

caArray 093 - How To Build caArray From Source Code - An Illustrated Tutorial

Unable to render {include} The included page could not be found.

Question: How can I build caArray from the source code in the NCI repository?

Topic: caArray Installation and Upgrade

Release: caArray v2.4.1.1 and higher

Date entered: 01/17/2012

Answer

Below is a step-by-step illustrated tutorial on how to build the application from source. It covers the following:

- How to check out the source code from the repository using a Subversion (SVN) client
- How to manually create the requisite database schema
- How to configure the 'properties' files from the source with local installation settings
- How to invoke the build process from the command line
- How to launch the caArray server once the build has completed successfully
- How to access the application's login page via your Web browser

The tutorial is designed for use in a Windows environment, but can easily be adapted to work in Linux as well.



Note

This tutorial is based on the **readme.txt** file in the NCI caArray source code repository at <https://ncisvn.nci.nih.gov/svn/caarray2/tags/2.4.1.1/>.

Prerequisites

1) Before proceeding to check out the code, ensure that the following are already installed on your machine:

- The Java 2 SE Development Kit (JDK) v1.5, available for download at http://download.oracle.com/otn/java/jdk/1.5.0_22/jdk-1_5_0_22-windows-i586-p.exe



Warning

The build process described in this tutorial only works with version 1.5 of the JDK; it will not work with any other versions, earlier or later.

- Apache Ant 1.7.0 or later, available for download at <http://mirrors.ibiblio.org/apache//ant/binaries/apache-ant-1.8.3-bin.zip>
- MySQL v5.0 or later, available for download at <http://filehippo.com/download/file/3518c972e6317bff1a5caacae83b83cf1057fceca08180ebd44ad9a0ba01444b/>
- A command-line Subversion (SVN) client of your choice. The one used in this tutorial is the Collabnet Subversion Command-Line Client v1.74, available for download at <http://www.open.collab.net/servlets/OCNDirector?id=CSVN1.7.4WINC32> (registration required)

2) Once all the above are installed, ensure they are configured as follows:

- The PATH and JAVA_HOME Windows environment variables must be set to the the installation path of the JDK binaries. The default path in Windows is:

C:\Program Files\Java\jdk1.5.0_22\bin

- The PATH and ANT_HOME Windows environment variables must be set to the installation path of the Ant binaries. The default path is:

C:\ant

- The PATH Windows environment variable must be set to the installation path of the SVN client binaries. The default path in Windows is:

C:\Program Files\CollabNet Subversion Client

- The ANT_OPTS environment variable must be set to the value "-Xmx256m"
- The MySQL server must be configured with a secure password for the root user, and this password must be recorded for future reference

