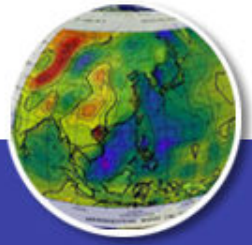
Development of e-Science Application Portal on GAP

WeiLong Ueng
Academia Sinica Grid Computing
wlueng@twgrid.org




Outline

- Introduction
- Grid Application Platform
- e-Science Application Portal
- Architecture and Components
- Conclusion

Introduction

- Computational scientists often develop large models and codes intended to be used by larger user communities or for repetitive tasks such as parametric studies. Lowering the barrier of entry for access to these codes is often a technical and sociological challenge.
- Portals help bridge the gap because they are well known interfaces enabling access to a large variety of resources, services, applications, and tools for private, public, and commercial entities, while hiding the complexities of the underlying software systems to the end-user.
- Computational science portals are emerging as useful and necessary interfaces for performing operations on the GRID.

Motivation and Purpose

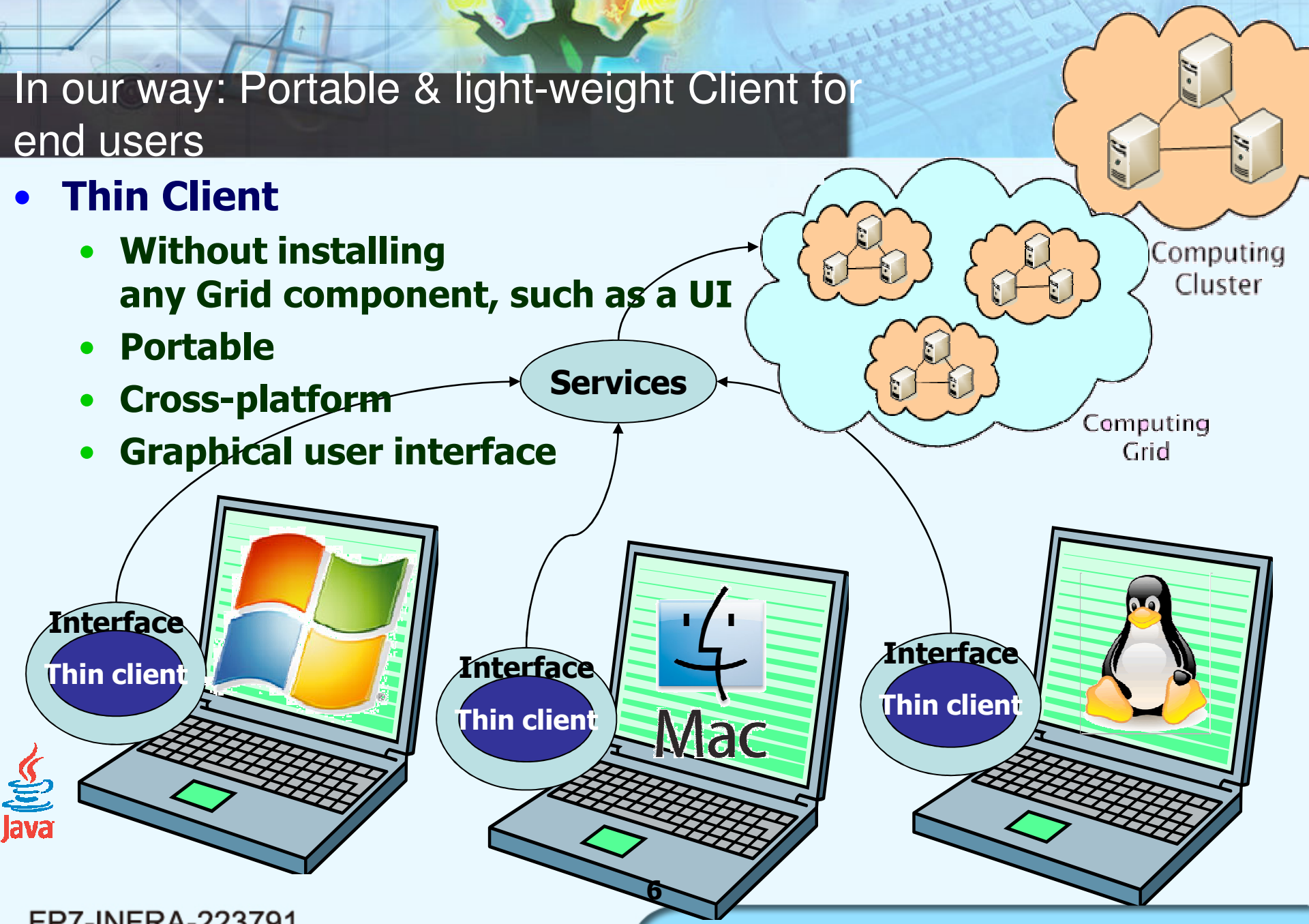
- **It is developed based on Grid Application platform (GAP), and provides a customizable interface allowing researchers to use a variety of GRID services including job submission, job monitoring, data management…etc.**
- **Through the grid-enabled computing portal, the end-users can easily take the advantage of GRID computing resources for large-scale scientific computing. Furthermore, they can even execute their own scientific computing, and obtain the computing result with this portal. In this way, the end-user can use the Grid environment more easily and securely.**

Difficulties for end users

- **Using a application on the grid, end users have to:**
  - Login to a remote Linux/Unix server which have the gLite UI component installed.
    - That UI component only provides command line interface instead of graphical user interface.
  - users have to familiar with commands of grid middleware due to lack of friendly user interfaces.
- **The whole application is tightly coupled with that UI component.**
- **UI is still a remote component, and it is not easy to be integrated with an existing application.**

