A Grid Infrastructure for Managing Workflows in Bioinformatics Applications

Maurizio Melato

<maurizio.melato@nice-italy.com>

NETTAB Workshop, July 10-13, 2006, Santa Margherita di Pula
Scenario

- Bioinformatics scientists have to execute complex tasks
- There is the need to orchestrate these services in workflows
- Bioinformatics scientists have to execute complex tasks
- There is the need to orchestrate these services in workflows
Today there is a high demand for workflow management.

Many current workflow management systems have strong limitations due to their client nature and too many standards.

With client nature, we mean fat client application running on the user’s workstation:
- low reliability
- no fault tolerance
- typically one workflow at a time
- ...

Grid infrastructures play a very important role in enacting workflows based on Grid services.
Gridified Scenario

- Grid technology leverages both the *computational* and *data management* resources.
- Providing optimisation, scalability, reliability, fault tolerance, QoS,...
Gridified Scenario

- Grid technology leverages both the *computational* and *data management* resources
- Providing optimisation, scalability, reliability, fault tolerance, QoS, ...
**Gridified Scenario**

- Grid technology leverages both the *computational* and *data management* resources.
- Providing optimisation, scalability, reliability, fault tolerance, QoS, …
Infrastructure proposal

- **Goal**: proposal of a *Grid infrastructure* able to provide the basic building blocks for *composition* and *enactment* of bioinformatics workflows for the life sciences community

- **Users can**
  - build workflows on top of exposed Grid services
  - run and monitor workflows via a standard Web browser
  - Exploit in a transparent way the computational power and data access capabilities provided by the backend Grid infrastructure…
We used the following tools to build the proposed architecture:

- **EnginFrame or Genius:**
  - As Grid services provider
  - As Web interface for managing workflow

- **Taverna:**
  - As Workflow designer

- **Moteur:**
  - As Workflows enactor
**Taverna & Moteur**

- **Taverna** is a graphical workbench tool for both creating and running workflows - in Scufl language - that allows the integration of resources/services that are published as Web Services.

- **Moteur** is a service based Scufl workflow engine developed in the AGIR project and optimized for dealing with data intensive applications.

  The current prototype is able to use Moteur as batch enactor of Scufl workflows of standard Web Services.
What is EnginFrame?

- It is a Web-based technology able to expose Grid services running on Grid infrastructures.
- It allows organizations to provide application-oriented computing and data services to both users (via Web browsers) and applications (via SOAP/WSDL and/or RSS), hiding all the complexity of the underlying Grid infrastructure.
- It greatly simplifies the development of Web portals exposing computing services that can run on a broad range of different computational Grid systems.
The Grid Portal / Gateway
EnginFrame adoption

- **Industries**
  - **Mechanical**: Ferrari, Audi, BMW, FIAT Auto, Delphi, Elasis, Magneti Marelli, P+Z, Swagelok, Toyota, TRW
  - **Manufacturing**: Bridgestone, Procter & Gamble, Galileo Avionica
  - **Oil&Gas**: Slavneft, Schlumberger, TOTAL, VNIIGaz
  - **Electronics**: STMicroelectronics, Accent, SensorDynamics, Motorola
  - **Biotech**: ENEA, EGEE LS community
  - **Telecom**: Telecom Italia

- **Research**
  - CERN, INFN, ASSC, CCLRC, CILEA, CINECA, CNR, CNRS/IN2P3, ENEA, FzU, ICI, IFAE, ITEP, JSC G.G.M., KU Leuven, SSC-Russia,

- EnginFrame is the technology on which is based GENIUS, that’s nowadays a standard of graphical user interface access to the EGEE grid infrastructure.
Usability & Input management

- User friendly, Application-oriented
  Job submission

- Flexible and efficient
  Input file management

- Hide complexity of
  Underlying scheduler
Usability & Input management

- User friendly, Application-oriented
- Job submission
- Flexible and efficient
- Input file management
- Hide complexity of
- Underlying scheduler