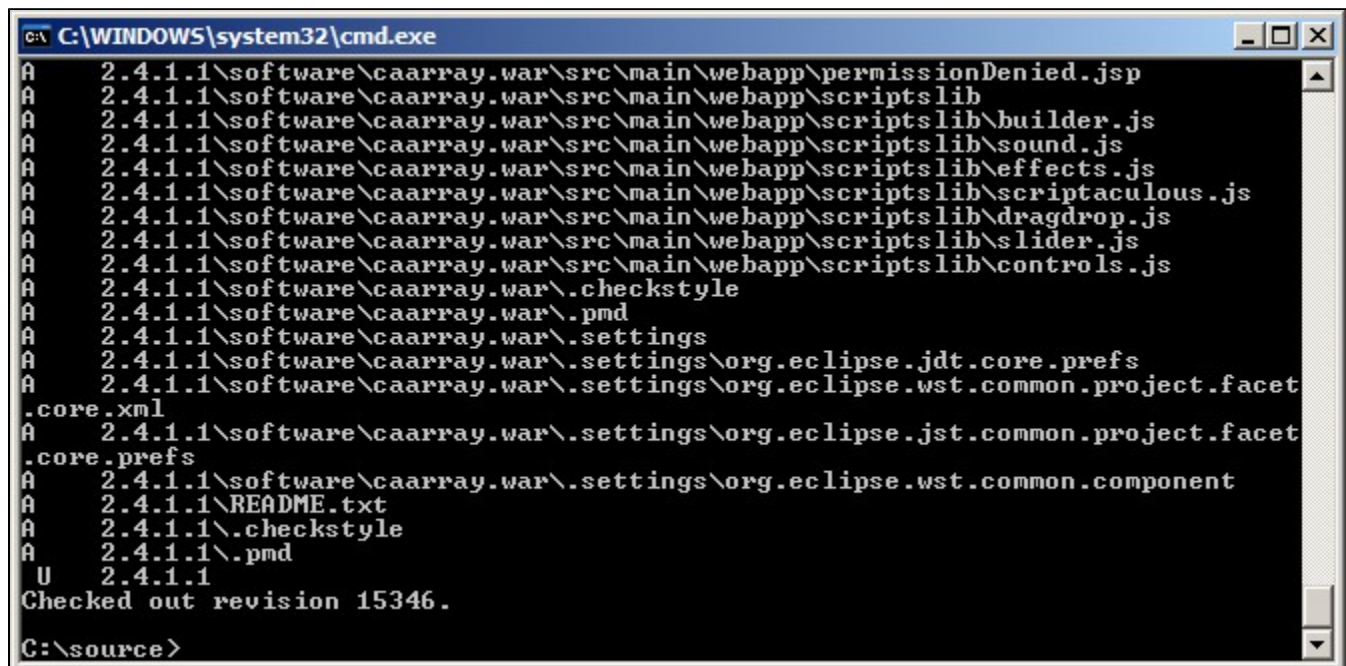
Checking Out The Source Code From The NCI Repository

To begin checking out the caArray source code from the NCI repository, first create a new folder (i.e., **C:\source**), then open up a command-line window, navigate to that folder, and enter the following:

```
svn checkout https://ncisvn.nci.nih.gov/svn/caarray2/tags/2.4.1.1/
```

The source files will then begin downloading from the repository server into the folder you created. Depending on the speed of your Internet connection, it may take over half an hour for the checkout to complete, as the source contains thousands of files distributed over hundreds of folders.

You'll know when the checkout is complete when the command-line window shows a message stating, 'Checked out revision x', where x is the revision number, as shown in the screenshot below:



```
C:\WINDOWS\system32\cmd.exe
A      2.4.1.1\software\caarray.war\src\main\webapp\permissionDenied.jsp
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\builder.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\sound.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\effects.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\scriptaculous.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\dragdrop.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\slider.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\controls.js
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\checkstyle
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\pmd
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\settings
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\org.eclipse.jdt.core.prefs
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\org.eclipse.wst.common.project.facet
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\org.eclipse.jst.common.project.facet
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\org.eclipse.wst.common.component
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\README.txt
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\checkstyle
A      2.4.1.1\software\caarray.war\src\main\webapp\scriptslib\pmd
U      2.4.1.1
Checked out revision 15346.
C:\source>
```

Manually Creating The Requisite Database Schema

It is possible to have Ant automatically generate the empty database schema required for the caArray installation. However, it is preferred to create these schema manually via the MySQL command line client.

Before doing so, you will need to derive a name for the database caArray will use to store genomic data, as well as a username and password for the user who is granted access to that database. In this example, the database name is **db1**, the username is **user1**, and the password name is **password1**.

