

Public

🔒 Log In



- Index
- Q Search
- F Changes
- Notifications
- Statistics
- Preferences

Webs

Public

TWiki

TWiki > Public Web > DpHipe > HipeDevGettingStarted (2012-10-19, DavideRizzo)

Selit <u>Attach</u>

Tags:

<u>create new tag view all tags</u>



Did you spot something wrong or missing on this page? If you have an account on this TWiki, you can fix it yourself by editing the page. If you don't have an account, you can leave a message at the bottom of the page to tell us about it. Thank you in advance!

Getting started with HIPE development

This tutorial will teach you the following:

- 1. How to obtain the source code of HIPE.
- How to use the Eclipse development environment to modify a component of HIPE.
- 3. How to put back the modified component into HIPE to see the result of your changes.

This tutorial has been tested with Eclipse 3.7 "Indigo". Steps may differ for other versions of Eclipse.

Getting HIPE source code

You can get HIPE source code in two ways:

- If you are installing a **user release**, make sure that the option *Would you like* to have the source code installed? is checked (see figure below).
- If you are downloading a developer build from the Continuous Integration System, run the installer with the option --src=yes Or --developer (--developer implies --src=yes).

☑ Would you like to have the source code installed?

You can find the source code in the same repository that contains HIPE binary files. It resides by default in the .hcss.d directory within your home directory.

In this tutorial you will make a tiny (but highly visible) change in the <u>ia_gui_apps</u> module. Go to this directory:

.hcss.d/repository/modules/hcss/ia_gui_apps

This directory contains one or more further subdirectories, each representing a version number of the module. For example:

8.71 8.73 8.77

If you install or update HIPE often you will likely have several entries. How do you find out which one corresponds to a certain HIPE version? Go to your HIPE installation directory and open the installed.userlibraries file. Search for ia_gui_apps and you will find an entry similar to the following:

<archive

path="/home/joe/.hcss.d/repository/modules/hcss/ia_gui_apps/8.77/ia_gui_apps_8.7'

This tells you that the right directory is 8.77. Among the files in this directory, ia_gui_apps.src_8.77.zip is the one containing the source.

Installing and configuring Eclipse

Now that you have the source, you need to set up the tools to modify it. We strongly recommend you use the Eclipser development environment. Besides being free and multi-platform, it has the crucial advantage of some Herschel-specific plugins that will make your life much easier.

To download Eclipse, go to http://www.eclipse.org@ and choose Eclipse IDE for Java Developers from the download page.

After installing Eclipse, install the HCSS Project plugin by following these steps:

- 1. In Eclipse, choose Help --> Install New Software.
- In the Install window, click Add. The Add Site dialogue opens. Write <u>http://herschel.esac.esa.int/hcss/eclipse/</u> in the Location field and any name you like in the Name field. Click OK.
- 3. Check the *Uncategorized* entry that has appeared in the *Install* window, and click *Next*. Follow the instructions on screen to complete the installation.

Creating an Eclipse project

First you have to unpack the source code. Create a directory to contain it, named for example <u>ia_gui_apps</u>, and a <u>main</u> subdirectory. In Linux this would be the command to issue:

mkdir -p ia_gui_apps/main

Unpack the zip file containing the *ia_gui_apps* source in the *main* directory.

Start Eclipse and choose *File --> New --> Java Project*. The *New Java Project* dialogue appears. You can write any name you like in the *Project name* field, but we'll use <u>ia_gui_apps_</u> for this tutorial.

- Eclipse 3.5 Galileo: In the *Contents* pane, select *Create project from existing* source. Then click *Browse* and select the <u>ia_gui_apps</u> directory that contains the sources.
- Eclipse 3.6 Helios and 3.7 Indigo: Uncheck the Use default location checkbox. The Location text field becomes active. Click the Browse button next to it and select the ia_gui_apps directory that contains the sources.

Click *Finish.* The project is created, and immediately gives an avalanche of errors, visible as little red icons in the *Package Explorer* view. This is normal, because Eclipse cannot yet find the auxiliary components it needs to build the project.

