Eclipse

- Install Eclipse & Plugins
  - Eclipse Kepler (3.9)
  - Eclipse Indigo (3.7)
    - Autoshare
    - m2e 1.0 known issue
    - Table of plugins which currently do not have a m2e connector
    - Other handy plugins
- Maven Repository
- Add SVN Repository
- Create/Checkout a Project
  - New Checkout
  - Importing an already checked out project from file system
- Setup the Server (Tomcat)
  - Create a Server
  - Deploy the Webapp project
  - Open Server Edit Pane
  - Server settings: Disable publishing
  - Server settings: Increase timeouts
- Freemarker specific settings
  - Taglib definitions (tld)
  - Auto completion
- Configure Code Style
- Tips and Tricks
  - Get the sources or a linked jar / module
  - Dependencies
  - Use company wide Java Code Style & Compiler preferences
- Known Problems
  - Wrong maven version is used for dependency checks
  - Freemarker can't find tags (tlds)
  - Project contains duplicate entry
  - Class Cast Exception: info.magnolia.cms.filters.MgnlMainFilter
  - Webapp project doesn't work
  - NPE when configuring tomcat with "don't publish" option
  - webapp is not recognized as webapp and doesn't show in tomcat when trying to add it
  - modules are not installed when installing the webapp
  - Module Dependency Exception

Install Eclipse & Plugins

Eclipse Kepler (3.9)

As of July 9th 2013 Kepler based image is available on file server with same set of plugins as those installed for 3.7 with addition of

- JBoss Tools (for editing BPMN workflow definitions)
- Vaadin plugin
- JRebel plugin

After installing this bundle

- reconfigure location of checkstyle header files according to Checkstyle for Eclipse (In difference to the page i would recommend to just checkout `magnolia-build-tools` directly from svn and point header properties for ce and ee checkstyle configurations directly to those files).
- it might be also necessary to click "Apply" again on formatting configuration to get it being used by eclipse (pbly related to workspace config?). See more details at Eclipse setup
- I had some performance issues with Indigo and git reindexing which was the primary reason for making Kepler bundle. Those issues were not solved by installing Kepler but only by trashing the workspace and recreating it again ... do the same if in doubt.
- If you have any problems with Kepler bundle, please report them along w/ solution or workaround here.

Complete list of installed plugins:

Installed Software | Installation History | Features | Plug-ins | Configuration

Name	Version	Id	Provider
AnyEditTools	2.4.4.201307082032	AnyEditTools.feature.group	Andrey Loskutov
▶ Apache Deltaspike Tools	1.5.0.Beta2–v2013...	org.jboss.tools.cdi.deltaspike.feature.feat...	JBoss by Red Hat
▶ Context and Dependency Injection Tools	1.5.0.Beta2–v2013...	org.jboss.tools.cdi.feature.feature.group	JBoss by Red Hat
Eclipse Checkstyle Plug-in	5.6.1.201306282206	net.sf.eclipsecs.feature.group	http://eclipse-cs.sf.net/
▶ Eclipse IDE for Java EE Developers	2.0.0.20130613–0...	epp.package.jee	
Forge Tools	1.2.0.Beta2–v2013...	org.jboss.tools.forge.feature.feature.group	JBoss by Red Hat
FreeMarker IDE	1.3.0.Final–v20130...	org.jboss.ide.eclipse.freemarker.feature.f...	JBoss by Red Hat
Hibernate Tools	3.7.0.Beta2–v2013...	org.hibernate.eclipse.feature.feature.group	JBoss by Red Hat
JBoss Archives Tools	3.4.100.Beta2–v20...	org.jboss.ide.eclipse.archives.feature.feat...	JBoss by Red Hat
▶ JBoss Central Community	1.2.0.Beta2–v2013...	org.jboss.tools.community.central.featur...	JBoss by Red Hat
JBoss JAX–RS Tools	1.5.0.Beta2–v2013...	org.jboss.tools.ws.jaxrs.feature.feature.group	JBoss by Red Hat
▶ JBoss Maven CDI Configurator	1.5.0.Beta2–v2013...	org.jboss.tools.maven.cdi.feature.feature....	JBoss by Red Hat
▶ JBoss Maven Endorsed Libraries Configurator	1.5.0.Beta2–v2013...	org.jboss.tools.maven.jdt.feature.feature....	JBoss by Red Hat
▶ JBoss Maven Hibernate Configurator	1.5.0.Beta2–v2013...	org.jboss.tools.maven.hibernate.feature.f...	JBoss by Red Hat
▶ JBoss Maven Integration	1.5.0.Beta2–v2013...	org.jboss.tools.maven.feature.feature.group	JBoss by Red Hat
▶ JBoss Maven Portlet Configurator	1.5.0.Beta2–v2013...	org.jboss.tools.maven.portlet.feature.feat...	JBoss by Red Hat
▶ JBoss Maven Project Examples	1.5.0.Beta2–v2013...	org.jboss.tools.maven.project.examples.f...	JBoss by Red Hat
▶ JBoss Maven Seam Integration	1.5.0.Beta2–v2013...	org.jboss.tools.maven.seam.feature.featu...	JBoss by Red Hat
▶ JBoss OpenShift Tools	2.5.0.Beta2–v2013...	org.jboss.tools.openshift.express.feature....	JBoss by Red Hat
▶ JBoss Portlet	1.5.0.Beta2–v2013...	org.jboss.tools.portlet.feature.feature.group	JBoss by Red Hat
JBoss Runtime Detection Core	2.1.0.Beta2–v2013...	org.jboss.tools.runtime.core.feature.featu...	JBoss by Red Hat
JBoss Runtime Seam Detector	3.5.0.Beta2–v2013...	org.jboss.tools.runtime.seam.detector.fe...	JBoss by Red Hat
JBoss Stacks Tools	1.0.0.Beta2–v2013...	org.jboss.tools.stacks.core.feature.featur...	JBoss by Red Hat
JBoss Tools Community Project Examples	1.5.0.Beta2–v2013...	org.jboss.tools.community.project.examp...	JBoss by Red Hat
JBoss Tools EGit Integration (Experimental)	2.5.0.Beta2–v2013...	org.jboss.tools.openshift.egit.integration....	JBoss by Red Hat
JBoss Tools Foundation	1.0.0.Beta2–v2013...	org.jboss.tools.foundation.feature.featur...	JBoss by Red Hat
▶ JBoss Tools Java Standard Tools	3.5.0.Beta2–v2013...	org.jboss.tools.jst.feature.feature.group	JBoss by Red Hat
JBoss Tools JDT Extensions	3.5.1.Beta2–v2013...	org.jboss.tools.common.jdt.feature.featu...	JBoss by Red Hat
▶ JBoss Tools JSF	3.5.0.Beta2–v2013...	org.jboss.tools.jsf.feature.feature.group	JBoss by Red Hat
JBoss Tools LiveReload	1.0.0.Beta2–v2013...	org.jboss.tools.livereload.feature.feature....	JBoss by Red Hat
▶ JBoss Tools Maven Packaging Configurator	1.5.0.Beta2–v2013...	org.jboss.tools.maven.jbosspackaging.fe...	JBoss by Red Hat
▶ JBoss Tools Maven Source Lookup	1.5.0.Beta2–v2013...	org.jboss.tools.maven.sourcelookup.feat...	JBoss by Red Hat
JBoss Tools Mobile Browser Simulator	3.5.0.Beta2–v2013...	org.jboss.tools.vpe.browsersim.feature.fe...	JBoss by Red Hat
▶ JBoss Tools RichFaces	3.5.0.Beta2–v2013...	org.jboss.tools.richfaces.feature.feature.g...	JBoss by Red Hat
JBoss Tools Usage Reporting	1.2.1.Beta2–v2013...	org.jboss.tools.usage.feature.feature.group	JBoss by Red Hat
JBoss Tools Visual Page Editor	3.5.0.Beta2–v2013...	org.jboss.tools.vpe.feature.feature.group	JBoss by Red Hat
▶ JBoss WebServices Tools	1.5.0.Beta2–v2013...	org.jboss.tools.ws.feature.feature.group	JBoss by Red Hat
▶ JBossAS Tools	2.4.100.Beta2–v20...	org.jboss.ide.eclipse.as.feature.feature.group	JBoss by Red Hat
JMX Console	1.4.0.Beta2–v2013...	org.jboss.tools.jmx.feature.feature.group	JBoss by Red Hat
JRebel	5.3.1.RELEASE–201...	org.zeroturnaround.eclipse.feature.featur...	ZeroTurnaround
▶ JRebel for Java EE	5.3.1.RELEASE–201...	org.zeroturnaround.eclipse.wtp.feature.fe...	ZeroTurnaround
JRebel m2eclipse	5.3.1.RELEASE–201307081557	urnaround.eclipse.m2e.feature.f...	ZeroTurnaround
Maven Integration for Eclipse JDT APT	1.0.1.201209200721	org.jboss.tools.maven.apt.feature.feature....	JBoss, by Red Hat
▶ Maven Profiles Management	1.5.0.Beta2–v2013...	org.jboss.tools.maven.profiles.feature.fea...	JBoss by Red Hat
Project Examples	1.5.0.Beta2–v2013...	org.jboss.tools.project.examples.feature.f...	JBoss by Red Hat
▶ Seam 3 Tools	1.5.0.Beta2–v2013...	org.jboss.tools.cdi.seam.feature.feature.g...	JBoss by Red Hat
▶ Seam Tools	3.5.0.Beta2–v2013...	org.jboss.tools.seam.feature.feature.group	JBoss by Red Hat
▶ Subversive Revision Graph (Optional)	1.1.0.I20130619–1...	org.eclipse.team.svn.revision.graph.featu...	Eclipse.org
▶ Subversive SVN Integration for the Mylyn Project (Optional)	1.1.0.I20130619–1...	org.eclipse.team.svn.mylyn.feature.group	Eclipse.org
Subversive SVN JDT Ignore Extensions (Optional)	1.1.0.I20130619–1...	org.eclipse.team.svn.resource.ignore.rule...	Eclipse.org
Subversive SVN Team Provider	1.1.0.I20130619–1...	org.eclipse.team.svn.feature.group	Eclipse.org
Subversive SVN Team Provider Localization (Optional)	1.1.0.I20130619–1...	org.eclipse.team.svn.nl1.feature.group	Eclipse.org
▶ Subversive SVN Team Provider Sources	1.1.0.I20130619–1...	org.eclipse.team.svn.source.feature.group	Eclipse.org
Vaadin Plug–in for Eclipse	2.2.0.201306261112	com.vaadin.integration.eclipse.feature.group	Vaadin Ltd