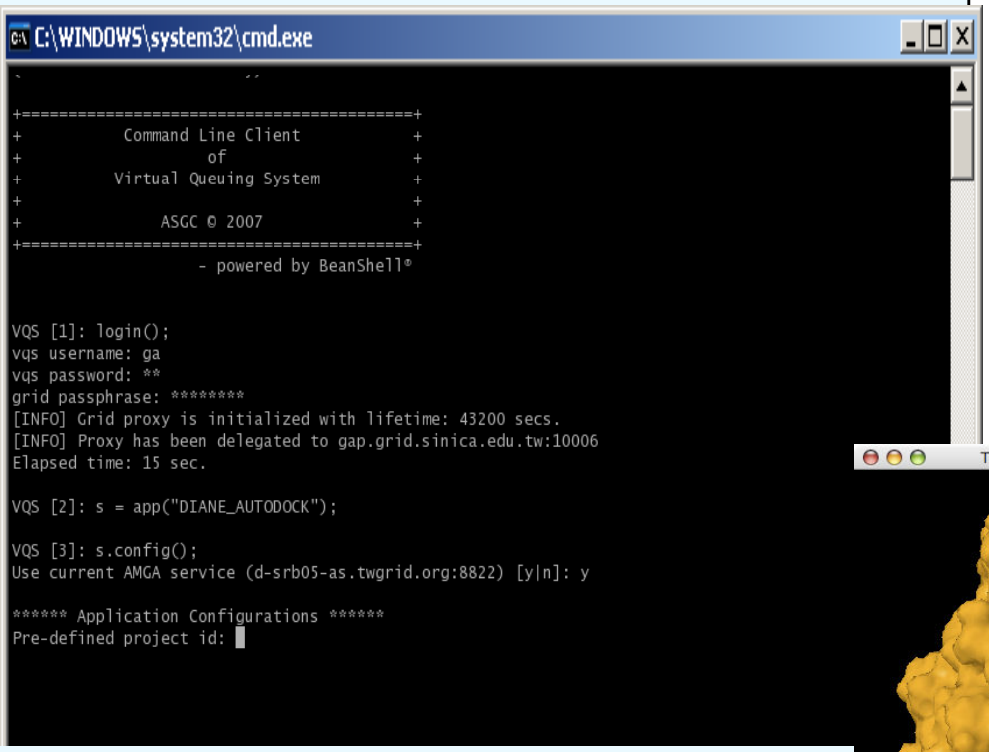
In our way: Portable & light-weight Client for end users

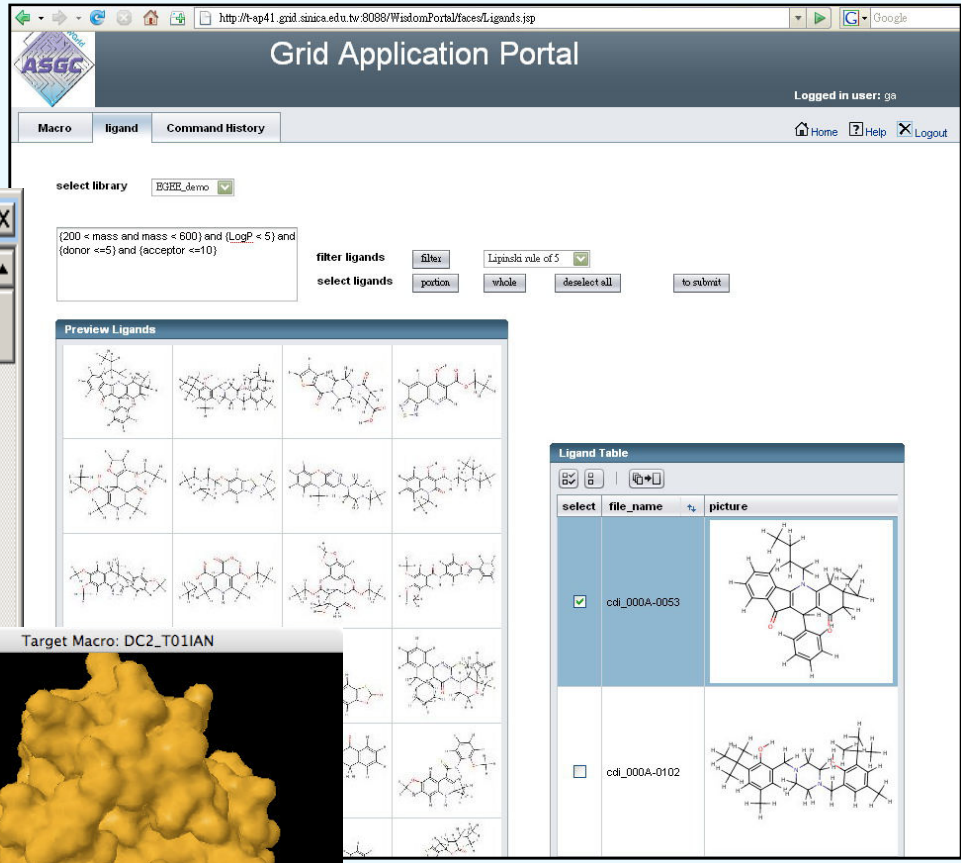
- **Thin Client**
  - **Without installing any Grid component, such as a UI**
  - **Portable**
  - **Cross-platform**
  - **Graphical user interface**

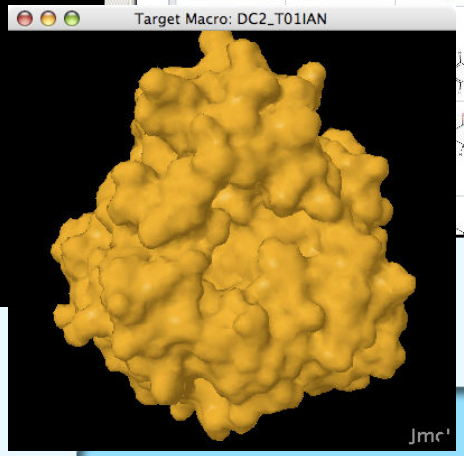


In our way: Intuitive Friendly User Interfaces for end users

- **Submitting jobs in an application oriented view is very easy.**







Grid Application Platform

- **Grid Application Platform (GAP) is a grid application framework developed by ASGC. It provides a vertical integration for developers and end-users**

  - **In our aspects, GAP should be**

    - Easy to use for both end-users and developers.
    - *Easy to extend for adopting new IT technologies, the adoption should be transparent to developers and users.*
    - *Light-weight in terms of the deployment effort and the system overhead.*

