caNanoLab 1.2.1

Installation Guide

October 19, 2007

Table of Contents

INTRODUCTION TO caNANOLAB 1
Targeted Developer 1
Important Background Knowledge 1
General System Requirements 2
Open Source Technologies 2
Grid Service 2
Grid Service Downloads 2
Obtaining the caNanoLab Source 3
Source Code 3

DATABASE TECHNOLOGY 3
Assumptions and Requirements 3
For a Previous Installation 3
For a New Installation 3
Initializing the MySQL Database 4
Follow these steps to initialize your MySQL database system: 4
Verification 5
Data Migration 5

caNANOLAB WEB APPLICATION 12
Assumptions and Requirements 12
Installation and Deployment 12
Common Security Module (CSM) 13
Before You Start caNanoLab 13
Verification 15
User Provisioning Tool (UPT) 15
Creating User Accounts 15
UPT Example 16

caNANOLAB GRID DATA SERVICE 18
Assumptions and Requirements 18
Installation and Deployment 18
Verification 20

CONTACTING APPLICATION SUPPORT 20

Introduction to caNanoLab

Welcome to the cancer Nanotechnology Laboratory Analysis Bench (caNanoLab) 1.2.1 Installation Guide. caNanoLab is a data sharing portal designed for laboratories performing nanoparticle assays. caNanoLab provides support for the annotation of nanoparticles with characterizations resulting from physical and *in vitro* nanoparticle assays and the sharing of these characterizations in a secure fashion.

As of release 1.1, the caNanoLab domain model has been caGrid enabled. In other words, a caNanoLab grid data service can be deployed and registered with caGrid 1.0 index server, allowing sharing of public characterizations across the caGrid. For more information, see https://cabig.nci.nih.gov/workspaces/Architecture/caGrid. The caNanoLab web application now allows remote searches against remote grid services hosting the caNanoLab data model.

Targeted Developer

Although the installation instructions have been simplified as much as possible, this installation guide is best suited for an experienced java developer who has some familiarity with the following J2EE and related technologies:

Important Background Knowledge

- Unix/Linux environment
- (Configuring environment variables; Installing Ant, JDK, Apache Tomcat and JBoss servers)
- Ant build scripts
- J2EE web application development using the Struts framework, Servlet/JSP's, JavaScript
- J2EE middle-ware technologies such as n-tier service-oriented architecture and software design patterns
- caGrid 1.0 infrastructure (for setting a caNanoLab grid data service)

General System Requirements

The following open source technologies power a caNanoLab web application:

Open Source Technologies

- Java Software Development Kit (JDK) version 5.0 http://java.sun.com/javase/downloads/index_jdk5.jsp
- JBoss version 4.0.5 http://labs.jboss.com/jbossas/downloads
- Jakarta Ant version 1.6.5 http://archive.apache.org/dist/ant/binaries/
- MySQL version 5.0.x http://dev.mysql.com/downloads/mysql/5.0.html

---

The caNanoLab web application has been tested within NCICB against JBoss servers hosted on Windows XP and RedHat Linux systems, and against MySQL 5.0.27 databases hosted on RedHat Linux systems and MySQL 5.0.45 on Windows XP systems. Prior to release 1.2.1, the caNanoLab web application had been tested against Oracle 9i databases hosted on Sun Solaris systems, and Oracle 10g XE database hosted on Windows XP.

Download each of the tools listed in the bulleted list and follow the installation instructions provided with each respective product for your environment. Assistance from a MySQL database administrator is expected to properly configure the MySQL database. For MySQL database configuration and maintenance, it is also helpful to download the MySQL 5.0 suite of GUI tools at http://dev.mysql.com/downloads/gui-tools/5.0.html.

Grid Service

Setting up a caNanoLab grid service is optional. It is recommended that you install the grid service only after you successfully install the web application and become familiar with submitting and searching data through the web application.

Grid Service Downloads

Once you have data that you would like share with the caBIG community and you intend to set up a caNanoLab grid service, download the following open source technologies that power a caNanoLab grid data service:

- The Globus Toolkit version 4.0.3 http://www.globus.org/
- Apache Tomcat version 5.0.28 http://tomcat.apache.org/

Obtaining the caNanoLab Source

Source Code

The caNanoLab web application source code is distributed as a zip file named `caNanoLab_1.2.1.zip` from the NCICB download center at http://ncicb.nci.nih.gov/download/.

There are no source updates related to the caNanoLab grid service. The caNanoLab grid service source code is distributed as a zip file named `caNanoLab_grid_1.1.zip` from the NCICB download center at http://ncicb.nci.nih.gov/download/.

Database Technology

Assumptions and Requirements

The caNanoLab source distribution `caNanoLab_1.2.1.zip` has been downloaded from the NCICB download site. A MySQL database has been set up on a system (local or remote) with an administrator (or root) account. You should be familiar with the process of creating InnoDB tablespaces, databases creating users and assigning privileges to users.

For a Previous Installation

If you have installed caNanoLab release 1.2 or release 1.1.1 or release 1.1 against an Oracle database and have associated production data in these schemas and you would like to continue to use the same schemas for caNanoLab release 1.2.1 in MySQL, review the following database initialization steps, then go directly to *Data Migration* on page *5*.

For a New Installation

If you are installing the caNanoLab application for the first time or want to install a new schema for release 1.2.1, follow the steps below to set up the required MySQL schema objects and the seed data for release 1.2.1.

Initializing the MySQL Database

Follow these steps to initialize your MySQL database system:

Step	Action
1	Extract the `caNanoLab_1.2.1.zip` to a location on your local system, for example, `C:\caNanoLab_1.2.1`. This location is referred as `<CANANOLAB_SOURCE>` throughout the document. Verify that the following seven SQL scripts exist in the directory `<CANANOLAB_SOURCE>/db/1.2.1`: • `app_schema.sql` • `app_priming_data.sql` • `csm_schema.sql` • `csm_priming_data.sql` • `app_csm_priming_data.sql` • `app_triggers.sql` • `mysql_seed.sql`

2	Log into the MySQL database as the database administrator (root) and go to the directory `<CANANOLAB_SOURCE>/db/1.2.1` and execute the script ***`mysql_seed.sql`*** to set up a schema named ***cananolab*** and its required seed data.
	
