Heliophysics Integrated Observatory

Project No.: 238969
Call: FP7-INFRA-2008-2

HELIO Processing Service Developers Guide

Version 0.2

Title:	**HELIO Processing Service – Developers Guide**
Document No.:	HELIO_TCD_S3_002_TN
Date:	19 June 2012
Editor:	**Dr Gabriele Pierantoni**, TCD
Contributors:	
Distribution:	Project

CAPACITIES

e-infrastructure

Revision History

Version	Date	Released by	Detail
0.1	19 June 2012	Dr. Gabriele Pierantoni	1st draft
0.2	01 October 2012	Dr. Gabriele Pierantoni	Small corrections

Note: Any notes here.

Table of Figures

Introduction

The HELIO Processing Service (HPS) is an interface to different computational resources that offers to HELIO two different kinds of services: the execution of a pre-defined set of applications and the execution of user-defined applications.

At the moment being, HPS supports two different computational resources. A fast execution engine suitable for short lived jobs and a more powerful (albeit with a longer overhead time) computational resource based on a gLite WMS.

According to its dual nature, the HPS can be deployed in two modalities as sketched in Figure 1.

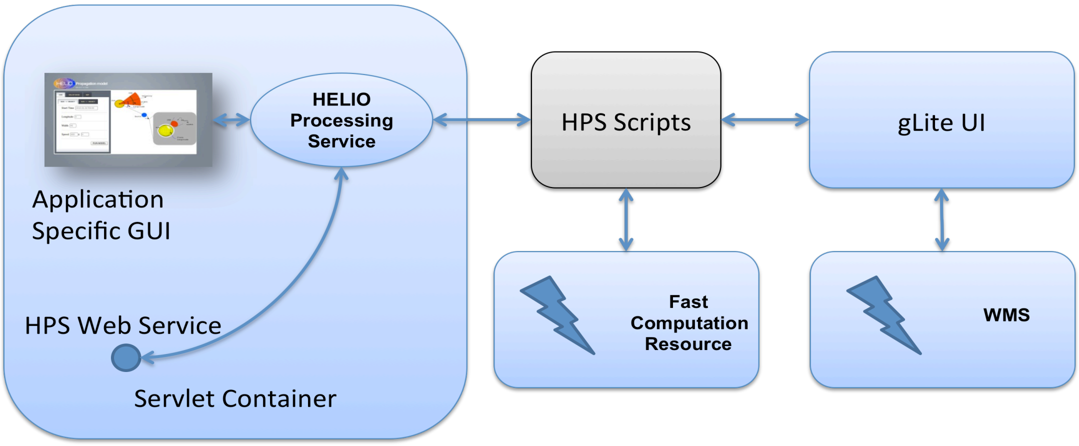

Figure 1, different deployment profiles of HPS

When the HPS only supports a simple, fast computational resource, its stack is described in Figure 2.

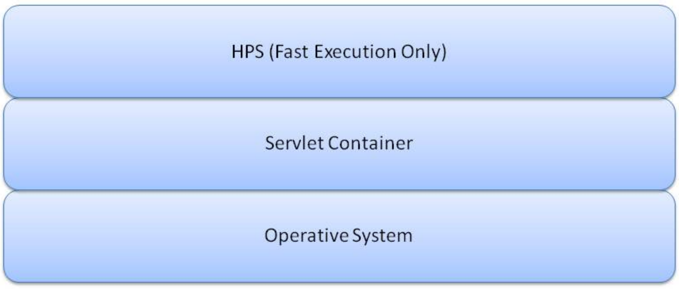

Figure 2, HPS stack in simple modality

In this case, the HPS simply consists of a web service application that comprehends an application repository, the main application, its web service interface and a GUI.

When a more powerful computational resource is required, the HPS can be used in conjunction with a gLite WMS service to execute its application on a Grid resources as sketched in Figure 3.

