the source code to run JUnit tests. The integration test clients are available from the openxds-core module. Use *ProvideAndRegisterDocumentSetTest* JUnit class to test document submission, *RegistryStoredQueryTes*
test stored query, and *RetrieveDocumentSetTest* JUnit class to test document retrieval
The testing data and configuration files are in the *src/test/resources* directory
The following properties need to be configured in the *test.properties* file
RepositoryUrl - the url used for the ProvideAndRegisterDocumentSet-b and RetrieveDocumentSet transactions
RegistryUrl - the url for the RegisterDocumentSet-b transaction
pixRegistryHostName - the host address or ip of the PIX Registry server of OpenXDS
pixRegistryPort - the port of the PIX Registry server of OpenXDS
patientId - the default test patient id
assigningAuthority - the default assigning authority of the test patients
validatePatient - true or false. Whether to validate the patient upon document submission
(Note: If validatePatient is set to true, the test clients must be on the same machine running the OpenXDS servers. We will provide remote testing capabilities in future versions.)
Be sure to start the OpenXDS repository and registry servers before executing the JUnit tests
To run document submission tests, execute the test scenarios in *ProvideAndRegisterDocumentSetTest* including
1. Test single document submission
2. Test multiple document submission
3. Test adding a document to a folder
4. Test adding a document addendum
5. Test document replacement
6. Test document transformation
7. Test document transformation with replacement

To run document stored query tests, execute the test scenarios in *RegistryStoredQueryTest* including
1. Test finding documents
2. Test finding folders
3. Test getting associations
4. Test getting documents and associations
5. Test get folders
6. Test get folders and contents
7. Test get folder for documents
8. Test get related documents
9. Test get submission set and contents
10. Test get submission sets
11. Test submission sets

To run *RetrieveDocumentSet* test, execute the *testRetrieveDocumentSetb* JUnit test method in *RetrieveDocumentSetTest*

Using the NIST XDS Toolkit

The NIST XDS Toolkit can be used as a test client to test the OpenXDS servers. To set up and run the NIST xdstest tool, please refer to the online resources here: NIST XDS Toolkit (http://ihewiki.wustl.edu/wiki/index.php/XDS_Main_Pag

Resources

OpenXDS Project Resources

- IHE (http://www.ihe.net/)
- OpenXDS Issue Tracking (/sf/tracker/do/listTrackers/projects.openxds/tracker)