	Example: On a Windows XP system hosting a MySQL 5.0.45 database, you would issue the following commands at the DOS prompt (assuming the ***root*** account has password ***rootpass***):
	
	`C:\>cd c:\caNanoLab_1.2.1\db\1.2.1`
	
	`C:\>mysql –h localhost –u root –prootpass <mysql_seed.sql`

3	Log into the MySQL database as the database administrator (root) and create a user account and grant this user the following privileges to the schema ***cananolab***: `select`, `insert`, `delete`, `update` and allow the user to connect from any hosts.
	
	Example: Create a user account named ***cananolab_app*** with password ***go!234***.
	
	`C:>mysql –h localhost –u root –prootpass`
	
	`mysql>grant select, insert, delete, update on cananolab.* to 'cananolab_app'@'%' identified by 'go!234';`
	
	Note: User administration and schema privileges can also be managed by the MySQL Administrator GUI as a part of the MySQL 5.0 suite of GUI tools.

Verification

Once the MySQL database has been created, either through new setup or through data migration (described in the next section), verify that the following numbers of database objects are created:

- Tables 101
- Views 1
- Triggers 18

Example: Issue the following queries at the MySQL prompt, logging in as ***cananolab_app***:

```
mysql> select count(*) from information_schema.tables
where table_schema='cananolab' and table_type='BASE TABLE'

mysql> select count(*) from information_schema.tables
where table_schema='cananolab' and table_type='VIEW'

mysql> select count(*) from information_schema.triggers
where trigger_schema='cananolab';
```

Data Migration

NOTE:	If you are installing caNanoLab 1.2.1 for the first time or installing a new caNanoLab release 1.2.1 schema, you can skip this section.

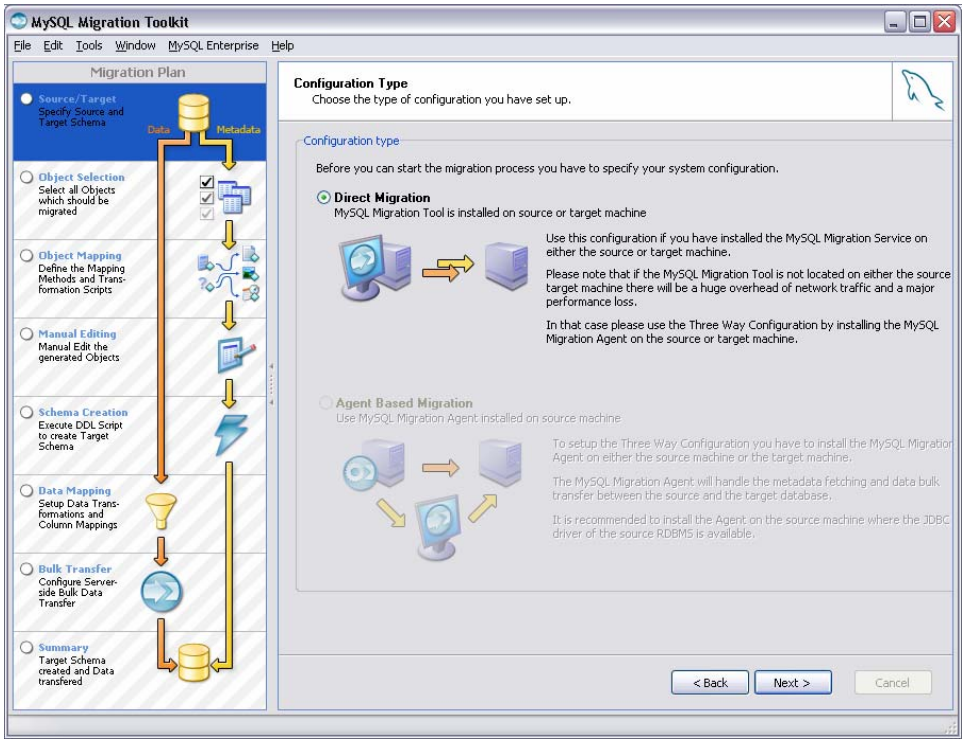
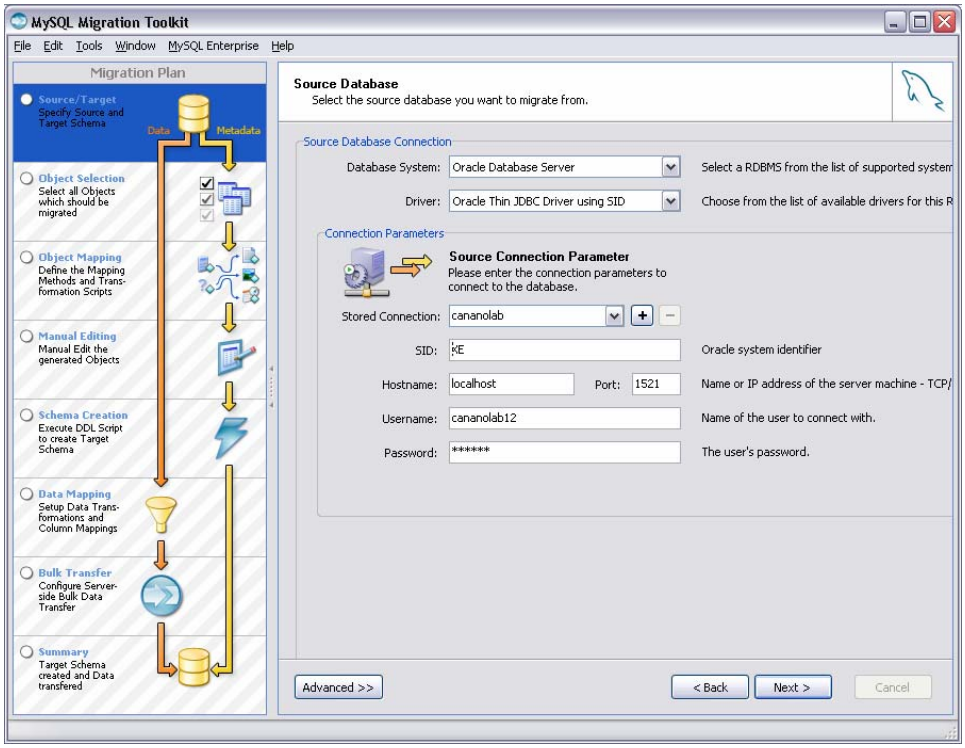
If you have already installed caNanoLab release 1.2 or release 1.1.1, or release 1.1 previously against an Oracle database and have production data associated with these schemas and you wish to keep the same data, you must perform a data migration to be compatible with MySQL. This installation guide only discusses the steps for migrating from release 1.2 in Oracle to release 1.2.1 in MySQL. If you are still on release 1.1 or release 1.1.1 in Oracle, you need to first migrate the database from release 1.1 or release 1.1.1 to release 1.2 in Oracle before migrating to MySQL. Refer to the release 1.2 installation guide for migration details from release 1.1 or release 1.1.1 to release 1.2 in Oracle.

