Cell Illustrator Server:

Installation Manual

Table of contents:

1 Introduction

Cell Illustrator Server (CI Server) is a companion product of Cell Illustrator. The CI Server plays the role of the calculation backend that is responsible for executing multiple simulation jobs submitted from one or more users of Cell Illustrator. In this mode of work, the Cell Illustrator is on the user's desktop machine ands it plays the role of the calculation front-end (client), where the user prepares a model and starts and controls remote simulations. The client's Cell Illustrator communicates with the CI Server over the standard HTTP protocol. The server mode of simulations does not use up the resources of the user's computer, and he or she may continue to work with Cell Illustrator or other applications after submitting many simulations to a server.

A remote simulation is executed with the following steps:

-   A model is prepared in CI Workspace and saved in a file in the CSML format (Cell Illustrator native format). In the model, the user selects model entities and processes whose states he would like to observe.

-   The model is submitted to the CI Server, where the simulation is executed. The state of the selected entities and processes is recorded in a log. The log contains the input model as well.

-   The simulation log file is downloaded to the local computer when the simulation finishes at the server. The log file is placed in the standard output directory of the input model and can be opened, viewed and analyzed with the CI Player application.

Technically, CI Server is a *web service* installed at the Apache Tomcat web server. This manual describes the process of the installation and configuration of CI Server which requires some system administration skills. Additionally, the option of replacing the default simulation engine at the server with a user-written program is described in the chapter Replacing the Simulation Engine.

The documentation for the Cell Illustrator software includes the manuals listed below:

Cell Illustrator User Manual: Introduction to Cell Illustrator system

Cell Illustrator Reference Manual: Detailed description of Cell Illustrator functionalities

Cell Illustrator Player Reference Manual: Detailed description of Cell Illustrator Player functionalities

Cell Illustrator Server Installation Manual: Installation instructions for Cell Illustrator Server product (for IT System Administrators)

This manual describes version 2.0 of Cell Illustrator (CI2.0).

2 Requirements

The following are the minimum requirements that must be met to install and run CI Server.

2.1 Operating System

- Linux

- Windows

- any other OS with Java 1.5, Apache Tomcat and Apache Axis

2.2 Web Server

- Apache Tomcat version 5.0 or higher

- Apache Axis version 1.2RC3

- Java SDK compatible with Apache Axis

2.3 Simulation Engine

- Java Runtime Environment, JRE version 1.5.0 or higher

2.4 Recommended Hardware

- Pentium® IV processor – 2 GHz or faster

- 512 MB RAM or more

- 1 GB available hard disk space

3 Installation Procedure

The installation of a web service requires administrative skills. You need to be familiar with such software packages and concepts as Java Runtime Environment, Apache Tomcat, and web services.

3.1 Simulation Engine Installation

The simulation engine used in the CI Server is the standard Cell Illustrator invoked in the command line mode. However, you will need a special license to run Cell Illustrator as the server engine. Besides installing Cell Illustrator, you will need to install and configure additional software components to make CI Server work.

3.1.1 Install Cell Illustrator

Start the executable that corresponds to your operating system (e.g. *CI2.0_lj.bin* for Linux or *CI2.0_lj.bin* for Linux without Java) and follow the instructions on the screen

For details on CI installation see the chapter Install Cell Illustrator in the Cell Illustrator User Guide.

3.1.2 Licensing Information

To run Cell Illustrator Simulation Engine, you need to obtain a license file from GNI or your local distributor of Cell Illustrator. Copy the license file to the directory where the Cell Illustrator is installed. For example, on Windows systems, the Cell Illustrator is installed to "C:\Program Files\GNI\Cell Illustrator" if the default installation settings are used.

The license required for the CI Server is different from the license for the Cell Illustrator Workspace.

3.2 Web Service Installation

Before beginning the installation procedure, make sure that Java Software Development Kit (Java SDK) and Apache Tomcat are installed on the machine that will serve as the Cell Illustrator Server.

3.2.1 Java SDK

Java SDK is available for download at: http://java.sun.com/downloads. You can install any Java SDK that is compatible with Apache Axis and Apache Tomcat. Apache Axis 1.2RC3 and Apache Tomcat 5.0 require Java SDK 1.4.2 or higher.