Eclipse Indigo (3.7)

As of 4th April 2012 **our SVN repository will be gradually migrated to GIT**. The new Eclipse bundle is based on latest (as of April 2012) Indigo and also contains EGit plugin, an Eclipse Team provider for the Git version control system. For any question concerning Eclipse and Git integration, please read the EGit User guide.
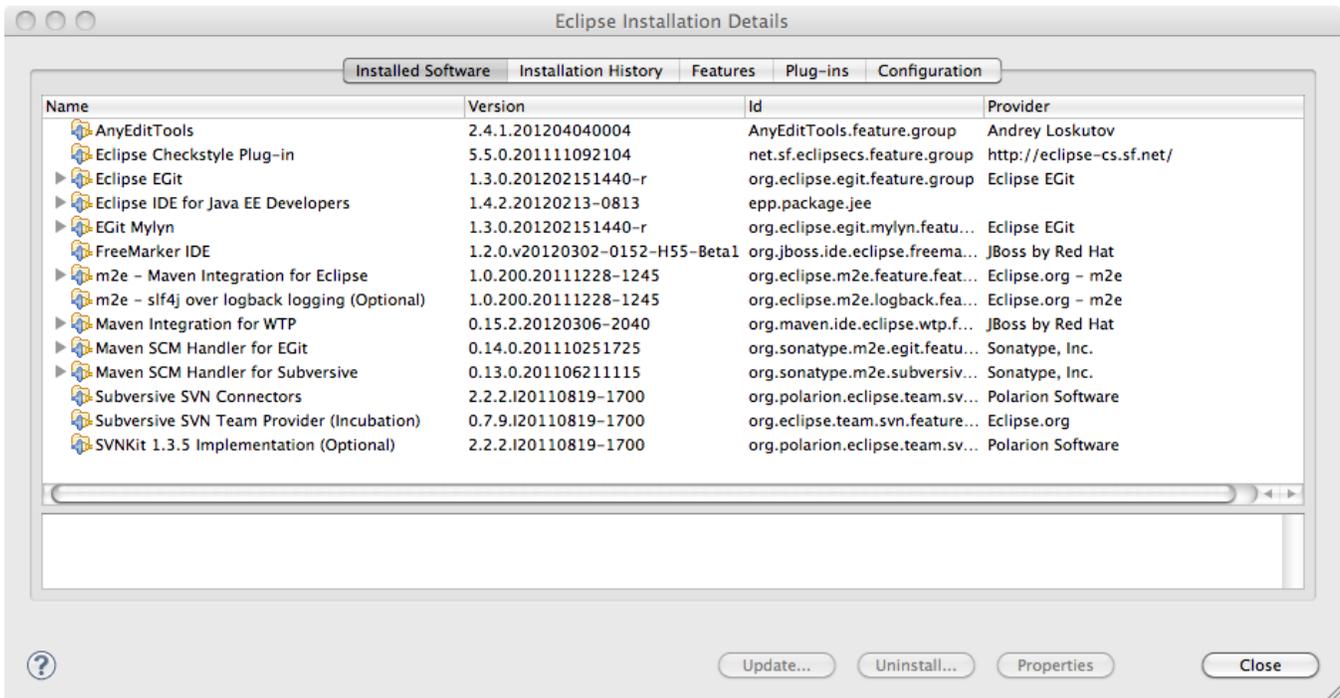
To know more about Git usage you can start by reading our wiki page

There's a bug in Indigo (see also related issue http://jira.magnolia-cms.com/browse/BUILD-72) which causes Eclipse to show an annoying error message on poms. Worry not, however, your IDE will work fine nonetheless. Read more below.

