

XSTAMPP Setup Guide

Lukas Balzer

May 9, 2017

Contents

1	How to build XSTAMPP	2
2	Run XSTAMPP in Eclipse	2
2.1	Create a new plugin:	2
3	Known Issues	3

1 How to build XSTAMPP

1. The build requires Maven (≥ 3.3) and java (≥ 1.7) either from <https://maven.apache.org/> or from the IDE
2. go to the xstamppp.parent directory in the root path of the xstamppp project (where this file is located)
3. open a command in the xstamppp.parent dir and execute
 - (a) 'mvn clean verify' to build xstamppp with xstpa and cast already included
 - (b) 'mvn clean install' as 3.1 but also installs xstamppp on `[user]/.m2` for usage as local dependency of other builds
4. the build artifacts are located in the `astpa.repository/target`

2 Run XSTAMPP in Eclipse

you need:

- Eclipse for RCP and RAP Developers (Plug-in Development)¹ ($> Lunar$)
- at least JavaSE 1.7
- to install gef (*help* \rightarrow *installnewsoftware* \rightarrow <http://download.eclipse.org/tools/gef/updates/releases/>)
- to install nebula grid from eclipse.org²
- to install maven³
- import/clone xstamppp projects using the included git
 1. Open the *Import* Dialog selecting *File* \rightarrow *Import*
 2. in the Import menu click *Git* \rightarrow *Projects from Git* and follow the steps of the import wizard
- to resolve upcoming error messages refer to Known Issues Section 3

-

2.1 Create a new plugin:

- create a new plugin by clicking *New* \rightarrow *Others..* \rightarrow *Plug-in Development* \rightarrow *Plug-in Project*
- add dependencies xstamppp and xstamppp.extension
- add the extension `xstamppp.extension.steppedProcess` to your plugin
- create a class implementing `IDataModel`
- create `stepEditors` which must extend `StandartEditorPart` and implement `IViewBase`
- xstamppp loads the files which are selected in the load Dialog or already located in the workspace by directly calling a load command registered as command in the `steppedProcess` extensionPoint herefore it needs:
 - a load job which extends `AbstractLoadJob`
 - a load Handler extending `AbstractHandler` which is registered as default handler for the load command
 - let your handler.execute() return a new instance of your load job

¹<http://eclipse.org/downloads>

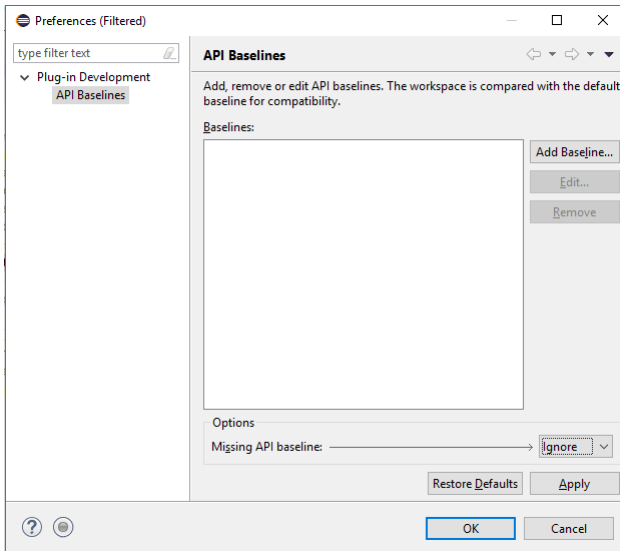
²<http://download.eclipse.org/technology/nebula/snapshot/>

³<https://maven.apache.org/download.cgi>

3 Known Issues

An API baseline has not been set for this Workspace

1. Go to the Eclipse Problems View (*Window → ShowView → Problems*)
2. Right click the 'API baseline' error
3. In the context menu select *QuickFix*
4. A Preference Window filtered for the API Baselines opens up
5. in that Dialog find the field *MissingAPIBaseline* and set it to *Ignore*



Plugin Execution not covered by lifecycle configuration

1. Go to *Window → Preferences → Maven → Error/Warnings*
2. find the line 'Plugin Execution not covered..'
3. Set the Value to ignore, by choosing selecting 'ignore' in the combo box
4. Click on Apply/Ok to rebuild the projects

When cloning into/ importing xstampp and its sub projects to eclipse the project dependencies must be located sometimes

1. In the Project Explorer right click on the project 'Build Path-¿Configure Build Path'
2. In the 'Java Build Path' Page click on 'Source', by doing that java relocates the source folders in the projects and sets the dependencies
3. hit Apply/Ok to store the settings