WLCG Monitoring for Managers

An introduction to WLCG monitoring for managers
Author: WLCG Monitoring Working Group[1]

Overview

This document is for all people who wish to gain a high level overview of the work being carried out by the WLCG Grid Services Monitoring working group.

This document describes the architectural principles behind Monitoring within WLCG, coming out of the requirements provided to the working group by its mandate and stakeholders.

After reading this document you should understand the overall architecture of the WLCG monitoring framework, and the immediate workplan for the WLCG monitoring group.

Why do we need monitoring?

The purpose of the WLCG monitoring group, is "… to help improve the reliability of the grid infrastructure, and to provide stakeholders with views of the infrastructure allowing them to understand the current and historical status of the service." [2]

All our work is to fulfill these two goals – improved reliability and improved comprehension of the underlying grid infrastructure - and is predicated on the fact that the availability of accurate, consistent data on service status and historical performance is essential for efficient problem determination, resolution and service planning.

Currently, in order to simplify the problem to achieve results in a shortened timescale, we consider both Information Systems (e.g. BDII) and Accounting (e.g. APEL, DGAS) to be out of scope; it is hoped that some of the solutions proposed may be of value to these two areas in the future.

Users of Grid Monitoring

In the mandate of the working group, four user groups, or stakeholders, are considered:

- Grid site administrators

[1] Editors: James Casey <james.casey@cern.ch>, Ian Neilson <Ian.Neilson@cern.ch>
[2] https://twiki.cern.ch/twiki/bin/view/LCG/GridServiceMonitoringWGMandate

- Grid service managers and grid operators
- Virtual Organization management
- Grid Project management

When analyzing the current reporting and visualization tools, it was seen that the Grid service manager/operators and project management are well served by tools like SAM and Gridview, while the VOs have the experiment dashboards. The group least well represented is the site administrators.

Site Administrators

We consider three classes[3] of site administrators

1. Site administrators of large sites, who have an existing fabric monitoring system, and have already integrated the grid services into their monitoring system
2. Site administrators who have an existing fabric monitoring system, but do not know what to monitor for the grid services at their site
3. Site administrators, normally of small sites, who have no local fabric monitoring and would like a packaged solution made available for them.

The requirements that came from the site administrators were that they would like:

- Better local monitoring of the grid services at their site by both the monitoring of the complete set Grid services and with increased detail to allow some level of problem diagnosis and the possibility of some recovery automation.
- Programmatic access to monitoring information gathered about their sites by external agencies to allow early local notification
- Descriptions of 'alarm conditions' for monitoring information gathered so that they can integrate this information with their local alarm notification systems.

These three classes have different requirements, from better sensors for sites already with monitoring, to a proposal for a complete monitoring solution for sites which have no existing fabric monitoring in place.

[3] An additional class, generally of small sites who are well supported by regional Grid infrastructures, we consider to be a hybrid of the three classifications.

Components of WLCG monitoring

Having discussed both the objectives and main users of the monitoring infrastructure, we now look at the main components we identify within the current WLCG monitoring framework (Figure 1).

1. Monitoring data is produced either from the **Site Fabric** or by **Remote Probes**.
2. A **Transport Layer** moves this data from where it is produced to **Repositories**.
3. **Repositories** are responsible for aggregating similar data for many sites and services.
4. **Repositories** may post-process the data, for example to calculate **Site Availability** or **Summary Information**.
5. **Visualization Systems** take the data, and provide it to users and administrators in visual form.

