Ultimate i2b2-PMN Deployment Guide

7/19/2012, rev. 8/8/2012, 10/24/12

# OVERVIEW

**Architecture**

The i2b2-PMN system has client, mid-tier and data server components that have to be all up and running in order for the server to work. Note that there are 9 components. During upgrades, these components may usually be upgraded independently as long as the interface between them are kept intact (see “Upgrade Process” section below).

On the client side, it requires:

1. i2b2 web client – Hosted on IIS7 with PHP5.
2. i2b2 client hive– Hosted on JBoss 4.2.2.GA with Axis2 v1.1.
3. PMN Client Adapter – Hosted on JBoss 4.2.2.GA. Replaces the CRC in the i2b2 client hive to interface with PMN (#6 below).

In the mid-tier is:

1. PopMedNet (PMN) – Implemented using Lincoln Peak’s DNS3 technology. This acts as a middleman through which all i2b2-style requests and responses are channeled and stored.

On the data server side:

1. DataMartClient (DMC) – Standalone Windows application installed on any desktop.
2. I2b2 data server hive – Hosted on JBoss 4.2.2.GA and Axis2 v1.1.
3. PMN Server Adapter – Hosted on JBoss 4.2.2.GA. The PMN Server Adapter converts DMC model processor calls into i2b2 messages and acts as an interface between DMC and i2b2 data server hive.

On both the client and data server side:

1. HQMF converter – Hosted on JBoss 4.2.2.GA. Converts i2b2 message to HQMF.
2. i2b2 ontology data – Stored in the ontology table in the i2b2 client database.

**Configuration**

On the client side, every user on i2b2 that can send a query through PMN needs an account on PMN with the username. The same password is not required. That is because all users are logged into PMN using the same proxy user and password.

On the data server side, every DMC logs into the data server using the same proxy user and password. If HQMF is deployed on a different JBoss server, then it can be configured with a different proxy user.

# TECHNOLOGY STACK

To support the above, the following foundation technologies are required to be deploy and run i2b2-PMN:

1. Windows OS
2. IIS7 with PHP5
3. SQL Server 2008 R2
4. DNS3
5. JDK 1.6
6. Ant 1.8
7. JBoss 4.2.2.GA (specific version required)
8. Axis2 v1.1 WAR Distribution (specific version required)
9. i2b2 v1.6

On top of these, we will deploy the i2b2-PMN integration packages consisting of:

* PMN Client Adapter (aka Protocol Translator)
* PMN Server Adapter
* HQMF Converter
* i2b2 Web Client modifications
* SHRINE ontology database (optional)

We are going to assume that setup of the technologies from 1-5 are familiar to the person installing this system and this documentation will proceed from #6 forward.

# PRELIMINARIES

**PMN (DNS3) - Additional Steps**

Install the I2B2Plugin.xml. This plugin will work for both redirect and remote plugin models. Be sure to modify the URLs in this file for your configuration.

**Java Environment Variables**

Set the following environment variables:

* JAVA\_HOME
* PATH=%JAVA\_HOME%\bin;%PATH%
* Verification: Type: java -version. It should display the Java version number installed.

**Install Ant**

1. Download Ant from Apache.
2. Unzip under [C:\](file:///C:\)
3. Set the environment variables:
   * ANT\_HOME
   * PATH=%ANT\_HOME%\bin;%PATH%
   * Verification: Type: ant. If installed correctly, Ant will complain that there is no build.xml file or attempt to run build.xml if it exists in the current directory.

# STANDARD JBOSS AND AXIS2

**Install JBoss and Axis2**

I2b2 works only with a specific version of JBoss, 4.2.2.GA and Axis2 WAR Distribution (Note: that's Axis2, not Axis), 1.1.

1. Download JBoss-4.2.2.GA and Axis2 v1.1 war version.
2. Unzip over [C:\.](file:///C:\)
3. In [C:\jboss-4.2.2.GA\server\default\deploy,](file:///C:\jboss-4.2.2.GA\server\default\deploy,) create i2b2.war folder.
4. Unzip axis2.war inside i2b2.war. That will deploy Axis2.
5. Configure JBoss listening port in C:\jboss-4.2.2.GA\server\default\deploy\jboss-web.deployer\server.xml.
6. To start JBoss, type: run.bat. In the scrolling display, watch for errors. There should not be any. A more extensive log is in [C:\jboss-4.2.2.GA\server\default\log.](file:///C:\jboss-4.2.2.GA\server\default\log.)
7. To stop JBoss, type: shutdown.bat -S.
8. Verification: Start JBoss. Go to URL: http://localhost:<port>/i2b2/services/listServices.
9. Optional: To run JBoss as Windows Service, download JBossNative 2.0.10. This is designed for JBoss 5, but should work for JBoss 4.2.2.
   * Copy service.bat to [C:\jboss-4.2.2.GA\bin.](file:///C:\jboss-4.2.2.GA\bin.)
   * Edit service.bat and modify SVCNAME, SVCDISP and SVCDESC.
   * Run: service.bat install.

