School of Engineering & Design

Electronic & Computer Engineering

MSc in Data Communications Systems

Brunel UNIVERSITY WEST LONDON

Grid Monitoring

Theofylaktos Papapanagiotou

Dr.Paul Kyberd

March 2011

A Dissertation submitted in partial fulfillment of the requirements for the degree of Master of Science

School of Engineering & Design

Electronic & Computer Engineering

MSc in Data Communications Systems

Grid Monitoring

Student's name: Theofylaktos Papapanagiotou

Signature of student:

Declaration: I have read and I understand the MSc dissertation guidelines on plagiarism and cheating, and I certify that this submission fully complies with these guidelines.

Abstract

Develop an interface which allows the customisable aggregation and display and information on the performance of a computational grid.

In EGI era of grid computing in Europe, MyEGEE and Nagios are taking the place of SAM in performance monitoring of the grid, in NGI oriented infrastructures. SAM Framework had a significant role in reporting the service availability. That monitoring model was needed to be replaced by a new Multi Level Monitoring architecture. The need of establishing a central point in regional level, where metrics gathered from each information system of the grid, lead to the adoption of MyOSG. MyEGEE is the tool that extends MyOSG to european NGI's and provides an interface to customize the display of aggregated metrics of the performance of NGI sites. Nagios is the main technological choice to provide that regional monitoring system.

Contents

List of Figures

Chapter 1

Introduction

1.1 Context

Performance monitoring of a grid is a key part in grid computing. Based on the reports of grid performance, decisions on capacity planning are being made. Customizable visualization of performance status in different levels helps scientists and managers focus on the exact point of the infrastructure where a bottleneck on service exists. Current interfaces delivers performance graphs without following the standard topology schema that is presented by the grid information system.

ATP standard and GLUE schema are examined to understand the gathering process of metrics. Ganglia's hierarchical delegation to create managable monitoring domains is an important aspect. MyEGI which is based in Django python framework, access easily the unified metrics database. Performance in the aspect of how many jobs are served by each site is not examined in this project. Whether it is possible to integrate Ganglia performance graphs in MyEGI and Nagios interfaces, using the standard Information System of ATP.

Build a lab to gather performance data and start working on the development of the integration parts. It is assumed that the environment is a grid site, that already have the components needed to work together. Ganglia daemons on each node, presented by the GLUE schema on site BDII, Nagios/MyEGI monitoring frameworks. A web interface is available to present the work of the integration of Ganglia into Nagios/MyEGI.

1.2 Aims & Objectives

Different role users are going to use a portal to get information about the performance status of the grid, to export the appropriate report for their job. This project aims to develop these particular pieces of code to support the aggregation of the metrics from nagios, to allow the web based customization of the visualization of the reports. These metrics are needed to report the availability and reliability of NGIs and particular sites of the grid.

The procedures that are going to be used in order to achieve the above aims should include at the beggining some opening and exploration of the environment where the interface is going to be placed. The usage of grid computing in the world should be well known, so a visibility of the importance and the possible uses of the software will be recognised. The appropriate access to the infrastructure should be gained, on different platforms and levels. Brunel University site and GridPP/NGS VO at the beggining, as long as the UKI ROC operations may be a good point of collaboration with researchers to reach the bests possible requirements and data to analyse. The middleware used in both these VOs should be examined so with the knowlege of running projects and global usage of them may target to export better specifications. Existing operations on the grid should also be discovered. The european initiative milestones on the operations of the regional level should be considered as a route, and registration to news about the upcoming research projects that are going to use the grid should also be take place.

After that wide-opening to get the whole picture, a targeted and focused view should follow. Existing monitoring tools must be used to check the problems and search for requirements. The experience of SAM, Gridview, Gridmap, Gstat, GridICE, etc should be taken in order to merge their functionality as possible as it is. Information systems that already reside over the infrastructure, must also be learned. Standards and specifications should be examined, on how the message bus works and delivers the data in an hierarhical manner. A contact with the CERN team working on MyEGEE and Indiana University's MyOSG team should be established, to collaborate on the core of MyOSG source. Changes submission to subversion system as long as ticket closure of the development project tool will help to get to know the core of MyEGEE and Nagios. It is possible to create and upstream a nagios customized web interface, to create different views of nagios resources scheme to grid topology oriented architecture. Nagios, NRPE and Ganglia installations should be deployed across the CE&SE nodes of Brunel's sites to have a working production environment to work on. Attention should be taken on the potential performance impact of these sensors deployment. UKI MyEGEE validation/testing portal will be used as a pre-production environment to check changes. PNP should be fixed in GridPP Nagios to be evaluated. Statistical access log analysis of existing tools may have results on trends of users/admins prefered views.

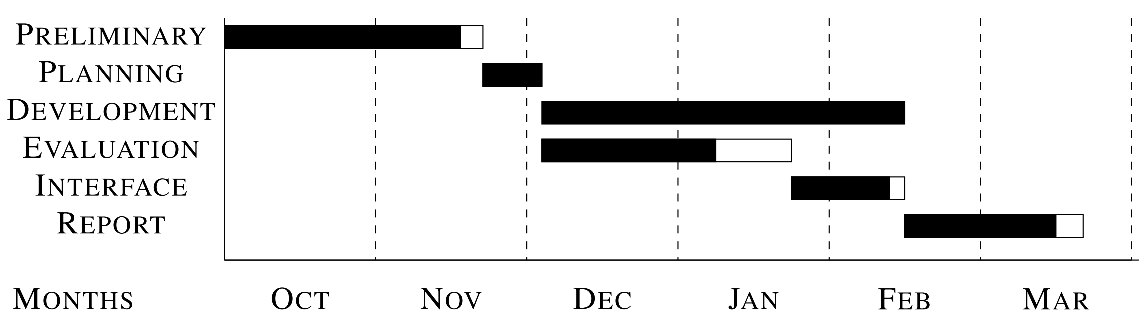
Various tools are going to be used to track changes and collaborate. Monitoring articles in GridPP wiki & CERN twiki should be made. Snippets upstream & status changes must be a regular operation in SVN/JIRA/Trac in CERN interfaces. Ongoing task through the disseration project is the reading of papers and methodical updates of Mendeley citation management tool to have the bibliography organized. Possible changes suggestions to MSc on DCS cource notes about grid monitoring may by made, as long as the EGI roadmap updates. Finally with the appropriate supervision and follow-up of meetings and presentations, a paper publishing might take place.

1.3 Organization

1.3.1 Tools

This project was developed in LaTeXusing Eclipse and TeXlipse plugin. Its releases may be found in Google Code, where Mercurial was used for source control. Citation management through Mendeley software. Papers obtained by becoming member of IEEE, ACM and USENIX. Operating Systems Laboratory of Technological Education Institute of Piraeus was used to build a testbed of grid site and tools to study existing monitoring tools.