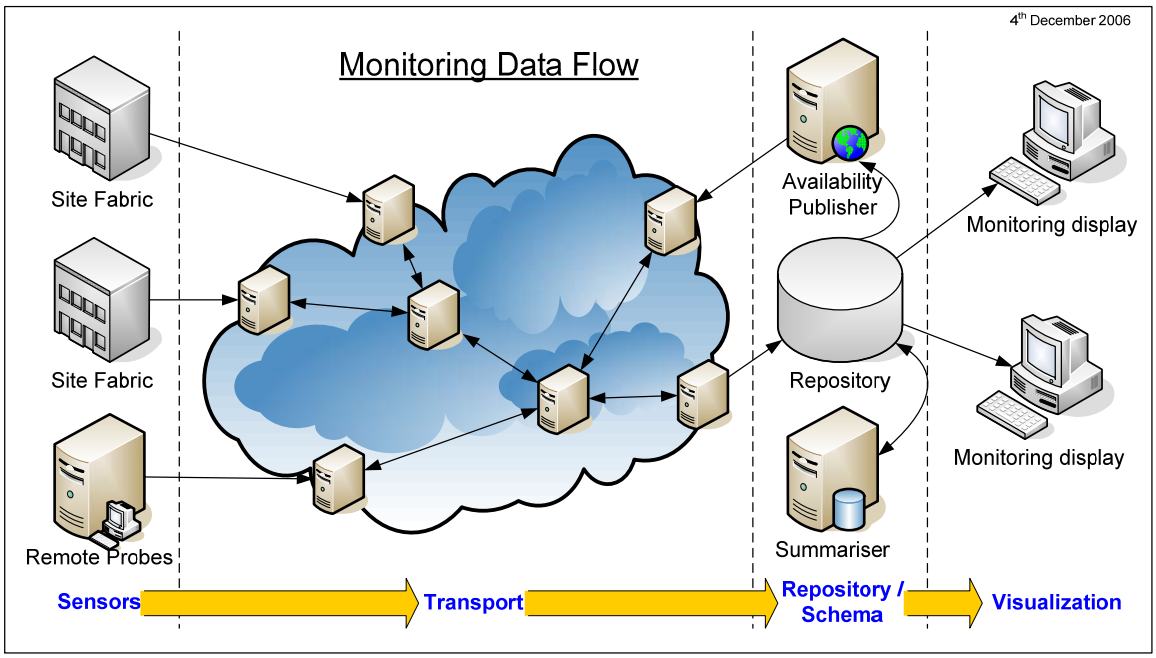

Figure 1: Monitoring Data Flow

It can be seen that many of these components have implementations within current tools used within WLCG. One of the tasks of the monitoring group is to compare the various features of these tools, and suggest areas where commonality may be achieved between components.

Architectural Principles

We have abstracted out some 'Architectural Principles' which should underpin the monitoring system.

"Site administrators are closest to the problems, and need to know about them first"

In our environment with highly distributed systems, and many different administrative domains, it is clear that the local site administrators are the people who have to fix the problems that occur at a site. Therefore the infrastructure should aim to provide as much information on problems at their site, direct to them, as quickly as possible.

"Site administrators don't need another site monitoring system – they need the information in their monitoring system"

Many sites already have a fabric monitoring system, and associated procedures and alarms. Rather than requiring another system to be use for grid services over the rest of the fabric, the information should be integrated into the normal fabric systems at a site.

"Problems at a site should be detected by the site monitoring before remote monitoring"

In a sense, this is an extension of the commonly agreed principle "The monitoring should detect the problem before the user". Within WCLG currently, we rely on remote monitoring (such as SAM) to do this, and the corresponding operational infrastructure to alert site administrators of the problems (i.e. CIC on duty). This can often lead to delays of several hours before the people who can fix the problem are actually notified of the problem.

"Local problems detected locally shouldn't require remote services to work out what the problem is"

This is basically a rephrasing of the fact that a site should be autonomous, where possible. For instance, if a calculation of service status or service availability can be done on the site and presented directly to the administrators, then it should be – without having to round-trip through an external service to do the calculation.

"If you're monitoring a site remotely, it's only polite to give the data to the site"

There exist many system that gather data on the grid services at sites. For instance:

- A Resource Broker knows about the success/failure rate for a Computing Element at a site
- The File Transfer Service knows about the reliability and data transfer rates for Storage Elements at a site

- The VO dashboards know about possible VO problems at a site e.g. software installation problems

While it is possible for the site admin to poll all these information sources on a regular basis (often by viewing web pages), it would be easier for the site for relevant data to be pushed to them.