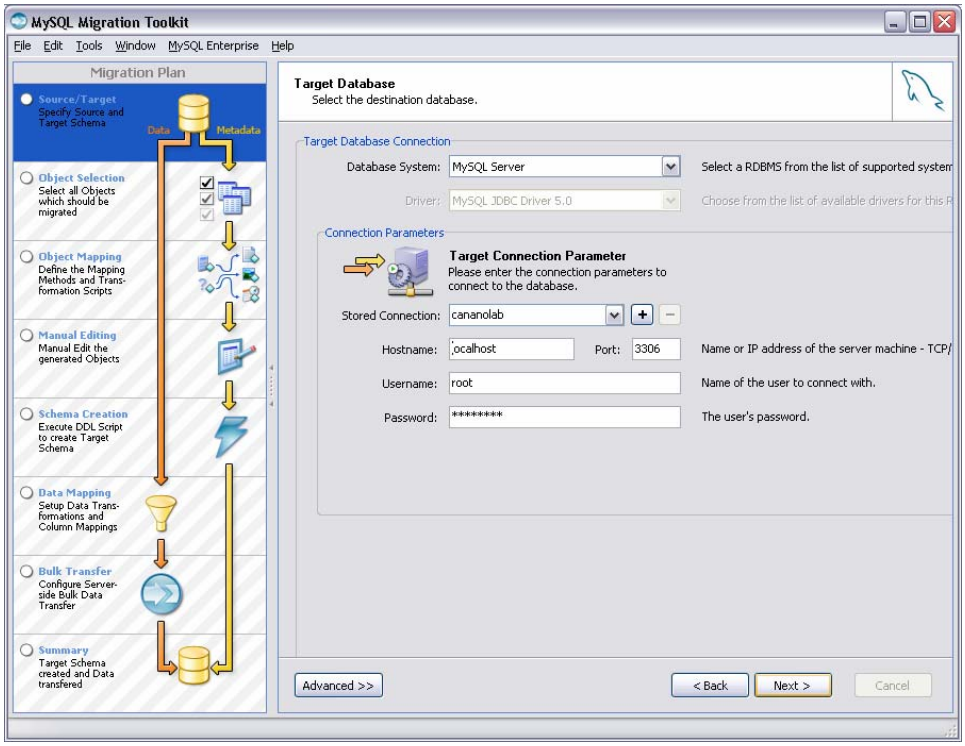
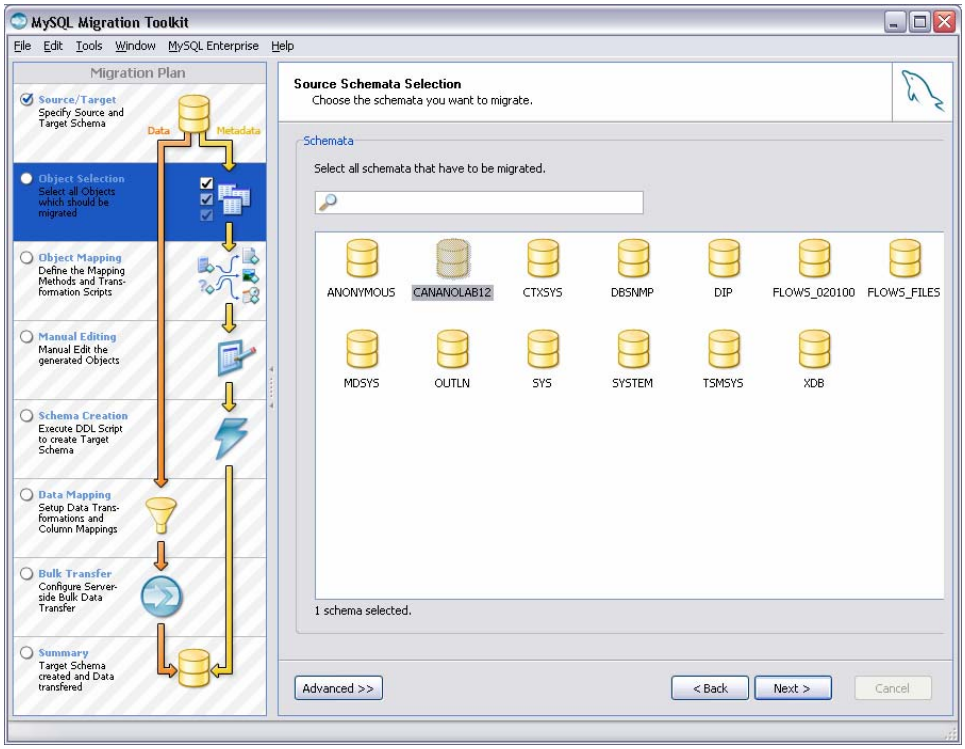
Follow these steps to complete the required data migration from release 1.2 in Oracle to release 1.2.1 in MySQL:

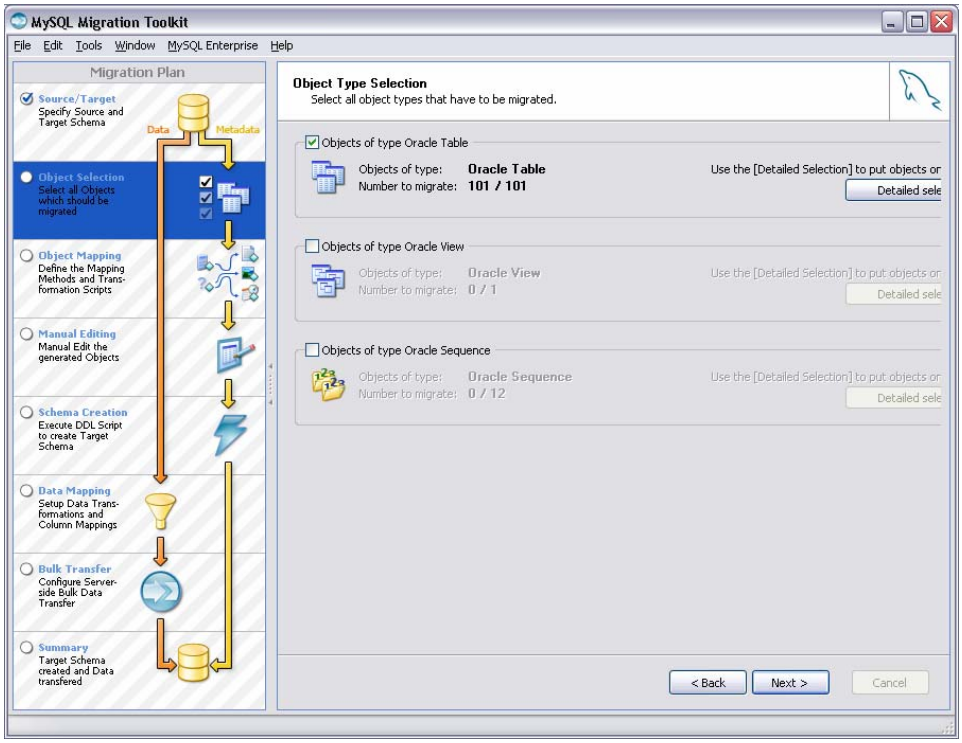
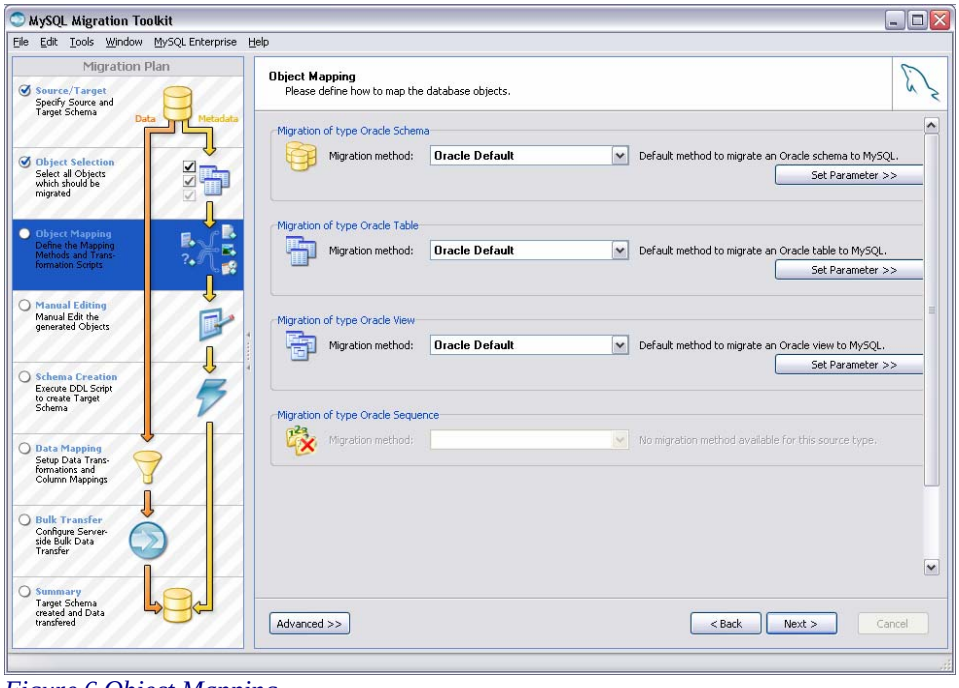
Step	Action
1	Extract the `caNanoLab_1.2.1.zip` to a location on your local system, for example, `C:\caNanoLab_1.2.1`. This location is referred as `<CANANOLAB_SOURCE>` throughout the document. Verify that the following three SQL scripts exist in the directory `<CANANOLAB_SOURCE>/db/1.2.1:` • `app_schema.sql` • `csm_schema.sql` • `app_triggers.sql` **Note:** For data migration, ignore the other SQL scripts in the same directory.