1.3.2 Time-plan (Gantt Chart)



Task	Start date	End date	Duration in days
Preliminary	09/29/10	10/24/10	20
- Identify Concepts	09/29/10	10/08/10	8
- Gain Access	10/08/10	10/24/10	12
Planning	11/12/10	12/04/10	17
- Explore existing technologies	11/12/10	11/28/10	12
- Write Interim Report	11/28/10	12/04/10	5
Experimental-Development	12/04/10	02/14/11	51
- Evaluate performance monitoring tools	12/04/10	12/25/10	15
- Information/topology databases	12/17/10	12/29/10	8
- Develop Customized Interface	12/29/10	02/14/11	34
— Coding of information aggregation	12/29/10	01/21/11	16
— Development of the frontend	01/21/11	02/10/11	14
— Complete the interface (auth, scale, etc)	02/10/11	02/14/11	4
Report	02/16/11	03/29/11	32
- Begin Writing	02/17/11	03/01/11	11
- Submit Draft & Make Changes	03/01/11	15/14/11	9
- Prepare Final	03/14/11	03/29/11	11

Table 1.1: Key activities necessary to complete the project

Chapter 2

Literature Review

2.1 Grid Computing

Grid computing [17] is the most recent decade's technology innovation in high performance computing. A large number of scientistcs working on the operations of this huge co-operative project of EU. Monitoring & information architecture [9] has been standardized in the initial state of that project, to succeed in today scale of 150.000 cores in production. Use of grid computing nowadays takes place in academic and research environments. Also, applications in industry-based needs such as promising Power Grid control [20] are emerging.

Cloud

stuff

here

2.2 Resource Brokers

Resource Brokers [16] where developed to manage the workload on Computer elements and Resource elements. Globus is a non-service based RB, and gLite RB which is service based. A Workload Management System (WMS) exists in gLite to do the distribution and management of the Computing and Storage oriented tasks.

Based on the middleware that resource brokers rely on, they use the equivalent information system. From resource broker's point of view, the relevant information is the data store and

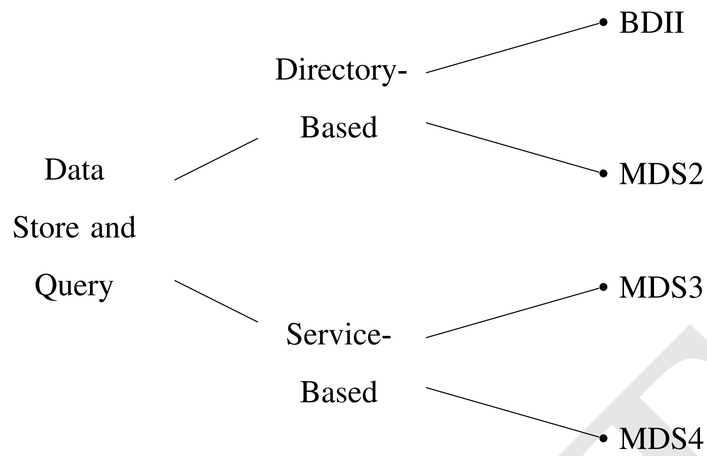


Figure 2.1: Grid Resource Brokers grouped by Information Systems[16]

query. There are two main categories of information systems in middlewares. The Directory-based and the Service-based. They are used for resource mapping by the brokers when they access the resource data.

2.2.1 Globus

Globus Toolkit is an open source toolkit used to build grids. It provides standards such as OGSA, OGSI, WSRF and GSI, and the implementations to OGF protocols such as MDS and GRAM.

Monitoring and Discovery Service (MDS) is part of Globus Toolkit. It provides the information for the availability and status of grid resources.

extra

extra

read all about it

2.2.2 gLite

gLite is a middleware which was created to be used in the operation of the experiment LHC in CERN. The user community is grouped in Virtual Organizations, and the security model is GSI. A grid using gLite consists of User Interface, Computer Element, Storage Element, Workload

Management System and Information Service.

The information service in version 3.1 of gLite is similar to MDS of Globus middleware, except that the GRIS and GIIS are provided by BDII (see Section BDII) which is an LDAP based service.

2.3 Information Services

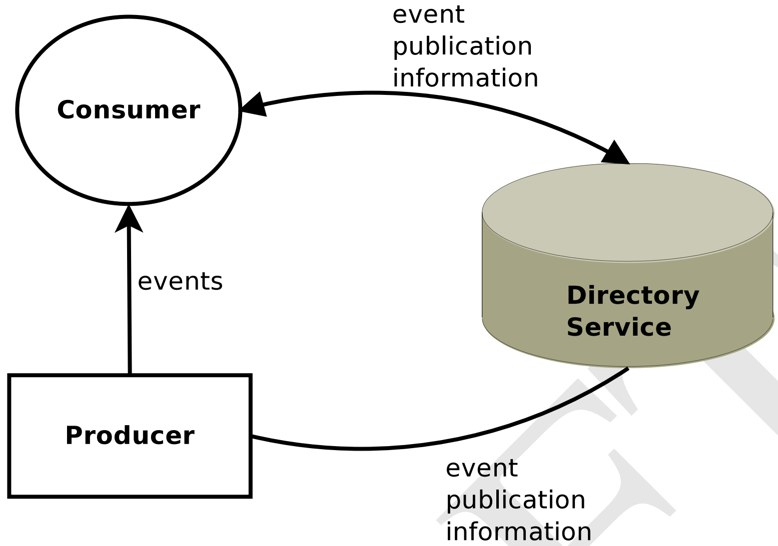


Figure 2.2: Grid Monitoring Architecture

A Grid Monitoring Architecture [21] was proposed in early 2000's. Information systems were developed to create repositories of information needed to be stored for monitoring and statistical reporting reasons. Such an organized system later was specified by the Aggregated Topology Provider (ATP) definition. The largest world grids adopt that model, forming OIM in OSG (USA) and GOCDB as that information base in EGEE (Europe). Message Bus was also defined as a mean to tranfer the undelying data, and well known tools came up such as Gstat, GOCDB and BDII with Glue specification. Grid performance monitoring and keeping of such an information system has also impact in the performance of the system itshelf [26], so various methods were developed to give the solution to the scalling and performance problem, such as MDS2 (GIIS & GRIS), GMA and R-GMA [24], which offers relational environment [8], has experience on production systems [5] and scales to reach huge needs such as CMS project [3, 6].

2.3.1 MDS

Monitoring and Discovery Services is about collecting, distributing, indexing and archiving information of the status of resources, services and configurations. The collected information is used to detect new services and resources, or to monitor the state of a system.

Globus Toolkit was using LDAP-based implementation for its information system since its early versions, back in 1998 [23]. MDS2 in Globus Toolkit fully implemented referral with a combined GRIS and GIIS, using mds-vo-name=local to refer to the GRIS and all other strings to refer to a GIIS. It was widely accepted as a standard implementation of a grid information system [7], with good scalability and performance [27].