"Site administrators are busy, so make it easy for them"

Site administrators often are not only managing grid services, but are running the rest of the computing fabric. Also, at many small sites, the administrator is often a part-time responsibility. Therefore, the emphasis should be on providing the simplest solution possible that solves their needs, and one that does not require a lot of training/specialist knowledge or ongoing support.

Other Concerns

Security

Many different kinds of data travel around in the grid, for example:

- user registration
- job accounting
- service availability data
- application input/output

It is recognized as important that the operational requirements for, and the handling of, such data are both appropriate and consistent with legal constraints such as a user's rights to privacy. Similarly, there may be constraints of access to usage data imposed by the application domain to prevent the build-up of usage patterns. The Joint Security Policy Group is developing policy for use in these areas.

Having said that, in general, the fabric and service monitoring data dealt with by the WLCG Monitoring Working Group is not of a kind which poses serious problems regarding legal constraints provided that job-level monitoring does not include information identifying the user. As a general framework it is proposed that data should be considered as falling within one of 5 classes:

1. Private data - data relating to an individual (?age, sex, address)
2. Personal data - data which can be used to identify the individual (name, email, institute)
3. Sensitive data - data which might be used to derive information about a user (account usage records related to one unidentified user)

4. Non-sensitive data - data which relates to an entity but does not relate to an individual (the total hours availability of a Grid site in a month)
5. Public data (the date)

Most monitoring data would fall into class 4 (non-sensitive). Appropriate authorization, transport and handling requirements should be recommended for data of each class.

One aspect of security that is of importance to monitoring data is *integrity*: the source of the data should be known (trust decision can be made) and the data should be known to be unaltered. To avoid the possibility of trivial “attacks”, transport protocols must be chosen to satisfy these objectives.

Interfaces and boundaries

The WLCG is highly heterogeneous as regards monitoring solutions in place at sites, and indeed regarding the systems within different grid infrastructures (e.g. EGEE, OSG, NorduGrid) to both produce and consume the monitoring data. Therefore we believe that a common approach for monitoring must be based on standardization of interfaces and data exchange formats rather than the mandating of common software solutions. We focus initially on two areas where we believe this standardization is both necessary, and possible.

Common Probes

A probe is something which can gather a sample for a set of metrics relating to a running service. These could either be metrics regarding service status or performance.

There exists a variety of probes for grid services in production use today. These include ‘SAM tests’ within the SAM framework, ‘LEMON sensors’ used at CERN within LEMON, ‘DaemonCheck’ tests for GridIce and Nagios probes developed and deployed at the EGEE CE and AP ROCs.

Indeed, there is often overlap between the functionality of the probes, which is currently necessary due to the different output formats required by the different monitoring frameworks. A common way to write sensors would promote better reuse of sensors, and lead to a set of ‘best of breed’ sensors being commonly available.

In order to do this, we have standardized on a way to call a named probe for a given grid service, and also the format of the message returned. This is based on the format currently used by SAM, but has alterations to allow easier interfacing with other monitoring systems. A single sensor "wrapper" for each fabric monitoring system is then required to use these common probes; Figure 2 shows the interaction of one of these probes with a fabric monitoring system (in this example LEMON).