The layered GAP architecture

Reduce the effort of developing application services

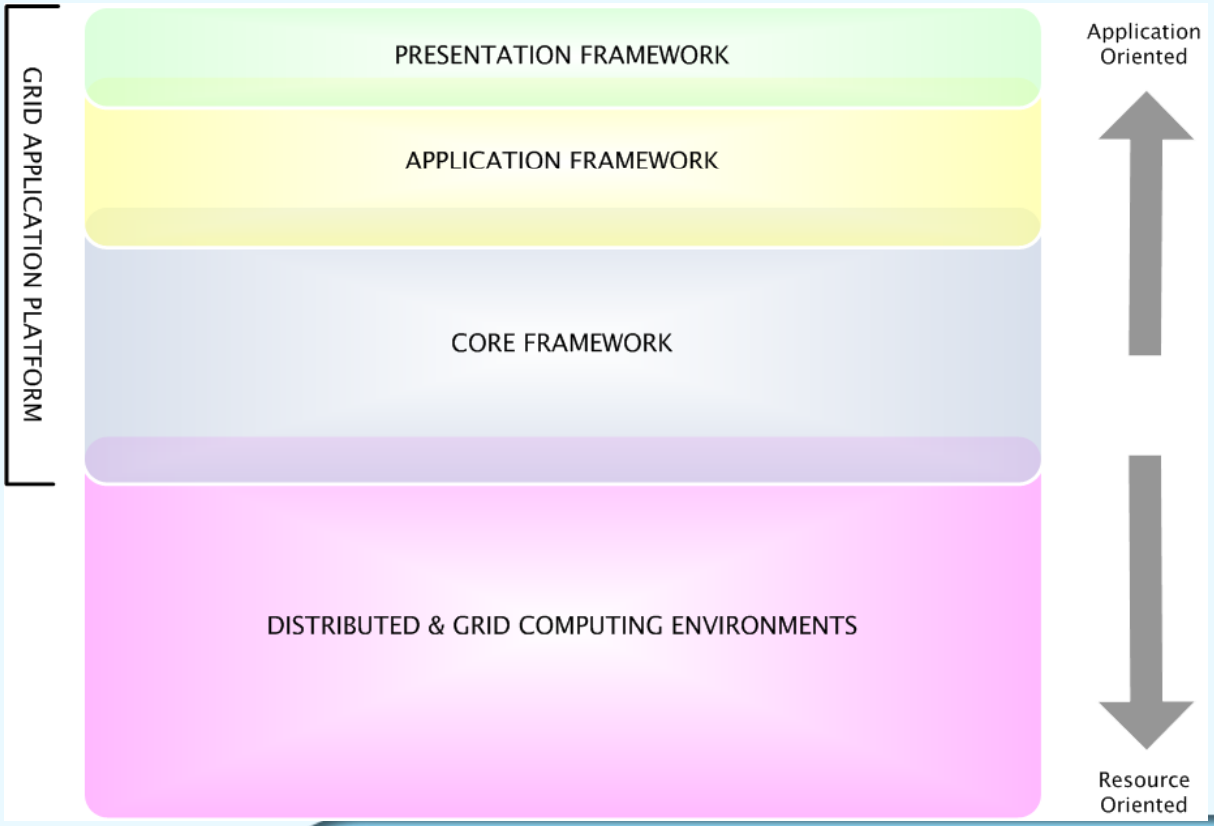
Reduce the effort of adapting new technologies

Concentrate efforts on applications

Re-usable interface components

High-level application logic

Interfacing computing resources

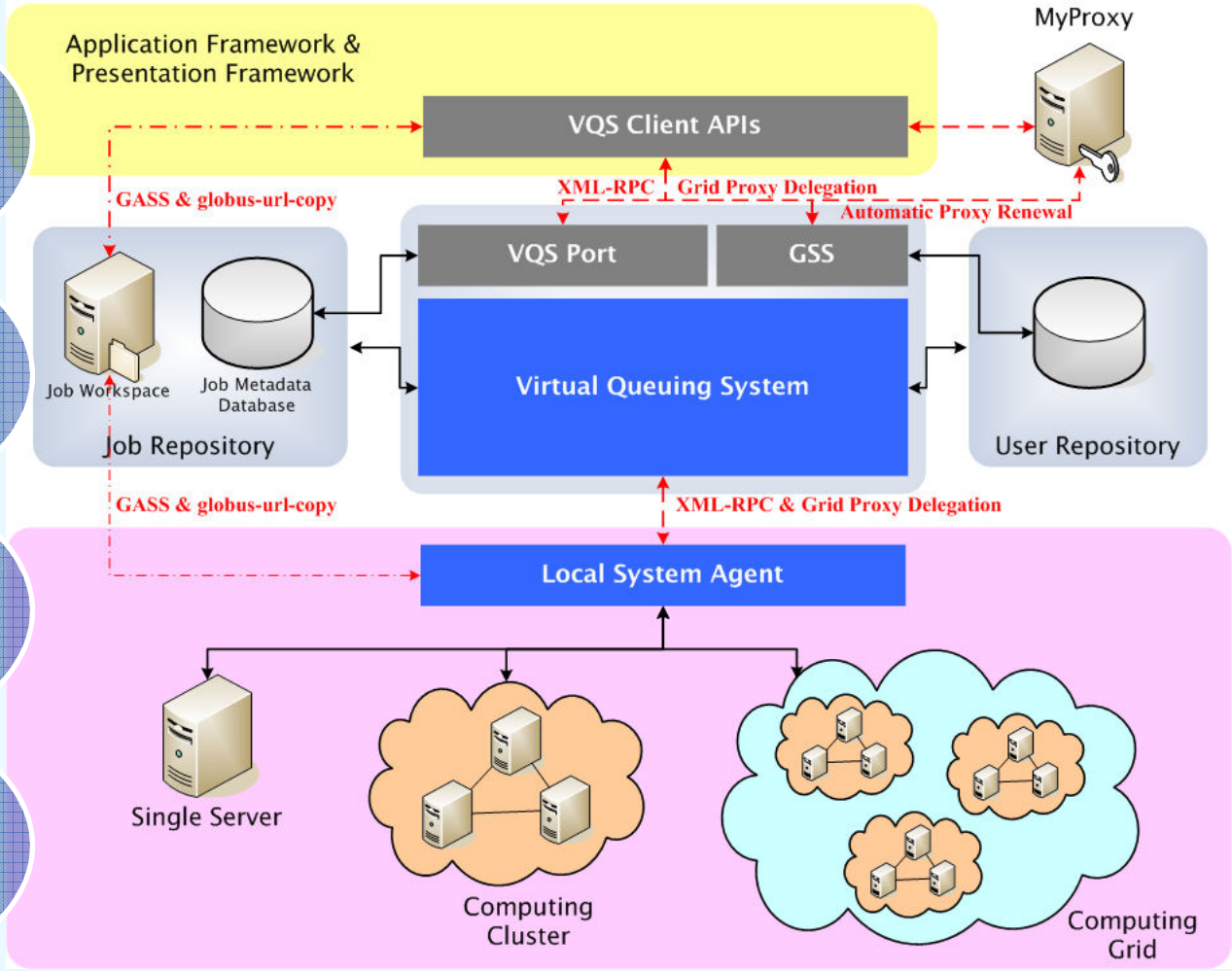


Components

- *Portable application package*: light-weight client-side package for managing jobs and running applications.
- *Virtual Queuing System*: high-level meta-schedule with application specific resource matching.
- *Local System Agent*: uniform interface for adapting heterogeneous computing environments.

The architecture overview

- Portable & light-weight Client
- Common Interface to Heterogeneous Environment
- Multi-user Environment
- Service Oriented Architecture



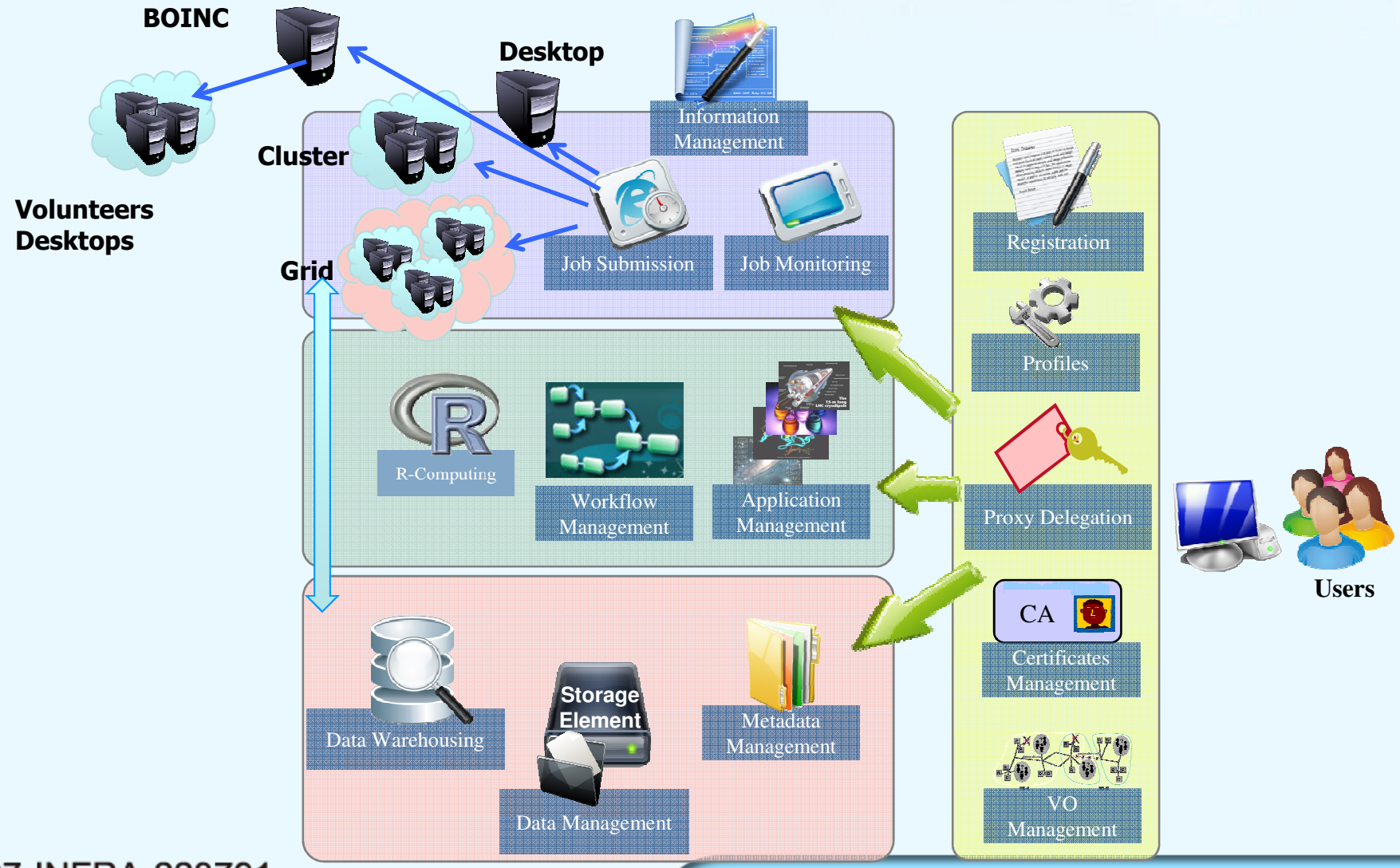
What is portal

- **A portal is a web based application that commonly provides personalization, single sign on, content aggregation from different sources and hosts the presentation layer of Information Systems"(JSR 168).**

What is Grid Portal

- **Grid Portals build upon the familiar Web portal model, such as Yahoo or Amazon, to deliver the benefits of Grid computing to virtual communities of users, providing a single access point to Grid services and resources.**
- **Grid portal is a web server that provides an interface to Grid services, allowing users to submit compute jobs, transfer files, and query Grid information services from a standard web browser. Figure1 shows the structure of portal. A there are four main components of a portal, include portal page, portlet window, portlet fragment and decorations and controls.**

e-Science Application Portal Design



Functions

- Personal Profile
- Personal Certification/VO Management
- Proxy Management
- Data Management (DPM)
- Job Management
- Information Management
- Resource Management
- Application Management
- Workflow Management

Sequence View of Computing Portal

Log in, service selection, parameter definition

Query

Dataset Series available

Data discovery

Available data

Service triggering

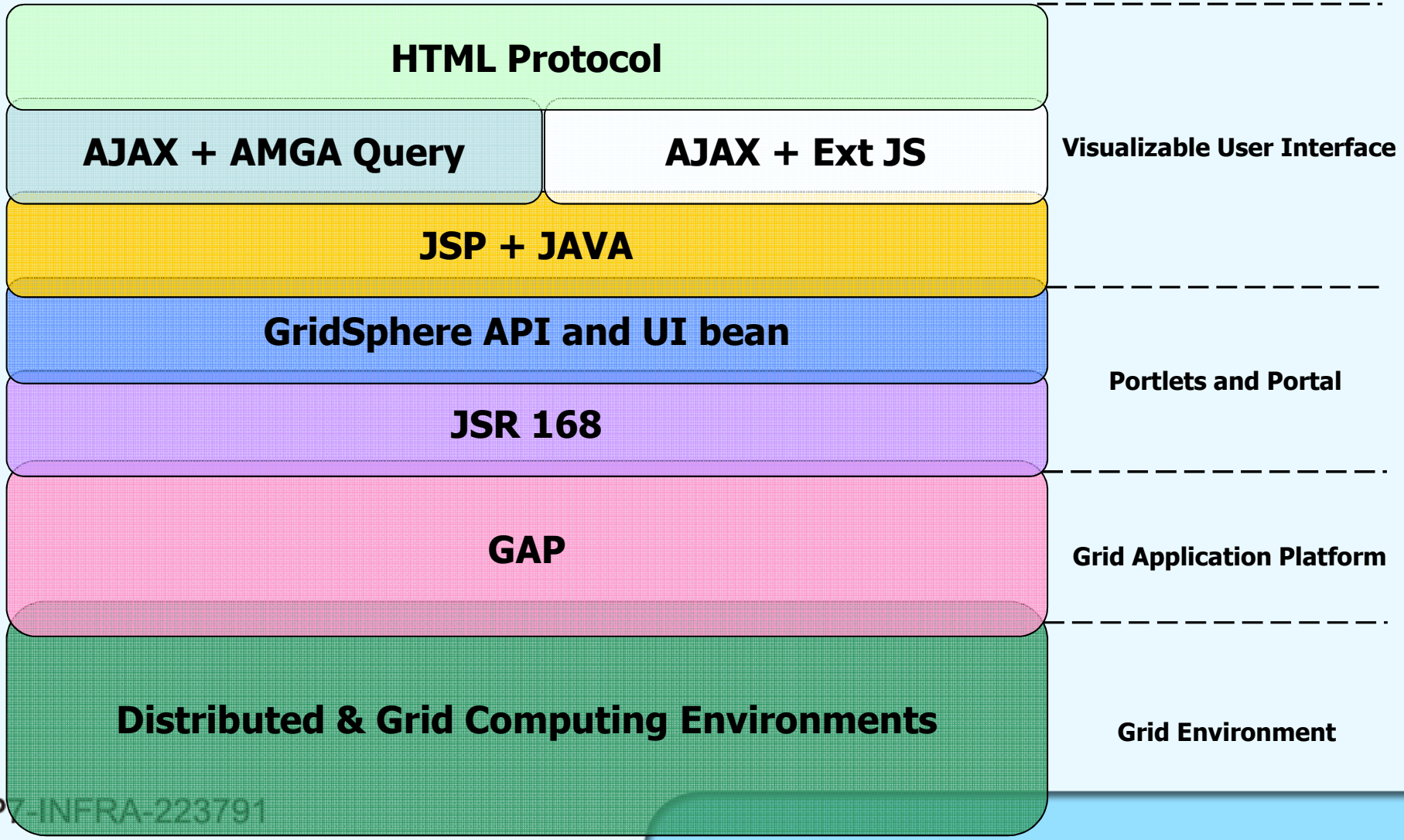
Send processing

Selected data

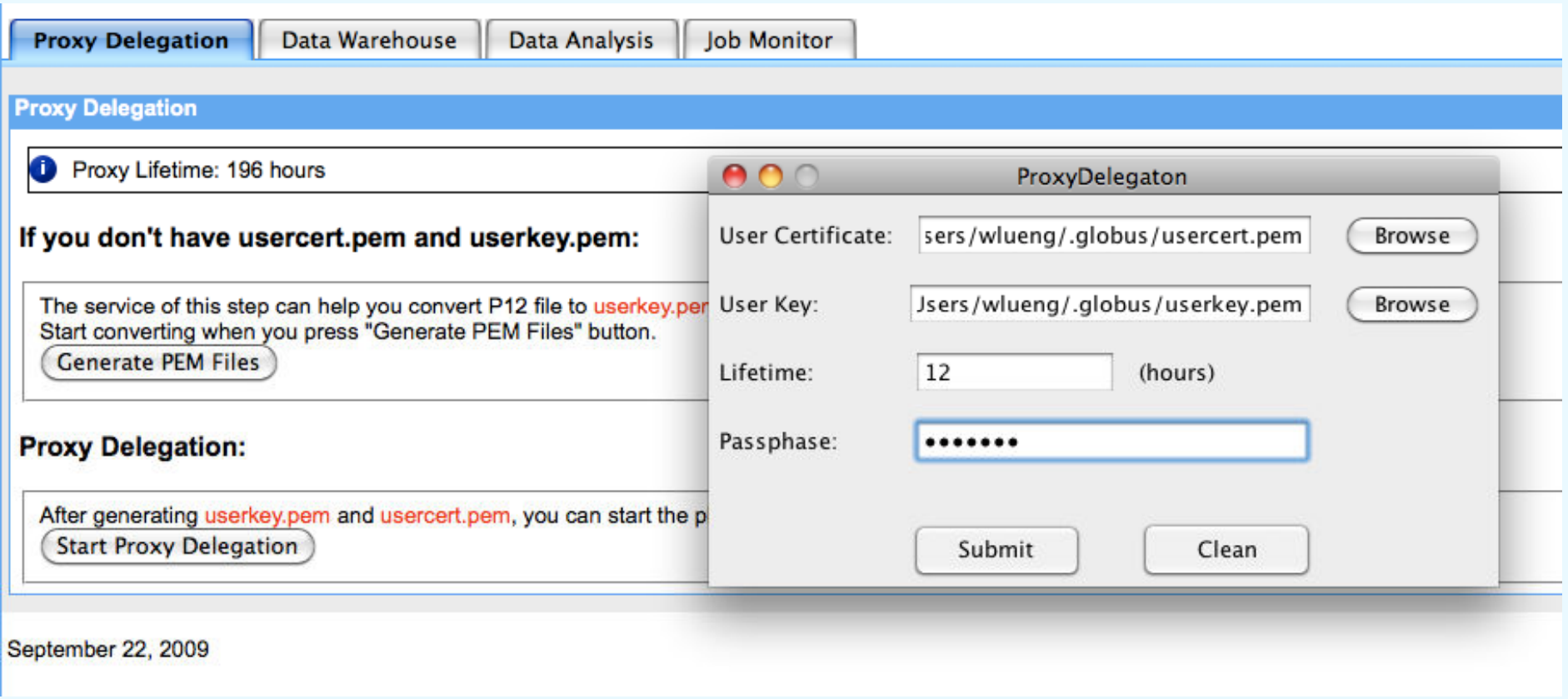
Task status

Results

Technical Stack Overview

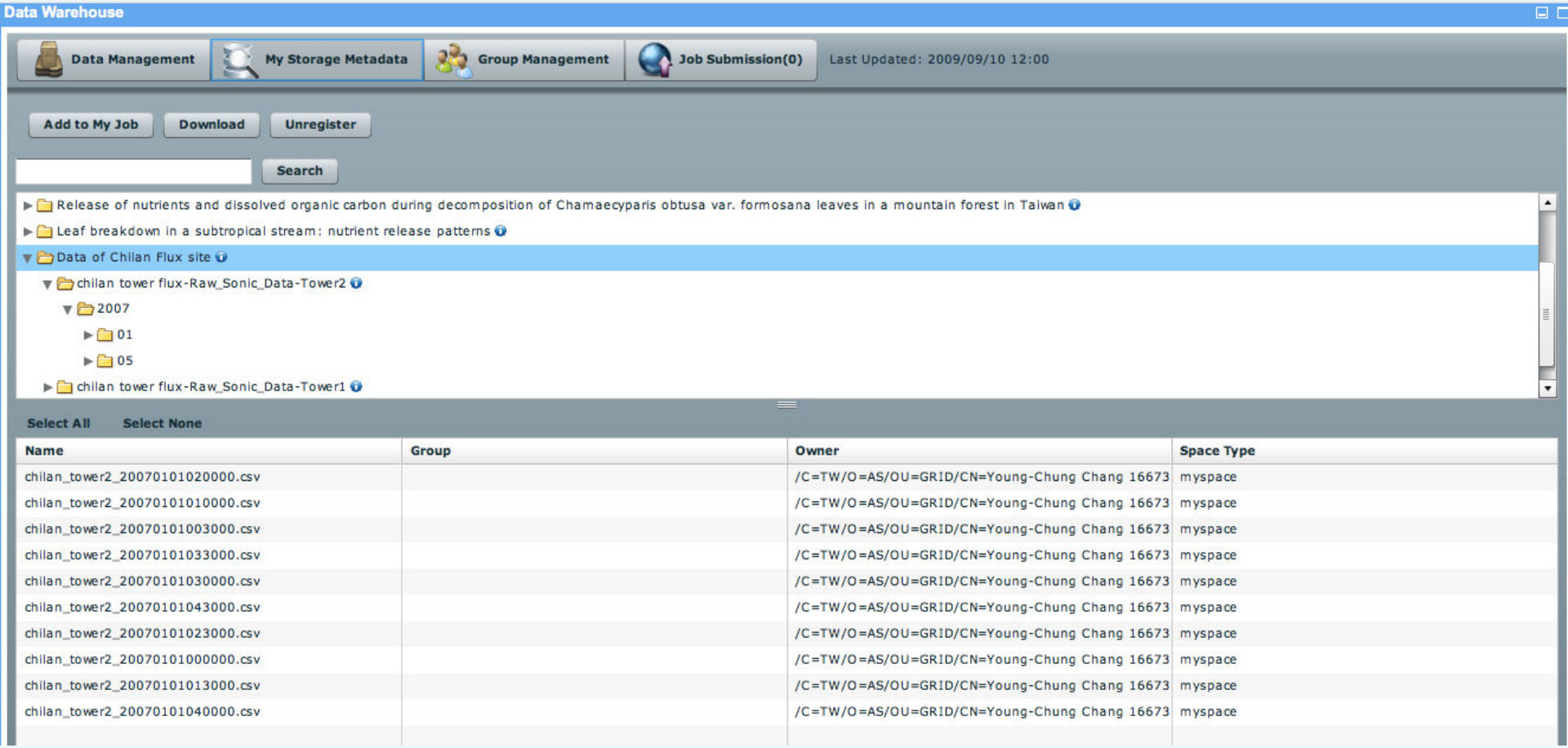


Proxy Delegation



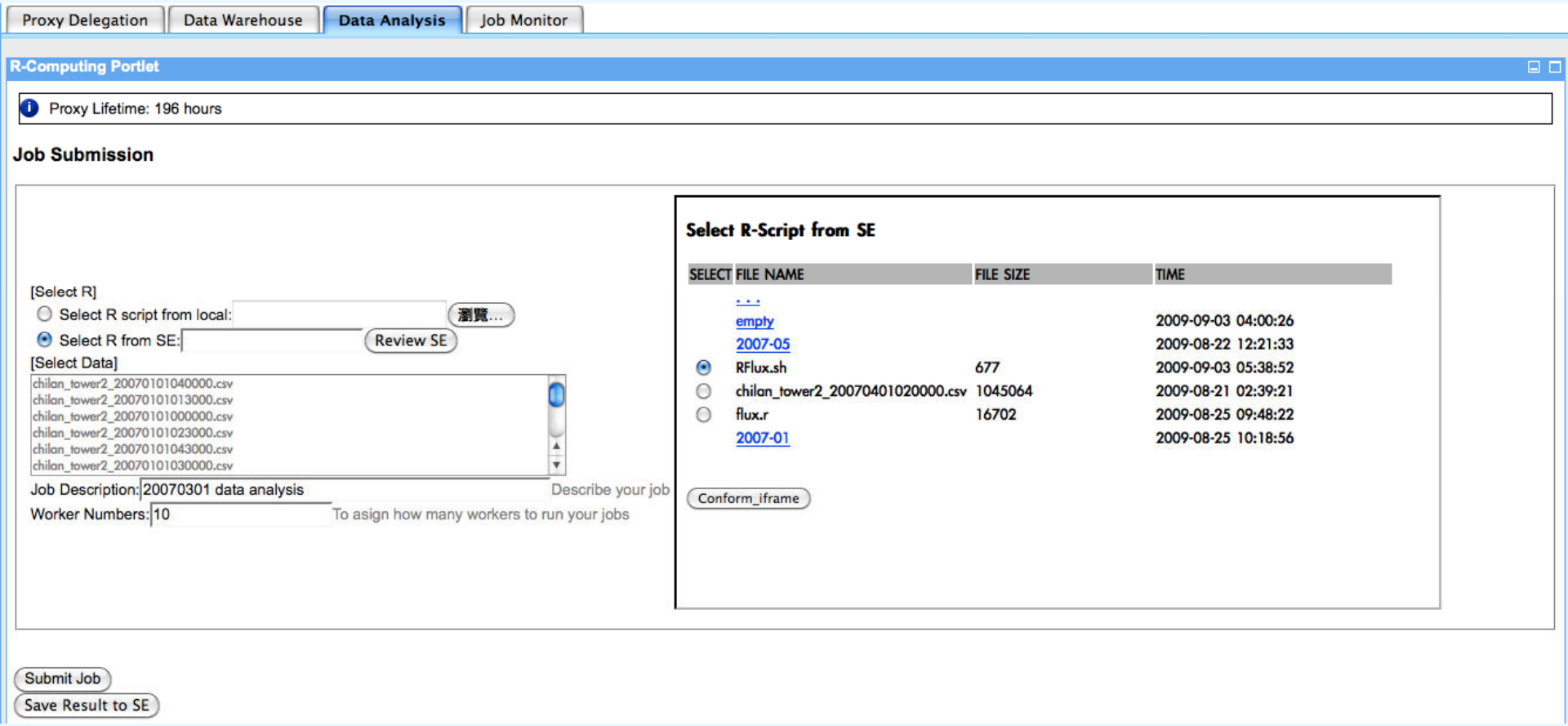
September 22, 2009

Data Management

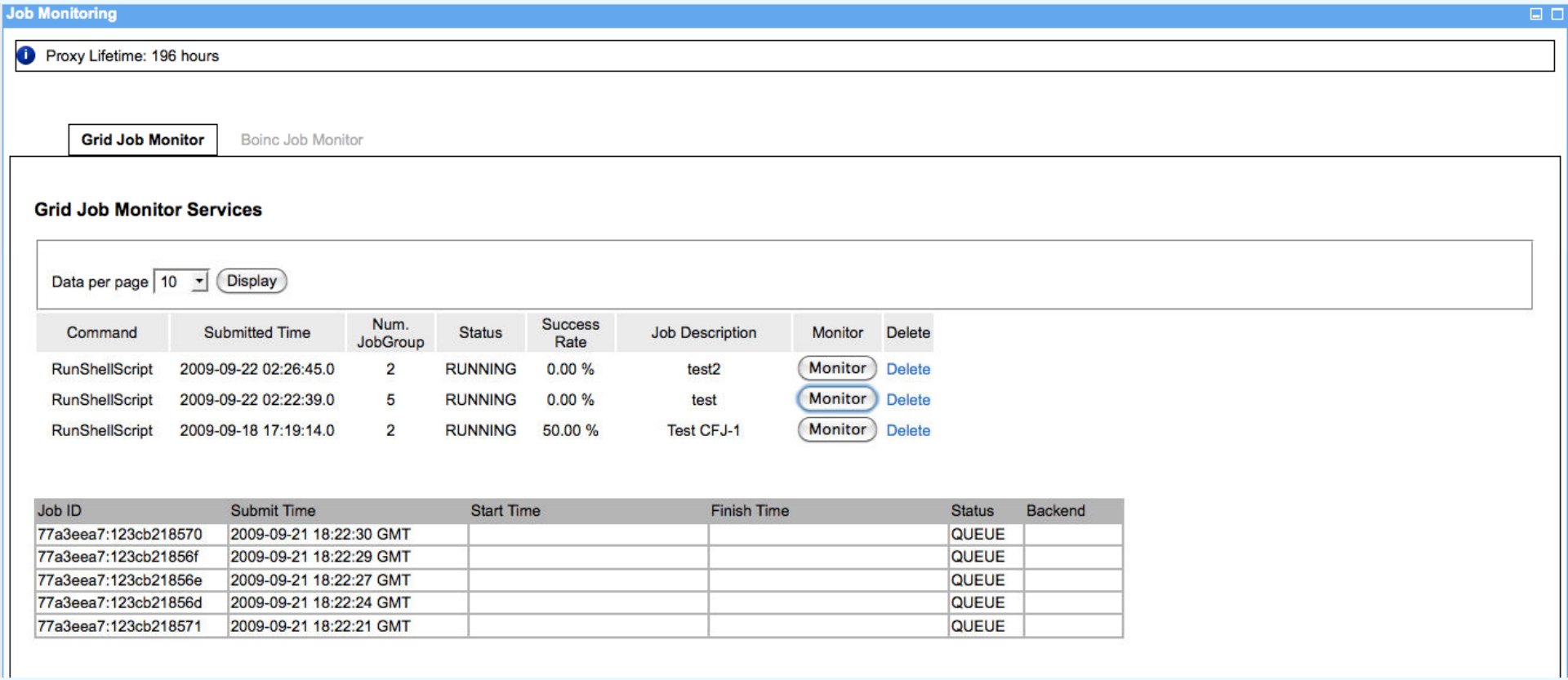


Name	Group	Owner	Space Type
chilan_tower2_20070101020000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101010000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101003000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101033000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101030000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101043000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101023000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101000000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101013000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace
chilan_tower2_20070101040000.csv		/C=TW/O=AS/OU=GRID/CN=Young-Chung Chang 16673	myspace