MDS 4 consists of the Web Services Resource Framework and a web service data browser, WebMDS. The WSRF Aggregator Framework includes:

1. MDS-Index, which provides a collection of services monitoring information and an interface to query such information.

2. MDS-Trigger, which provides a mechanism to take action on collected information.

3. MDS-Archive, is planned for future release of MDS, to provide access to archived data of monitoring information.

External software components that are used to collect information (such as Ganglia)[18] are called Information Providers.

2.3.2 Glue

As long as Information Services are used to connect different infrastructures, the schema of its structure had to be standardised. To interoperate EU and USA grids, DataTAG developed the GLUE schema implementation. GLUE specification quickly adopted by the communities and currently its recommended LDAP DIT is specified in GLUE specification v.2.0 from GLUE Working Group of OSG.

Many objectclasses of the Glue schema define a Computer Element, a Storage Element, etc. As seen in Figure 3.1 in later chapter, performance monitoring attributes such as processor load are defined in objectclasses that extends Computer Element objectclass.

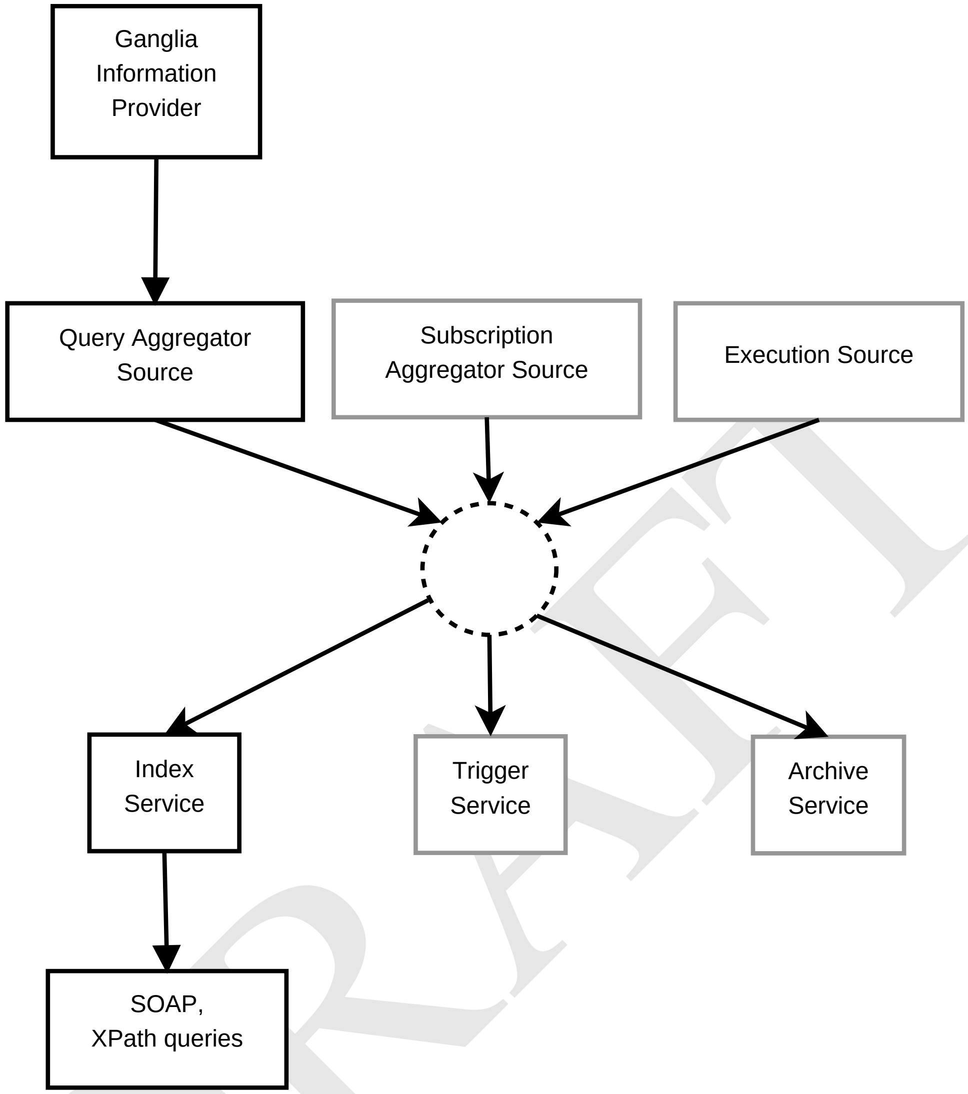


Figure 2.3: Web Service Resource Framework

2.3.3 BDII

BDII is used by gLite as the Information Index Service of the LHC experiment. It is LDAP based and may be at top-level or site-level. The GIIS has been replaced by site BDII, which is fundamental for a site in order to be visible in the grid.

Top-level BDII contains aggregated information about the sites and the services they provide. Site BDII collects the information from its Computer Elements, Storage Elements, etc as long as every configured service that is installed on the site.

Information about the status of a service and its parameters is pushed on BDII using external processes. An information provider is also used (such as in WSRF) to describe the service attributes using the GLUE schema.