3.2.2 Apache Tomcat

Apache Tomcat web server can be downloaded from http://jakarta.apache.org/tomcat. After installing Apache Tomcat, make sure that it is

working: start the web server, open your web browser and try to open the Apache Tomcat start page (normally at: http://localhost:8080/) in your browser. If the page is not displayed, read the error message and fix the configuration:

- You may need to define the *JAVA_HOME* environment variable to point to the Java SDK home folder

- You may need to download and add *servlet.jar* to the directory Tomcat/Common/Lib

3.2.3 Install Apache Axis

Apache Axis is one of the standard Java packages for Web Services. You can download Axis 1.2 from http://ws.apache.org/axis/java.

- Download Apache Axis and uncompress it.

- Copy the unzipped folder webapps/axis to the Tomcat webapps folder

- Restart Tomcat server

- Go to the Apache-Axis home page. It should be accessible at: http://localhost:8080/axis/

If the page is not displayed, read the error message and fix the configuration. If you try to access this page for the first time, Tomcat will require the Java compiler (javac) to compile the jsp page. If the Java compiler cannot be found by Tomcat verify whether the *JAVA_HOME* environment variable is correct.

In case of any problems, consult the Axis installation manual which is available at the Apache Axis page http://ws.apache.org/axis/java.

3.2.4 Apache Axis Installation Test

The Apache Axis requires packages (jar files) which are not included in the standard Apache Tomcat installation. Therefore you will need to find the missing components and copy them to the Tomcat common folder. The Apache Axis installation can be validated by visiting the Apache Axis Happiness Page:

- Go to the Apache-Axis home page. Normally, it can be accessed at: http://localhost:8080/axis/

- Click on the *Validation* hyperlink. This will validate the Axis installation and display the report.

- Read the validation report and make sure that all required and optional Axis components are installed. If a component is missing, find it on the web and download and copy to the Tomcat/common/lib folder. For example, you may need to download the missing jars: *activation.jar* and *mail.jar*.

3.2.5 Deploy the Web Service

- Open the folder CIServer in the Cell Illustrator installation CD.

- Copy the *net* folder from the CD to the Tomcat subfolder *webapps/axis/WEB-INF/classes*

- Check and make sure that the file *CIRemoteSimulation.class* exists in *webapps/axis/WEB-INF/classes/net/genomicobject/server/services/* (the letter case matters in Apache Tomcat)

- Start Apache Tomcat.

- Copy the file *deploy.wsdd* to *axis/WEB-INF* folder

- Deploy the web service by executing the command, like that:

cd webapps/axis/WEB-INF/

java -cp "classes;lib/activation.jar;lib/axis-ant.jar;lib/axis.jar;lib/commons-discovery.jar;lib/commons-logging.jar;lib/jaxrpc.jar;lib/log4j-1.2.8.jar;lib/saaj.jar;lib/wsdl4j.jar;lib/xmlsec.jar" org.apache.axis.client.AdminClient deploy.wsdd

The file *deploy.wsdd* can be found in the Cell Illustrator installation CD in the folder CIServer.

- Restart Apache Tomcat

3.2.6 Web Service Test

Check that the web service was successfully deployed, by executing the following actions:

- Open the page http://localhost:8080/axis/services/CIRemoteSimulation with your web browser.

- The message "CIRemoteSimulation. Hi there, this is an AXIS service!" should be displayed.

If the web service is password protected you will be asked for the user name and password, before the page is displayed.

3.2.7 Password Protection

It is recommended to protect the web service by a password. This canbe done by using the standard BASIC-AUTH (Basic Authentication), which is offered by the Tomcat Web Server. To set up password protection, please do the following:

- Add a user and role to the TOMCAT web server. Edit the *TOMCAT HOME/conf/tomcat-users.xml file*. In this file you should place the name of each user who will have access to protected resources, and his "role". In the following example, roles named "CIrole", and a "CIuser" which belong to that role are created:

```xml
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="CIrole"/>
  <role rolename="tomcat"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="CIuser" password="passwd" roles="CIrole"/>
</tomcat-users>
```

- Set the authentication. Add to the web application descriptor at */axis/Web-inf/web.xml* the following information:

```xml
<security-constraint>
  <web-resource-collection>
    <web-resource-name>CIrole</web-resource-name>
    <url-pattern>/services/CIRemoteSimulation</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>CIrole</role-name>
  </auth-constraint>
 </security-constraint>
 <login-config>
   <auth-method>BASIC</auth-method>
   <realm-name>CIrole</realm-name>
 </login-config>
```

where :
```
<url-pattern> points to the protected resource.
<role-name> refers to the role defined in tomcat-users.xml file.
```

To ensure that the password protection is active, repeat the test described in section Web Service Test.