If you can access Magnolia's internal network, a preinstalled eclipse for **Mac OS X(Cocoa 64)** called *eclipse-indigo-macosx-cocoa-64-egit.zip* can be found at our file server under *Product Development/Eclipse*
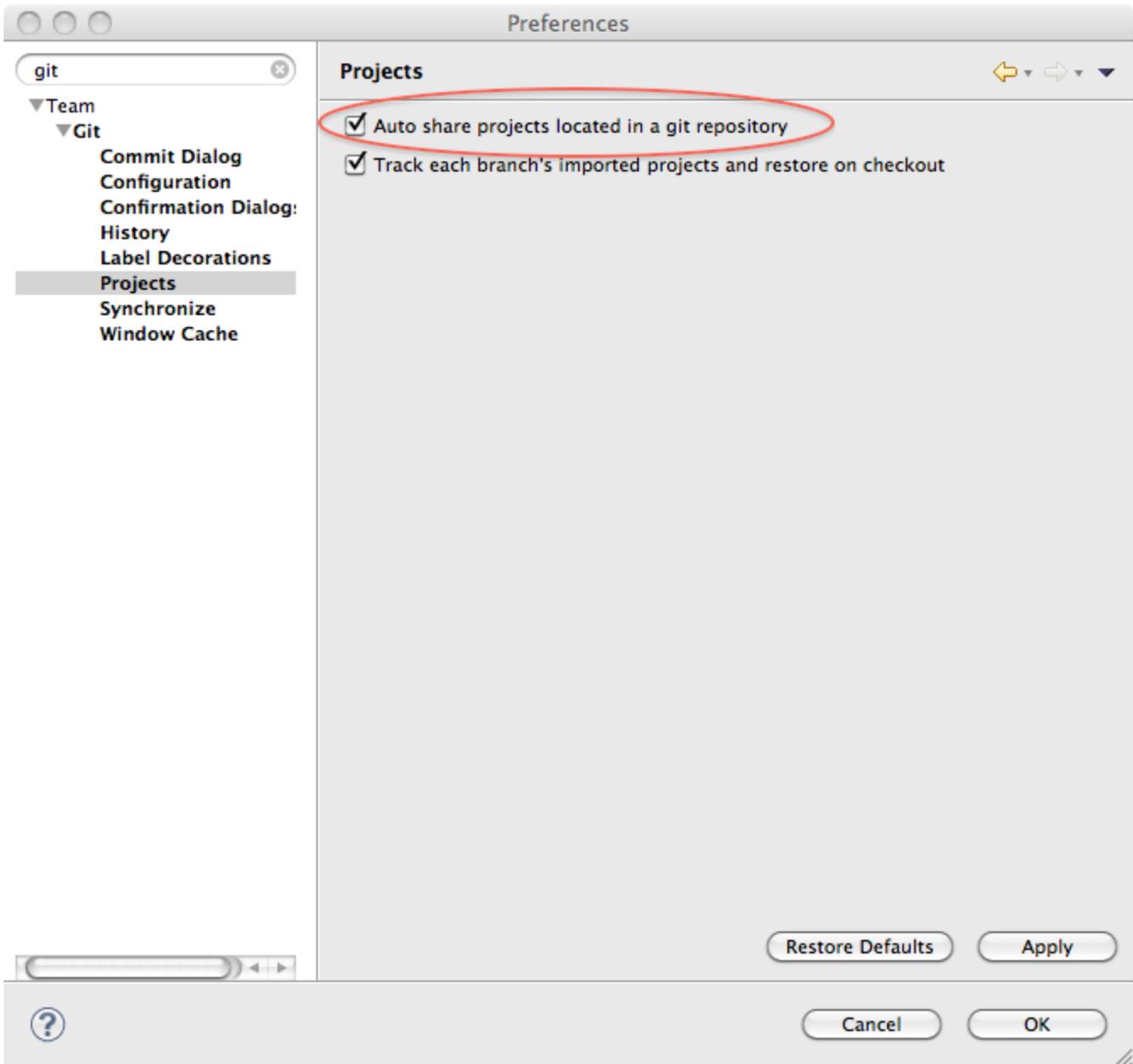
There is also preinstalled eclipse for **Linux-64b**. It can also be found in Product Development/Eclipse folder. This version supports Git DVCS.

The *eclipse-indigo-macosx-cocoa-64-egit.zip* bundle was created on 10 April 2012 and consists of the following platform, plugins and related versions:
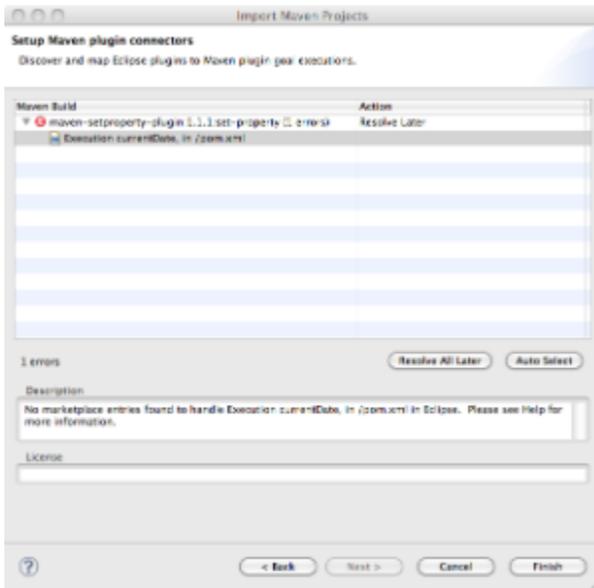
- **Autoshare**