NOTES:

* Setting up a second JBoss server on the same machine also requires changing:
  + In server\default\deploy\jboss-web.deployer\server.xml, the HTTP, HTTPS and AJP bind address ports.
  + In server\default\conf\jboss-service.xml, the webservice mbean, JNP, RMI ports.
  + In server\default\deploy\ejb3.deployer\META-INF\jboss-service.xml, the InvokerLocator port.
  + In server\default\deploy\jms\uil2-service.xml, the ServerBindPort.
* When installing as a service, change the service name and display name to differentiate it from the other JBoss instances running.

# I2B2, SHRINE and CEDD

**Preparation**

I2B2 has to be built from source (see note below for "Deploy i2b2 Hive" for deploying by cloning an existing i2b2 Hive).

1. Download i2b2 source. There are 4 sub-directories for: doc, src, db and webclient.
2. Download SHRINE source using Subversion (optional). For SHRINE, we only need their ontology database and their ontology mapping files (XMLs).

Notes:

* I2b2 consists of "cells", which are essentially web services. In the standard implementation, these cells are implemented in Axis2. A collection of these cells is called a "hive".
* Five of these cells are critical: Project Management (PM), Ontology, Workplace, Data and Hive.

**Deploy i2b2 Databases**

Standard i2b2 instructions are biased towards a Linux/Oracle setup. For i2b2-PMN integration, we are deploying on Windows and will be creating SQL Server databases. For each i2b2 hive:

1. Create 5 databases i2b2data, i2b2hive, i2b2metadata, i2b2pm, i2b2workdata. (For example, for NYC, we have nyc\_i2b2data, nyc\_i2b2hive, nyc\_i2b2metadata, nyc\_i2b2pm, nyc\_i2b2workdata, and then for FDA, we have another set: fda\_i2b2data, fda\_i2b2hive, fda\_i2b2metadata, fda\_i2b2pm, fda\_i2b2workdata).
2. Create a database login user with SQL authentication. (For example, nyc\_i2b2 and fda\_i2b2).
3. Map login user to the 5 databases. Default schema is dbo.
4. Under each database, go to Security > Schemas > dbo. Click "Permissions". Click the link "View database permissions". Grant the login user all permissions explicitly.
5. Read and follow the i2b2 "Data Installation Guide" to setup the databases. Basically the steps are as follows:
   1. Modify the db.properties for each database accordingly.
      1. db.type=sqlserver
      2. db.username=<sql server login name>
      3. db.driver=com.microsoft.sqlserver.jdbc.SQLServerDriver
      4. db.url=jdbc:sqlserver://localhost:1433;databasename=<database name for the cell>
   2. Run ant scripts. Most will simply create tables in the databases and load data. For some databases, scripts will have to be run to also created stored procedures and triggers.

**Deploy SHRINE Ontology**

I2B2 can use other ontologies other than the default. SHRINE is one example.

To install SHRINE ontology (note that SHRINE installer scripts are also biased towards Oracle):

1. From $shrine\_src\code\install\i2b2, run: ontology\_create\_tables.sql
2. From $shrine\_src\code\install\i2b2, run: ontology\_table\_access.sql
3. From $shrine\_src\ontology\core, run: ShrineWithSyns.sql

**Deploy CEDD Ontology**

CEDD is another ontology we support. To deploy:

1. Create a cedd\_ont database.
2. Create the CEDD, SCHEME and TABLE\_ACCESS tables. Create SCHEME and TABLE\_ACCESS manually. Modify the CreateCEDDTableSQLServer.sql script in $i2b2rdp\_src\3rdParty\cedd-mbuck so that it points to the cedd\_ont database instead of the generic i2b2metadata database. Run the script.
3. CEDD is provided to us as a “pipe” delimited CSV file. We have converted it to Excel as it seems to be the more reliable way of importing into SQL Server. These files are compressed and checked into $i2b2rdp\_src\3rdParty\cedd-mbuck. Unzip the Excel version.
4. Using SQL Server Management Studio, right click on “cedd\_ont” > Tasks > Import…
5. Select “Microsoft Excel” as Data source. Choose the Excel file extracted in step 3. Ensure that the checkbox for “First row has column names” is checked. Click Next until you get to the “Select Source Tables and Views” page.
6. For Destination, choose the CEDD table created in step 2. Check the mappings and decide whether to delete or append new data. Preview and visually verify that all are in order. Click Next until you can Run Immediately. There should be no errors.
7. Repeat steps 1-6 on the server hive. And also run ConceptMigration and ModifierMigration scripts (after modification) on the database containing the query data.