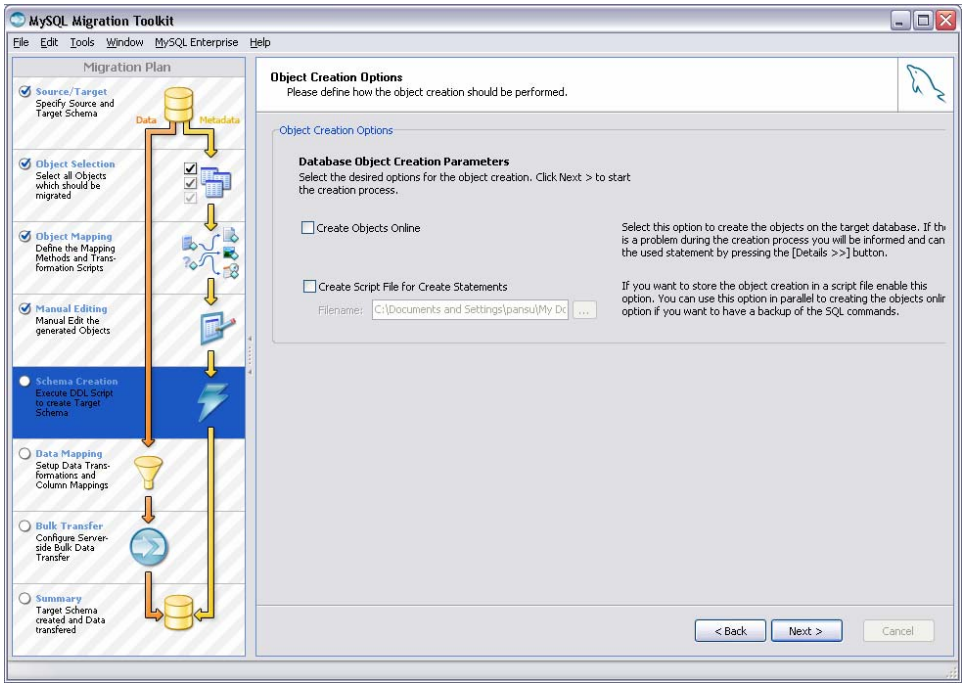
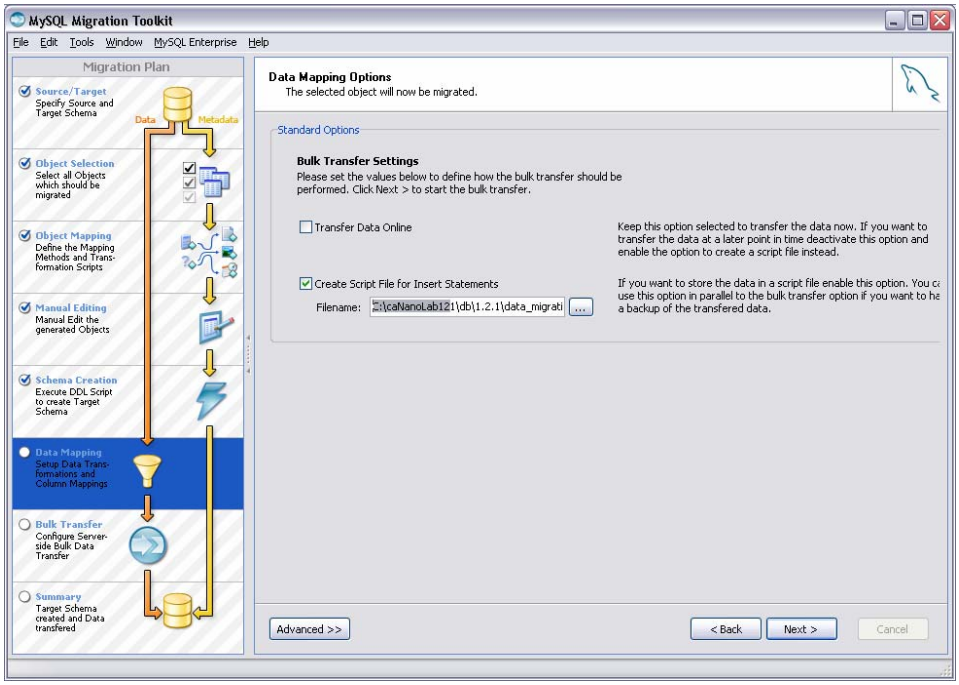
2	Log into the MySQL database as the database administrator (root) and go to the directory `<CANANOLAB_SOURCE>/db/1.2.1 and` execute the above three scripts in the order specified to create a database named ***cananolab***.
	
	Example: On a Windows XP system hosting a MySQL 5.0.45 database, you would issue the following commands at the DOS prompt (assuming the ***root*** account has password ***rootpass***):
	
	`C:\>cd c:\caNanoLab_1.2.1\db\1.2.1`
	`C:\>mysql –h localhost –u root –prootpass <app_schema.sql`
	`C:\>mysql –h localhost –u root –prootpass <csm_schema.sql`
	`C:\>mysql –h localhost –u root –prootpass <app_triggers.sql`

3	Download the MySQL 5.0 GUI Tools from http://dev.mysql.com/downloads/gui-tools/5.0.html and install them.
4	Start the MySQL 5.0 Migration Toolkit GUI and follow the instructions to generate the required insert statements for the MySQL database from the Oracle database. Click **Next** in the bottom right-hand corner of each page to proceed through the steps. *Example*: The following screenshots from the MySQL Migration Toolkit GUI illustrate the required steps for generating the insert statements from an Oracle 10g XE database to a MySQL 5.0.x database on a Windows XP system.

Step	Action
4a	 *Figure 1 Configuration Type*
4b	 *Figure 2 Source Database*

Step	Action
4c	 *Figure 3 Target Database*
4d	 *Figure 4 Source Schemata Selection*

Step	Action
4e	 *Figure 5 Object Type Selection*
4f	 *Figure 6 Object Mapping*

Step	Action
4g	 *Figure 7 Object Creation Options (Make sure to uncheck the checkboxes.)*
4h	 *Figure 8 Data Mapping Options (Make sure to uncheck **Transfer Data Online** and check **Create Script File** for Insert Statements.)* . **Note:** The Migration Toolkit GUI is used for generating insert statements only; there is no need to generate schema creation scripts.