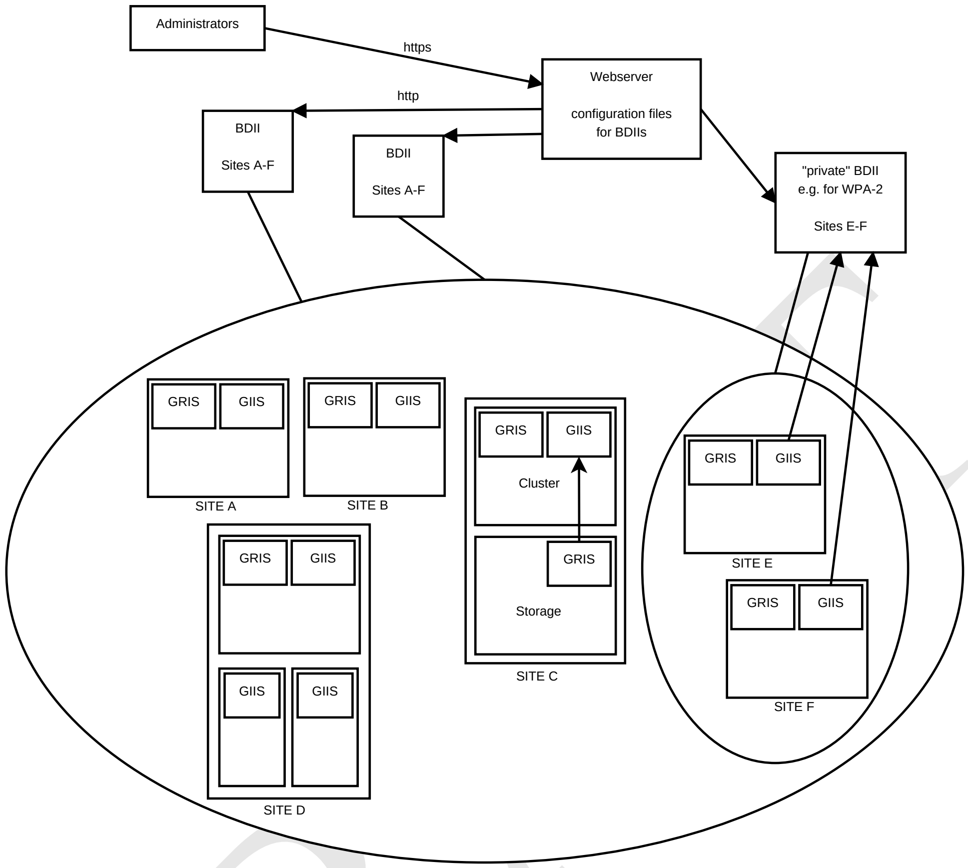


Figure 2.4: Berkeley Database Information Index

2.4 Performance Monitoring

Standards are being published about the operational models that the grid computing initiative will use. Last decade the EGEE I, II & III was adopted by european universities to fund and establish a collaborative community of researchers under a central point, the CERN oriented research project in Particle Physics. After EGEE, the European Grid Initiative were formed to lead to the explode of that community into regional initiatives. Performance and availability monitoring tools and views also follow that format, phasing out commonly used SAM [19] and having the adoption of Nagios as the monitoring of regional performance tool.

A taxonomy effort has been made [11] to present the differencies of performance monitoring systems of the grid, and later a more general [25] taxonomy paper was published to give a more general visibility of these tools. GridICE was generaly used to aggregate the performance metrics of ROCs in high level reports [1]. Later GridICE was left as long as the SAM left, to meet the milestone of EGI to have a regional monitoring tool (Nagios) to report the reliability of the joined sites and report the values for SLA reasons.

2.4.1 Ganglia

2.5 European Grid Infrastructure

Latest EGI directive to form regional operation tools pushed the use of Nagios [15] as the main tools of availability & performance (an so reliability) monitoring of the grid. Each NGI/ROC (regional level) has its own interface, and hierarhicaly there is a Super Nagios interface to report the top level view of general system availability. Nagios offers extensions such as NRPE to remotely invoke check commands in inaccesible/private installations. Another important add-on to Nagios is the NdoUtils, which offers an SQL store of history data to the monitoring interface. Nagios Configuration Generator was introduced to help the automaticaly generation of the configuration based on the information system of nodes and services. Finally, there has been proposed an integration of SAM views to a Nagios customized interface, to offer the last good known SAM interface to the old users. Nagios also integrates with GGUS, a ticketing system that european grid initiative uses.

By the end of the project the monitoring infrastructure (regional Nagios servers and the corresponding regional MyEGEE portals [R52], [R53]) will complete the transition process to a fully distributed infrastructure. The Nagios-based system will totally replace the Service Availability Monitoring (SAM) infrastructure. At the time of writing two monitoring environments are operated: a production one and a test one.

2.5.1 NGS

Brunel Univesity takes part in regional and european initiatives. 4 different Computer Elements exist, and 3 Storage Elements, consisting the UKI-LT2-Brunel site. LT2 stands for London Grid, a co-operation with other London Universities. GridPP and NGS are two collaboration groups that Brunel Univesity is member of, and papers on the web interface [13] and real time visualization of the grid status were presented [14] by GridPP

Chapter 3

Design/Methods

3.1 Approach Adopted

edw

to keimeno

pou tha leei

panw katw

ti ginetai

s'auto to kefalaio

DHLADH THA ESTIAZEI STO WS 'H LDAP

3.1.1 implementation of performance analysis of distributes systems in GMA

na perigrapsw to paper: [2] requirements

very large volume of data

measurement data must be accurate and consistent

problems

time synchronisation

time measurments must be accurate

time measurements must be consistent.

