Grid web services and Virtual Observatory: VisIVO and new capabilities

INAF–Catania Astrophysical Observatory
A. Costa¹, M. Comparato¹, U. Becciani¹
C. Gheller²

¹ INAF – Catania Astrophysical Observatory
² CINECA - Italy
VisIVO
Visualization Interface for the Grid & Virtual Observatory

VisIVO is a visualization package developed in collaboration between INAF (Catania Astrophysical Observatory) and CINECA (the largest Italian academic supercomputing center) with the specific object of supporting visualization and analysis of astrophysical data. The package is written in C++.

VisIVO: Italian Contribution to the VO-TECH – DS6: Data Exploration (Data Mining + Visualization)

The VOTech project aims at completing all technical preparatory work necessary for the construction of the European Virtual Observatory (Euro-VO).
VisIVO can be used both as a stand-alone application, that acts on local files, and as an interface to the Virtual Observatory framework, from which it can retrieve the data.
NEW 0.9 release is now available:

http://visivo.cineca.it/
http://visivo.oact.inaf.it/
VisIVO & INFN Production Grid
(EGEE compliant)

The interface to the grid

- Apache Tomcat 5.5.9
- Axis 1.3
- JSSE (Java Secure Socket Extension)
- GILDA User Interface Plug & Play

(fully compatible with gLite 3.0)
Web services: overview

Application

Authentication & Authorization

Message Parser

Application Handling
STD in & out

Logging

soap messages

GRID
Web services: overview

- Authentication & Authorization
  - OpenLDAP
  - Plain text file
  - XML document

- Logging
  - Multilevel Log Writer:
    - Debugging
    - Production
    .......


Web services:
overview

- **Application Handling**
  - Creates a new process
  - Retrieves its Standard input & output
  - Creates “ad hoc” jdl files and bash scripts

- **Message Parser**
  - checks syntax and range for numeric input parameters
VisIVO Web Service

UML Use Case Diagram

Application

myproxy-get-delgat

glite-job-submit

glite-job-status

glite-job-output

ApplicationHandler

«uses»

«uses»

«uses»

«extends»

GRIDs

Logging

MessageParser

Auth&Auth

«uses»

«extends»
HOP Group finder

HOP is an algorithm for finding groups of particles based on the one developed and coded by Daniel Eisenstein & Piet Hut, Institute for Advanced Study, Princeton, NJ

HOP was distributed as RPM Package and installed in the Worker Nodes Elements

http://cmb.as.arizona.edu/~eisenste/hop/hop.html
HOP Group finder RPM Package
hop-1.0-1.rpm

- hop
- regroup
- hop_to_xml
  - Command Line Interface:
    creates a VoTable from hop-regroup output files.
- hop_doc.txt
- hop_doc.html
FLY

• FLY is a parallel tree code that runs, with a very high resolution, N-Body simulations of the Large Scale Structure of the Universe. It runs on multi processors Unix-like systems where a full implementation of the MPI-2 communication standard is available and on IBM SP systems using LAPI. The code is designed to minimize data replication among processors and to maximize the number of bodies which can be used.

FLY has been developed by:
U. Becciani, V. Antonuccio-Delogu
(INAF Astrophysical Observatory of Catania)

http://www.oact.inaf.it/fly/
Upload your credentials

First of all you need a digital X509 certificate issued by INFN Certification Authority, or most simply a digital certificate issued by GILDA (Grid Infn Laboratory for Dissemination Activities); you can follow the link https://gilda.ct.infn.it/users.html for the instructions to request and obtain a digital certificate and a User Interface account.

Then you have to upload credentials to a myproxy-server from a User Interface everywhere in the world.

The command line is:
$ myproxy-init -s "myproxy server" -l "myproxy username" (you have to provide a myproxy passphrase).
Get Myproxy Delegation

- The OK button will invoke the web service and will allow you to retrieve credentials from the myproxy opening your session in the grid environment.
HOP-Regroup

- The OK button will invoke the web service and will run a HOP-REGROUP session.
- The web server will create the ad hoc jdl file reporting all the user specifications: “guid” and arguments and the job will be executed in the grid nodes nearest to the physical input grid file or its replicas.
- Finally the grid job identifier will be prompted to the user.
- VisIVO can create a snapshot of your working session making a backup of all your parameters values and grid job identifiers. This is useful for exploit grid-web-services power by working off-line.
- Each snapshot contains the parameters and the job status on the grid: this can be updated through a specific VWS by a mouse click.
Output Results as VoTable

- Point Coordinates
- Density values (from Hop Algorithm)
- Group ID
Output Results as VoTable