

# Medical-Image Processing Workflow Support on EGEE with Taverna

*Ketan Maheshwari*, Paolo Missier, Carole Goble, Johan Montagnat

UNS / CNRS-I3S Laboratory and The University of Manchester

August 3, 2009



# Overview

- Optimize the enactment of Data Intensive Workflows on Grid Infrastructures.
- Extend Taverna user community to biomedical applications.
- Grids enabling data intensive medical image analysis applications.
- This work attempts to achieve this objective by employing a 'Grid plugin' to the Taverna workbench.
- This presentation details the Taverna gLite plugin Design, Implementation and its Usage.

- A popular scientific workflow manager with 1000+ strong user base.
- Advanced enactment capabilities including pipelining and data parallel enactment mode.
- Extendible Architecture of Taverna enables custom plugin development.
- Sophisticated User Interface but lack of grid integration.
- Taverna workbench ease the access to Grid infrastructure.

# EGEE and gLite

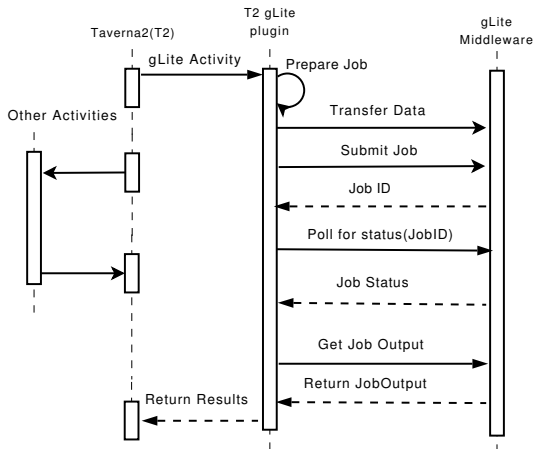
- EGEE (Enabling Grids for E-scienceE) is a Premier European Grid Infrastructure.
- 125 Virtual Organizations, 9000+ users in 50 countries, 20 petabyte of storage and 80000 processor cores.
- Computation abstracted to Compute Elements (CE) and Storage to Storage Elements (SE).
- gLite is a de facto middleware program to access the EGEE batch system environment.



# Design Challenges

- The main design challenge constitutes of coupling Taverna and EGEE environments.
- Transforming asynchronous Taverna calls to the batch-oriented, poll-based EGEE is a challenge.
- Overcome the Grid reliability issues.

# Taverna EGEE Interaction via gLite Plugin



# Implementation

- Process Description: Auto-generation of gLite Job Description Language.
- Data Transfer: Auto-generation of wrapper script.
- Job Status Polling : Configurable Frequency.

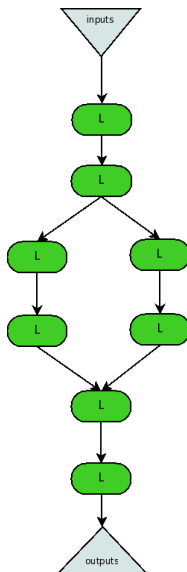
# Addressing Reliability

- Job Resubmission on Error, long Wait or Abort state.
- Round Robin selection of Resource Brokers.
- Reliable data transfer: Repeat transfer in case of failure, rotate Storage Elements.

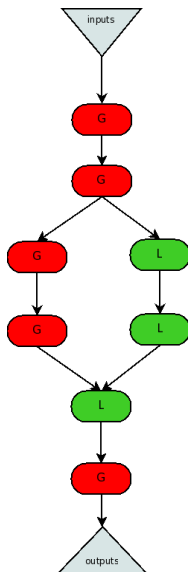


- Workflow Composition panel of Taverna provides a gLite processor.
- Configurable properties of the gLite processor.
- Automatic proxy delegation.
- Configurable polling frequency.
- Readily alterable execution mode.

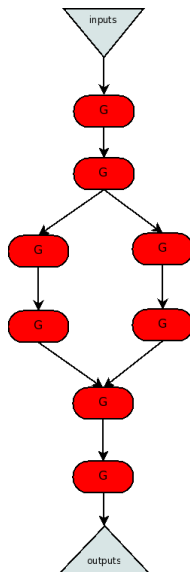
# Execution Modes: Pure Local



# Execution Modes: Local/Grid Mixed



# Execution Modes: Pure Grid



# Screenshots(1)!

**Taverna Workbench 2.0**

File Edit Activities Workflows Advanced Help

Design Results

Enter search here Search

type None

Available activities (702)

- String Constant (1)
- gLite (1)
  - gLite
- Beanshell (1)
- Rshell (1)
- Workflow (1)
- WSDL (106)
- Localworker (50)
- Biomart (253)
- Soaplab (288)

