
AstroGrid-D

Deliverable



Requirement analysis for specific components and services of the Astro community for the GACG portal

Deliverable	7.2
Authors	Oliver Wehrens
Editors	
Date	11/28/2006
Document Version	1.0.0
Current Version	1.0.0
Previous Versions	

A: Status of this Document

Draft

B: Reference to project plan

This deliverable refers to the task TA VII-2 and milestone M13 in the project plan.

C: Abstract

This document describes the requirements of the astro community for components and services to be integrated into the GACG portal.

D: Changes History

Version	Date	Name	Brief summary
0.1.0	11/28/2006	O. Wehrens	Working Draft Creation
0.2.0	12/6/2006	O. Wehrens	Updates for Globus Toolkit 4
0.3.0	12/7/2006	O. Wehrens	Updates for Grid Monitoring
0.4.0	12/11/2006	O. Wehrens	Updates for Dynamo
0.5.0	12/13/2006	O. Wehrens	Updates for Task farming
1.0.0	12/24/2006	O. Wehrens	Final

E:

Contents

1	Introduction	4
2	Use cases	5
2.1	Cactus	5
2.1.1	Querying simulation data	5
2.1.2	Comparing simulation data	6
2.1.3	Testing	6
2.2	Task farming jobs	6
3	Infrastructure	8
3.1	Globus Toolkit 4	8
3.2	Grid Monitoring	8
	References	9
	List of Figures	10

1 Introduction

The requirements analysis started in summer/fall 2006. All the partners documented in their use cases that they intent to use a web portal as one of the options to present the interfaces and data to the end user. However for many applications it is not clear yet how to integrate the portal. The community members are still building software and services and in many cases it is too early to have a clear vision how this should be done. Just a few selected application groups have already submitted their requirements on how to integrate their work with the portal.

All work will be based on GridSphere[?]. The GridSphere portal framework provides an open-source portlet based Web portal. GridSphere enables developers to quickly develop and package third-party portlet web applications that can be run and administered within the GridSphere portlet container. GridSphere is the leading scientific grid portal in research institutions around the world.

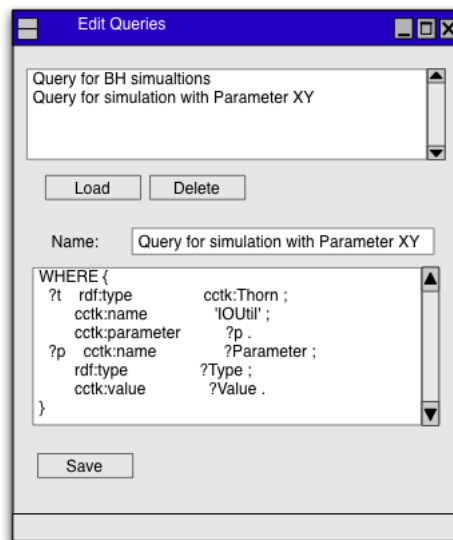


Figure 1: Edit cactus metadata queries

2 Use cases

2.1 Cactus

The use case Cactus[?] integrates some part of their work with the information service developed by WG 2[?]. Metadata are send directly from their simulations to the service to be stored. These data include static information like host, cpu time and date. Furthermore the user can define their own data to be send.

This data is send to the information service and stored on that machine. This service is using RDF to communicate with clients. To present this data to the users so they can take advantage of it (e.g. profiling applications and sharing data between collaborators) web interfaces are required.

2.1.1 Querying simulation data

The first requirements is to build an interface to query the collected data in a very flexible way. Users should be able to

- create
- update
- delete

their queries (see figure ??). These queries should be persistent on a per users basis. It should be possible to query data from colleagues to foster collaboration.

The results should be presented in table format providing all the queried information (see figure ??).

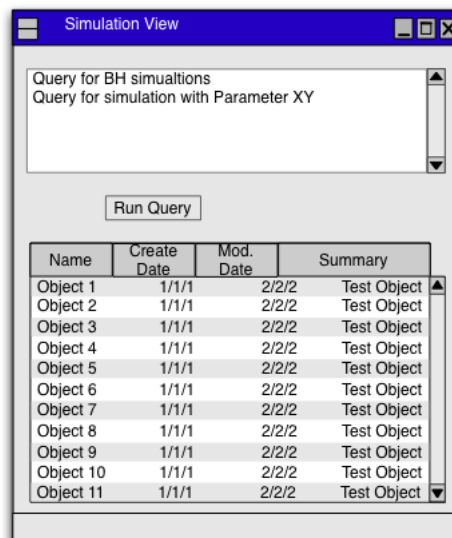


Figure 2: Cactus simulation view

2.1.2 Comparing simulation data

An important feature is announcing the cactus configuration to the RDF server. To take advantage of the transmitted data it is desirable to compare the configurations in different ways. It should be possible to use two or more configuration files from the RDF server to compare available cactus thorns in them and present the result in a way so the scientist can see very quickly what the differences are. The more advanced use case is to compare the cactus parameter files and their values. Differences between the files should be highlighted (see figure ??).

