Integrate Virtual Screening Service with Desktop Grid by EDGeS infrastructure

Chun-Wei SHEN <waynesan@twgrid.org>,
Yu-Ting CHEN <yuting.chen@twgrid.org>
Academia Sinica Grid Computing
Outline

• Introductions
• GVSS System & Service Architecture
• Integration of SG and DG by EDGeS
• Master-Worker Pilot-Job with BOINC
• Introduction of Java Pilot-Job Framework
• Web UI System & Service Architecture
• Prototype Screenshots
• Future Plan
• Questions
Introduction of Master-Worker Pilot-Job model

- Job submission in Grid rely on using metaschedulers, which directly interact with the Local Resource Management Systems (LRMS) installed at the different sites offering resources. Therefore, this process requires, at least, two queuing and scheduling processes, which leads to an additional overhead.

- Strategies of piloting or master / slave submission, in which the jobs submitted, are the executors that communicate with central managers that dynamically dispatch several jobs that are executed without additional queuing and scheduling.

Introduction of GVSS

- Grid-enabled Virtual Screening Service
- Base on Grid Application Platform (GAP)
- Using DIANE as Pilot-Job framework
- Using AMGA as storage metadata catalog
GVSS System Architecture

GVSS

GAP-VQS
- Database

GAP-LSA (DIANE)
- gLite

AMGA
- Database

GridFTP
GVSS Service Architecture
Integration of SG and DG by EDGeS
Scenario 1 – DG to SG via bridge

- User entry point is DG – using SG is completely transparent from user’s point of view
Scenario 2 – SG to DG via bridge

- Desktop Grid 1
- Desktop Grid n
- EGEE VO
  - Edges Services
  - DG CE + EDGeS AR
  - WMS and other EGEE services
- SG (EGEE) user (using EGEE UI machine or portal)
  - User entry point is SG
  - Using DG is transparent from user’s point of view
SG/DG resources but not through EDGeS bridges
Master-Worker Pilot-Job with BOINC

Three models:

1. BOINC as a Job Queue
2. BOINC as a Worker Manager
3. Super Worker above BOINC
1. BOINC as a JOB QUEUE

- **Pilot-Job Master**
  - Push jobs
  - Recruit workers

- **Job Queue**
  - Pull jobs

- **Service Grid**
  - Pull jobs

- **BOINC Server**
  - Push jobs
  - Pull jobs

- **Desktop Grid**
1. BOINC as a JOB QUEUE

- BOINC work unit as a "Job"
- Master must split jobs into SG and DG parts
- Job(worker) in SG cannot be "translated" to work unit(job) in BOINC
2. BOINC as a WORKER MANAGER

- **Pilot-Job Master**
  - Push jobs
  - Recruit workers

- **Job Queue**
  - Pull jobs

- **BOINC Server**
  - Pull jobs
  - Pull workers

- **Service Grid**
  - Pull workers

- **Desktop Grid**
  - Recruit workers
2. BOINC as a WORKER MANAGER

- BOINC worker unit as a worker
- Both in SG and BOINC, job means the worker program
- Automatic load balanced
- Can NOT use BOINC validator on jobs
3. Super worker above BOINC

Recruit workers

Push jobs

Pull jobs

Service Grid

Pilot-Job Master

Job Queue

BOINC Server

Desktop Grid

Super worker

Push jobs

Pull jobs
3. Super worker above BOINC

- BOINC work unit as a "Job"
- Automatic load balanced
- Can use BOINC validator
Introduction of Java Pilot-Job Framework (Cell)

• A new project from Dec. 2010

• Major components:
  • Authentication
    - GridAuthentication, JLiteAuthentication
  • Worker Manager
    - JLiteWorkerManager, 3gBridgeWorkerManager
  • Information Service
    - JPAInformationService, HSQLDBInformationService
  • Task Ticket Window
    - EmbeddedTaskTicketWindow (base on ActiveMQ)
Cell-based Web UI
System Architecture

Web UI
  ├── Cell-Master
  │     ├── HSQL DB
  │     ├── Active MQ
  │     └── jLite
  └── GASS
      └── GridFTP

or

3G-bridge WS
Cell-based Web UI
Service Architecture
Prototype Web UI
Screenshots

Choose and Submit Jobs!

Job Status

Dengue Fever NS3:

Please choose docking ligands (DIRECTORY):
Future Plan

- Finish implement the Web-based Virtual Screening Service with Cell and EDGeS infrastructure.
- Propose Cell-VSS Worker to EDGeS Application Repository.
- Recruit other EDGeS BOINC projects to support Cell-VSS Worker.
- Porting Cell-VSS Worker on more platforms.
Questions?

Chun-Wei SHEN <waynesan@twgrid.org>,
Yu-Ting CHEN <yuting.chen@twgrid.org>
Academia Sinica Grid Computing