XML

```
- <ef:agent xmlns:ef="http://www.enginframe.com/2000/EnginFrame" authority="os"
  id="com.enginframe.system">
  <ef:name>EnginFrame System Services</ef:name>
  <ef:location host="127.0.0.1" port="9999"/>
</ef:agent>

- <ef:spooler server="${EF_SPOOLER_DIR}" ttl="0">
  <ef:folder id="root">
    <ef:name>EnginFrame System Services</ef:name>
    <ef:folder id="data">
      <ef:name>Data management</ef:name>
      <ef:service id="list.spoolers">
        <ef:name>List your spoolers</ef:name>
        <ef:option id="countHiddenFiles" type="hidden">false</ef:option>
        <ef:action id="submit" label="Details">
          ${EF_ROOT}/plugins/ef/bin/ef.list.spoolers
        </ef:action>
      </ef:service>
      <ef:service id="show.spooler">
        <ef:name>Browse spooler contents</ef:name>
        <ef:option id="uri" label="URI:" type="text" />
        <ef:option id="sub" label="Sub Directory:" type="text" />
        <ef:action id="submit" label="Details">
          ${EF_ROOT}/plugins/ef/bin/ef.show.spooler $uri $sub
        </ef:action>
      </ef:service>
    </ef:folder>
  </ef:folder>
</ef:spooler>
```

NETTAB Workshop, July 10-13, 2006, S.ta Margherita di Pula
Monitoring and Output management

Data lifecycle management

Comprehensive output management
File manipulation (view, edit, delete, zip, ...)

Job details & control
Double role of EnginFrame

EnginFrame plays a double role at two different levels of the proposed architecture:

1. **Grid services provider**: Grid services exposed as Web services by EnginFrame can be used in the workflow as standard nodes

2. **Grid portal Web interface**: for managing workflow submission and management on the Grid

In both of its roles EnginFrame exploits the computational power and data access capabilities provided by the backend Grid infrastructure.
The proposed Grid Infrastructure

- WS provider
- MOTEUR (Scufl wf enactor)
- Storage and Data
- Computational Power
- Grid Portal / Gateway
- Web Portal
- Services (SOA)
- HTML/HTTP
- SOAP
- WSDL
- Tools

NETTAB Workshop, July 10-13, 2006, S.ta Margherita di Pula
The service interface implemented provides the following functionalities:

- **Upload of Scufl workflow and related inputs**: users can upload their own workflows in Scufl language and insert input data for execution.
- **Submission of Moteur as Grid job**: the workflow is executed by Moteur on the Grid infrastructure as a standard Grid job.
- **Monitoring of workflow**: it is possible to check both Moteur submission and the workflow processing status together with data produced by intermediate results.
- **Results visualization**: when jobs are terminated, workflow results are staged in the EnginFrame spooler area and made available through the portal for visualization, post-processing or download.
Bioinformatics application

- Implementation of a bioinformatics application workflow exploiting the proposed architecture
- Workflow version of an innovative bioinformatics application developed by the Bio-Lab team of the University of Genoa
- Based on DChip, one of the most complete and diffuse free software for the microarray data analysis
- Composed of different modules:
  1. data set opening and normalization
  2. model based gene expression
  3. extraction of differentially expressed genes
  4. clustering
Application Workflow design
Scufl submission and enactment
Workflow status
Output Management
Conclusions

- We have proposed a Grid infrastructure supporting workflow design with Grid services as building blocks, workflow enactment and life cycle management.

- A workflow of a bioinformatics application has been described to show how it is possible to exploit this type of infrastructure in bioinformatics area

- Limits: Moteur appears as a performing enactor tool, but it imposes some limitations about processors and data inputs → many standard incompatibilities with most common used bioinformatics services

- Future steps: the proposed infrastructure is still a prototype.
  - Stay tuned on the evolution of MOTEUR…
  - Improvements to DChip modules interfaces to make easier to wrap them as EF services
  - Improve Web interface to provide users with dynamically generated forms for workflow inputs definition
A-WARE FP6 Funded Project

Project goals
- Simplify users’
  - life (focus on problems)
  - way of perceiving the GRID
- Fill an existing gap
  - between middleware and portals
- EnginFrame + A-WARE + UNICORE/GS
  aim to be a completely integrated solution
- Contribute to the standards
A-WARE: Key technology advancements

- Key Advancements
  There is a requirement for a higher level access point to the Grid, as an entry at the Target System is too low level. What is missing is a high-level service, which can manage the multiple invocations of TSS, and other services. The project will supply this as a Workflow Orchestrator Service (WOS).
A-WARE: Key technology advancements

- Os/Storage
  - Unicore/GS
  - GTK4
  - gLite
  - Other

- Non OGSA middlewares
  - Other
  - LSF

- OGSA interface
  - UAS interface

- EU-1 technology components
  - Workflow orchestrator service (WOS)
  - Workflow validation service (WVS)
  - Workflow designer application (WDA)
  - Workflow repository Service (WRS)

- Other consumers

- Portal standard compliant (open source)
Thanks for your attention!

Q&A