By default cloning a Git repo and then importing it as a Java or Maven project does not show it as *"shared"*, meaning that the Eclipse project looks disconnected from your local Git repo. If you're used to work with SVN you might find this baffling, as you're accostumed to checkout a project from the central scm repository and see it connected to it, so that changes to your local copy are visually decorated by a special label or icon. In order to save you some clicks to achieve the same with Git you need to check the following option under `Eclipse -> Preferences -> Team -> Git -> Projects`

- **m2e 1.0 known issue**

Indigo introduces **m2e (Maven 2 Eclipse) 1.0**, which seems to cause some problems with plugins that are not "*known*" by m2e. This hits Magnolia with the **s etproperty** Maven plugin and other Maven plugins as well (See list below). If you have m2e 1.0 installed you will likely get the following error upon importing a Maven project (i.e. a Magnolia module) into Eclipse:

> No marketplace entries found to handle Execution currentDate, in /pom.xml in Eclipse. Please see Help for more information.

Just choose to **resolve the issue later** and import the project.

- ## Table of plugins which currently do not have a m2e connector

groupId	artifactId	version	goal	execution	phase
org.apache.maven.plugins	maven-javadoc-plugin	2.8	javadoc	default	generate-resources
info.magnolia	maven-setproperty-plugin	1.1.1	set-property	currentDate	generate-resources
org.codehaus.mojo	gwt-maven-plugin	2.4.0	resources	default	process-resources

(Last update April 2012)

- ## Other handy plugins

The following plugins might come in handy. These are not part of the Indigo bundle for now.
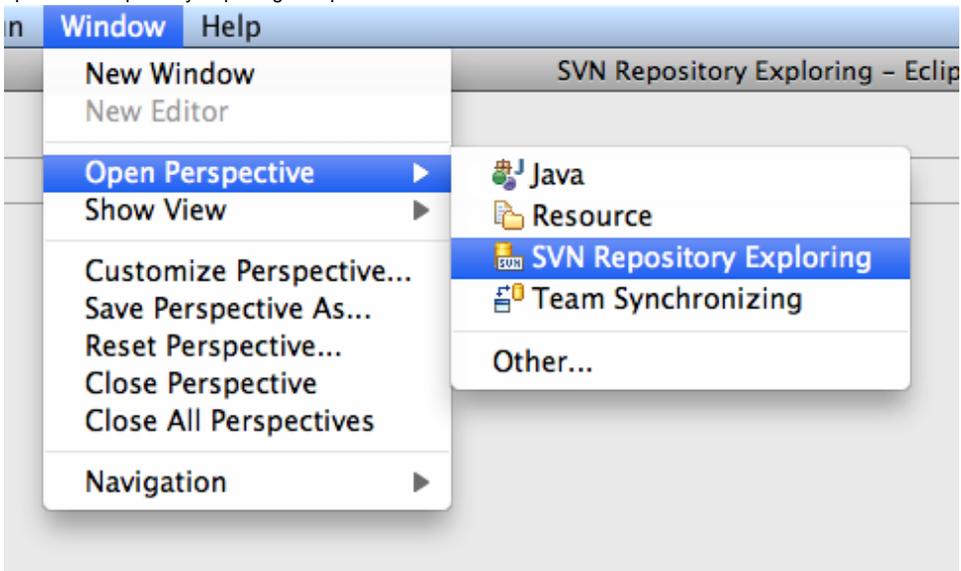
- Eclipse ResourceBundle Editor

Maven Repository

See Maven setup. Be sure to follow these steps first as else you'll not be able to checkout magnolia properly!
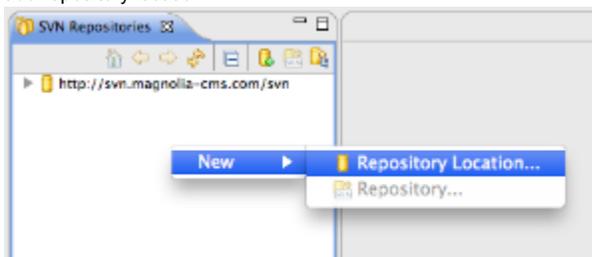
Add SVN Repository