NOTE: To import into SQL Server on a Windows PC without Office installed:

1. You may first deploy it by following the steps above on a local PC with Office.
2. Detach the database cedd\_ont (use master; sp\_detach\_db ‘cedd\_ont’;).
3. Copy cedd\_ont.mdf and cedd\_ont\_log.ldf to the target PC. (You may want to compress them first.)
4. Attach the database on the target PC.
5. Check security settings and permissions on the target PC for the newly transferred cedd\_ont database.

**Deploy i2b2 Hive**

In i2b2, a collection of web services called "cells" is known as a "hive".

1. Read the i2b2 installation documentation for the 5 required cells (corresponding to the databases) of an i2b2 hive - CRC, Ontology, Workspace, Hive and Project Management. Edit the property files as specified and deploy each one.
2. When editing the $i2b2\_src\etc\spring\\*.properties file, ensure that the schema names ends with ".dbo" to match SQL Server requirements.
3. The \*-ds.xml are copied to $jboss\_home\server\default\deploy and \*.properties files are copied to $jboss\_home\server\default\conf and JAR-ed and copied into $jboss\_home\server\default\deploy\\*Server.jar.
4. Verification: http://localhost:<port>/i2b2/services/listServices - make sure services from all 5 cells are listed: OntologyService, PMService, WorkplaceService, QueryToolService, FRService

NOTE: JBoss installation and deployment can almost be replicated wholesale and made to work elsewhere with minor changes. Just ensure the following:

* i2b2 is sensitive to the location and credentials of its databases.
  + In $jboss\_home\server\default\deploy, check all the \*-ds.xml and \*-jms.xml and ensure that the database names and credentials are correct.
  + In $jboss\_home\server\default\conf, check all \*.xml and \*.properties and ensure that database names and credentials are correct.
  + In $jboss\_home\server\default\deploy, un-jar all \*Server.jar files elsewhere, make appropriate changes and re-jar them back.
* Check the i2b2hive database's \*\_DB\_LOOKUP tables and ensure the database names are correct.

**Deploy i2b2 Web Client**

I2B2 has a web-based query builder deployable in IIS7 with PHP5.

1. Create [C:\websites\i2b2](file:///C:\i2b2) folder.
2. Copy $i2b2\_src\i2b2webclient-1603\webclient into [C:\websites\i2b2.](file:///C:\i2b2.)
3. Modify [C:\websites\i2b2\webclient\i2b2\_config\_data.js](file:///C:\i2b2\webclient\i2b2_config_data.js) to add a "LincolnPeak" domain with the following properties:
   * domain: "i2b2demo" - can be changed, but has to be changed in the PM database's PM\_HIVE\_DATA table.
   * urlCellPM: http://localhost:<port>/i2b2/rest/PMService
4. In IIS7 Manager, create a new web site and point it to [C:\websites\i2b2\webclient.](file:///C:\i2b2\webclient.)

I2B2 also has a separate administration web client.

1. Copy $i2b2\_src\i2b2core-src-1603\admin into [C:\websites\i2b2.](file:///C:\i2b2.)
   1. Modify [C:\websites\i2b2\admin\i2b2\_config\_data.js](file:///C:\i2b2\webclient\i2b2_config_data.js) to add a "LincolnPeak" domain with the following properties:
      * domain: "i2b2demo" - can be changed, but has to be changed in the PM database's PM\_HIVE\_DATA table.
      * urlCellPM: http://localhost:<port>/i2b2/rest/PMService
2. In IIS7 Manager, create a new web site and point it to [C:\websites\i2b2\admin.](file:///C:\i2b2\admin.)

Verification:

1. Verify that you can login to the admin web site using the URL configured for it. Login as "i2b2" and password "demouser".
2. Verify that you can login to the webclient web site using the URL configured for it. Login as "demo" and password "demouser".

