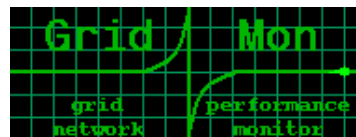


Network Performance Monitoring for the Grid (UK e-Science, 2005 All Hands Meeting)



Mark Leese (m.j.leese@dl.ac.uk), Rik Tyer (r.p.tyer@dl.ac.uk) and Robin Tasker (r.tasker@dl.ac.uk) on behalf of GridPP and the JISC CCLRC Daresbury Laboratory, Warrington, Cheshire, WA4 4AD
<http://gridmon.dl.ac.uk/>

Abstract:

Network performance monitoring has traditionally been important to the operation of networks of any significant size as an aid to fault detection and for determining expected performance. The need for such monitoring is enhanced for Grid computing. Without it Grid middleware and applications cannot optimise their performance by adapting to changing network conditions, networks cannot be debugged for efficiency, and the Grid cannot support the measurable SLAs required of the “utility computing” model. To address some of these issues, a UK e-Science Grid network performance monitoring project began in June 2002. The work continues, and is currently funded by GridPP2 and the JISC.

Progress has been presented at each All Hands Meeting, and this continues in 2005 with an update focusing on the European and wider international Grid network monitoring work which the UK has contributed to, and in some cases led, since the last AHM. Our work within the JRA4 (development of network services) group of the EGEE project will be outlined, concentrating on the development of ‘mediator’ software which provides unified access to heterogeneous network performance infrastructures. Our work leading the Network Measurements Working Group of the GGF is also discussed, covering their development of XML schemas for communicating with network monitoring systems. Finally, some consideration is also given to work closer to home, such as evolution of the UK Grid network monitoring infrastructure.

Glossary:

EDG	European Data Grid	NREN	National Research & Educational Network
EGEE	Enabling Grids for E-scienceE	PFN	Physical File Name
GGF	Grid Global Forum	piPEs	performance initiative Performance Environment system
GOC	Grid Operations Centre	RDBMS	Relational Data Base Management System
IEPM	Internet End-to-end Performance Monitoring	R-GMA	Relational Grid Monitoring Architecture
JRA	Joint Research Activity	SLA	Service Level Agreement
LDAP	Lightweight Directory Access Protocol	SOAP	Simple Object Access Protocol
LFN	Logical File Name	TCP	Transmission Control Protocol
NOC	Network Operations Centre	WP	Work Package
		XML	eXtensible Mark-up Language

Introduction

This paper summarises the work undertaken by the ‘GridMon’ Grid network performance monitoring project since the last All Hands Meeting.

To put the work in context, we will first consider the motivation for Grid network performance monitoring, before describing our international, European and UK work.

Motivation

Network performance monitoring is crucial to the Grid. Measurements are required for:

- Debugging networks for efficiency, an essential step for those wishing to run data intensive applications.
- Grid middleware and applications to make intelligent use of the network, optimising their performance by adapting to changing network conditions (including the ability to be “self healing”).
- Supporting the Grid “utility computing” model and ensuring that the differing network performances required by particular Grid applications are provided, via measurable SLAs.

The concepts and practice of network monitoring are well understood and are widely used to identify problems, quantify performance and set expected levels of service. Monitoring for the Grid however is a special case, and must be given special treatment. Debugging networks for efficiency has increased importance in the Grid arena, while publication to middleware and SLA support are relatively new concepts. These topics are now discussed in more detail.

Debugging networks for efficiency

Networks have always been debugged for performance. The difference in the Grid world is that it becomes an essential step for those wishing to use data intensive applications, that is, projects with large data sets such as high energy and particle physics experiments, radio astronomy, and medical applications, and high-bandwidth dependant projects, such as real-time remote visualisation applications.

Its importance results from the fact that simple over-provisioning of networks is not greatly benefiting the end user. While in many cases WANs are un-congested and operating below capacity, problems in the “last mile” mean that end users are separated from these fast flowing rivers by silted tributaries. Efficient WAN performance does not necessary translate into equally efficient performance across the full end-to-end path. This issue also explains Grid network monitoring’s focus on end-to-end performance.

Publication to middleware and Grid applications

Making network performance data available to middleware and Grid applications is a significant new development. The very idea that performance data published to man and machine will allow the middleware and applications to achieve optimum performance by adapting to changing network conditions is a major driver for this work. It also relates to the Grid’s much publicised self-healing capability.

An adaptive Grid could take many forms, such as an application varying its transport strategy by tuning TCP parameters, or a more distributed case involving Grid middleware, as described below.