2.1.3 Testing

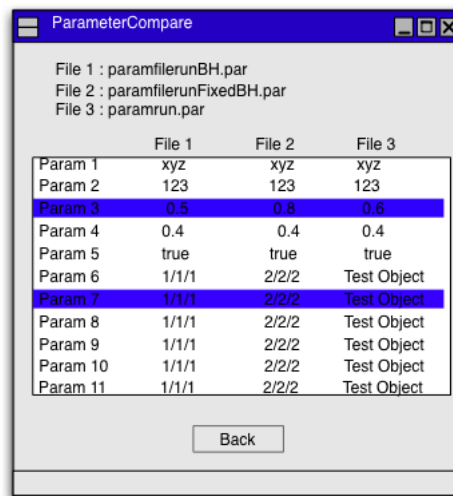
It should be possible to show the list of Cactus integration test results as obtained from corresponding metadata queries, clearly structured by different user-selectable views on the metadata, eg.

- an overview of all recent integration tests
- a detailed view of all metadata for a specific integration test
- derived from a specific Cactus testsuite, a history of that across all recent integration tests

2.2 Task farming jobs

A general interface should be developed to make it easier to run task farming jobs from a portal environment. This will satisfy use cases like Dynamo and Amiga.

The following requirements were given:



	File 1	File 2	File 3
Param 1	xyz	xyz	xyz
Param 2	123	123	123
Param 3	0.5	0.8	0.6
Param 4	0.4	0.4	0.4
Param 5	true	true	true
Param 6	1/1/1	2/2/2	Test Object
Param 7	1/1/1	2/2/2	Test Object
Param 8	1/1/1	2/2/2	Test Object
Param 9	1/1/1	2/2/2	Test Object
Param 10	1/1/1	2/2/2	Test Object
Param 11	1/1/1	2/2/2	Test Object

Figure 3: Cactus compare parameters

- Information about the resources available should be shown (an extract from AGD-IS containing: CPU-Speed, Size of RAM, Diskspace, 15 min Load as a list)
- Information about the service availability should be displayed
- It should be possible to select from this list of available hosts the one for
 - job execution (multiple hosts)
 - host containing the start files
 - host which should take the results
- The user should be able to specify the directory which contains the start files and specifying the name of the directory, which will contain the results. The job can directly be executed from the portlet.
- Monitor-Information: The job is logging some data to a file as well as to standard out, this information should be shown in the portlet.
- The portlet should contain as well a (sortable) list of executed and running jobs, i.e a history

3 Infrastructure

3.1 Globus Toolkit 4

All deployed grid installations facilitate Globus Toolkit 4. At the moment the standard deployment of GridSphere includes Globus 2.x. This works with Globus 4 but does not use the web services approach. To ensure compatibility with the newer standard it is required that the portal will support Globus 4 as standard deployment. This is mostly relevant to any workpackage using job submission to globus installations in the Astrogrid-D community.

3.2 Grid Monitoring

As a general service to the AstroGrid-D Community a portlet should be developed to monitor

- the availability of all default grid services (GSIftp, GSIssh, GSIscp, GRAM, gatekeeper, MDS)
- the regular update of gridmap files and revocation lists for certificates
- for central services running on special hosts (AstroGrid-D-MDS, AstroGridD-IS) their availability

to ensure a healthy and reliable grid infrastructure. This is only the first stage, later we want to include more and detailed information about the performance of these services and resources. This development will take advantage of the AstroGrid-D Information service.

F: References / Bibliography**References**

- [1] Cactus Computational Toolkit
<http://www.cactuscode.org>
- [2] GridSphere Portal Framework
<http://www.gridsphere.org>
- [3] Working Group 2 of Astrogrid
<http://www.gac-grid.de/project-overview/workpackages/wp2.html>

G: List of Figures**List of Figures**

1	Edit cactus metadata queries	5
2	Cactus simulation view	6
3	Cactus compare paramaters	7