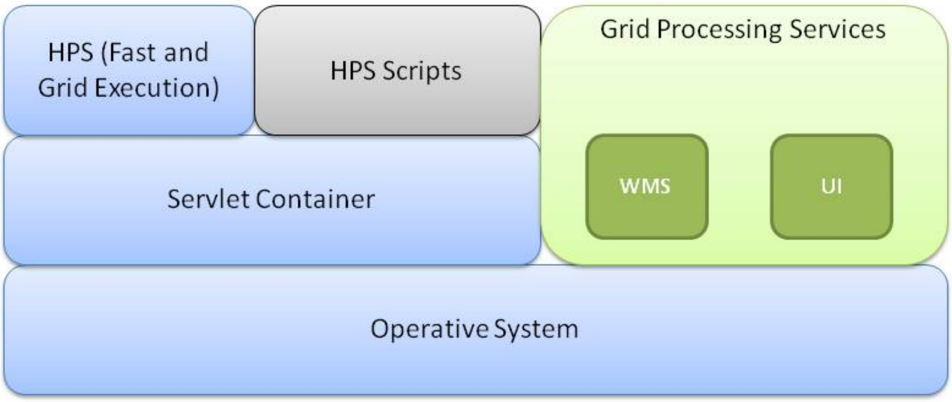

Figure 3, HPS stack in full (fast and Grid) modality

Suggested Reading

To build the service	
Java (compile service)	http://www.oracle.com/technetwork/java/javase/overview/index.html
Tomcat (web container to host the service)	http://tomcat.apache.org/
Maven (build system)	http://maven.apache.org/ (or use a plug-in to your development environment)
To install the service	
WMS (Main web page)	http://web.infn.it/gLiteWMS/
gLite UI (Main page)	http://glite.web.cern.ch/glite/packages/R3.2/sl5_x86_64/deployment/glite-UI/3.2.1-0/glite-UI-3.2.1-0-update.html
gLite UI (Documentation page)	https://edms.cern.ch/document/722398/

System Requirements

- Java 1.6
- Servlet Container i.e. Tomcat greater than 5.5, GlassFish, JBOSS. For the purpose of this document, installation is in Tomcat.
- gLite WMS (Only if grid support is required)
- gLite 3.2 UI (Only if grid support is required)

Service Installation

HPS primary installation is in TCD, these instructions will create a mirror service. The service can be installed as simple (fast only) application execution or full (fast and grid) application execution.

Downloading the Source

- HELIO Processing Service is installed as a war file on a servlet container:

Building the Service

No building is necessary.

Installing the Service

It is assumed installation of a servlet container Tomcat.

Simple Installation

The simple installation requires that the war file is deployed in the servlet container.
The HPS configuration is contained in the HPS.conf file that must be deployed in the tomcat directory. When the HPS is deployed in simple modality, the HPS.conf file contains the following entries:

- The location of the application repository, a directory that can be read by the tomcat user.
- The location of the output directory, a directory that will contain the output of the applications' execution. The output directory must be readable and writable by the TOMCAT user. Also this directory is usually exposed by a HELIO Storage Service (HSS).
- The directory of all the HPS scripts defined in the HPS.conf file be readeable and executable by the TOMCAT user but cannot be written.

Full Installation

In addition to the steps described in the Simple Installation section, requires that:

- An instance of a gLite UI 3.2 is installed on the machine as per https://twiki.cern.ch/twiki/bin/view/LCG/GenericInstallGuide320 and https://twiki.cern.ch/twiki/bin/view/LCG/GenericInstallGuide320#The_UI
- An instance of a WMS server is installed (as per https://twiki.cern.ch/twiki/bin/view/LCG/GenericInstallGuide320#Installing_the_Middleware) and can be reached by the machine hosting the HPS.

Installation on an IDL/Solarsoft environment

If the applications require an IDL/Solarsoft environment, this could be installed with the following instructions.

- Install IDL and its license using the instructions http://www.exelisvis.com/ProductsServices/IDL.aspx. Version used in thte HPS: 7.1
- Complete (all packages) installation of Solarsoft following the instructions detailed in http://www.lmsal.com/solarsoft/ssw_install_howto.html on the machine[1].
- The applications that use this environment have to be configured to the proper Cshell environment variables used by solarsoft.

Structure of the Code

The HPS code is divided into different packages:

- ***helio-shared*** - Utilities classes shared among different components of HPS and other services are in the shared project,
- ***helio-hps-server-lib*** - This package contains all the core functionalities of the HPS server,
- ***helio-hps-server-ws*** – This package contains the web-service interface of the HPS
- ***PropagationModel*** – This package contains the graphical user interface specific to the propagation model. One application running in the HPS.
- ***helio-hps-client*** – This package contains the client to the web-service interface of the GPS
- ***Applications*** – These are the applications available to the HPS. At the moment SHEBA propagation model is installed. Two algorithms (CHARM and SMART) for feature detections are also in the system and are usually executed in the Grid environment.

- [1] This is a library that is subjected to changes. Code that relies on it may break.

Dependencies

The HPS dependencies are managed by the maven project of HELIO. They are described package by package by their pom.xml files.

Logging

HPS logs to the ***catalina.out*** log file.

Maintaining & Extending the Service

To add an application, it has to be added to the application repository.

Fixing problems

Contact mailto://helio-services@majordomo.mssl.ucl.ac.uk

Download, Build, Install

If the requirement is to view the source code to make an extension or correct a possible error, follow instructions below for building from the source code repository.

Downloading Source

The HPS source code is part of the HELIO-vo project in sourceforge. The main page is: http://sourceforge.net/projects/helio-vo/.
You can download the complete source code using svn issueing the command
svn co https://helio-vo.svn.sourceforge.net/svnroot/helio-vo helio-vo

Building HPS

HPS is built using Maven. It is recommended to issue a ***maven clean*** command before building with a ***maven package*** command. This will build a war file helio-hps-server/target directory.