Activiti Designer Developer Guide

This page describes how to get started if you want to build Activiti Designer yourself or if you want to extend Activiti Designer's features. It assumes you have a working knowledge of Eclipse, Maven and RCP development.

- Start by downloading an Indigo or Juno version of Eclipse for RCP and RAP Developers from here: http://www.eclipse.org/downloads/
- Download and install Maven 3 from here: http://maven.apache.org/download.html . Using Maven 2.x will not work, so make sure you have a compliant version. Commands listed further on will assume that you have the mvn command linked to Maven 3. Make sure your Maven settings are such, that Maven can access the Internet to reach public repositories and download dependencies
- Add the Eclipse Modelling Framework (EMF) SDK
- Add Eclipse Graphiti SDK 0.9.1 from update site http://download.eclipse.org/graphiti/updates/0.9.1.
- Add a team provider for Subversion, such as Subversive
- Add the following location as source code repository: https://github.com/Activiti/Activiti-Designer

As you can see, there are quite a number of projects in this tree. You will need to have all of them in your Eclipse workspace to build Designer yourself. It's important that you perform a checkout of all of these projects as **top level** projects in your workspace. Obviously, you can do without the examples folder, but you need the other projects.

- Checkout all of the folders in the source control location as top level Eclipse projects.
- Open a console and navigate to the directory of your Eclipse workspace and then into the org.activiti.designer.parent directory, which is of course the directory of the project with the same name. Now perform the following Maven commands:

```
mvn clean eclipse:clean
mvn eclipse:eclipse
```

This might take a while because Maven will now download a lot of dependencies to your local repository and setup the projects. Make sure the command ends with a Build Successful so you know everything's OK as far as Maven is concerned.

- Go back to Eclipse, select all of the projects, right click and choose Refresh. The information set by Maven will now be picked up by Eclipse. If you like being thorough, you can perform a Project > Clean command as well.
- Now you have a environment for working with Activiti Designer. To run Designer, right click the org.activiti.designer.eclipse project and choose Run as > Eclipse application.

The projects

This section briefly describes the different projects used in Activiti Designer.

- org.activiti.designer.eclipse: contains the main extensions of Eclipse extension points and a lot of shared code for working with the model, saving resources and utilities.
- org.activiti.designer.gui: contains most of the UI code, mainly functionality that utilizes Graphiti for drawing diagrams.
- org.activiti.designer.help: help files for end users.
- org.activiti.designer.export.bpmn20: provides an extension of the ExportMarshaller extension point to generate BPMN 2.0 xml files.
- org.activiti.designer.validation.bpmn20: provides an extension of the ProcessValidator extension point to validate diagrams according to BPMN 2.0 rules.
- org.activiti.designer.integration: provides APIs to developers creating CustomServiceTask extensions.

- org.activiti.designer.parent: Maven parent project, used mainly for building the project.
- org.activiti.designer.feature: contains an Eclipse feature definition, which groups the various projects into the installable Activiti Designer feature.
- org.activiti.designer.updatesite: contains an Eclipse update site definition, which allows you to build a website that can be used as an Eclipse update site.