Job Submission



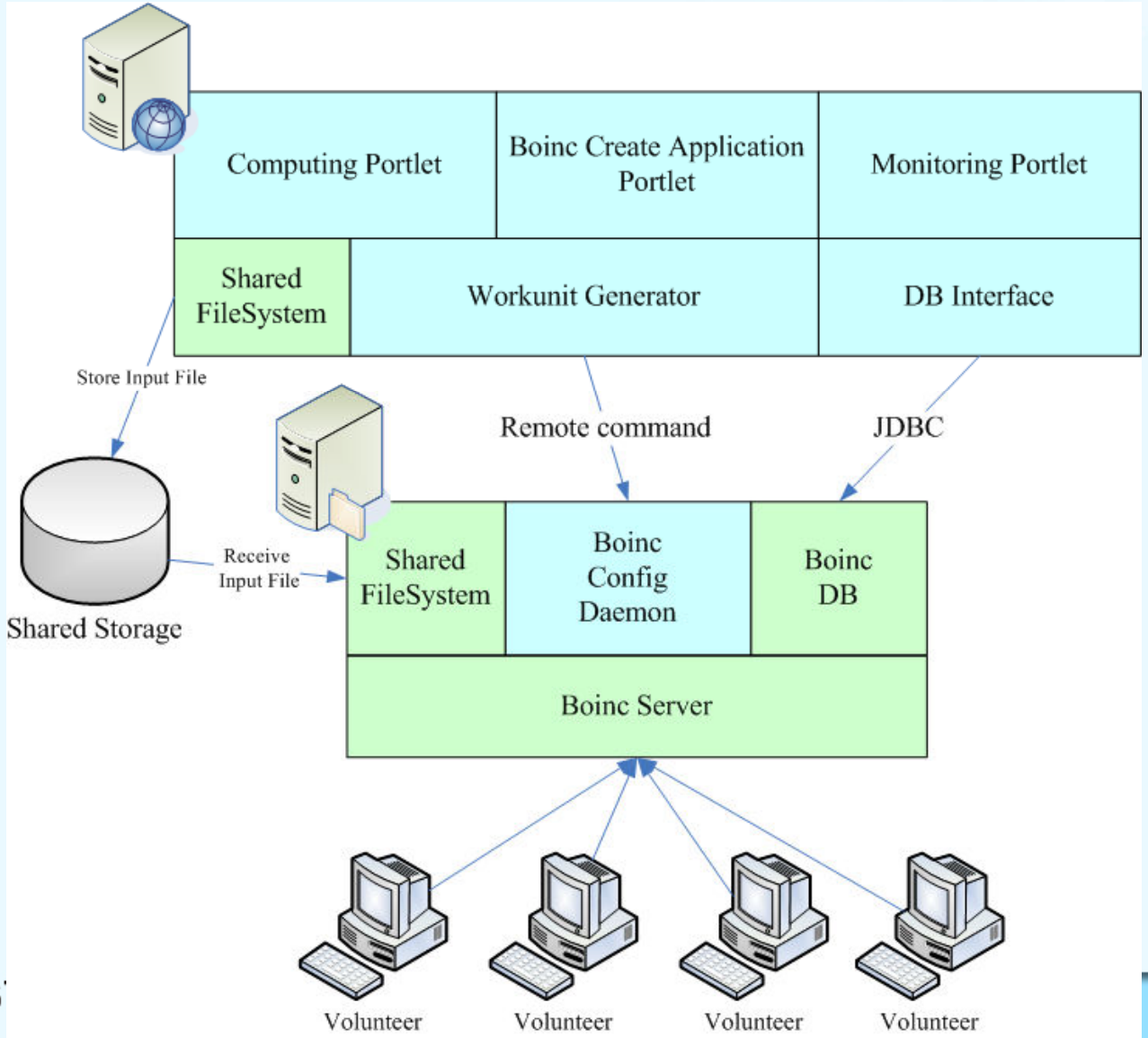
Job Monitoring



Command	Submitted Time	Num. JobGroup	Status	Success Rate	Job Description	Monitor	Delete
RunShellScript	2009-09-22 02:26:45.0	2	RUNNING	0.00 %	test2	Monitor	Delete
RunShellScript	2009-09-22 02:22:39.0	5	RUNNING	0.00 %	test	Monitor	Delete
RunShellScript	2009-09-18 17:19:14.0	2	RUNNING	50.00 %	Test CFJ-1	Monitor	Delete

Job ID	Submit Time	Start Time	Finish Time	Status	Backend
77a3eea7:123cb218570	2009-09-21 18:22:30 GMT			QUEUE	
77a3eea7:123cb21856f	2009-09-21 18:22:29 GMT			QUEUE	
77a3eea7:123cb21856e	2009-09-21 18:22:27 GMT			QUEUE	
77a3eea7:123cb21856d	2009-09-21 18:22:24 GMT			QUEUE	
77a3eea7:123cb218571	2009-09-21 18:22:21 GMT			QUEUE	

Desktop Grid Integation



Conclusions

- GAP was designed by modular approach where re-usable and service-based components as well as portlet frameworks were integrated.
- The result GRID computational portal provides a customizable interface allowing scientists to use a variety of GRID services including job submission, job monitoring, data management, computing pipeline, analysis, and workflow management etc.
- Volunteer computing model and desktop computing services is consolidated for flexible computational application purposes.

Many thanks for your attention