

DAR Software Development Notes

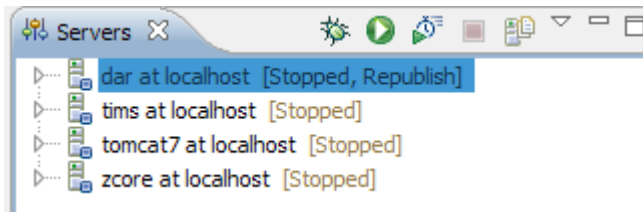
The target server for DAR development should be Apache Tomcat 6. Dar/web/WEB-INF/web.xml uses the 2.4 api.

My usual process is to first install the DAR executable - <https://www.ictedge.org/files/zcore/dar/dar-1.1s-windows-installer.exe> - and then use that version of Tomcat as the target server. I usually install that version at C:\dar instead of the default C:\Program Files\... in order to avoid any path issues.

To setup the server, here is one way:

- Right-click on the project name in Eclipse and choose Properties
- In Targeted Runtime, click Targeted Runtimes and choose New... In the New Server Runtime Environment dialogue box, choose Apache Tomcat 6 and click Next.
- In the tomcat Server dialogue box, Browse to the installation directory – in my case, it is at C:\dar. The JRE is set to Workspace JRE. (My Workspace JRE is currently JDK1.6.0_23.)

The server should now display in the Servers view below Project Explorer (assuming you are in Java EE perspective).



Double-click on this server. This will reveal the server Overview. Configure as below:

Overview

General Information
Specify the host name and other common settings.

Server name:

Host name:

Runtime Environment:

Configuration path:

[Open launch configuration](#)

Server Locations
Specify the server path (i.e. catalina.base) and deploy path. Server must be published with no modules present to make changes.

Use workspace metadata (does not modify Tomcat installation)

Use Tomcat installation (takes control of Tomcat installation)

Use custom location (does not modify Tomcat installation)

Server path:

[Set deploy path to the default value](#)

Deploy path:

Server Options
Enter settings for the server.

Serve modules without publishing

Publish module contexts to separate XML files

Enable security

Enable Tomcat debug logging (not supported by this Tomcat version)

Publishing
Modify settings for publishing.

Never publish automatically

Automatically publish when resources change

Automatically publish after a build event

Publishing interval (in seconds):

Select publishing actions:

Update context paths

Timeouts
Specify the time limit to complete server operations.

Start (in seconds):

Stop (in seconds):

Ports
Modify the server ports.



Port Name	Port Number
Tomcat admin port	8005
HTTP/1.1	8088
AJP/1.3	8009

MIME Mappings

Click the Modules tab at the bottom of this view and select "Add Web Module..." and add dar. This is how it looks once it is setup:

Web Modules

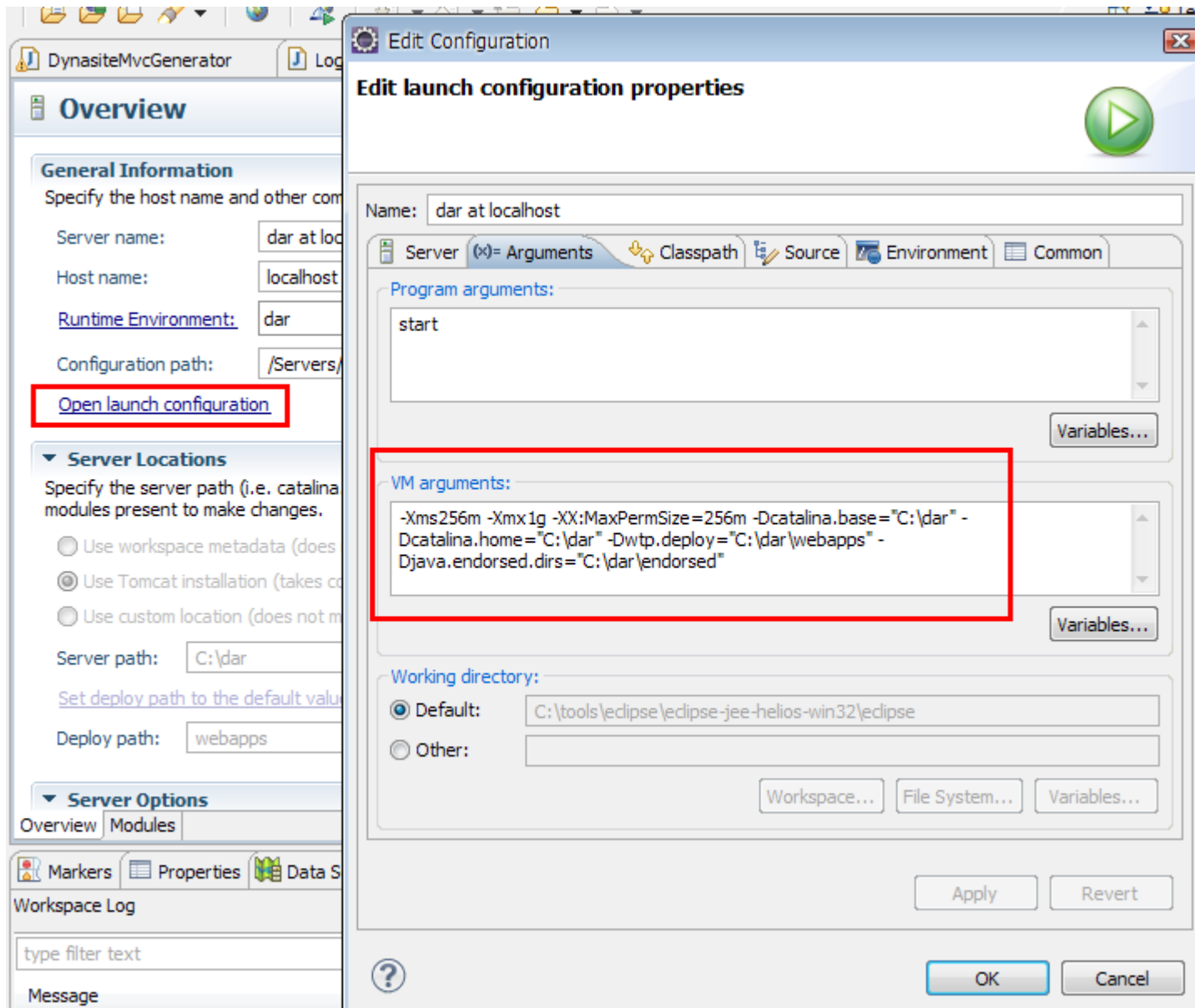
Configure the Web Modules on this server.

Path	Document Base	Module	Auto Reload
 /dar	dar	 dar	Enabled

Buttons: Add Web Module..., Add External Web Module..., Edit..., Remove

Overview | **Modules**

Here are the tomcat server VM args in the launch configuration:



Once you have the server configured, check your Build path in the project properties (Right-click on the project name, select Properties -> Java Build Path). Here is my build path:

Properties for dar

type filter text

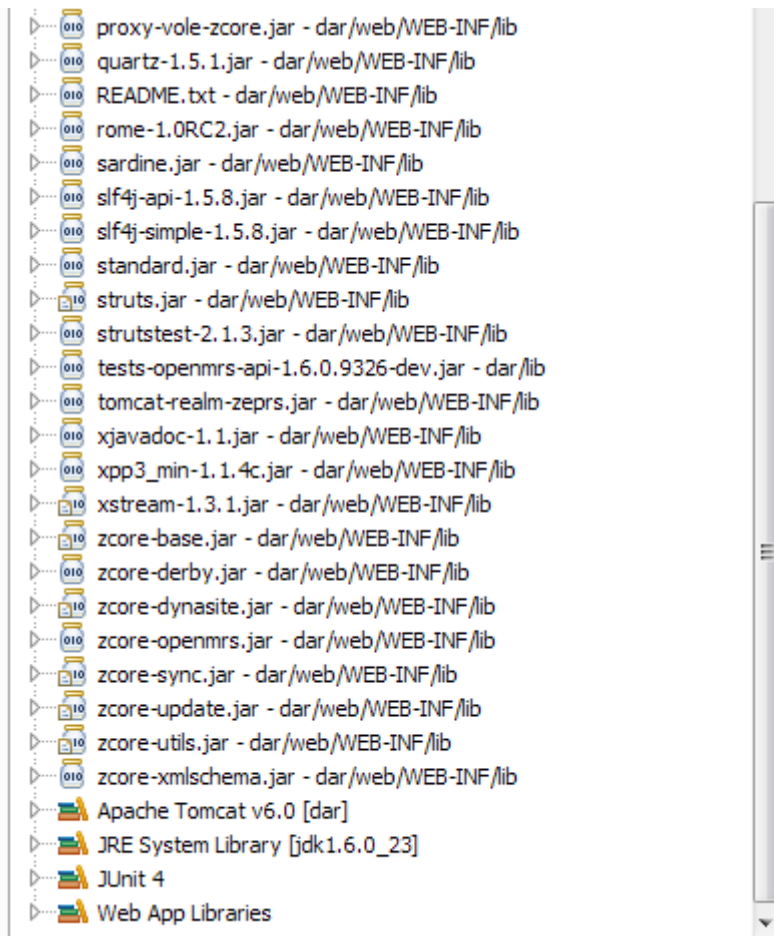
- ▶ Resource
- ... AnyEdit Tools
- ... Builders
- ... Deployment Assembly
- ... Generate Test Cases
- ▶ **Java Build Path**
- ▶ Java Code Style
- ▶ Java Compiler
- ▶ Java Editor
- ... Javadoc Location
- ▶ JavaScript
- ... JSP Fragment
- ... Project Facets
- ... Project References
- ... Refactoring History
- ... Run/Debug Settings
- ... Server
- ... Service Policies
- ... SVN Info
- ... Targeted Runtimes
- ▶ Task Repository
- ... Task Tags
- ▶ Validation
- ... Web Content Settings
- ... Web Page Editor
- ... Web Project Settings
- ... WikiText
- ▶ XDodet

Java Build Path

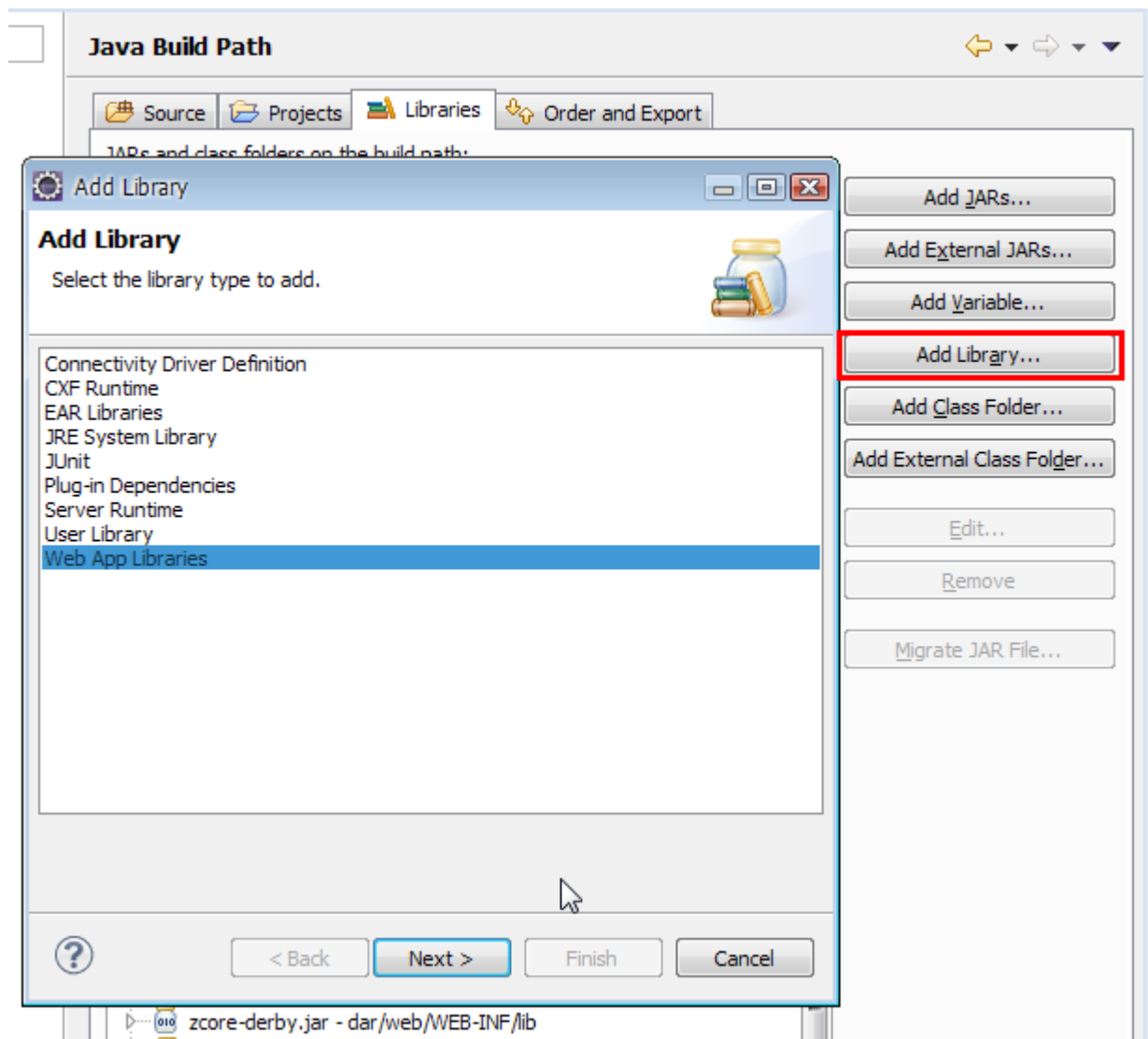
Source Projects Libraries Order and Export