# I2B2-PMN INTEGRATION

**Preparation**

The i2b2-PMN integration code is in the DNS subversion project, under the Plugins\Lpp.Dns.HealthCare.I2B2RDP folder. Each of the subfolders in I2B2RDP are Eclipse projects.

**Configure i2b2 for PMN Integration**

Login to i2b2 admin site as "i2b2". (Database information below for reference.)

1. Modify CRC cell to point to: http://localhost:<port>/i2b2/rest/PMNCRCBridgeService/ (note: URL may change in the future). (PM database's PM\_CELL\_DATA table)
2. Add new user: Investigator. (PM database's PM\_USER\_DATA table)
3. Associate "Investigator" to the "i2b2 Demo" project. (PM database's PM\_PROJECT\_USER\_ROLES)
4. If using a different Ontology, modify the hive database's ONT\_DB\_LOOKUP table's C\_DB\_FULLSCHEMA.

All the above assumes the default i2b2 deployment domain of "i2b2demo" and project id of "Demo" - which may be modified.

Alternatively,

1. Modify domain name if desired. Web client's configuration will have to be modified to conform.
2. Add a new project (e.g., "pmn" with path "/pmn"). (PM database's PM\_PROJECT\_DATA)
3. Add a new user to the project.
4. Add datasource entries into hive database's CRC\_DB\_LOOKUP, ONT\_DB\_LOOKUP, WORK\_DB\_LOOKUP for new domain and project. (Watch out for the slashes - they are not consistent for each database. Mimic them carefully from existing examples.)

**Deploy i2b2 Web Client Modifications**

Replace these JavaScript files from $i2b2rdp\_src\i2b2WebClientMod:

1. Copy i2b2\_cell\_communicator.js into [C:\websites\i2b2\webclient\js-i2b2\hive.](file:///C:\websites\i2b2\webclient\js-i2b2\hive.)
2. Copy CRC\_ctrlr\_QryStatus.js into [C:\websites\i2b2\webclient\js-i2b2\cells\CRC.](file:///C:\websites\i2b2\webclient\js-i2b2\cells\CRC.)
3. Copy PM\_ctrlr.js into [C:\websites\i2b2\webclient\js-i2b2\cells\PM.](file:///C:\websites\i2b2\webclient\js-i2b2\cells\PM.)

**Deploy HQMF Converter**

To support HQMF conversion, deploy hqmf.war from $i2b2rdp\_src\3rdParty\hqmf-jboss-service:

* Read the README file.
* Copy the files in lib\ in $jboss\_home\server\default\lib.
* Copy the folder hqmf.war to $jboss\_home\server\default\deploy.
* Configure hqmf.properties:
  + xslloc - $jboss\_home\server\default\deploy\hqmf.war\xsl
  + ontloc - URL of OntologyService (e.g., http://localhost:<port>/i2b2/rest/OntologyService)
  + baseurl - Base URL of hqmf service (e.g., http://localhost:<port>/jersey
  + fulli2b2 - Must be false. Converts HQMF to full i2b2 or just the query definition.
  + rootkey - Must be set to Ontology database's TABLE\_ACCESS's C\_TABLE\_CD code. (E.g., [\\\\SHRINE](file:///\\SHRINE\) or [\\\\I2B2](file:///\\I2B2)\)) HQMF supports only one root key per i2b2 domain.
* Verification:
  + http://localhost:<port>/jersey/application.wadl
  + http://localhost:<port>/jersey/hqmf/test
  + http://localhost:<port>/jersey/hqmf/getCodeInfo/{domainName}/{projectId}/{username}/password/{password} (e.g., http://localhost:9090/jersey/hqmf/getCodeInfo/i2b2demo/Demo/demo/password/demouser)
  + <http://localhost:9090/jersey/hqmf/getTermInfo/i2b2demo/Demo/demo/password/demouser?key=\\I2B2\i2b2\Demographics\Gender\Female\>

**Deploy PMN Server Adapter**

This adapter is a web service implemented using RESTlet that services DataMartClient's RESTModelProcessor, performs pre-processing (i.e., HQMF-SHRINE conversion, SHRINE-BI ontology translation) and forwards it to the designated hive's CRC cell for query processing.