Now, log into the client using the root password you set when installing the database server, then enter the following SQL commands line-by-line, substituting your own database name, username, and password as needed:

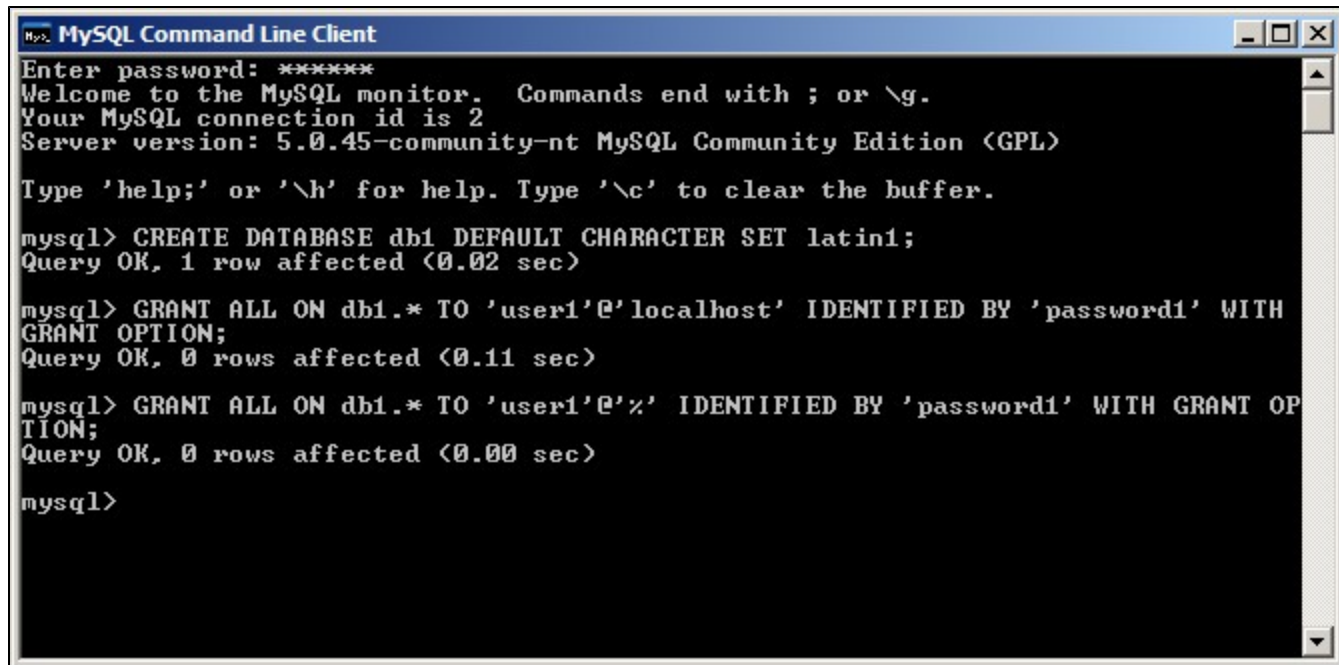
```
CREATE DATABASE db1 DEFAULT CHARACTER SET latin1;
GRANT ALL ON db1.* TO 'user1'@'localhost' IDENTIFIED BY 'password1' WITH GRANT OPTION;
GRANT ALL ON db1.* TO 'user1'@'%' IDENTIFIED BY 'password1' WITH GRANT OPTION;
```



Warning

The single quotes around the username and password in the code snippet above are not optional and can not be omitted -- they are part of the MySQL syntax.

The client will respond with a confirmation that the issued queries were successful, as shown in the screenshot below:



```
MySQL Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.0.45-community-nt MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> CREATE DATABASE db1 DEFAULT CHARACTER SET latin1;
Query OK, 1 row affected (0.02 sec)

mysql> GRANT ALL ON db1.* TO 'user1'@'localhost' IDENTIFIED BY 'password1' WITH
GRANT OPTION;
Query OK, 0 rows affected (0.11 sec)

mysql> GRANT ALL ON db1.* TO 'user1'@'%' IDENTIFIED BY 'password1' WITH GRANT OP
TION;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

Configuring The 'Properties' Settings Files

The caArray installation settings are mainly specified by a source file, **install.properties**, whose path is:

```
$CAARRAY_HOME/software/master_build/install.properties
```

where **\$CAARRAY_HOME** represents the root folder into which the caArray source was checked out (**C:\source** in this example).

A second file, **local.properties**, specifies additional settings. This file is not included in the repository -- you must manually create it yourself by opening a plain text editor and saving a blank file with the filename **local.properties** to the following path:

```
$CAARRAY_HOME/software/build/local.properties
```

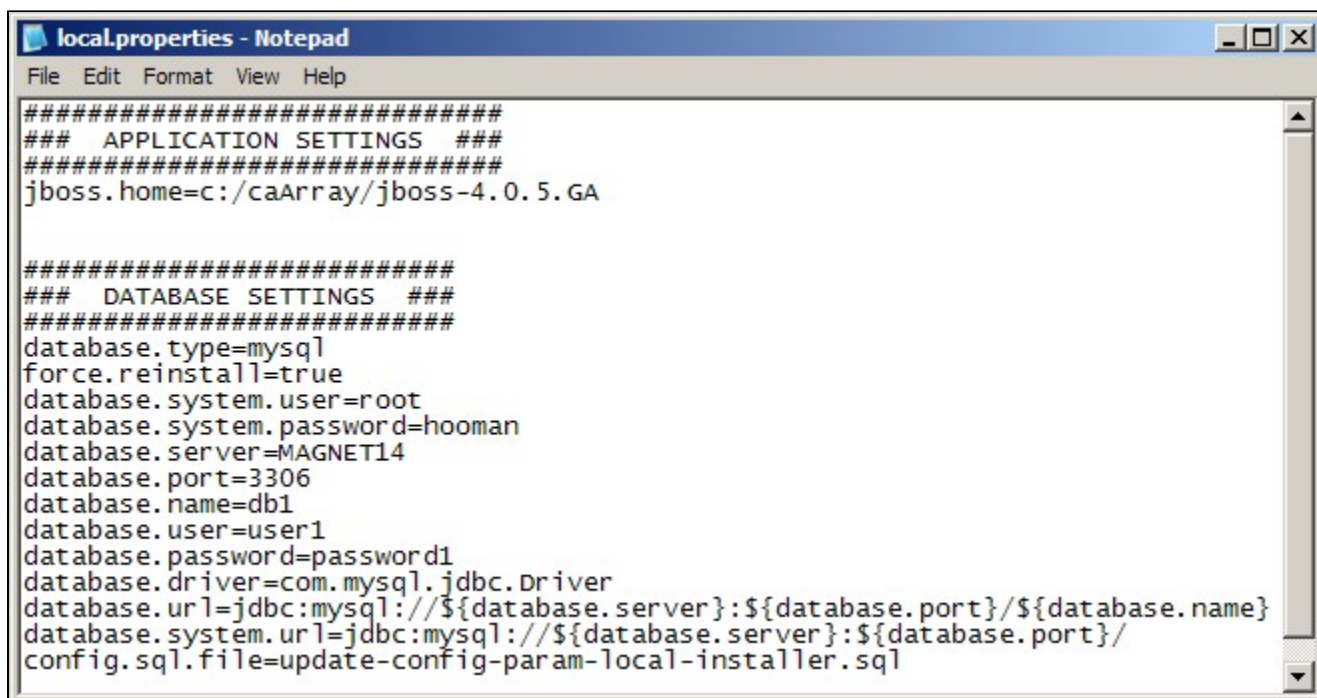
These files must be customized with the settings specific to your local environment before starting the build process. These settings include:

- The username of the root account you established when installing MySQL ('root' by default), as specified by the **database.system.user** property
- The password of the root account you established when installing MySQL, as specified by the **database.system.password** property
- The name of your database server (usually the same as the Windows name of your workstation or server), as specified by the **database.server** property
- The port your database server is installed on (3306 by default), as specified by the **database.port** property
- The database name you established when manually creating the database schema, as specified by the **database.name** property
- The username of the user to whom you granted access to this database, as specified by the **database.user** property
- The password of the user to whom you granted access to this database, as specified by **database.password** property
- The path to which caArray should be installed, as specified by the **application.base.path** property

The **install.properties** file contains several other properties aside from the ones listed above, but these others do not need to be customized and can be left at their default values.

The blank **local.properties** file you created previously must be populated with all the properties listed above and their respective values, plus an additional field, **jboss.home**, which specifies the name of the subfolder within the root installation folder to which the JBoss server will be installed. The value of the **jboss.home** property is derived from the value of the **application.base.path** property in **install.properties**, with the text 'jboss-4.0.5.GA' appended. For example, if the value of the **application.base.path** property is **C:\lcaArray**, then the corresponding value of the **jboss.home** property would be **C:\lcaArray\jboss-4.0.5.GA**.

Refer to the screenshot below for an example of how the **local.properties** file should be configured:



```
#####
### APPLICATION SETTINGS ###
#####
jboss.home=c:/caArray/jboss-4.0.5.GA

#####
### DATABASE SETTINGS ###
#####
database.type=mysql
force.reinstall=true
database.system.user=root
database.system.password=hooman
database.server=MAGNET14
database.port=3306
database.name=db1
database.user=user1
database.password=password1
database.driver=com.mysql.jdbc.Driver
database.url=jdbc:mysql://${database.server}:${database.port}/${database.name}
database.system.url=jdbc:mysql://${database.server}:${database.port}/
config.sql.file=update-config-param-local-installer.sql
```



Note

The **install.properties** file contains additional properties which pertain to grid services. These are not covered here, as configuring caArray to utilize a grid service is beyond the scope of this tutorial. For more information, please refer to the [caArray FAQ](#) and [in-depth articles](#).

Invoking The Build Process From The Command Line

Now that you've checked out the source code, created the database schema, and configured the properties files, you're now ready to start the actual build process. First, open a command-line window and navigate to the following path:

```
$CAARRAY_HOME/software/master_build
```

Then, enter the following command:

```
ant -Dproperties.file=<absolute path to install.properties file> deploy:local:install
```

For example, if the path to the **install.properties** file is:

```
C:\source\2.4.1.1\software\master_build\install.properties
```

Then the command is:

```
ant -Dproperties.file=C:\source\2.4.1.1\software\master_build\install.properties deploy:local:install
```

The amount of time needed for the build process to complete can vary anywhere from under 30 minutes to several hours depending on your hardware configuration. The screenshot below shows the command line window after the build process has completed successfully, with the **'BUILD SUCCESSFUL'** message at the bottom.

```
C:\WINDOWS\system32\cmd.exe

[exec] [echoproperties] web-console.role.matched=admin\=JBossAdmin\n
[exec] [echoproperties] web-console.user.file=c\:/Temp/jboss-4.0.4.GA/serve
r/default/conf/props/web-console-users.properties
[exec] [echoproperties] web-console.user.matched=admin\=admin\n
[exec] [echoproperties] web-console.web.xml.file=c\:/Temp/jboss-4.0.4.GA/se
rver/default/deploy/management/console-mgr.sar/web-console.war/WEB-INF/web.xml
[exec] [echoproperties] working.dir=E:\caArraySource\2.4.1.1\software\
target\dist\exploded\working
[exec] [echoproperties] wscore.download.url=https\://ncisvn.nci.nih.gov/svn
/commonlibrary/trunk/techstack-2006/os-independent/ws-core-enum-4.0.3.zip
[exec] [echo] os.temp.dir - c:/temp/caarray
[exec]
[exec] install:jboss:validation:post-install:
[exec] [echo] Waiting for http://MAGNET14:47210/caarray to respond.
[exec] [echo] ***** INSTALLATION COMPLETED SUCESSFULLY *****
[exec] [echo]
[exec] [echo] To view your application goto http://MAGNET14:47210/caar
ray.
[exec]
[exec] BUILD SUCCESSFUL
[exec] Total time: 144 minutes 26 seconds