**gLite Plugin**

Contextual View: Processor Image\_Crop

Input Port Name	Depth
ConfigIn	0
DataIn	0

Output Port Name	Depth	Granularity
DataOut	0	0
ConfigOut	0	0

Configure

List handling

Advanced

Workflow Explorer

- dataflow0
  - Inputs
  - Outputs
  - Results
  - Processors
    - Image\_Crop
    - ConfigIn
    - DataIn
    - DataOut
    - ConfigOut
    - Interpolation
    - ConfigIn
    - DataIn
    - DataOut
    - ConfigOut
    - Image\_Pyramid\_Decomposition

Inputs

Image\_Crop

Interpolation

Image\_Pyramid

Gradient\_Computation

Motion

Work

Properties Ports jdl

VO biomed

CADir /etc/grid-security/certificates

VOMSDir /home/kean/gliteui-3.1/glite/etc/vomses

VOMSCertDir /etc/grid-security/vomsdir/

ProxyPath /tmp/x509up\_u501

UI egee1.unice.fr

WMProxyURL ee-wms-01.cnaf.infn.it:7443/glite\_wms\_wmproxy\_server

WMSDir home/kean/gliteui-3.1/glite/etc/biomed/glite\_wms.conf

Output Path /tmp/

Poll Frequency Path where you want your output to be written (usually /tmp/)

SE hepgid11.ph.liv.ac.uk

☐ Execute Locally

**Execute Local Job Properties**

OK Cancel

# Screenshots(2)!

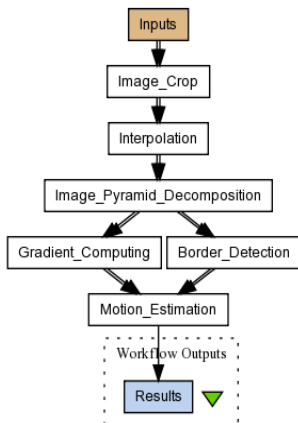
The screenshot displays the Taverna Workbench 2.0 interface. On the left, the 'jdl Properties' dialog box is open, showing configuration for a job named 'Image\_Crop'. The 'Nodes' field is set to 'Normal or MPICH'. The 'InputSandbox' is 'Image\_Crop', and the 'OutputSandbox' is 'stdout'. The 'Executable' is 'Image\_Crop.sh'. The 'Job Requirements' are set to 'other GlueCEStateStatus == "Production"'. The 'Retry Count' is 3, and the 'Inputs Path' is '/xetian/ManchesterWork/gliworkflows/Inputs/'.

The main workspace shows a workflow diagram with the following steps:
 

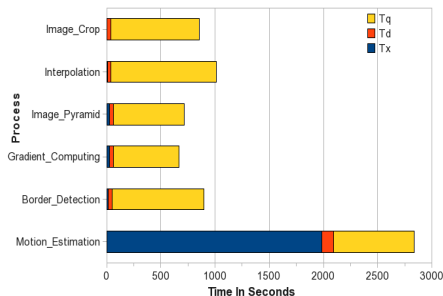
- Inputs** (brown box)
- Image\_Crop** (blue box, highlighted with a red border)
- Interpolation** (white box)
- Image\_Pyramid\_Decomposition** (white box)
- Gradient\_Computing** (white box)
- Border\_Detection** (white box)
- Motion\_Estimation** (white box)
- Workflow Outputs** (dashed box containing):
  - Results** (blue box)

The 'Workflow Explorer' on the right lists the workflow components and their connections. The 'Contextual View: Processor Image\_Crop' at the bottom left shows the activity details for the 'Image\_Crop' processor, including input and output port names and depths.

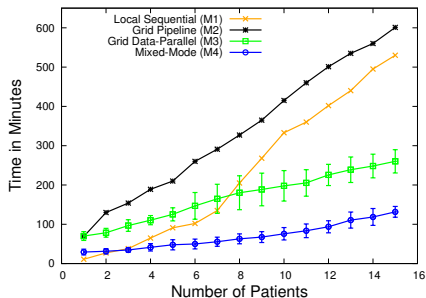
# The Workflow and Results (1)



Orange(Tq)=queueing overhead,  
Red(Td)=data transfer overhead,  
Blue(Tx)=execution time



## The Workflow and Results (2)



- Patients: 15
- Data: 20-30M/patient
- Peak Load: 65 concurrent threads



# Conclusions

- gLite plugin is one of the first development in Taverna to interface with the grid and *the* first with EGEE.
- Emphasis on ease of workflow composition and Grid execution & data transfer reliability.
- 'Mixed-mode' enables easy empirical tests of data-intensive workflows.
- Can be easily applied to workflows from other domains involving data pipelines.

# Thanks!, Questions?

This work is supported by the French ANR GWENDIA project under contract number ANR-06-MDCA-009. We are thankful to *myGrid* developers team for their excellent technical support and CREATIS (CNRS-INSERM) laboratory for providing application components. We thank Oleg Sukhoroslov for providing jLite java API.

