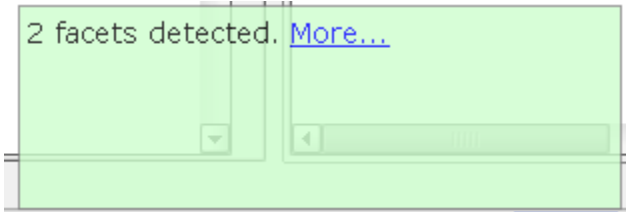


Getting Started

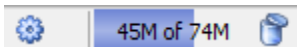
Module Settings

Auto detection

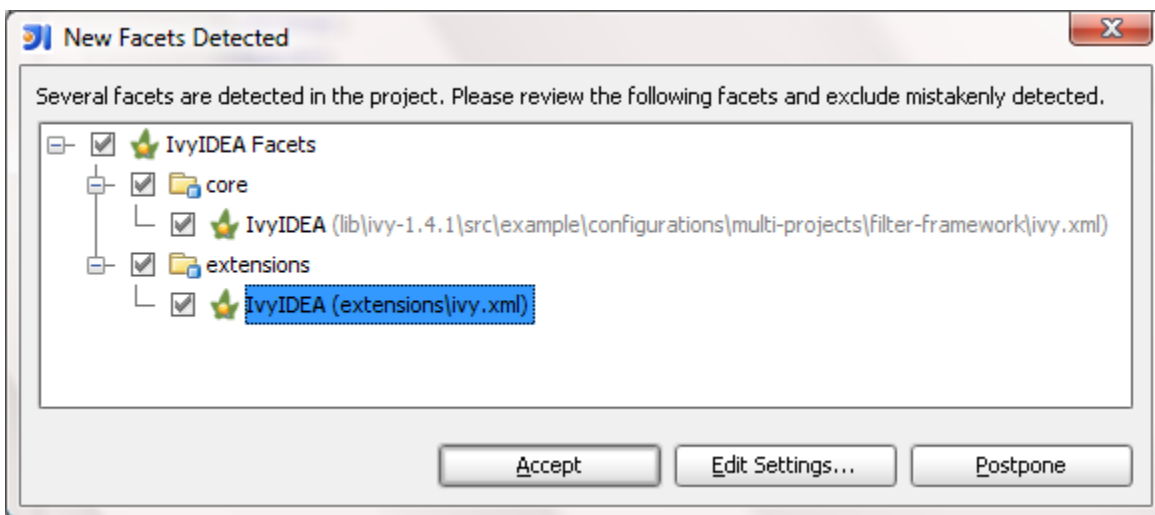
With a bit of luck, your ivy-enabled modules will be detected automatically by IvyIDEA. You'll see a popup stating that one or more new facets are detected on opening your project.



Note: it is possible that you missed the popup as it is only displayed briefly. In this case you can still see if a new facet was detected by the presence of a blinking gear icon in the statusbar:



If you click on the 'More...' link (or on the blinking gear icon), you'll get a screen that looks like this:

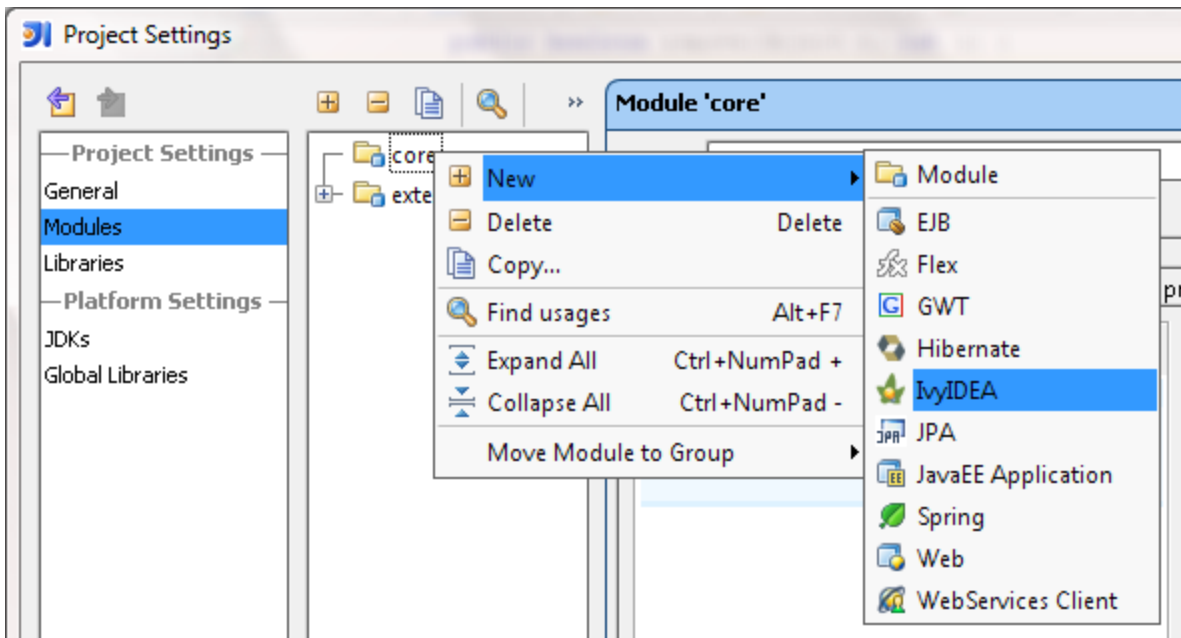


Click 'Edit Settings...' to make sure that the correct ivy.xml file is chosen for each module. This will take you to the Module Settings screen, which is described in greater detail below. Note: there is a known issue that the ivy file displayed here may differ from the one that is configured in the module settings, so make sure to check this if your module contains multiple ivy.xml files!