implementation

consumer registers interest

trace events generation

producers query IS for consumer

producers subscribe to consumers

consumer get events

3.2 Design Methods

3.2.1 Recommendations and standards

Grid Monitoring Architecture

R-GMA

XML schema

GLUE

gained wide acceptance given its adoption by Globus MDS3

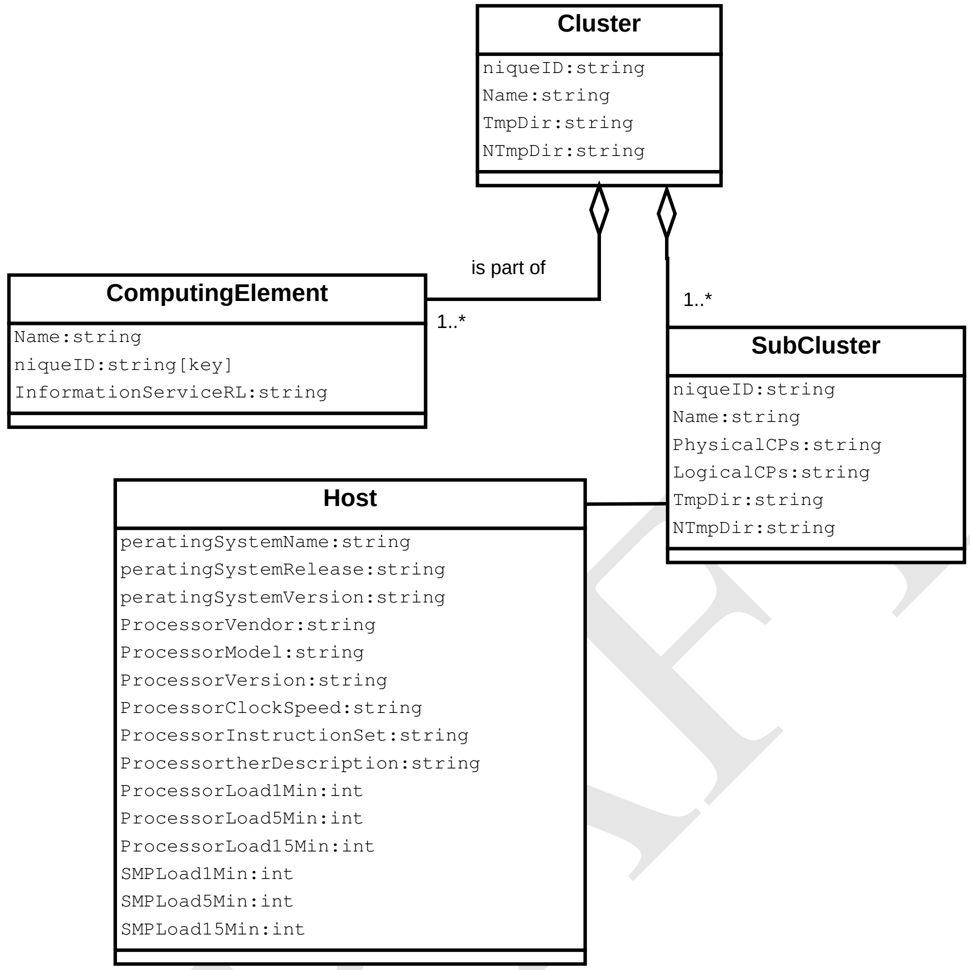


Figure 3.1: GLUE schema 2.0 extention for Host and SMP Load

3.2.2 Information Infrastructure

“Performance”. The applications of interest to us frequently operate on a large scale (e.g., hundreds of proces- sors) and have demanding performance requirements. Hence, an information infrastructure must permit rapid access to frequently used configuration information. It is not acceptable to contact a server for every item: caching is required. “Scalability and cost”. The infrastructure must scale to large numbers of components and permit concurrent access by many entities. At the same time, its organization must permit easy discovery of information. The human and resource costs (CPU cycles, disk space, network bandwidth) of creating and maintaining information must also be low, both at individual sites and in total. “Uniformity”. Our goal is to simplify the development of tools and applications that use data to guide configuration decisions. We require a uniform data model as well as an application programming interface (MI) for common operations on the data represented via that model. One aspect of this uniformity is a standard representation for data about common resources, such as processors and networks. “The X.500 standard” defines a directory service that can be used to provide extensible distributed directory services within a wide area environment. A directory service is a service that provides read-optimized access to general data about entities, such as people, corporations, and computers. X.500 provides a framework that could, in principle, be used to organize the information that is of interest to us. [10]

3.3 Data-acquisition Systems

plugins that take metrics (SAM) and send results to nagios:

https://tomtools.cern.ch/confluence/display/SAMDOC/Grid+probes

http://nationalgridservice.blogspot.com/2010/10/nagios-myegee-and-myegi.html

4 layers to performance investigation:

1. Storage elements
2. Sites
3. VOs
4. Middleware

3 benchmarking categories

1. micro-benchmarks
2. micro-kernels
3. application kernels

Benchmarking

HPL [22]

CE performance free processors MFLOPS MIPS (instructions per second) free RAM

SE performance IOPS free space

EGI accounting portal: CPU usage metrics aggregated for accounting.

3.3.1 Metrics

CPU load is taken using the pseudo /proc/loadavg file which in turn is filled by Linux kernel's CALC_LOAD macro. This function takes 3 parameters. The load-average bucket, a y constant that is calculated using formula

$$y = \frac{2^{11}}{2^{((5log_2(e))/60x)}}$$

for values $x = 1$, $x = 5$ and $x = 15$ (where x represent the minutes and y the exponent constant), and the number of how many processes are in the queue, in running or uninterruptible state.