Step	Action
5	Edit the file generated in step 4 as follows: • Replace all occurrences of word ***cananolab12*** with word ***cananolab*** (assuming the Oracle schema that the MySQL database migrated from was ***cananolab12***).

6	Log into the MySQL database as the database administrator (root) and execute the edited insert statements resulted from step 5.
	
	Example: On a Windows XP system, assuming the SQL file resulted in step 5 is located at `c:\caNanoLab1.2.1\db\1.2.1\insert_data_from_oracle.sql`.
	
	`C:\>cd c:\caNanoLab_1.2.1\db\1.2.1`
	`C:\>mysql –h localhost –u root –prootpass <insert_data_from_oracle.sql`

7	Log into the MySQL database as the database administrator (root) and create a user account and grant this user the following privileges to the schema ***cananolab***: **select**, **insert**, **delete**, **update** and allow the user to connect from any hosts.
	
	Example: Create a user account named ***cananolab_app*** with password ***go!234*** .
	
	`C:>mysql –h localhost –u root –prootpass`
	
	`mysql>grant select, insert, delete, update on cananolab.* to 'cananolab_app'@'%' identified by 'go!234';`
	
	Note: User administration and schema privileges can also be managed by the MySQL Administrator GUI as a part of the MySQL 5.0 suite of GUI tools.

After data migration, refer to the Verification section on page 5 to verify that the migration has been successful. The same number of schema objects should be generated as described in that section.

---

caNanoLab Web Application

Assumptions and Requirements

The database has been installed and verified, as described on pages 3 and 5. Ant has been installed. The JBoss application server has been installed on a system (local or remote) and can be started at a designated port. JBoss install directory is referred as `<JBOSS_HOME>` in the document. The JBoss application server host URL is referred as `<SERVER_URL>`. It is assumed that the default configuration is used for deploying caNanoLab related web archive files; it is recommended that you change the designated port from the default port 8080 to another unused port.

For example, in the case of JBoss 4.0.5, the default configuration is located at the directory `<JBOSS_HOME>/server/default`. The caNanoLab web archive file shall be deployed at the directory `<JBOSS_HOME>server/default/deploy`. For changing the default port 8080 and associated ports, locate the file `server.xml` at the directory `<JBOSS_HOME>/server/default/deploy/jbossweb-tomcat55.sar`. Replace all occurrences of 8080, 8009 and 8443 with 18080, 18009 and 18443, respectively. For details, refer to the appropriate JBoss documentation.

Installation and Deployment

Follow these steps to install and deploy caNanoLab.

Step	Action
1	Edit the Ant build properties file `build.properties` at `<CANANOLAB_SOURCE>` by specifying values for the following five properties: a. `file.repository.dir`: A directory on the system that hosts the JBoss application server for storing uploaded files, for example, `C:/temp/caNanoLab/fileupload` **Note:** This directory should be writable by the user that starts the JBoss server, and this directory should be created prior to starting the application. Either double back slashes \\ or a single forward slash / should be used as the file separator if working on Windows platform. b. `sample.prefix`: The prefix used for auto-accessioning samples, for example, `NCL` is used for sample accessions `NCL-1, NCL-2`, etc at the Nanoparticle Characterization Laboratory. c. `application.owner`: the owner of the caNanoLab release 1.2 installation instance, for example, `NCL`. d. `grid.indexserver`: the grid index server hosting the caNanoLab grid data services, for example, `cagrid-index-qa.nci.nih.gov:8080`, the NCICB's QA caGrid index server. e. `database`: This property specifies whether the application works with an Oracle database or a MySQL database. For release 1.2.1, use `mysql`.

Step	Action

2	Execute the Ant build script `build.xml` located at `<CANANOLAB_SOURCE>` with the default target `build-application` and the default `build.properties` file configured in step 1.
	
	Example: Issue the following commands to execute the Ant script:
	
	`C:\>cd c:\caNanoLab_1.2`
	`C:\>ant`
	
	Successful execution of the Ant script creates three deployable web archive war files in the directory `<CANANOLAB_SOURCE>/output/war`: `caNanoLab.war`, `upt.war`, and `caNanoLabSDK.war`. Place these war files into the default JBoss application server deploy directory, `<JBOSS_HOME>/server/default/deploy`.

3	Copy the file `mysql-connector-java-5.0.7-bin.jar` from `<CANANOLAB_SOURCE>/lib` folder and place it in the directory `<JBOSS_HOME>/server/default/lib/.`

Common Security Module (CSM)

 Before You Start caNanoLab	Before starting the JBoss application server and running the caNanoLab application, you must configure the server with required security configurations for user authentication and authorization. The caNanoLab web application relies on a modified version of NCICB CSM (Common Security Module) version 3.1 for user authentication and role-based authorization. For more information, see the CSM documentation at http://gforge.nci.nih.gov/frs/?group_id=12.

Follow these steps to complete the security configurations:

Step	Action
1	Copy the file `ApplicationSecurityConfig.xml` from the directory `<CANANOLAB_SOURCE>/conf/csm`. Place it in the directory `<JBOSS_HOME>/server/default/conf`:
2	Copy the file `caNanoLab.csm.hibernate.cfg.xml` from the directory `<CANANOLAB_SOURCE>/conf/csm/mysql`. Place it in the directory `<JBOSS_HOME>/server/default/conf`:

Step	Action
3	Edit the file `ApplicationSecurityConfig.xml` at `<JBOSS_HOME>/server/default/conf` as follows: • Replace the token `@ABSOLUTE_PATH@` with the value `<JBOSS_HOME>/server/default/conf`, where `<JBOSS_HOME>` should be replaced by the actual full path JBoss install directory name, `C:/jboss-4.0.5.GA`. **Note:** Either double back slashes \\ or a single forward slash / should be used as the file separator if working on Windows platform.
4	Copy the file `mysql-ds.xml` from the directory `<CANANOLAB_SOURCE>/conf` and place it in the directory `<JBOSS_HOME>/server/default/deploy`, if such a file does not already exists. If a file named `oracle-ds.xml` exists in the directory `<JBOSS_HOME>/server/default/deploy`, remove it.
5	Edit the copied file `mysql-ds.xml` at `<JBOSS_HOME>/server/default/deploy` as follows: a. Replace the token `@DBSERVER@` with the URL or the IP address of the system that hosts the MySQL database, for example, ***127.0.0.1***. b. Replace the token `@PORT@` with the port number of the database server, for example, ***3306***. c. Replace the token `@DBNAME@` with the name of the database service to which to connect, for example, the value is ***cananolab*** for the MySQL Database created in the Database Technology Section Check with your Oracle DBA for your database service name configurations. d. Replace the token `@DBUSER@` with the user name configured in step 1 of the Database Technology instructions on page 3, for example, ***cananolab_app***. e. Replace the token `@DBPASSWORD@` with the password associated with the value for the token `@DBUSER@`, for example, ***go!234***.
6	Locate the last `</application-policy>` tag in the file `login-config.xml` at `<JBOSS_HOME>/server/default/conf`, and insert the content of the file `login-config-segment.xml` at `<CANANOLAB_SOURCE>/conf/csm/mysql` into the file `<JBOSS_HOME>/server/default/conf/login-config.xml` just below the last `</application-policy>` tag.
7	Edit the inserted file `login-config.xml` at `<JBOSS_HOME>/server/default/conf` by doing similar replacements as in step 4: a. Replace the token `@DBSERVER@` with the URL or the IP address of the system that hosts the MySQL database, for example, ***127.0.0.1***. b. Replace the token `@PORT@` with the port number of the database server, for example, ***3306***. c. Replace the token `@DBNAME@` with the name of the database service to which to connect, for example, the value is ***cananolab*** for the MySQL Database created in the Database Technology Section Check with your Oracle DBA for your database service name configurations. d. Replace the token `@DBUSER@` with the user name configured in step1 of the Database Technology instructions on page 3, for example, ***cananolab_app***. e. Replace the token `@DBPASSWORD@` with the password associated with the value for the token `@DBUSER@`, for example, ***go!234***.

Step	Action
8	Locate the last `</mbean>` tag in the file `properties-service.xml` at `<JBOSS_HOME>/server/default/deploy`, and insert the content of the file `properties-service-segment.xml` at `<CANANOLAB_SOURCE>/conf/csm` into the file `<JBOSS_HOME>/server/default/deploy/properties-service.xml` just above the last `</mbean>` tag.
9	Edit the inserted file `properties-service.xml` at `<JBOSS_HOME>/server/default/deploy` as follows: • Replace the token `@ABSOLUTE_PATH@` with the value `<JBOSS_HOME>/server/default/conf`, where `<JBOSS_HOME>` should be replaced by the actual full path JBoss install directory name, `C:/jboss-4.0.5.GA`. **Note:** Either double back slashes \\ or a single forward slash / should be used as the file separator if working on Windows platform.

Verification

Once the JBoss application server is correctly configured with the CSM settings, you can now start the JBoss application server and start the caNanoLab application. Open the URL http://<SERVER_URL>/caNanoLab/ (for example, http://localhost:18080/caNanoLab). You should see a disclaimer page. Click on **CLICKING HERE** to go to the Welcome/Login page.

User Provisioning Tool (UPT)

Creating User Accounts

Before users can log in to the caNanoLab application to submit and search data, you must first create their user accounts and assign them to the pre-defined user groups with the pre-defined roles to access the pre-defined protection groups. The caNanoLab application makes use of the NCICB's User Provisioning Tool (UPT), a separate web application, for user account management. The concepts of users, groups, roles, protection groups are defined according the CSM/UPT principles. See the CSM documentation at http://gforge.nci.nih.gov/frs/?group_id=12 for details on these concepts and the use of the UPT tool.

NOTE:	As a part of the database seed data, a default user group Public has been created and has been assigned access to the SEARCH and the REMOTE SEARCH functions of the caNanoLab application. During the caNanoLab application start up, three more default users groups are auto-generated:
	
	`<application.owner>_Administrator`
	`<application.owner>_Researcher`
	`<application.owner>_PI`
	
	Where the value for `<application.owner>` is specified in the file `build.properties` prior to building the application as described on page 6.

In release 1.2, during the caNanoLab application start up, automatic roles are assigned for these three default user groups as follows:

1. `<application.owner>_Administrator` is assigned access to the ADMINISTRATION function,

2. `<application.owner>_Researcher` is assigned access to the WORKFLOW and the INVENTORY functions, and

3. `<application.owner>_PI` is assigned access to the WORKFLOW, the INVENTORY and the SUBMIT functions.

UPT Example The following steps illustrate an example use of the UPT tool to create a new user, to assign the user to be a caNanoLab administrator, and to assign the user to two pre-defined user groups.

Step	Action
1	Launch the UPT tool at http://<SERVER_URL>/upt and login as ***superadmin/password***. Use ***caNanoLab-upt*** as the application name when prompted at the UPT log in. **Note:** The user **superadmin** was created as a part of the database setup.
2	Once logged into the UPT tool, follow these steps: a. Select **User > Create a New User**. b. Create a new user account named **admin**. c. Select **Application > Select an Existing Application**; click **Search**. d. Select **caNanoLab** from the application list. e. Click **View Details**, then **Associated Admins**. f. Assign this user to be an administrator for the caNanoLab application. g. Click **Update Association** to commit the change.