BUILD SUCCESSFUL
Total time: 256 minutes 51 seconds
E:\caArraySource\2.4.1.1\software\master_build>
```



Note

The time it takes for the Ant build process to complete also depends on the speed of your Internet connection, as Ant retrieves all the JAR dependencies from the NCI Ivy repository during the build. This retrieval is very bandwidth-intensive, so ideally the build should be run on a connection with at least 100 Mbps throughput.



Warning

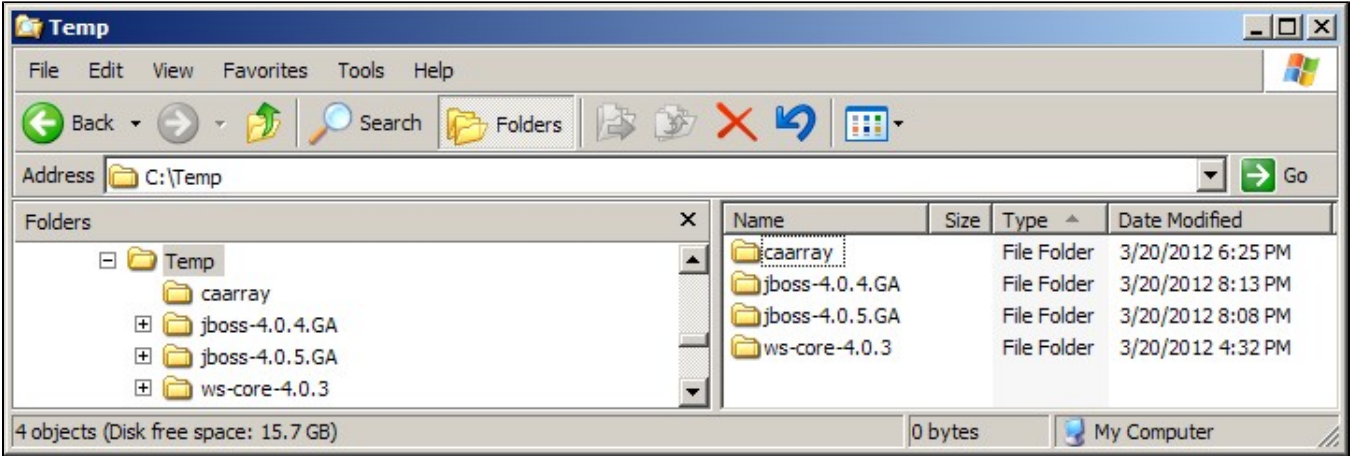
Even if the command line window shows a 'BUILD SUCCESSFUL' message at the end, it is still possible that the build process did not complete successfully. The build process launches several sub-processes, each of which must complete successfully in order for the entire build to complete successfully. If any one of these sub-processes fails, it will display its own 'BUILD FAILED' error message, but the overall build process may still display the 'BUILD SUCCESSFUL' message at the end. The screenshot below illustrates such a case in which a sub-process failed due to an incorrect version of the JDK installed; note the indented 'BUILD FAILED' error message (*highlighted in red*) several lines above the 'BUILD SUCCESSFUL' message (*highlighted in yellow*) at the bottom.