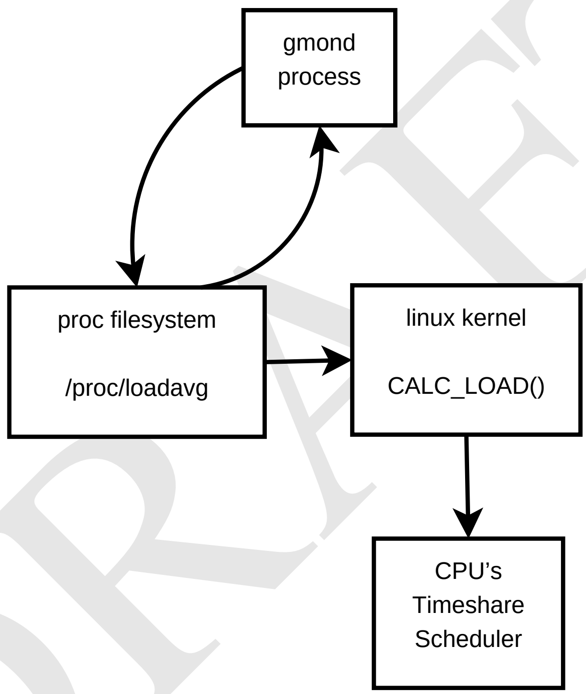


Figure 3.2: Load Average calculation

3.3.2 Ganglia

enw to ganglia to kanei etsi

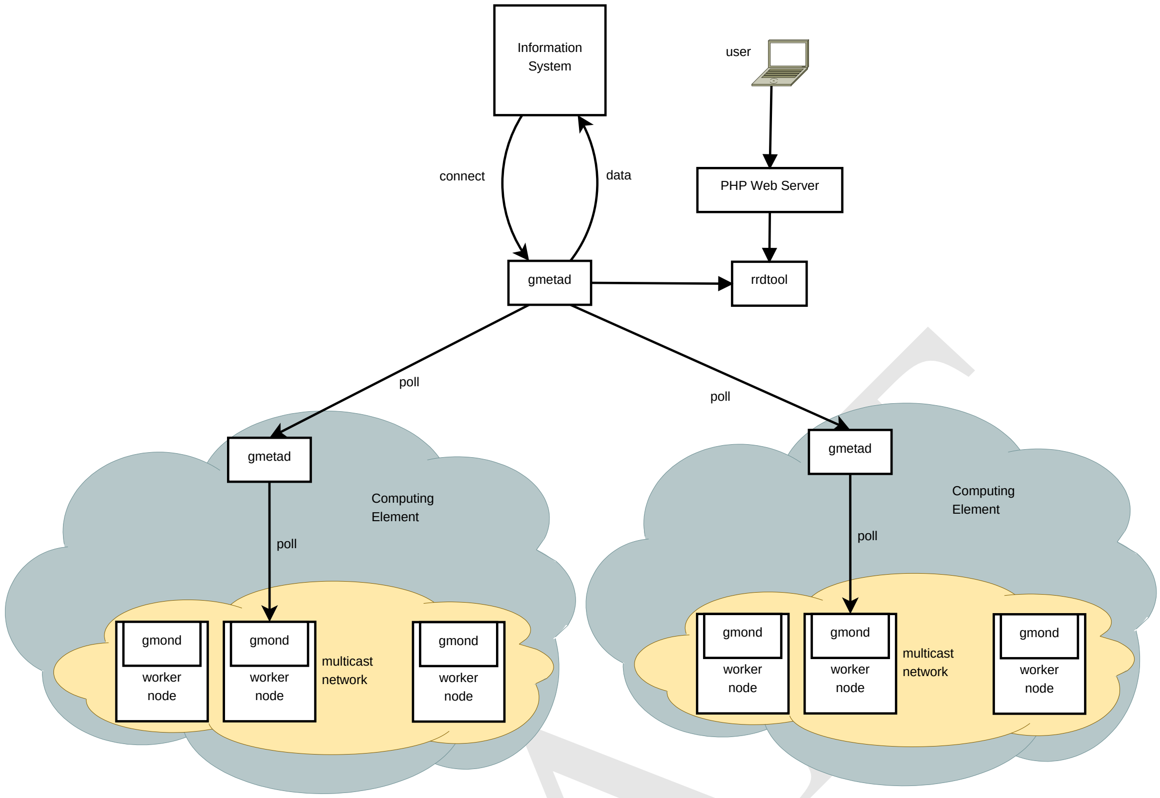


Figure 3.3: Ganglia Data Flow

3.3.3 Nagios

msg-to-handler msg-nagios-bridge

nagios2metric

3.3.4 Ganglia to Nagios

check_ganglia

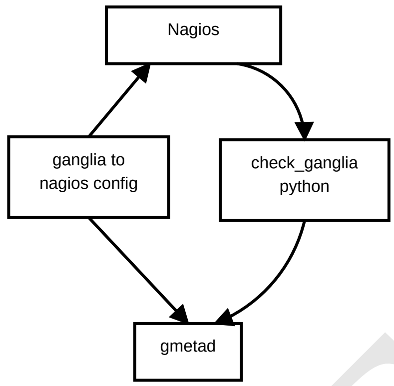


Figure 3.4: Nagios configuration and check ganglia values

3.3.5 pnp4nagios

Bulk Mode with NPCD NPCD: spool directory to process bulk data create graphs using RRD-TOOL

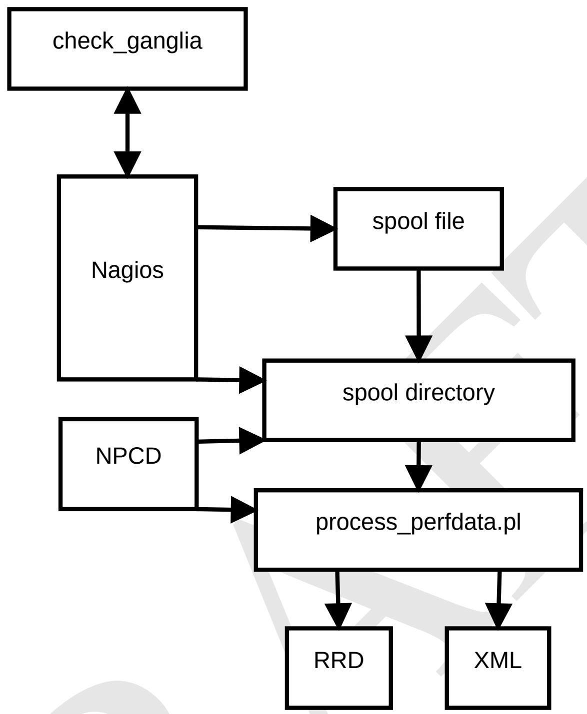


Figure 3.5: PNP 4 Nagios data flow

3.4 Range of cases examined - Metrics

Grid performance can be measured using benchmark tools in different levels of the grid architecture, using the micro-benchmarks at the Worker Node level, the Site (CE) level and the Grid VO level. Various benchmarks exist in these levels, using different libraries and algorithms, such as This project focuses on mathematically compute of the performance of a grid based on the metrics that are taken at the Worker Node level.

Different metrics and benchmarks exist, such as the measurement of the performance of CPUs in **MIPS using EPWhetstone** and the evaluation of the performance of a CPU in **FLOP/s and MB/s using BlasBench**. GridBench [22] provides a framework to collect those metrics using its own description language, **GBDL**.

GcpSensor [12] introduce a new performance metric called WMFLOPS. It uses PAPI [4] (Performance API) to access the hardware performance counters. For data distribution it uses MDS information system which provides dynamic metrics for CPU load average, one for 1, for 5 and for 15 minutes load.

3.5 Publish to Information System

3.5.1 LDAP based - MDS/BDII

1. Python ganglia client script: http://globus.org/toolkit/docs/2.4/mds/gangliaprovider.html

2. Perl gaglia-IP tool:

   http://www.star.bnl.gov/public/comp/Grid/Monitoring/SimpleGangliaIP.html

   http://www.star.bnl.gov/public/comp/Grid/Monitoring/ganglia_ip

Glue schema

GlueCluster GlueClusterTop GlueGeneralTop GlueHost GlueHostArchitecture GlueHostMainMemory GlueHostNetworkAdapter GlueHostOperatingSystem GlueHostProcessor GlueHostProcessorLoad GlueHostSMPLoad GlueSubCluster GlueTop

Perl:

```
$[root@mon ~]# ./ganglia_ip -h mon -p 8649 -o mds
```

Python:

```
$[root@mon ~]# /opt/ganglia/bin/ganglia --format=MDS
```

```
[root@osweb ~]# cat /opt/glite/etc/gip/provider/glite-info-provider-service-
#!/bin/bash
/opt/bin/ganglia_ip -h 195.251.70.54 -p 8649 -o mds
```

3.5.2 Web Service based - WSRF

container

using WSRF (GT4, information services, information providers) Ganglia Resource Provider MDS Index Service GLUE CE

OASIS standard

container

3.5.3 information provider

wssd

kai

rp xml

XPath

example queries

XSLT

my note: WSRF is GLUE 2.0 schema CE compatible

```
/opt/globus/etc/globus_wsrf_mds_usefulrp/ganglia_to_glue.xslt

<glue:ProcessorLoad>

<xsl:attribute name="glue:Last1Min">
  <xsl:call-template name="emitProperNumeric">
    <xsl:with-param name="numeric"
    select="floor(100 * METRIC[@NAME='load_one']/@VAL)"/>
  </xsl:call-template>
</xsl:attribute>

<xsl:attribute name="glue:Last5Min">
  <xsl:call-template name="emitProperNumeric">
    <xsl:with-param name="numeric"
    select="floor(100 * METRIC[@NAME='load_five']/@VAL)"/>
  </xsl:call-template>
</xsl:attribute>

<xsl:attribute name="glue:Last15Min">
  <xsl:call-template name="emitProperNumeric">
    <xsl:with-param name="numeric"
    select="floor(100 * METRIC[@NAME='load_fifteen']/@VAL)"/>
  </xsl:call-template>
</xsl:attribute>
```

```
</glue:ProcessorLoad>
```