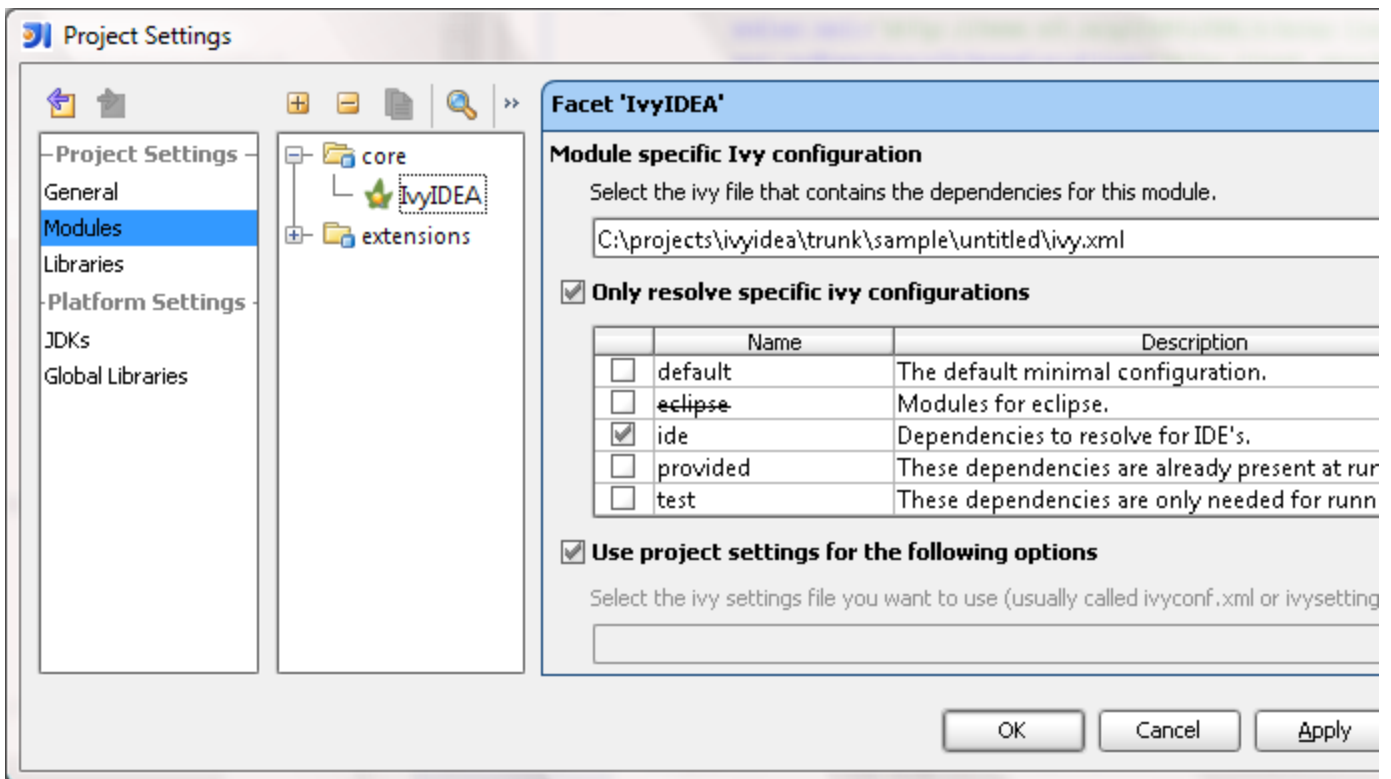
Manual setup

If the autodetection did not work for some reason, or you want to change the settings that were automatically chosen, you can manually setup an IvyIDEA facet for each module that needs it. This can be done by opening the project settings and clicking on the 'Modules' item.

You can add a facet to a module simply by clicking the [+] icon above the module list, or by right clicking on the module name as shown here:



Once the facet is added to the tree, you can change its configuration by clicking on it, you will get the following screen:

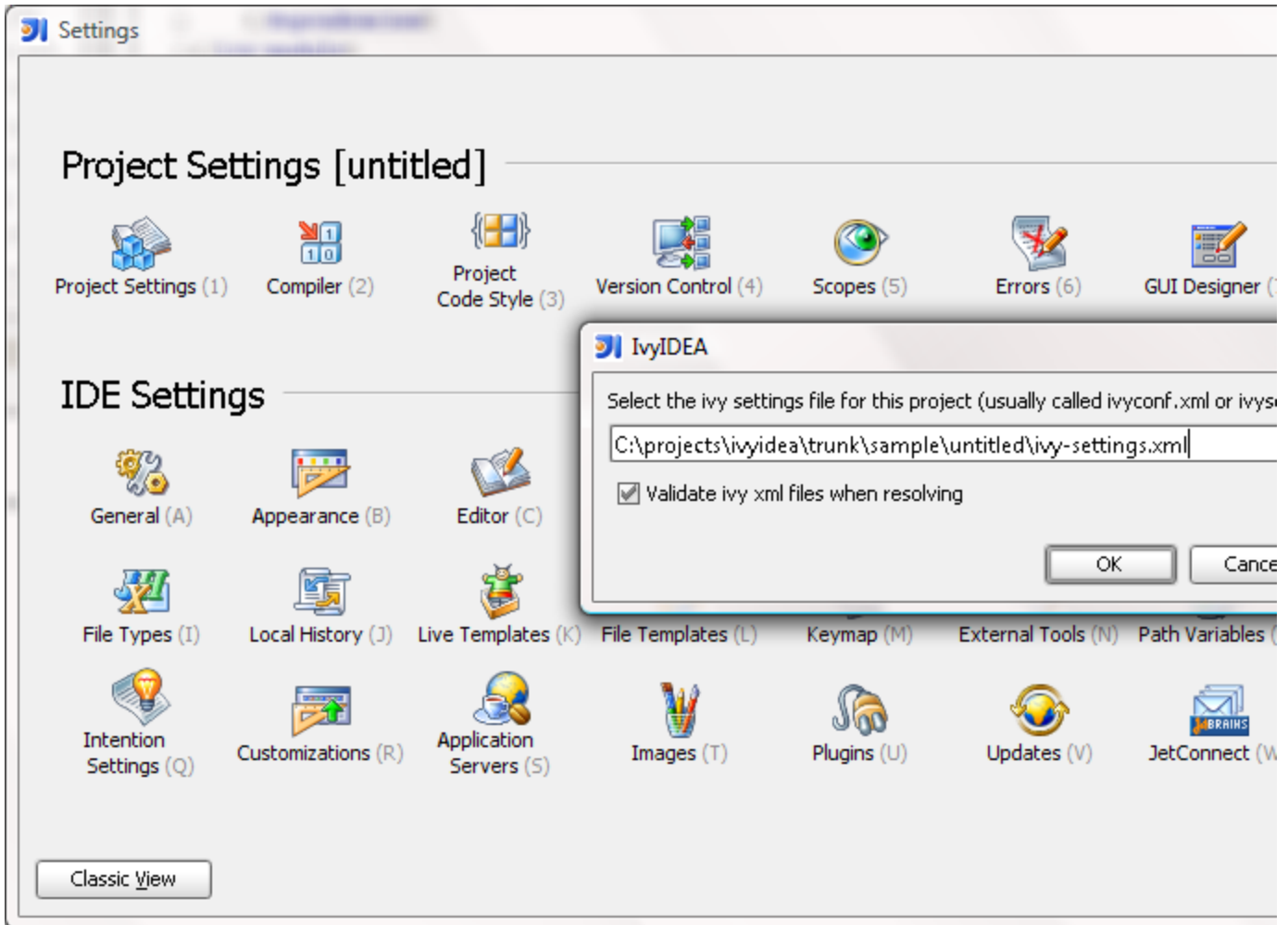


Here you need to specify the ivy file for your module. This is the file which holds the dependency information for your module. More information about ivy files can be found in the apache ivy quick start guide

You can also select the configuration(s) that you want to resolve in IntelliJ; the configurations table will be filled automatically when you have selected a valid ivy.xml file. Note that if you use an include file for your configurations in your ivy.xml and you use a `{}` style property inside the filename, the ivy settings file needs to be set as well (here or in the project settings) in order to correctly lookup the available configurations. The current version of the plugin will not report this, it will simply not show any data in the table.

Project Settings

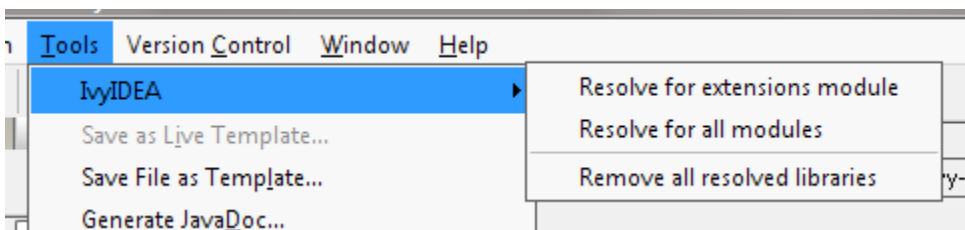
If you do not specify an ivy settings file with your module (meaning that you left the 'Use from project settings' checkbox checked), you obviously need to specify one at the project level. The following screenshot shows you how to do this:



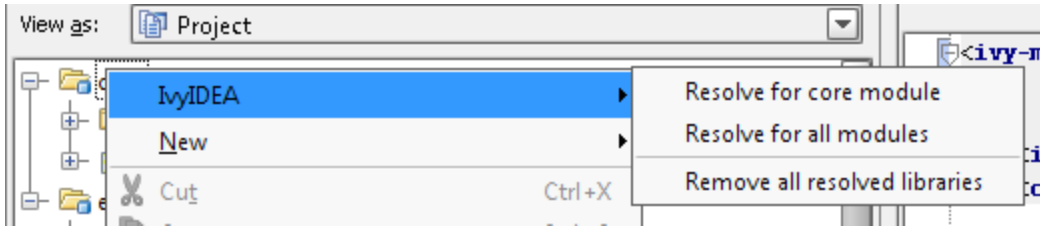
You can also choose to enable or disable the validation of ivy.xml files; this can be useful if you have minor errors in ivy.xml files that you do not control but which do not cause the resolve process to fail. Disabling this will also probably be a bit faster.

Resolving Dependencies

This can be done through the Tools menu



but also through the right-click menu in the project view or in the editor (note that the IvyIDEA menu option is not necessarily at the top of the menu)



It is possible to do a resolve for the current module (if you are inside the context of a module) or for all modules.

There is also an option to remove all IntelliJ module libraries that were a result of the ivy resolve process. This does not remove any files from the filesystem, it will simply remove the IvyIDEA-resolved module library from all modules that have an IvyIDEA facet configured. It will also not remove module to module dependencies that were made by IvyIDEA!