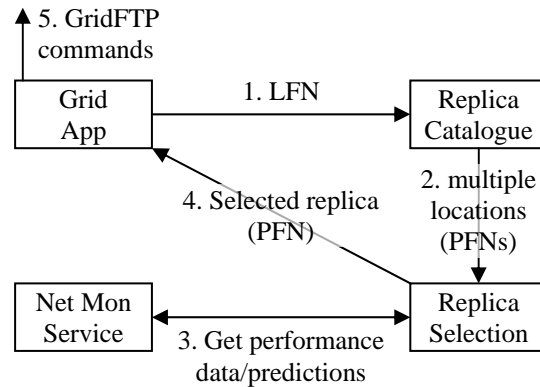


Figure 1: Replica Selection

File replication is a proven technique for improving data access, involving distribution of multiple copies of a file across a Grid. When Grid resources such as computing elements subsequently require the file, they have access to a choice of copies, some of which may be more desirable to use than others.

A replicated file has a Logical File Name (LFN) which maps to one or more PFNs (physicals). A Replica/Replication Manager is responsible for replication issues, including maintaining the mapping between logical and physical filenames, and deciding which replicas should exist and what their locations should be (normally based on recent usage patterns). The Replica Manager includes a Replica Selection Service which uses network performance data (from somewhere) to find the “best” replica. “Best” could be defined in several ways. The most obvious definition of “best” would be the “quickest obtainable”, but it could just as equally be that which will have the least impact on other network users, or the geographically closest. A simplified representation of the process is given in figure 1.

Support for measurable SLAs

Without periodically monitoring the performance of a network it is not possible to ensure that you are receiving expected levels of service or are providing the levels which you are contracted to deliver. Service levels can be based on many characteristics, including network availability (uptime), available bandwidth and network latency (delay).

The Grid as “utility computing”, with its associated SLAs, may currently have little application in the academic world, but interest is

growing in the commercial sector. We mention it briefly for completeness.

Having described the basic needs for network performance monitoring for the Grid, we will now the work carried out since the last AHM.

Publication of Performance Data

There are currently two major drivers for publishing network performance data in a manner in which it can be easily obtained by other software:

1. So that Grid middleware and applications can access the data for decision making over intelligent use of the network, as exemplified in the previous section.
2. To allow NRENs and other network providers to gain access to the data to facilitate early detection and diagnosis of network problems, speeding up problem resolution.

In the future, we expect the drivers to extend to include SLA monitoring, and advanced networking functions, such as modelling the network as a Grid manageable resource, as proposed by the GGF Grid High Performance Networking Research Group (GHPN-RG) [1]. These are however beyond the scope of this year's paper.

We have commented in previous years that during the lifetime of the project, various methods of publishing data to the Grid middleware have been touted, including LDAP and R-GMA, and further that Web Services are now seen as the accepted forward path.

Web Services are essentially "online" applications accessed via XML messages. The use of agreed XML messaging allows web services to interface with each other.

In June 2003 the GGF Network Measurements Working Group (NM-WG), which the UK now co-chairs through Daresbury, produced a draft XML schema for publishing network monitoring data.

The group had previously proposed a network measurement classification system [2] to assist in data portability. The NM-WG "hierarchy document" describes a set of network

characteristics together with a classification hierarchy aimed at Grid applications and services. Application of the hierarchy is designed to facilitate creation of common schemata for describing network monitoring data. Using a standard classification for measurements maximises the portability of data.

In October 2003 Daresbury suggested that the publication schema should be complemented by a schema for requesting network performance data, whether that request referred to on-demand/future tests or historic data. The request and response (publication) schemas have been evolving ever since.

The idea is illustrated in figure 2, where in fully interactive systems clients will be able to request historic data, future or on-demand tests or predictions (as popularised by such monitoring tools as the Network Weather Service [3]). Results can then be returned. All request and result messages are formatted using standardised schemas, crucially allowing heterogeneous monitoring systems to interact providing that they use the same schemas.

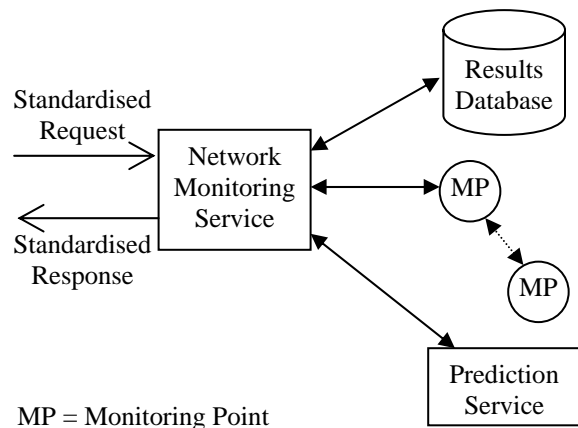


Figure 2: request-response mechanism

Although a slight aside, it should be noted that the prediction service does not predict how the network will react to certain traffic, but predicts the likely value of network characteristics, e.g. what bandwidth will be available. Middleware and Grid applications can then make decisions based on those predictions.

Discussing the schemas in detail is beyond the scope of this paper. However, it worth considering what types of request can be made. Three textual examples are given below. They highlight the

flexibility of requests, most notably that requests can range from very simple to very complex.

Simple: Give me the last value for available bandwidth to host X. I'm only interested in something in the last 30 minutes. If there isn't any data, don't run any new tests.

More Complex: Give me the last value for available bandwidth to host X. I'm only interested in something in the last 30 minutes. If there isn't any data, run an on-demand test, using

- between two and four TCP streams
- a 4MB TCP buffer size
- the iperf tool if available, pathchar as a second choice, and any other bandwidth tool as a third

Historical Query: Give me a maximum of 20 results for available bandwidth to host X from the last 24 hours. If there are more than 20 matches, give me the 20 values closest to the start of the period. Report what parameters were used in the tests, and provide a mean of the results.

By mid-2004 the schemas had reached sufficient stability to allow example implementations to begin, culminating in demonstrations of the EGEE JRA4 Mediator and Internet2 piPEs [4] software at GGF13 in March this year.

This was a major step for NM-WG. The piPEs demonstration was significant because it showed the schemas being used to request and satisfy queries for historic data **and** requests for on-demand tests. The EGEE JRA4 Mediator prototype (to be described in more detail in the next section) was significant because it allowed historic queries to be made of separate end-to-end **and** backbone monitoring infrastructures. This was a first demonstration of obtaining performance data from multiple administrative network domains using a single method.

Given this success it may be a surprise to learn that these schemas will soon be replaced by new versions, which are not backwards compatible. The reasons for this will be explained shortly, but it is first worth mentioning that the "version 1/V1" schemas have served a useful purpose, not only educating the group with respect to schemas, but in highlighting the best form network monitoring schemas should take, and most importantly of all,

winning people over to the idea of sharing data using common schemas.

The "version 2/V2" schemas were formally proposed in October 2004, and were aimed at addressing limitations in the V1 schemas that had come to light whilst producing demonstration software.

The V1 schemas were designed as monolithic "one size fits all" solutions. The motivation for this was to limit the number of schemas requiring development and maintenance, and generation and parsing when in actual use.

Whilst this worked in practice, it was perhaps not formally acceptable because it was not possible to encode all relevant "rules" and business logic within the schemas. As an example, a request could be generated containing the following elements:

```
<toolName>ping<toolName>  
<protocol>UDP</protocol>  
<tcpBufferSize>1024K<tcpBufferSize>
```

While this request message containing these elements could be schema compliant and pass schema validation, it would not be semantically valid since the ping tool uses the ICMP not UDP or TCP protocols.

V2 sees us move to a model in which separate schemas using certain NM-WG defined base elements are developed for specific tools and/or specific characteristics. As a result, schemas will only contain elements relevant to that particular tool or characteristic.

V2 also sees us move to a model in which requests and data publication are broken down into data and metadata. All measurements have some value and associated time, constituting a measurement's data. Further, a measurement is described by its metadata, identifying the who, what and how of the measurement. In more detail, V2 metadata consists of:

- Subject: the measured/tested entity
- Characteristic (verb): what type of measurement was taken?
- Parameters (adjectives and adverbs): how, or under what conditions, was this measurement taken?

Referring back to the previous point concerning characteristic/tool specific schemas, this implies that the specific structures of Data and Metadata elements will depend on the measurement to which they relate.

In addition to being a “neater” solution, the separation of data and metadata can minimise the actual data sent on the wire. Providing it has been transmitted at least once (e.g. at the beginning of a ‘conversation’) metadata can then be referred to by an ID, and does not need to be re-sent with every related message.

Further advantages of the V2 schemas are that:

- The Characteristic (verb) could be extended to support the recording and notification of events.
- They lend themselves to supporting more than the direct request-response model. E.g. a V2 request could actually be a request to register for event notifications, while there could be specific V2 schemas for reporting those events.

The group has already developed sample schemas for the iperf, ping, and traceroute tools, with sample implementations for dealing with such request and result messages.

One problem to be resolved however is who will be responsible for maintaining the set of “official” schemas, and validating requests to have new schemas submitted to the “approved set”.

As a final point, the V1 schemas, once finalised will be recorded in a GGF experimental document, summarising “results of Grid related experiments, implementations, or other operational experience”. In addition to providing a record of the history of the NM-WG schema development and lessons learned, documenting a final version of the V1 schemas supports the development groups (such as GridMon and EGEE JRA4) who have produced software based on V1 which needs to be demonstrated or put into service whilst V2 reaches maturity.

EGEE JRA4 Mediator

EGEE is the successor to the European Data Grid (EDG) project, while JRA4 is the EGEE group responsible for “Development of Network Services”. We have worked within JRA4’s

Network Performance Monitoring (NPM) activity, helping develop “Mediator” software, and at one point leading this effort.