WebMDS

webmds

Chapter 4

Results

4.1 Tables and Plots

4.1.1 Unix stuff

example processes/load monitoring

top/uname

cat /proc/loadavg

code from linux kernel about CALC_LOAD

4.1.2 Gmond

Installation and configuration

Transport and sample

code from gmond client

example output

port and xml staff

multicast examples

Gmetad

XDR

gstat -al1

4.1.3 Visualization

Drill down and levels of visualization

1. local resource
2. site
3. regional
4. grid

ganglia gui

rrdtool

4.2 Methods of Presentation

4.2.1 WSRF

```
/opt/globus/bin/wsrf-query \
-s https://osweb.teipir.gr:8443/wsrf/services/DefaultIndexService \
"//*[local-name()='Host']"
```


```
<ns1:GLUECE xmlns:ns1="http://mds.globus.org/glue/ce/1.1">
 <ns1:Cluster ns1:Name="OSLAB" ns1:UniqueID="OSLAB">
  <ns1:SubCluster ns1:Name="main" ns1:UniqueID="main">
   <ns1:Host ns1:Name="gr03.oslab.teipir.gr"
    ns1:UniqueID="gr03.oslab.teipir.gr"
    xmlns:ns1="http://mds.globus.org/glue/ce/1.1">
    <ns1:Processor ns1:CacheL1="0" ns1:CacheL1D="0"
     ns1:CacheL1I="0" ns1:CacheL2="0" ns1:ClockSpeed="2392"
     ns1:InstructionSet="x86"/>
    <ns1:MainMemory ns1:RAMAvailable="299" ns1:RAMSize="1010"
     ns1:VirtualAvailable="2403" ns1:VirtualSize="3132"/>
    <ns1:OperatingSystem ns1:Name="Linux"
     ns1:Release="2.6.18-194.26.1.el5"/>
    <ns1:Architecture ns1:SMPSize="2"/>
    <ns1:FileSystem ns1:AvailableSpace="201850"
     ns1:Name="entire-system" ns1:ReadOnly="false"
     ns1:Root="/" ns1:Size="214584"/>
    <ns1:NetworkAdapter ns1:IPAddress="10.0.0.33"
     ns1:InboundIP="true" ns1:MTU="0"
     ns1:Name="gr03.oslab.teipir.gr" ns1:OutboundIP="true"/>
    <ns1:ProcessorLoad ns1:Last15Min="45" ns1:Last1Min="337"
     ns1:Last5Min="126"/>
   </ns1:Host>
  </ns1:SubCluster>
```

```
  </ns1:Cluster>
</ns1:GLUECE>
```

XSLT

WebMDS and XPath

query examples

interface and forms description

raw XML output from WebMDS

4.2.2 BDII

/opt/glite/etc/gip/provider/glite-info-provider-service-xxx create a wrapper to present ganglia url and status to the BDII

Ganglia official python client:

```
[root@mon ~]# /opt/ganglia/bin/ganglia --format=MDS | grep -A 30 host=gr03
```

```
dn: host=gr03.oslab.teipir.gr, scl=sub2, cl=datatag-CNAF, \
 mds-vo-name=local, o=grid
objectclass: GlueHost
GlueHostName: gr03.oslab.teipir.gr
GlueHostUniqueID: RDLAB-TEIPIR-gr03.oslab.teipir.gr
objectclass: GlueHostArchitecture
GlueHostArchitecturePlatformType: x86-Linux
GlueHostArchitectureSMPSize: 2
objectclass: GlueHostProcessor
GlueHostProcessorClockSpeed: 2392
objectclass: GlueHostMainMemory
GlueHostMainMemoryRAMSize: 1035104
GlueHostMainMemoryRAMAvailable: 306280
objectclass: GlueHostNetworkAdapter
GlueHostNetworkAdapterName: gr03.oslab.teipir.gr
GlueHostNetworkAdapterIPAddress: 10.0.0.33
GlueHostNetworkAdapterMTU: unknown
GlueHostNetworkAdapterOutboundIP: 1
GlueHostNetworkAdapterInboundIP: 1
objectclass: GlueHostProcessorLoad
GlueHostProcessorLoadLast1Min: 2.57
GlueHostProcessorLoadLast5Min: 1.48
GlueHostProcessorLoadLast15Min: 0.58
objectclass: GlueHostSMPLoad
```

```
GlueHostSMPLoadLast1Min: 2.57
GlueHostSMPLoadLast5Min: 1.48
GlueHostSMPLoadLast15Min: 0.58
objectclass: GlueHostStorageDevice
GlueHostStorageDeviceSize: 209555000
GlueHostStorageDeviceAvailableSpace: 197120000
GlueHostStorageDeviceType: disk
```

Perl script to export MDS format:

```
[root@mon ~]# ./ganglia_ip -h mon -p 8649 -o mds | grep -A 22 host=gr03
```

```
dn: host=gr03.oslab.teipir.gr, cl=RDLAB, \
 mds-vo-name=local, o=grid
objectclass: GlueHost
GlueHostName: gr03.oslab.teipir.gr
GlueHostUniqueID: RDLAB-TEIPIR-gr03.oslab.teipir.gr
objectclass: GlueHostProcessorLoad
GlueHostProcessorLoadLast1Min: 2.57
GlueHostProcessorLoadLast5Min: 1.48
GlueHostProcessorLoadLast15Min: 0.58
objectclass: GlueHostSMPLoad
GlueHostSMPLoadLast1Min: 2.57
GlueHostSMPLoadLast5Min: 1.48
GlueHostSMPLoadLast15Min: 0.58
objectclass: GlueHostArchitecture
GlueHostArchitectureSMPSize: 2
objectclass: GlueHostProcessor
GlueHostProcessorClockSpeed: 2392
objectclass: GlueHostNetworkAdapter
GlueHostNetworkAdapterName: gr03.oslab.teipir.gr
GlueHostNetworkAdapterIPAddress: 10.0.0.33
```

```
objectclass: GlueHostMainMemory
GlueHostMainMemoryRAMSize: 1035104
GlueHostMainMemoryRAMAvailable: 306280


[root@osweb ~]# ldapsearch -H ldap://osweb.teipir.gr:2170 -x \
-b GlueHostName=ainex.local,Mds-Vo-name=local,o=grid \
GlueHostProcessorLoadLast1Min GlueHostProcessorLoadLast5Min \
GlueHostProcessorLoadLast15Min

# ainex.local, local, grid
dn: GlueHostName=ainex.local,Mds-Vo-name=local,o=grid
GlueHostProcessorLoadLast1Min: 27
GlueHostProcessorLoadLast15Min: 22
GlueHostProcessorLoadLast5Min: 20
```

4.3 Description of Information

4.3.1 DOM

php code example

WSRF -¿ DOM example

screenshot of output (green/yellow/red) colored pages from `http://people.brunel.ac.uk/~dc09ttp`

Chapter 5

Analysis

5.1 Methods Adopted

5.1.1 Peformance Metrics

CALC_LOAD - load average

extra

5.1.2 Information Systems

BDII

WSRF

5.2 Interpretation of Results

Discussion about performance results based on load average.

some unix internals, processes, scheduler not a percentage counter of cpu usage

Difference between these metrics and the availability of a grid based on the queue of jobs have been submitted.

5.3 Specific Interpretations

5.3.1 Scalling

LDAP

there is a paper about how information systems perform in large scale

WSRF

5.4 Enveloping Interpretations

Chapter 6

Conclusions

6.1 Conclusions

Conclusions should be based on an in depth critical analysis of the information presented in the dissertation and should be related to the objectives stated in the introduction.

do not simply summarise the dissertation

do not recapitulate the analysis or discussion

do not introduce new ideas

identify specific points that have been clarified or discovered, and specific actions to be taken

identify specific additional investigation that is required (and why)

It is important to remember that conclusions should only be drawn on the basis of the information presented in the dissertation. Generalised conclusions without supporting evidence are to be discouraged.