Step	Action
3	Log out of the UPT tool and log back in as **admin/admin**. Use **caNanoLab** as the application name when prompted at the UPT log in. **Note:** When a new user account is created, its initial default password is the same as its login name. Given that the UPT tool is lacking a function for users to manage their own passwords, users can only update their passwords within the caNanoLab application. See the Notes below on how to update user passwords.
4	Select **User > Select an Existing User**, and click **Search**. Select **admin** from the User list. a. Click **View Details**, then **Associated Groups**. b. Select **<application.owner>_Adminstrator** and **Public** groups from the pre-defined group list and assign them to the user. c. Click **Update Associations** to commit the change.

Follow similar steps to create other application user accounts and to assign them to different users groups, as appropriate.

NOTES:	
	• All users should be assigned to the Public group so everyone has access to the SEARCH and the REMOTE SEARCH functions. • A user can be assigned to multiple user groups, for example, **admin** is assigned to both `<application.owner>_Administrator` and `Public` groups. • The initial passwords for user accounts are set to be the same as their user accounts. When a user first logs into the caNanoLab application at http://<SERVER_URL>/caNanoLab, he/she would be prompted to change the initial password. Users can update their passwords at any time within the caNanoLab application.

For more information about how to use the UPT tool for managing user accounts, contact NCICB Application Support at ncicb@pop.nci.nih.gov and request that the caNanoLab technical team give you a demonstration of the UPT tool in the context of the caNanoLab application.

caNanoLab Grid Data Service

Assumptions and Requirements

Apache Tomcat has been downloaded and installed with the `CATALINA_HOME` environment variable set pointing to the Tomcat install directory. The Tomcat server URL is referred to as `<TOMCAT_SERVER_URL>` throughout the document.

Globus Toolkit has been downloaded and installed with the `GLOBUS_LOCATION` environment variable pointing to the Globus install directory. Apache Tomcat is assumed to be the grid service container.

The caNanoLab grid data service requires the caNanoLab web application war file `caNanoLab.war` be deployed, as well as caNanoLab SDK application service war file `caNanoLabSDK.war` be deployed to the same application server. Refer back to the caNanoLab web application installation and deployment section on page 12 for details on how to deploy these war files.

Installation and Deployment

Before deploying grid services, Tomcat must be pre-configured with a Globus web application that can host a grid service. Issue the following commands to complete the pre-configuration required for Tomcat to host a caNanoLab grid service:

```
C:>cd "%GLOBUS_LOCATION%"
C:>ant -f share\globus_wsrf_common\tomcat\tomcat.xml
deployTomcat -Dtomcat.dir="%CATALINA_HOME%" –
Dwebapp.name=wsrf-canano
```

This generates a directory `<CATALINA_HOME>/webapps/wsrf-canano` with the required Globus webapp information. Note that by specifying a `webapp.name` property (in the above command) to have a different value than the default value `wsrf`, multiple caCORE SDK backed grid services can be deployed to the same Tomcat server and function properly and independently without interference. This circumvents the current known issue with multiple caCore SDK-backed grid services interfering with each other if deployed to the same Globus webapp `wsrf` under the same Tomcat server.

Once the Tomcat server is pre-configured, complete the following steps to create and deploy a caNanoLab grid data service:

Step	Action
1	Extract `caNanoLab_grid_1.1.zip` to a location on your local machine such as the C drive. This location is referred as `<CANANOLAB_GRID_SOURCE>` throughout the document.

Step	Action
2	By default, the caNanoLab grid data service is registered to the NCICB caGrid 1.0 QA index service. To change the index service URL, open the `file deploy.properties` at `<CANANOLAB_GRID_SOURCE>` and edit the property `index.service.url` to be your choice of caGrid index server.
3	Edit the file `serviceMetadata.xml` at `<CANANOLAB_GRID_SOURCE>/etc/` as follows: • Replace the contents in the tag `<ns1:hostingResearchCenter>` with your hosting center information. **Note**: The value for the attribute `displayName` is used in the caNanoLab web application as the grid node host name to distinguish different remote caNanoLab grid services.

4	Execute the Ant build file (`build.xml`) located at `<CANANOLAB_GRID_SOURCE>` with the following two build properties and the task `createTomcatDeploymentTar`:
	
	`cananolab.sdk.host =<SERVER_URL> (e.g. localhost:8080)`
	
	`webapp.name = wsrf-canano`
	
	Execution of the Ant build script generates a file `CaNanoLabSvc.tar` at the `<CANANOLAB_GRID_SOURCE>/output` directory.

5	Place the file `CaNanoLabSvc.tar` to the directory `<CATALINA_HOME>` and extract the tar file within the same directory.

6	For the caNanoLab grid service to successfully register with the index server, edit the file `server-config.wsdd` at `<CATALINA_HOME>/webapps/wsrf-canano/WEB-INF/etc/globus_wsrf_core as follows`:
	
	• Add a `logicalHost` parameter, with the value being the public IP address or the public hostname for the Tomcat server. For example:
	
	`<globalConfiguration>`
	`...`
	`<parameter name="sendXsiTypes" value="true"/>`
	
	`<parameter name="logicalHost"`
	`value="<TOMCAT_SERVER_URL>"/>`
	
	`<parameter name="publishHostName" value="true"/>`

7	Start the tomcat server to deploy the grid service.

Verification

To verify the creation of the caNanoLab grid data service, open the URL http://<TOMCAT_SERVER_URL>/wsrf-canano/services/cagrid/CaNanoLabSvc. You should see a page that says:

```
cagrid/CaNanoLabSvc

Hi there, this is an AXIS service!
```

Log into the caNanoLab application, and select **REMOTE SEARCH > Nanoparticles**. You should see the display name for the research center you specified earlier listed as one of the remote grid node hosts.

For more information and help on caGrid service configuration and deployment, refer to the caGrid wiki page: http://www.cagrid.org.

If you need further assistance on setting a caNanoLab grid service, contact NCICB Application Support at ncicb@pop.nci.nih.gov.

Contacting Application Support

NCICB Application Support

http://ncicb.nci.nih.gov/NCICB/support
Telephone: 301-451-4384
Toll free: 888-478-4423