You will have to register our SVN repository in case you want to checkout magnolia modules/projects
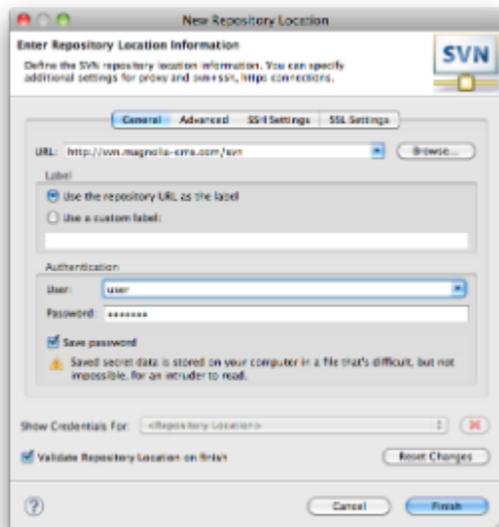
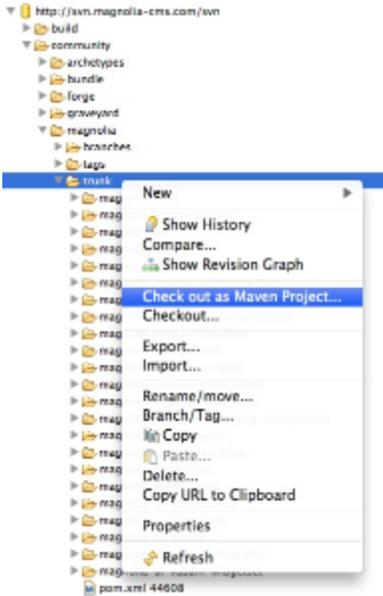1. open SVN Repository Exploring Perspective

2. add repository location

3. http://svn.magnolia-cms.com/svn

Create/Checkout a Project

New Checkout

1. open "SVN Repository" perspective (Window -> Open perspective)
2. select module/project to check out
3. click "Check out as Maven project..." in the context menu

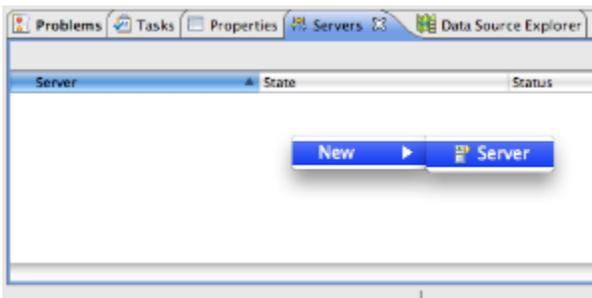## Importing an already checked out project from file system

Sometimes it is more convenient to checkout a project manually and then use the eclipse import function. This is especially helpful if the module to be imported has submodules.

1. File -> Import
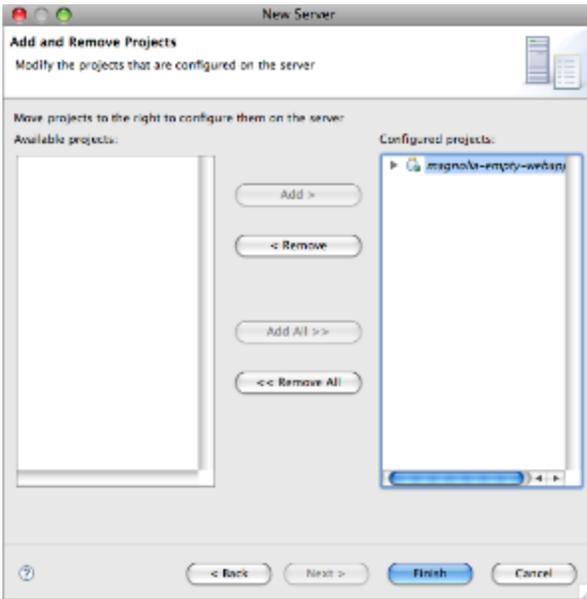2. in the import dialog select 'Maven Projects'

Setup the Server (Tomcat)

Create a Server
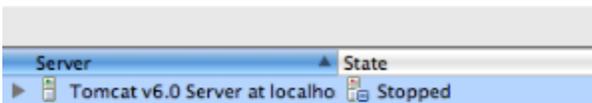
If not yet done create a server.
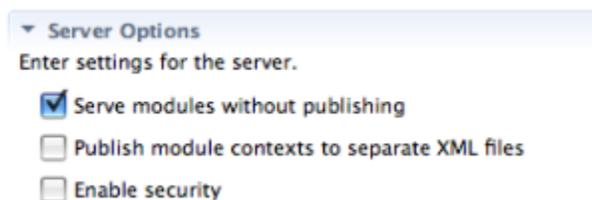

Deploy the Webapp project

Open Server Edit Pane

Doubleclick on the server item

Server settings: Disable publishing

Make sure that the option Serve modules without publishing is selected. This ensures that the classes are taken from the target directories.

Server settings: Increase timeouts

In these same settings, increase both timeout values by a factor of 10.