```
C:\WINDOWS\system32\cmd.exe

[exec] [echoproperties] wscore.download.url=https\://ncisvn.nci.nih.gov/svn
/commonlibrary/trunk/techstack-2006/os-independent/ws-core-enum-4.0.3.zip
[exec] [echo] Validating Ant version...
[exec] [echo] Ant Version: PASSED
[exec] [echo] Validating Java version...
[exec] BUILD FAILED
[exec] E:\caArraySource\2.4.1.1\software\target\dist\exploded\build.xml:664
: The following error occurred while executing this line:
[exec] E:\caArraySource\2.4.1.1\software\target\dist\exploded\bda-utils\bda
-build-utils-1.7.5.caarray.xml:7795: The following error occurred while executin
g this line:
[exec] E:\caArraySource\2.4.1.1\software\target\dist\exploded\bda-utils\bda
-build-utils-1.7.5.caarray.xml:707:
[exec] Your Java SDK version must be 1.5. The version you have installe
d is 1.6.0_30. Please install the correct version of the Java SDK and update you
r System PATH to point to directory where you installed it.
[exec]
[exec]
[exec] Total time: 3 minutes 9 seconds

BUILD SUCCESSFUL
Total time: 102 minutes 43 seconds
E:\caArraySource\2.4.1.1\software\master_build>
```

The caArray application is now installed in the path you specified in the `install.properties` file via the `application.base.path` property. In our example, the installation path is `C:\temp`, whose contents are displayed in the screenshot below:



If the contents of your installation path do not match those shown above, it is likely that your build process failed.

Launching The caArray Server Upon Build Completion

To launch the caArray server, navigate to the following path:

```
$INSTALLATION_HOME\jboss-4.0.5.GA\bin
```

where `$INSTALLATION_HOME` is the installation path you specified in `install.properties`, and run the file `run.bat`. A command line window will launch showing the progress of the launching of the caArray server. The launch process is not instant and may take anywhere from 20 to 120 seconds, after which the command line will display a message indicating that the JBoss MX Microkernel has started, as shown at the bottom of the screenshot below:

```
C:\WINDOWS\system32\cmd.exe
rnal.v1_0.data.impl.DataServiceBean ejbName: DataServicev1_0
14:45:33,212 INFO [JmxKernelAbstraction] installing MBean: jboss.j2ee:ear=caarray.ear,jar=caarray-ejb.jar,name=SearchServicev1_0,service=EJB3 with dependencies
14:45:33,243 INFO [EJBContainer] STARTED EJB: gov.nih.nci.caarray.services.external.v1_0.search.impl.SearchServiceBean ejbName: SearchServicev1_0
14:45:33,243 INFO [JmxKernelAbstraction] installing MBean: jboss.j2ee:ear=caarray.ear,jar=caarray-ejb.jar,name=FileRetrievalServiceBean,service=EJB3 with dependencies
14:45:33,259 INFO [EJBContainer] STARTED EJB: gov.nih.nci.caarray.services.file.FileRetrievalServiceBean ejbName: FileRetrievalServiceBean
14:45:33,259 INFO [JmxKernelAbstraction] installing MBean: jboss.j2ee:ear=caarray.ear,jar=caarray-ejb.jar,name=CaArraySearchServiceBean,service=EJB3 with dependencies
14:45:33,274 INFO [EJBContainer] STARTED EJB: gov.nih.nci.caarray.services.search.CaArraySearchServiceBean ejbName: CaArraySearchServiceBean
14:45:33,290 INFO [EJB3Deployer] Deployed: file:/C:/Temp/jboss-4.0.5.GA/server/default/tmp/deploy/tmp6271192806359580549caarray.ear-contents/caarray-ejb.jar
14:45:33,290 INFO [TomcatDeployer] deploy, ctxPath=/caarray, warUrl=.../tmp/deploy/tmp6271192806359580549caarray.ear-contents/caarray-exp.war/
14:45:36,273 INFO [EARDeployer] Started J2EE application: file:/C:/Temp/jboss-4.0.5.GA/server/default/deploy/caarray.ear
14:45:36,398 INFO [Server] JBoss (MX MicroKernel) [4.0.5.GA (build: CVSTag=Branch_4_0 date=200610162339)] Started in 1m:43s:392ms
```

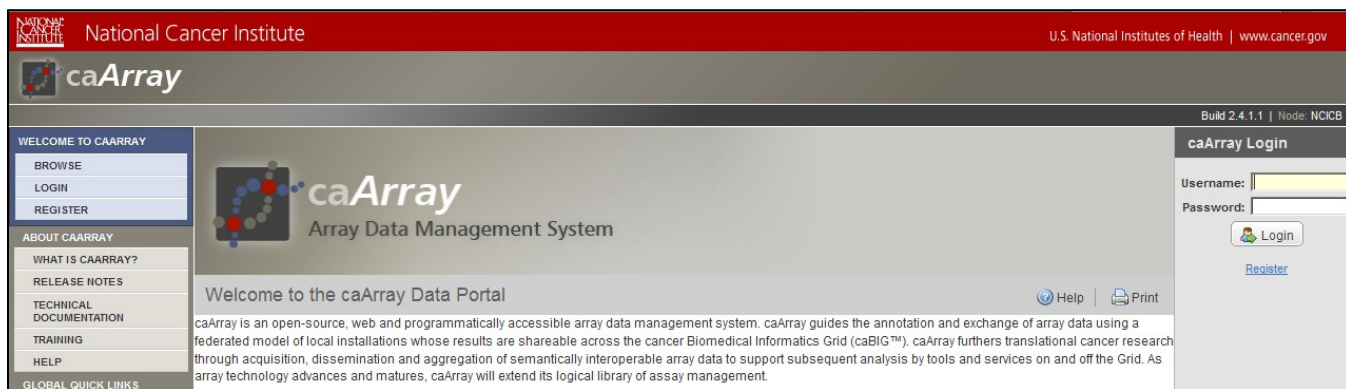
Do not attempt to access the application until you see the message stating that the JBoss kernel has started.

Accessing The caArray Login Page Via Your Web Browser

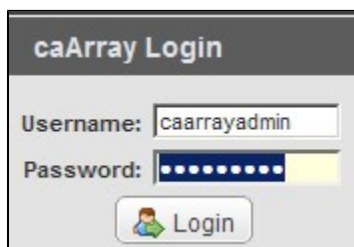
To access the caArray login page, open up a new tab in your Web browser and enter the following URL in the address bar:

```
http://localhost:38080/caarray/
```

The login page will now load as shown in the screenshot below from a Mozilla Firefox tab:



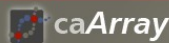
The default installation of caArray comes with several pre-configured user accounts built-in. We can log into the application via any of these accounts. In this tutorial, the one we will log in with is 'caarrayadmin', whose corresponding password is 'caArray2!'. You can enter this username and password into their respective fields under the caArray Login panel at the right of the login page, then click on the 'Login' button beneath to log into the application, as shown in the screenshot below:



You will now be directed to the caArray homepage, which shows the 'My Experiment Workspace', as shown in the screenshot below. Congratulations on successfully building and logging into caArray!

 National Cancer Institute

U.S. National Institutes of Health | www.cancer.gov



Please Select... Search

Build 2.4.1.1 | Node: NCICB | Welcome, caarrayadmin | [Logout](#)

WELCOME TO CAARRAY

HOME

BROWSE

EXPERIMENT MANAGEMENT

MY EXPERIMENT WORKSPACE

CREATE/PROPOSE EXPERIMENT

JOB QUEUE

My Experiment Workspace

[Help](#) | [Print](#)

No items found.

Experiment ID	Experiment Title	Assay Type	Samples	Status	Public Access	Permissions	Reimport/Reparse	Edit	Delete
Nothing found to display.									



Note

caArray comes with a user provisioning tool (UPT) that allows you to modify the built-in accounts as well as to create additional accounts and assign them varying privileges and access levels. For more information on how to install and use UPT, please visit the UPT user guide at [caCO RE_CSM_v421_ProgrammersGuide.pdf](#).

Have a comment?

Please leave your comment in the [caArray End User Forum](#).