1. Go to $i2b2rdp\_src\PMNServerAdapter.
2. Run: ant deploy-exploded.
3. Copy $i2b2rdp\_src\PMNServerAdapter\deploy\pmn.war to $jboss\_home\server\default\deploy.
4. Modify $jboss\_home\server\default\deploy\pmn.war\WEB-INF\classes\com\lincolnpeak\i2b2\restlet\PMNAdapter.properties:
   * crcUrl - The URL of the CRC cell of the data source.
   * i2b2pmnProxy\* - Credentials and other information for authenticating to the CRC cell.
5. Modify $jboss\_home\server\default\deploy\pmn.war\WEB-INF\classes\com\lincolnpeak\i2b2\utils\PMNCommons.properties:
   * hqmfUrl - The URL of the HQMF service.
   * convertHQMF - Boolean whether to convert to HQMF or not.
   * translateConcepts - Boolean whether to translate ontology terminologies or not.

Note: The decision to use RESTlet instead of Axis2 like the rest of i2b2 and the PMN Client Adapter is because the RestModelProcessor issues REST requests in the form of a path, which is hard to implement with Axis2 1.1.

**Deploy PMN Client Adapter**

This adapter is the CRC replacement implemented using Axis2 that accepts request from the i2b2 web client and determines to respond to it or submits it as a query request to PMN.

1. Copy $i2b2rdp\3rdParty\httpcomponents-client-4.2\\*.jar to $jboss\_home\server\default\deploy\i2b2.war\WEB-INF\lib.
2. Go to $i2b2rdp\PMNServerAdapter.
3. Run: ant deploy-exploded.
4. Copy $i2b2rdp\PMNServerAdapter\deploy\i2b2.war\WEB-INF\services\PMNCRCBridgeService to $jboss\_home\server\default\deploy\i2b2.war\WEB-INF\services.
5. Modify $jboss\_home\server\default\deploy\i2b2.war\WEB-INF\services\PMNCRCBridgeService\com\lincolnpeak\i2b2\crc\axis2\QueryService.properties:
   * serviceUrl - The URL of PMN remote plugin web service.
   * i2b2HiveUrl - The URL of the local CRC cell.
   * proxyUser - The username to login to PMN.
   * proxyPassword - The password to login to PMN for the proxyUser.
   * pmnRequestType - GUID matching the request type install for the i2b2 model on PMN.
6. Modify $jboss\_home\server\default\deploy\pmn.war\WEB-INF\classes\com\lincolnpeak\i2b2\utils\PMNCommons.properties:
   * hqmfUrl - The URL of the HQMF service.
   * convertHQMF - Boolean whether to convert to HQMF or not.
   * translateConcepts - Unused.

**DataMartClient Setup**

1. Select I2B2 Query Builder as the model processor, which is a Rest Model Processor.
2. Enter http://localhost:<port>/pmn as the URL. This is pointing the the PMN Server Adapter web service.

# VERIFICATIONS

* PMN Remote Plugin Web Service:  <http://fdapilotquerytool.lincolnpeak.com/api/rest/remote/help>
* JBoss, Axis2, i2b2 Hive: http://localhost:<port>/i2b2/services/listServices
* HQMF: http://localhost:<port>/jersey/hqmf/test

# UPGRADE PROCESS

As mentioned in the beginning of this document, there are 9 components that may or may not be upgraded at the same time as long as the interfaces between components are kept intact. We will address each upgrade point in turn here.

**I2b2 Web Client**

1. Follow the steps in “Deploy i2b2 Web Client” above.
2. Follow the steps in “Deploy i2b2 Web Client Modifications” above. You may want to compare the LPP modified JavaScripts against the new i2b2 JavaScripts to ensure there is no conflict between our changes and theirs.

**Ontology Data**

1. Create a new database. Choose a database name that’s different from the old one.
2. Convert data from CSV to Excel if necessary. Import Excel to database using Import Wizard via Excel. Note mapping and preview for errors.
3. Detach database.
4. Copy database mdf and ldf files to target server.
5. Detach old database and attach new one using the old database name.

**I2B2 Hive (JBoss)**

1. If the database is not relocated, it should be possible to transfer the configuration information from the old i2b2 build and re-build.
2. Copy the appropriate war files to the target server.

# DOWNLOADS

* JDK 1.6 (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>)
* JBoss 4.2.2.GA (<http://www.jboss.org/jbossas/downloads>)
* JBoss Native 2.0.10 (<http://www.jboss.org/jbossweb/downloads/jboss-native-2-0-10>)
* Axis2 v1.1 WAR Distribution (<http://axis.apache.org/axis2/java/core/download.cgi>)
* i2b2 v1.6 <https://www.i2b2.org/software/index.html>)
* SHRINE 1.13 (Subversion repository: <https://open.med.harvard.edu/svn/shrine/releases/1.13>)
* Ant 1.8 (<http://ant.apache.org/bindownload.cgi>)