Freemarker specific settings

Taglib definitions (tld)

FreeMarker does not search the classpath to find taglib definition. It uses only `WEB-INF/lib` and the entries defined in `web.xml` directly. There are two workarounds:

A) Register taglib definitions in `web.xml`

1. Edit `web.xml` and add the taglib definitions as follows. It is important to add them before `jsp-property-group`.

```
<jsp-config>
   <taglib>
     <taglib-uri>cms-taglib</taglib-uri>
     <taglib-location>cms-taglib.tld</taglib-location>
   </taglib>
   <taglib>
     <taglib-uri>cms-util-taglib</taglib-uri>
     <taglib-location>cms-util-taglib.tld</taglib-location>
   </taglib>
   <taglib>
      <taglib-uri>cmsfn-taglib</taglib-uri>
      <taglib-location>cmsfn-taglib.tld</taglib-location>
    </taglib>

    <jsp-property-group>
      <url-pattern>*.jsp</url-pattern>
      <page-encoding>UTF-8</page-encoding>
    </jsp-property-group>
</jsp-config>
```

2. Copy the files to `webapp/WEB-INF`.

B) Put minimal JARs into `WEB-INF/lib`

1. In the `webapp`folder, execute:

```
  mvn war:inplace
```

This generates a complete webapp in your `src/webapp` directory as opposed to `target/webapp-name-1.0-SNAPSHOT`. This is an Eclipse specific workaround.
2. Delete all JARs except the taglib JARs in `src/main/webapp/WEB-INF/lib`
3. Minimize the taglib JAR

```
  zip -d magnolia-taglib-cms-4.1.1-SNAPSHOT.jar /info/*
  zip -d magnolia-taglib-utility-4.1.1-SNAPSHOT.jar /info/*
```

Auto completion

Freemarker templates do not provide automatically auto completion of the provided objects.

The provided objects like ctx, stk or mgnl must be declared to the IDE, that they can be auto completed.

There for a configuration file must be provided in each project in your workspace. The file is named '.freemarker-ide.xml'. I have attached it to this page without the dot at the beginning, or I can't upload it on the Mac otherwise (hidden).

The important part of this configuration file is the declaration of the objects the auto completion should know. I think these four declarations I use make sense:

```
<value key="mgnl" object-class="info.magnolia.module.templating.MagnoliaTemplatingUtilities"/>
<value key="ctx" object-class="info.magnolia.context.MgnlContext"/>
<value key="stk" object-class="info.magnolia.module.templatingkit.util.STKUtil"/>
<value key="state" object-class="info.magnolia.cms.core.AggregationState"/>
```

Configure Code Style

If you are an internal developer or a contributor please follow the instructions found here: Coding Conventions.

Tips and Tricks

Get the sources or a linked jar / module

- select the jar (in the Maven Dependencies library)
- right click -> Maven -> Download Sources

Dependencies

The dependencies are resolved inside the workspace. This is quite helpful. But this will only work if the version you reference in the project and the version of the module you checked out has the same version. In some cases you will have to adapt the pom files locally (to point to the current snapshot for instance).

Use company wide Java Code Style & Compiler preferences

Simply import the preferences file into your Eclipse.

Known Problems

Wrong maven version is used for dependency checks

If the project can't build because of the following message (printed in Maven Console)

```
[WARN] Rule 0: org.apache.maven.plugin.enforcer.RequireMavenVersion failed with message: Detected Maven
Version: 2.1-SNAPSHOT is not in the allowed range [2.0.9,2.0.9].
```

The expected stacktrace on startup is:

```
SEVERE: Exception sending context initialized event to listener instance of class info.magnolia.cms.servlets.
MgnlServletContextListener
java.lang.NullPointerException
        at info.magnolia.cms.beans.config.PropertiesInitializer.loadAllModuleProperties(PropertiesInitializer.
java:92)
        at info.magnolia.cms.beans.config.PropertiesInitializer.loadAllProperties(PropertiesInitializer.java:79)
        at info.magnolia.cms.servlets.MgnlServletContextListener.contextInitialized(MgnlServletContextListener.
java:176)
        at org.apache.catalina.core.StandardContext.listenerStart(StandardContext.java:3843)
        at org.apache.catalina.core.StandardContext.start(StandardContext.java:4350)
```

You can either add the following to the pluginManagement section in your main project pom to disable the enforcer plugin or select the **eclipse profile**:

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-enforcer-plugin</artifactId>
  <configuration>
    <skip>true</skip>
  </configuration>
</plugin>
```

Then clean all projects

To ensure that everything is fine you have to check that magnolia-core/target/classes/mgnl-bean.properties exists.

Freemarker can't find tags (tlds)

See #Freemarkerspecificsettings

Project contains duplicate entry

Sometimes the same jar is references twice (by maven and EAR dependencies). This sometimes happens if you open eclipse freshly. To make the project compile again you must remove the ear dependency:

- right click project -> Java EE -> Update EAR libraries

Class Cast Exception: info.magnolia.cms.filters.MgnlMainFilter

Symptom:

```
SEVERE: Exception starting filter magnoliaFilterChain
java.lang.ClassCastException: info.magnolia.cms.filters.MgnlMainFilter
        at org.apache.catalina.core.ApplicationFilterConfig.getFilter(ApplicationFilterConfig.java:255)
        at org.apache.catalina.core.ApplicationFilterConfig.setFilterDef(ApplicationFilterConfig.java:397)
        at org.apache.catalina.core.ApplicationFilterConfig.<init>(ApplicationFilterConfig.java:108)
        at org.apache.catalina.core.StandardContext.filterStart(StandardContext.java:3709)
        at org.apache.catalina.core.StandardContext.start(StandardContext.java:4356)
        at org.apache.catalina.core.ContainerBase.start(ContainerBase.java:1045)
```

This typically depends on the servlet-api **AND/OR** the gwt-user dependencies in your pom.xml(s) and Tomcat (i.e. never seen this problem with jetty). The solutions provided below are not mutually exclusive and you might need to apply a combination or all of them.

Solution 1: exclude the servlet api jar. Verify that you don't have the servlet api jar in your maven dependencies.

Example:

```
<dependency>
        <groupId>info.magnolia</groupId>
        <artifactId>magnolia-empty-webapp</artifactId>
        <version>4.0.1</version>
        <type>pom</type>
        <exclusions>
                <exclusion>
                        <groupId>javax.servlet</groupId>
                        <artifactId>servlet-api</artifactId>
                </exclusion>
        </exclusions>
</dependency>
```

Solution 2: make the servlet-api dependency scope `provided`.

Example:

```
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>servlet-api</artifactId>
    <version>2.5</version>
    <scope>provided</scope>
</dependency>
```

Solution 3: make the gwt-user dependency scope `provided`.

Example:

```
<dependency>
    <groupId>com.google.gwt</groupId>
    <artifactId>gwt-user</artifactId>
    <version>2.4.0</version>
    <scope>provided</scope>
</dependency>
```

Webapp project doesn't work

1. make sure that you have only one webapp project open
2. add the pom dependency (not only the war dependency) otherwise the dependencies defined in the referenced war file won't be seen
3. use mvn war:inplace to overlay the webapp project (make sure that the WEB-INF/web.xml exists

4. ensure that you delete WEB-INF/lib after a war:inplace
5. check settings `project/project facets/dynamic web module` ==> 2.4 works (helios, wtp 2.4), 3.0 doesn't (helios, wtp 3.0)

NPE when configuring tomcat with "don't publish" option

1. delete Servers, delete .project and .settings from the webapp and reimport the webapp

webapp is not recognized as webapp and doesn't show in tomcat when trying to add it

1. go to `project/project facets` and make sure `dynamic web module` and `java` and `javascript` facets are checked.
2. rebuild the project

modules are not installed when installing the webapp

1. cause: modules are not visible if the faceted build version of java doesn't match the current version or that used by the webapp project
2. solution: update project config and change java version

Module Dependency Exception

Symptom:

```
ERROR info.magnolia.init.MagnoliaServletContextListener : Oops, Magnolia could not be started
info.magnolia.module.model.reader.ModuleDependencyException: Module Magnolia Mail Module (version 4.5.0-
SNAPSHOT) is dependent on myCoolModule version 4.5/*, which was not found.
    at info.magnolia.module.model.reader.DependencyCheckerImpl.checkSpecificDependency(DependencyCheckerImpl.
java:79)
```

Solution: ensure that no resource is excluded under myCoolModule -> Build path -> Configure Build Path... This might happen when i.e. updating the Maven dependencies if the m2Eclipse plugin is installed.

Trouble shooting step by step

1. check if there are any build issues denoted in the problem view
2. check if there are any problems denoted in the pom.xml (missing parents, ..)
3. check if the project builds in the terminal (by using mvn only)
   - are the parent poms available?
4. check the maven log in eclipse (select the maven console in the console view)
   - are the logs fine?
   - verify that there is no version conflict thrown by the enforcer plugin
     - if so configure the project to use the **eclipse** profile in the project settings (maven tab)
5. clean the project
   - this ensures that maven is used to build the resources
   - verify that the resources have been built (module descriptor in `target/classes/META-INF/magnolia`
6. update project configuration
   - right click project -> maven -> update project configuration
   - check the maven console (some errors are not reported to the GUI)
   - one last step can be added to the above Eclipse voodoo dance: ensure that no resource is excluded under

Install Sun JDK to Ubuntu

By default Ubuntu packagers are not supporting sun jdk. You have to add new software source:

```
deb http://ppa.launchpad.net/ferramroberto/java/ubuntu oneiric main
```