Right click on the new project entry in the *Package Explorer* view and choose *Properties*. The properties dialogue appears. Go to the *Java Build Path* section and select the *Libraries* tab. Click *Add library*. The *Add Library* dialogue appears. Select *HCSS Project Container* from the list (this is the plugin you installed earlier) and click *Next*. You can now configure the plugin by setting the following fields:

- **Project:** projects are listed in the *Developer Tracks* pages of the <u>Continuous</u> Integration system, and have names such as hcss.dp.core. If you are developing for the core system only (no instrument-specific software) write hcss.dp.core. If you are developing for an instrument, write one of hcss.dp.hihi, hcss.dp.pacs and hcss.dp.spire. Since ia_gui_apps is part of the core system, you can write hcss.dp.core for this tutorial.
- **Track:** corresponds to the major version of HIPE you are developing for. Write 7.0, 8.0 or whatever version of HIPE you have installed. You can check by choosing *Help --> About* in HIPE. If you see for example *8.0.375* you have to write 8.0 in this field.
- **Build:** you can choose to download a specific build or click *Fetch latest* to get the most recent one for the major version you have selected. For this tutorial, to achieve maximum compatibility, write the build number corresponding to the HIPE version you have installed (375 in the example above).
- **Build Updates:** in this pane you can choose how often Eclipse will look for and install updated builds. The default is not to look for updates.

Click *Finish* and then *OK* once you are done. Eclipse downloads the necessary files and builds the project. All the red error icons should disappear.

Note: if not all the red icons disappear, it may mean that the project still needs external resources. For example, the <u>ia_document</u> module needs the <u>tools.jar</u> file included in the Java Development Kit. To include an external JAR file, choose *Project --> Properties*, open the *Libraries* tab and click *Add External JARs*.

Modifying the source

You will not change any Java or Jython code in this tutorial, but just an HTML file. The point of the tutorial is to show how to incorporate your changes back into HIPE. In the *Package Explorer* view of Eclipse, open the ia_gui_apps project by clicking the small white arrow on the left of its name. Do the same with the main folder and then with the herschel.ia.gui.apps.views.welcome.contents node. Right click on welcome.html and choose *Open With --> Text Editor* from the context menu. This is the page that is displayed in the *Welcome* view of HIPE.

Change the following line to anything you like:

<h1>Welcome to Herschel Interactive Processing Environment!</h1>

Just a suggestion:

<hl>Welcome to Sputnik Interactive Processing Environment!</hl>

Save the file.

Creating a JAR file

JAR (Java ARchive) files are the building blocks HIPE is made of. You now have to create a JAR file with your modified <u>ia_gui_apps</u> module.

Right click on the project name and choose *Export* from the context menu. The *Export* dialogue appears. Select *Java --> JAR file* and click *Next*. The *JAR Export* dialogue appears.

Write a name in the *JAR file* field, for example <u>ia_gui_apps_new.jar</u>. You can leave all the other settings unchanged. Click *Finish*. Your JAR file is created.

At the end of the process you may get the message *JAR export finished with warnings*. This usually refers to harmless compilation warnings and can be ignored.

Including the changes into HIPE

Now you need to tell HIPE to look for your JAR file. You do this by editing the **installed.userlibraries** file in your HIPE installation directory. Open the file in your favourite editor and insert the following line as the *first* <archive path line, just below the <library name line:

<archive path="/home/joe/ia_gui_apps_new.jar"/>

You should of course put the correct path and name of your JAR file. Save the file and start HIPE. You should see your change.

Where to go from here

This tutorial showed you how to set up Eclipse for HIPE development and make a small change to HIPE.

The quickest way to add new features to HIPE is to write *tasks* and add them to HIPE as *plug-ins*. Read these two tutorials to learn more:

- My first Jython HIPE task
- My first Java HIPE task

blog comments powered by Disgus

Edit | Attach | Watch | Print version | History: r14 < r13 < r12 < r11 < r10 | Backlinks | Raw View | Raw edit | More topic actions

Topic revision: r14 - 2012-10-19 - DavideRizzo