4 Configuration

The Installation Procedure of the CI Server was described in the previous section. If you have successfully installed both CI Server components: the simulation engine and the web service, you are ready to start the CI Server configuration. The configuration ties the web service and the simulation engine together in the local environment, i.e. it specifies where to find and how to start the simulation engine. For this you must set two obligatory variables in the *CIServer.properties* file:

- *engineDirectory* is the path to the simulation engine directory. Under normal circumstances, this is the directory, where you have installed Cell Illustrator – see Simulation Engine Installation. This path will be used by the web service to start the simulation engine.

- *uploadDirectory* is the path to the upload directory. The upload directory is the working directory where the simulation input files will be uploaded and simulation output files are written.

To set the abovementioned environment variables, do the following:

- Copy the file *CIServer.properties* from the installation CD to the *webapps/axis* folder.

- Open the *CIServer.properties* file.

- Set the obligatory variables: *engineDirectory* and *uploadDirectory*.

Note that properties file format uses the ISO 8859-1 character encoding. Characters that cannot be directly represented in this encoding can be written using Unicode escapes. E.g. the ASCII character \ should be written as \\.

- Save and close the file.

The variables *engineDirectory* and *uploadDirectory* must be defined in the *CIServer.properties* file. Optionally, you can also set the following variables:

- *javaExePat*h – the path to the java executable.

- *javaXMX* – the –Xmx parameter (maximum memory) for the java process.

- *javaXMS* - the –Xms parameter (initial memory) for the java process.

- *javaClassPath* – the classpath for the java executable.

- *javaMainClass* – the main class, the simulation engine entry point.

Note that the parameters j*avaClassPath* and j*avaMainClass* should be changed only if you wish to use a different simulation engine – see chapter Replacing the Simulation Engine.

5   Installation Validation

This chapter describes how to validate whether the CI Server has been properly installed and configured. At first, the validation test should be performed on the same computer where the CI Server has been installed. If the test passes successfully, it is recommended to repeat the test on another computer that is connected via LAN to the server.

5.1   URL Test

Repeat the test described in the chapter Web Service Test. If you are connecting to a remote server use the name of the remote computer instead of the local host.

5.2   Connection Test

- Open the Cell Illustrator application

- In the Cell Illustrator Workspace, select the menu command *Simulation | Setup Servers*. This will open the corresponding dialog box.

- In the server list, select the server that you want to test and press the Test button. The message box "Connection successful" should be displayed.

If the server that you want to test is not in the server list, press the Add button to register the server in the CI Workspace – for more information refer to the Cell Illustrator Reference Manual.

5.3   Remote Simulation Test

- Open the standard sample file *notch.gon* in Cell Illustrator workspace. This file can be found in the subfolder *sample/notch* of the CI installation directory.

- Open the Element List dialog box (Dialog | Element Lists) and make sure, that at least one entity is selected for logging.

- Select Simulation | Send to Server | <server_name> to submit the file to the server for simulation.

The Submitted Simulations dialog should be displayed and you should see the job progress. The simulation should finish successfully.

6 Replacing the Simulation Engine

This chapter is intended for a user who wants to integrate his own simulation program with Cell Illustrator front-end. This can be done by replacing the default CI Simulation Engine component in the CI Server with the custom engine. The user-created engine must fulfill the following requirements:

- read input file from a CI model file, i.e., the model must be in the CSML format

- write simulation log file in the CSML format to the output directory

- write the simulation status file in the format compatible with the CI Server

- check for the stop file (optionally)

To install such an engine within the CI Server environment, one needs to do the following:

- Install the engine at the computer where CI Server is installed

- Configure CI Server so that it will start the engine. To accomplish this, edit the CI Server configuration file which is described in the chapter Configuration. Beside setting the obligatory variables, you will have to set the variables: *javaExePath, javaClassPath, javaMainClass*