JForum2 Developer Quick-Start Guide v2.0

*Import Source Code from SVN into Eclipse and Build with Maven*

Andowson Chang, 2017/9/16

Environment:

OS:Windows 10 64-bit

JDK: Oracle Java SE JDK 8.0 Update 144 x64

IDE: Eclipse IDE for Java EE Developers 64-bit(eclipse-jee-oxygen-R-win32-x86\_64.zip)

Install Subclipse

https://dl.bintray.com/subclipse/releases/subclipse/latest/













Modify Installed JREs

Windows > Preferences > Installed JREs



Add…



Next



Directory…



Choose the jdk directory



Windows > Preferences > Installed JREs











Import source code from SVN into Eclipse

File > Import …









URL: svn://svn.code.sf.net/p/jforum2/code

Ignore the error message, just click on the OK button.



Expand trunk





Select trunk and click Next





Checking out source code from SVN



Switch to the Progress view



Now, we have downloaded the source code successfully.



And we have some Maven Dependency Problem



Failed to read artifact descriptor for com.jhlabs:imaging:jar:01012005

org.eclipse.aether.resolution.ArtifactDescriptorException: Failed to read artifact descriptor for com.jhlabs:imaging:jar:01012005

at org.apache.maven.repository.internal.DefaultArtifactDescriptorReader.loadPom(DefaultArtifactDescriptorReader.java:282)

at org.apache.maven.repository.internal.DefaultArtifactDescriptorReader.readArtifactDescriptor(DefaultArtifactDescriptorReader.java:198)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.resolveCachedArtifactDescriptor(DefaultDependencyCollector.java:535)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.getArtifactDescriptorResult(DefaultDependencyCollector.java:519)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.processDependency(DefaultDependencyCollector.java:409)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.processDependency(DefaultDependencyCollector.java:363)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.process(DefaultDependencyCollector.java:351)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.doRecurse(DefaultDependencyCollector.java:504)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.processDependency(DefaultDependencyCollector.java:458)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.processDependency(DefaultDependencyCollector.java:363)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.process(DefaultDependencyCollector.java:351)

at org.eclipse.aether.internal.impl.DefaultDependencyCollector.collectDependencies(DefaultDependencyCollector.java:254)

at org.eclipse.aether.internal.impl.DefaultRepositorySystem.collectDependencies(DefaultRepositorySystem.java:316)

at org.apache.maven.project.DefaultProjectDependenciesResolver.resolve(DefaultProjectDependenciesResolver.java:172)

at org.apache.maven.project.DefaultProjectBuilder.resolveDependencies(DefaultProjectBuilder.java:215)

at org.apache.maven.project.DefaultProjectBuilder.build(DefaultProjectBuilder.java:188)

at org.apache.maven.project.DefaultProjectBuilder.build(DefaultProjectBuilder.java:119)

at org.eclipse.m2e.core.internal.embedder.MavenImpl.readMavenProject(MavenImpl.java:636)

at org.eclipse.m2e.core.internal.project.registry.DefaultMavenDependencyResolver.resolveProjectDependencies(DefaultMavenDependencyResolver.java:63)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryManager.refreshPhase2(ProjectRegistryManager.java:530)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryManager$3.call(ProjectRegistryManager.java:492)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryManager$3.call(ProjectRegistryManager.java:1)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.executeBare(MavenExecutionContext.java:177)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.execute(MavenExecutionContext.java:151)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryManager.refresh(ProjectRegistryManager.java:496)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryManager.refresh(ProjectRegistryManager.java:351)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryRefreshJob$1$1.call(ProjectRegistryRefreshJob.java:89)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryRefreshJob$1$1.call(ProjectRegistryRefreshJob.java:1)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.executeBare(MavenExecutionContext.java:177)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.execute(MavenExecutionContext.java:151)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.execute(MavenExecutionContext.java:99)

at org.eclipse.m2e.core.internal.embedder.MavenImpl.execute(MavenImpl.java:1351)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryRefreshJob$1.call(ProjectRegistryRefreshJob.java:87)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryRefreshJob$1.call(ProjectRegistryRefreshJob.java:1)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.executeBare(MavenExecutionContext.java:177)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.execute(MavenExecutionContext.java:151)

at org.eclipse.m2e.core.internal.embedder.MavenExecutionContext.execute(MavenExecutionContext.java:99)

at org.eclipse.m2e.core.internal.embedder.MavenImpl.execute(MavenImpl.java:1355)

at org.eclipse.m2e.core.internal.project.registry.ProjectRegistryRefreshJob.run(ProjectRegistryRefreshJob.java:81)

at org.eclipse.core.internal.jobs.Worker.run(Worker.java:56)

Caused by: org.eclipse.aether.resolution.ArtifactResolutionException: Failure to transfer com.jhlabs:imaging:pom:01012005 from http://forge.octo.com/archiva/repository/3rd-party was cached in the local repository, resolution will not be reattempted until the update interval of 3rd-party has elapsed or updates are forced. Original error: Could not transfer artifact com.jhlabs:imaging:pom:01012005 from/to 3rd-party (http://forge.octo.com/archiva/repository/3rd-party): forge.octo.com

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.resolve(DefaultArtifactResolver.java:444)

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.resolveArtifacts(DefaultArtifactResolver.java:246)

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.resolveArtifact(DefaultArtifactResolver.java:223)

at org.apache.maven.repository.internal.DefaultArtifactDescriptorReader.loadPom(DefaultArtifactDescriptorReader.java:267)

... 39 more

Caused by: org.eclipse.aether.transfer.ArtifactTransferException: Failure to transfer com.jhlabs:imaging:pom:01012005 from http://forge.octo.com/archiva/repository/3rd-party was cached in the local repository, resolution will not be reattempted until the update interval of 3rd-party has elapsed or updates are forced. Original error: Could not transfer artifact com.jhlabs:imaging:pom:01012005 from/to 3rd-party (http://forge.octo.com/archiva/repository/3rd-party): forge.octo.com

at org.eclipse.aether.internal.impl.DefaultUpdateCheckManager.newException(DefaultUpdateCheckManager.java:238)

at org.eclipse.aether.internal.impl.DefaultUpdateCheckManager.checkArtifact(DefaultUpdateCheckManager.java:206)

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.gatherDownloads(DefaultArtifactResolver.java:585)

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.performDownloads(DefaultArtifactResolver.java:503)

at org.eclipse.aether.internal.impl.DefaultArtifactResolver.resolve(DefaultArtifactResolver.java:421)

... 42 more

Change to the user’s .m2 repository and navigate to the com\jhlabs directory

Path: C:\Users\%username%\.m2\repository\com\jhlabs



Download the imaging.zip from jforum.andowson.com and extract these files into the directory





Right click on project name and run Maven > Update Project





Now, the Maven dependency problem is solved.













Maven Build





Goals: package

Check Skip Tests for now. We just want to verify that all the codes can be built in Eclipse.

Apply->Run







Maven clean





OK, now you can start to develop new programs.