JARs and class folders on the build path:

- ▶ ant-launcher.jar - dar/web/WEB-INF/lib
- ▶ ant.jar - dar/web/WEB-INF/lib
- ▶ bsh-1.3.0.jar - dar/web/WEB-INF/lib
- ▶ commons-beanutils.jar - dar/web/WEB-INF/lib
- ▶ commons-codec-1.4.jar - dar/web/WEB-INF/lib
- ▶ commons-collections-3.1.jar - dar/web/WEB-INF/lib
- ▶ commons-configuration-1.1.jar - dar/web/WEB-INF/lib
- ▶ commons-dbc-1.2.1.jar - dar/web/WEB-INF/lib
- ▶ commons-dbutils-1.3.jar - dar/web/WEB-INF/lib
- ▶ commons-digester.jar - dar/web/WEB-INF/lib
- ▶ commons-io-1.4.jar - dar/web/WEB-INF/lib
- ▶ commons-lang-2.0.jar - dar/web/WEB-INF/lib
- ▶ commons-logging-1.1.jar - dar/web/WEB-INF/lib
- ▶ commons-logging-api-1.1.jar - dar/web/WEB-INF/lib
- ▶ commons-pool-1.2.jar - dar/web/WEB-INF/lib
- ▶ commons-validator.jar - dar/web/WEB-INF/lib
- ▶ dom4j-1.6.1.jar - dar/web/WEB-INF/lib
- ▶ dwr.jar - dar/web/WEB-INF/lib
- ▶ hibernate-tools.jar - dar/web/WEB-INF/lib
- ▶ hibernate3.jar - dar/web/WEB-INF/lib
- ▶ httpclient-4.1.jar - dar/web/WEB-INF/lib
- ▶ httpcore-4.1.jar - dar/web/WEB-INF/lib
- ▶ jackson-core-asl-1.3.0.jar - dar/web/WEB-INF/lib
- ▶ jackson-mapper-asl-1.3.0.jar - dar/web/WEB-INF/lib
- ▶ jakarta-oro.jar - dar/web/WEB-INF/lib
- ▶ jdom-1.1.jar - dar/web/WEB-INF/lib
- ▶ jettison-1.0.1.jar - dar/web/WEB-INF/lib
- ▶ jmx.jar - dar/web/WEB-INF/lib
- ▶ ioda-time-1.6.jar - dar/web/WEB-INF/lib



To add Libraries:



Note the location of test-openmrs-api – I do not deploy that library since it is only used for BDD unit tests.