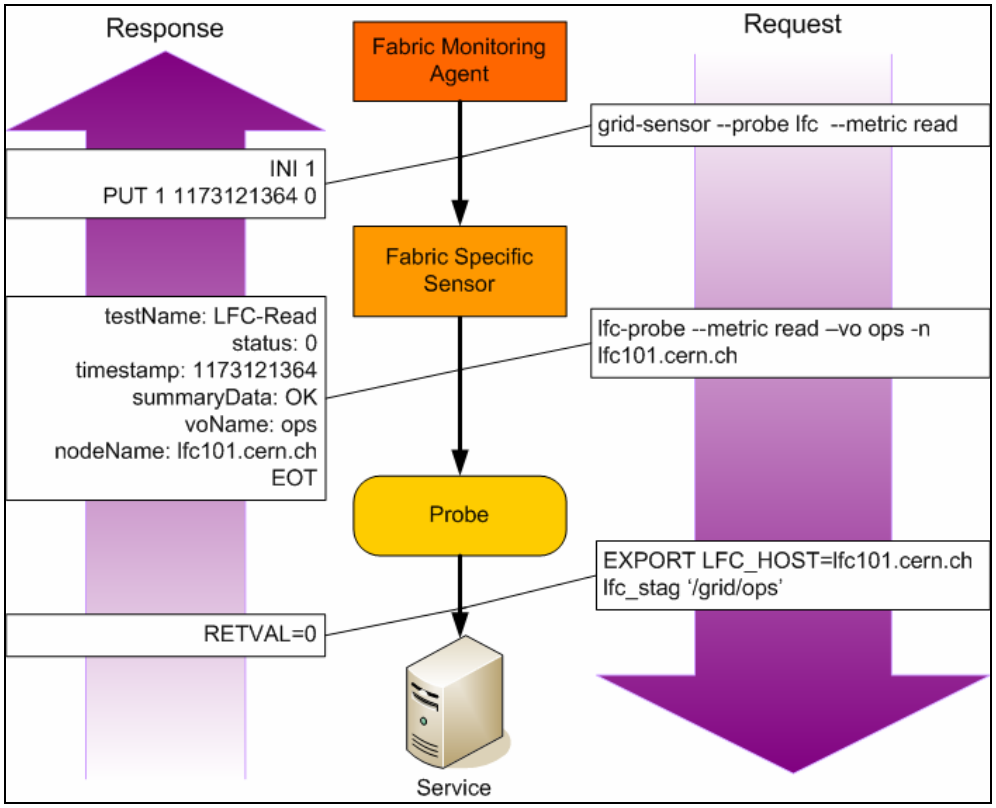

Figure 2 : Interactions between monitoring framework, sensors and probes

Standard Data Exchange Format

When a remote service has some information it has gathered about a site, it must provide a way for a site to query for the information relevant to the site. In order to not have a wide range of interfaces that a site manager must code against in order to get this information, we have created a standard data exchange format[4]. This is based on the current mechanism that SAM used to expose SAM tests results. It is a protocol which uses URL-encoded queries and returns XML over HTTP. It also describes the encoding for common data types, such as dates, numbers and arrays within the query and response.

[4] https://twiki.cern.ch/twiki/bin/view/LCG/GridMonitoringDataExchangeStandard

Publication of data from a site

While not all data gathered at a site is suitable or required to be published outside, there is some data that should be published from the site. For instance, publishing summary availability data at the same level of detail as SAM allows grid managers to verify that a site is actually monitoring their fabric at a suitable level.

This implies that a **grid publisher** component runs on each site, which is responsible for extracting (and possibly calculating) a subset of metrics which describe the availability and performance of the site at a high level.

WLCG Monitoring proof-of-concept

In order to further develop and validate the ideas presented in this paper, we intend to create a proof of concept prototype monitoring installation, using as many as possible of the technologies currently used within WLCG monitoring and those outlined above.

This prototype will consist of a Nagios service monitoring at least two services on a grid site, using a common set of probes, and a Nagios-specific adaptor. This Nagios service will also collect SAM test data from the central SAM repository via the standard data exchange format.

Finally, the installation will re-publish service availability data, calculated from the locally collected data, using the standard data exchange format to expose this data.

We aim to have this proof of concept demo ready for the end March 2007.

In parallel with the practical validation of the concepts we have initiated the gathering of more accurate and complete monitoring definitions, starting with service developers. Gathering this information and mapping it to the standardized interfaces and formats described above, should facilitate the more rapid deployment of a set of definitive sensors per grid service when the proof-of-concept phase is completed.