Telosys 1.0.2 Starter Kit

How to create a new Telosys project in Eclipse 3.2 WTP and how to generate the Data Access Layer
Sommaire
1 – Install the “TelosysTools plug-in” in Eclipse......................................................... 3
2 – Define the JRE............................................................................................................. 3
3 – Define the Tomcat server .......................................................................................... 5
4 – Create a standard Dynamic Web Project................................................................. 7
5 – Test projet with Tomcat.............................................................................................. 10
6 – Import the Telosys Starter Kit in the project .......................................................... 13
7 – Run the project ........................................................................................................... 15
8 – Configure the project................................................................................................. 17
9 – Install the “Bookstore” DERBY database for tests ............................................... 18
10 – Connect to DERBY database .................................................................................. 19
11 – Test the project “Data Access Layer” with the database........................................ 23
1 – Install the “TelosysTools plug-in” in Eclipse

Stop Eclipse

Download telosys-tools-plugin_X.X.X.zip from the Telosys web site http://telosys.ow2.org/ → Download

Unzip file telosys-tools-plugin_X.X.X.zip in the Eclipse « plugins » directory

Restart Eclipse

Check the “Plugin details” : “Help” → “About Eclipse” → “Plug-in details”
The Telosys Tools plug-in must be in the list.

2 – Define the JRE

Window → Preferences …

Select « Add » button and browse JRE home directory
Choose a full Java SDK, because a Java compiler is required ( to compile the JSP )

Telosys supports all the Java versions from 1.4
Select the JRE just added (other unused JRE can be removed)

Select « OK » button
3 – Define the Tomcat server

Select Window → Preferences …

Select « Add » button
Select « Next » button and:
- browse for Tomcat installation directory
- select java version

Select « Finish » button
4 – Create a standard Dynamic Web Project

File → New (Alt+Shift+N) → Project …

Web → Dynamic Web Project   (with Tomcat 5.0)
Select « Next » button
Select « Next » button

Select « Finish » button
5 – Test projet with Tomcat

Create the « test.jsp » file in WebContent directory

Right click on WebContent directory → New → JSP

![New JavaServer Page dialog box](image)

Select « Finish » button

Insert “hello” in the <body> tag of the JSP file
Right click on test.jsp file → Run As → Run on Server

Select « Next » button
Make sure that your project is in the « Configured projects » part.

Select « Finish » button

Verify the good execution of your JSP file.
6 – Import the Telosys Starter Kit in the project

Stop Tomcat server

Right click on your project → File → Import → Import …

Select General → Archive File

Select « Next » button
Browse to select the file `telosys-x.x.x-StarterKit.zip` ...

Select « Finish » button

Overwrite `web.xml` in folder `xxx/WebContent/WEB-IN` ? → Yes

That’s all !
Your project is now a “Telosys project”.
7 – Run the project

Right click on your project → Run As → Run on server
Telosys test page

Test servers presence

- Test screenmap server without screen name
- Test actions server without action name
- Test services server with nonexistent service

Test servers with real resources

- Test screenmap server with an existing screen ('test')
- Test services server with an existing RPC service ('ping')

Check there’s no error in the console.

Stop Tomcat Server
8 – Configure the project

Edit the Telosys project properties:

Select your project + right click → “Properties” → Telosys

You can define here:
- The repository’s folder
- The templates folder
- The names of the Java packages
- The construction of the classes names
- etc ...
9 – Install the “Bookstore” DERBY database for tests


Unzip the file in a folder (this folder will be the DERBY_HOME folder)

Start DERBY by running the command file

    DERBY_HOME/start.bat

or

    DERBY_HOME/bin/startNetworkServer.bat (Windows)
    DERBY_HOME/bin/startNetworkServer (shell Unix/Linux)

The database engine is running in a command window, and is now ready to accept connections.

To stop the Derby database: just close this window.
10 – Connect to DERBY database

Edit the “databases.dbcfg” file with a “text editor” and uncomment DERBY part

Save and close databases.dbcfg file an open it with the “Dbconfig editor”
Select database

Select « **Test connection** » button to test connection to database

Select « **Generate repository** » button

Repository is generate in **TelosysTools/repos** directory

To allow the dynamic association between the entity name and its Value Object bean and its DAO edit **WebContent/WEB-INF/conf/telosys.properties** and uncomment these two properties “daoClass” and “recordClass”:

```properties
# DAO class name pattern for dynamic DAO registry/regist
daoclass = vo.dao.\$(BEANNAME)DAO
# daoClass = vo.dao.db$\$(DBID).\$(BEANNAME)DAO
# Value Object Bean class name pattern to determine the
# ( N5 : can be overridden in 'databases.dbcfg' )
recordclass=vo.\$(BEANNAME)
```

Be sure that the classes names are conform to the Eclipse project properties ( packages and classes names )
Double click on the repository file to edit database tables

**Generate tables one by one:**
You can generate files for a table by selecting « Generate » button. These files are create in packages defines in `telosys-tools.cfg` file (if packages definition are modify in this file, you have to regenerate repository) Select « Config » tab to see defined packages.

**Generate all the tables:**
Select « Bulk generation » tab
Select « Select All » buttons
Select « Launch bulk generation » button

The project “Data Access Layer” is now ready to use.
11 – Test the project “Data Access Layer” with the database

Right click on your project → Run As → Run on server

By default each DAO can be called in REST mode (via an URL):

<table>
<thead>
<tr>
<th>How to call a DAO in REST mode (request examples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dao/Author/loadlist</td>
</tr>
<tr>
<td>/dao/Author/loadlist?where=id &lt; 5</td>
</tr>
<tr>
<td>/dao/Author/load?id=1</td>
</tr>
</tbody>
</table>

Select one of the links to see the results