6.2 Further Work

Identify specific additional investigation that is required to be carried out.

Bibliography

[1] Sergio Andreozzi, Natascia De Bortoli, Sergio Fantinel, Antonia Ghiselli, Gian Luca Rubini, Gennaro Tortone, and Maria Cristina Vistoli. Gridice: a monitoring service for grid systems. *Future Gener. Comput. Syst.*, 21:559–571, April 2005.

[2] Z. Balaton, P. Kacsuk, N. Podhorszki, and F. Vajda. Use cases and the proposed grid monitoring architecture. Technical report, Tech. Rep. LDS-1/2001, Computer and Automation Research Institute of the Hungarian Academy of Sciences, 2001. http://www. lpds. sztaki. hu/publications/reports/lpds-1-2001. pdf, 2001.

[3] D. Bonacorsi, D. Colling, L. Field, S.M. Fisher, C. Grandi, P.R. Hobson, P. Kyberd, B. MacEvoy, J.J. Nebrensky, H. Tallini, and S. Traylen. Scalability tests of R-GMA-based grid job monitoring system for CMS Monte Carlo data production. *IEEE Transactions on Nuclear Science*, 51(6):3026–3029, December 2004.

[4] S Browne, J Dongarra, N Garner, G Ho, and P Mucci. A portable programming interface for performance evaluation on modern processors. *International Journal of High Performance Computing Applications*, 14(3):189, 2000.

[5] R Byrom, B Coghlan, A Cooke, R Cordenonsi, L Cornwall, M Craig, A Djaoui, S Fisher, A Gray, S Hicks, and Others. Production services for information and monitoring in the grid. *AHM2004, Nottingham, UK*, 2005.

[6] R. Byrom, D. Colling, S.M. Fisher, C. Grandi, P.R. Hobson, P. Kyberd, B. MacEvoy, J.J. Nebrensky, and S. Traylen. *Performance of R-GMA for Monitoring Grid Jobs for CMS Data Production*. IEEE, 2005.

[7] K Czajkowski, S Fitzgerald, I Foster, and C Kesselman. Grid information services for distributed resource sharing. In *High Performance Distributed Computing, 2001. Proceedings. 10th IEEE International Symposium on*, 2001.

[8] S Fisher. Relational model for information and monitoring. In *Global Grid Forum, GWD-Perf-7*, volume 1, 2001.

[9] S Fisher. DataGrid information and monitoring services architecture: design, requirements and evaluation criteria'. Technical report, Technical Report, DataGrid, 2002.

[10] S. Fitzgerald, I. Foster, C. Kesselman, G. von Laszewski, W. Smith, and S. Tuecke. A directory service for configuring high-performance distributed computations. In *High Performance Distributed Computing, 1997. Proceedings. The Sixth IEEE International Symposium on*, pages 365 –375, August 1997.

[11] M Gerndt, R Wismuller, Z Balaton, G Gombás, P Kacsuk, Z Németh, N Podhorszki, H L Truong, T Fahringer, M Bubak, and Others. Performance Tools for the Grid: State of the Art and Future, APART White Paper, 2004.

[12] Dong Guo, Liang Hu, Meng Zhang, and Z. Zhang. Gcpsensor: a cpu performance tool for grid environments. In *Quality Software, 2005. (QSIC 2005). Fifth International Conference on*, pages 273 – 278, 2005.

[13] P R Hobson, E Frizziero, Chenxi Huang, M R Irving, T Kalganova, P Kyberd, F Lelli, A Petrucci, R Pugliese, G Taylor, and R Taylor. *Grid Computing Technologies for Renewable Electricity Generator Monitoring and Control*. IEEE, June 2007.

[14] Chenxi Huang, Peter R. Hobson, Gareth A. Taylor, and Paul Kyberd. *A Study of Publish/-Subscribe Systems for Real-Time Grid Monitoring*. IEEE, March 2007.

[15] E Imamagic and D Dobrenic. Grid infrastructure monitoring system based on Nagios. In *Proceedings of the 2007 workshop on Grid monitoring*, page 28. ACM, 2007.

[16] Attila Kertész and Péter Kacsuk. A Taxonomy of Grid Resource Brokers. In *6th Austrian-Hungarian Workshop on Distributed and Parallel Systems (DAPSYS*, pages 318–366. Basil Blackwell, 2006.

[17] M Li and M Baker. *The grid: core technologies*. John Wiley & Sons Inc, 2005.

[18] Yi Liu and Shu Gao. Wsrf-based distributed visualization. In *Cluster Computing and the Grid, 2009. CCGRID '09. 9th IEEE/ACM International Symposium on*, pages 615 –619, May 2009.

[19] Luciano Gaido. DSA1.2.2: Assessment of production service status, 2010.

[20] G.A. Taylor, M.R. Irving, P.R. Hobson, C. Huang, P. Kyberd, and R.J. Taylor. *Distributed monitoring and control of future power systems via grid computing*. IEEE, 2006.

[21] B Tierney, R Aydt, D Gunter, W Smith, M Swany, V Taylor, and R Wolski. A grid monitoring architecture. In *The Global Grid Forum GWD-GP-16-2*. Citeseer, 2002.

[22] G. Tsouloupas and M.D. Dikaiakos. Gridbench: a tool for benchmarking grids. In *Grid Computing, 2003. Proceedings. Fourth International Workshop on*, pages 60 – 67, November 2003.

[23] G Von Laszewski and I Foster. Usage of LDAP in Globus. *Mathematics and Computer Science Division, Argonne National Laboratory*, 1998.

[24] A J Wilson, R Byrom, L A Cornwall, M S Craig, A Djaoui, S M Fisher, S Hicks, R P Middleton, J A Walk, A Cooke, and Others. Information and monitoring services within a grid environment. In *Computing in High Energy and Nuclear Physics (CHEP)*. Citeseer, 2004.

[25] S Zanikolas. *Importance-Aware Monitoring for Large-Scale Grid Information Services*. PhD thesis, the University of Manchester, 2007.

[26] X Zhang, J L Freschl, and J M Schopf. A performance study of monitoring and information services for distributed systems. In *12th IEEE International Symposium on High Performance Distributed Computing, 2003. Proceedings*, pages 270–281, 2003.

[27] X Zhang and J M Schopf. Performance analysis of the globus toolkit monitoring and discovery service, mds2. *Arxiv preprint cs/0407062*, 2004.

School of Engineering & Design

Electronic & Computer Engineering

Industrial Mentor's Report on a MSc Dissertation

Student's name: Theofylaktos Papapanagiotou

Dissertation Title: Grid Monitoring

I confirm / cannot confirm* that the MSc dissertation submitted by the above student truly reflects the respective contributions of the student and others, with whom the student has worked during the MSc project.

∗ delete as appropriate

Comments:

Signed:

Name: Dr.Paul Kyberd

Date: