Mango development platform installation instructions

The Mango development platform allows you to extend Mango functionality and/or fix bugs. If you are new to Mango, you may wish skip this section for now and go directly to the "Hello World" example for Mango.

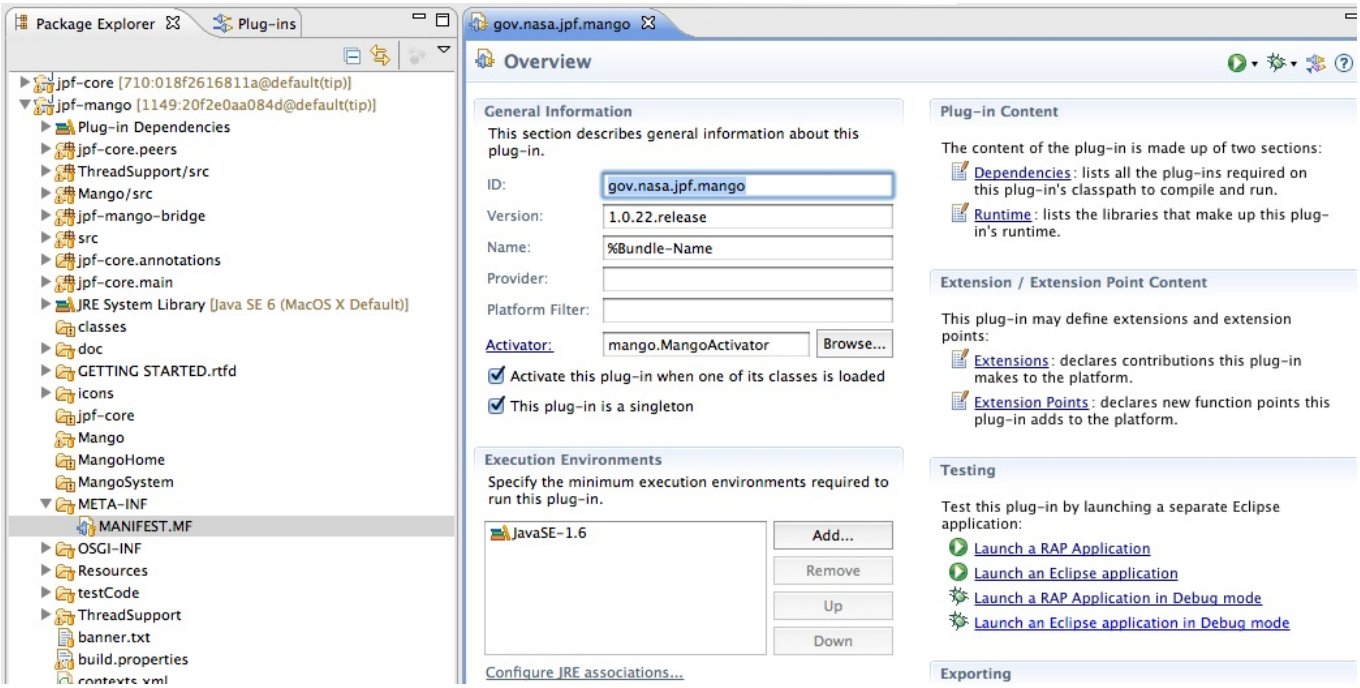
1. Install the most recent version of Mercurial.
2. Install the Mercurial Eclipse plugin into the Eclipse workbench. The site for this plugin is http://cbes.javaforge.com/update
3. Install the GEF/Zest plugin for 2D graphics. This plugin site is http://download.eclipse.org/tools/gef/updates/releases

4. Add the **jpf-core** and **jpf-mango** projects.

   1. Go to File>New>"Project..."
   2. Open the Mercurial folder. Select "Clone Existing Mercurial Repository". Hit "Next".
   3. Enter the URL: http://babelfish.arc.nasa.gov/hg/jpf/jpf-core Hit "Next".
   4. The default directory revision will appear in the window. Hit "Next".
   5. The "Import Projects" window will appear. Hit "Finish".
   6. The jpf-core project now appears in the Package Explorer. Observe the project build messages in the console. If there is an indication that javac cannot be found, then the build has failed. The other possible error messages are normally harmless. If javac cannot be found, it is probably an installation problem with java or a failure to set the proper environment variables. Googling for instructions particular to your operating system should yield a solution.
   7. Repeat steps a-e for jpf-mango, using the URL http://babelfish.arc.nasa.gov/hg/jpf/jpf-mango. Sometimes **jpf-core** and **jpf-mango** get out of sync, and sometimes the most current version of **jpf-mango** is unstable. In this case, the Mercurial commit notes for **jpf-mango** should be consulted.

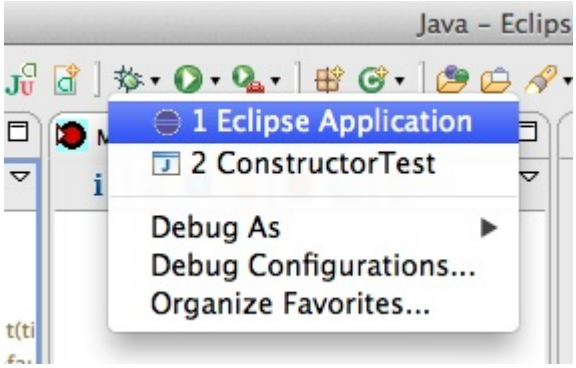
- 5. You must now set up the runtime for the Mango development platform. Click on "Launch an Eclipse application in Debug mode".

  1. In the **jpf-mango** project, locate "META-INF/MANIFEST.MF" and open this file in an editor (see figure below).



- 5. continued:

  1. The Mango development platform will open. This is just an Eclipse application which recognizes the **jpf-mango** project as one of its plugins. You can set break-points in the **jpf-mango** project, and the Eclipse debugger in the base workbench will honor them as the development platform runs. This is the manner in which Mango is developed.
  2. You may need to switch to the Java perspective and adjust the Mango views as in step 5 of the Mango plugin installation.
  3. Close the Mango development platform. Henceforth, use the "Debug as..." icon to launch the development platform (see figure below).



- 5. conclusion:
  1. In the directory containing your workspace, Eclipse has created a runtime folder for the Mango development platform. Locate this folder, it should have a name such as "runtime-EclipseApplication". This folder should appear empty. In the next step you will provide content for this folder.
- 6. Add the MangoHome and MangoSystem projects to the Mango development platform. If you have already installed the plugin and built a specification, then versions of these projects already exist in a base workspace. However, these versions may not be in sync with the current jpf-mango project. Therefore, it is always best to get fresh versions from this site.
  1. At the bottom of the main page, expand "Attachments". Double-click on "MangoNucleus.zip" and then on "downloading".
  2. Unzip the file. Ignore the MacOSX artifact if it appears, and locate the MangoSystem and MangoHome folders.
  3. Move MangoSystem and MangoHome to the Mango development platform runtime, located in step 4e.
  4. Re-open the Mango development platform, see step 4d.
  5. Select File>New>Project...>"Java Project". Hit "Next".
  6. Make sure "Use default location" is checked. This is the runtime folder.
  7. Enter "MangoHome" for the project name. The message "The wizard will automatically configure ..." should appear because MangoHome is in fact already in the runtime folder. Hit "Finish".
  8. Repeat steps e-g for "MangoSystem."

Thats all there is too it! Keep this in mind: " **jpf-mango** is to Mango development platform as Mango plugin is to base workbench. That is, whatever you can do in the base workbench with the Mango plugin installed, you can do in the Mango development platform while running/debugging **jpf-mango**. Because the base workbench is observing the development platform, you can also get much better feedback for diagnosing problems.