The purpose of the Mediator can be quickly explained if you consider figures 3 and 4, below.

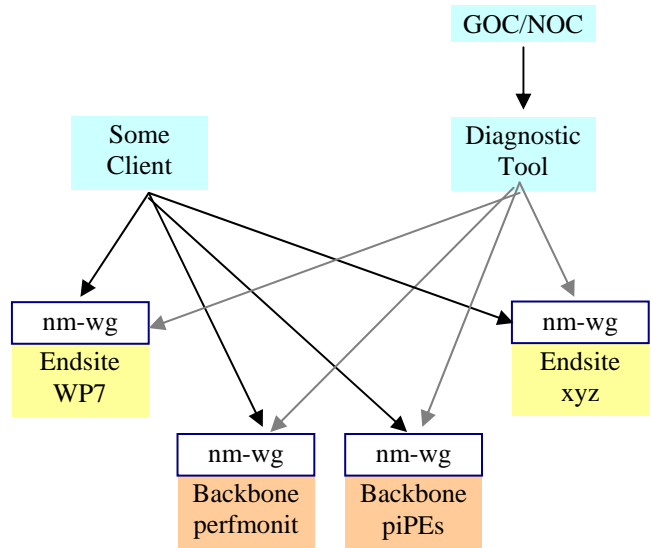


Figure 3: Current network monitoring

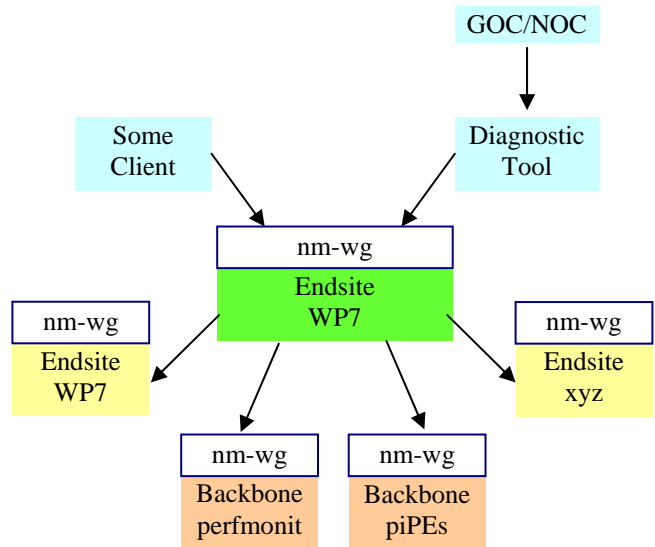


Figure 4: Monitoring with the Mediator

Currently if a client, whatever that may be, wishes to receive detailed information concerning a network path, or wishes to concatenate data from several backbone networks to produce a picture of the full backbone path, it will have to make several requests, and if necessary, perform its own

data aggregation. Therefore, the role of the Mediator, as shown in figure 4, is to greatly simplify this process by unifying access to NPM data. An in-depth description is beyond the scope of this paper, however, design documents are available [5], and the pertinent details are explained below with the aid of figure 5.

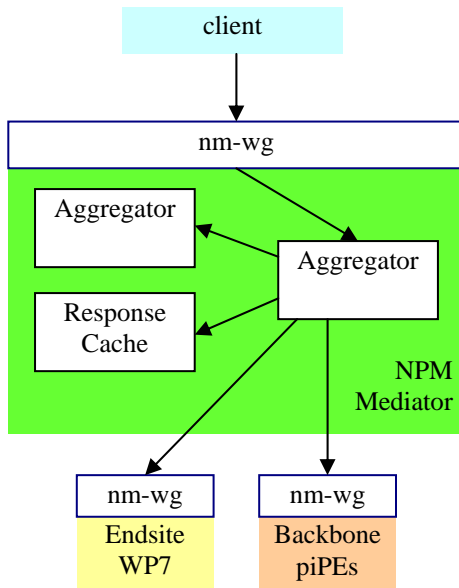


Figure 5: Mediator internals

1. Human and machine users interact via a client application, “speaking NM-WG”.
2. The Discoverer locates the MP(s) or infrastructures that can answer the client’s query. Currently, this is a static list of known MPs, however a means for dynamic update is obviously required, and a means for achieving this is being discussed.
3. The aggregator obtains the results of the query from the chosen MP or infrastructure, and if necessary, aggregates any partial path data to produce one result for the client.
4. To improve performance and reduce loading, results of recent requests will be cached.

In this respect, the Mediator can be thought of as an intelligent forwarding machine; intelligent because it can deal with basic aggregation of data, and therefore acts as more than just a router for NM-WG requests.

Our work in JRA4 continues, with a current focus on producing a diagnostic tool aimed at helping NOC and GOC operatives make the most of the

capabilities of the Mediator for routine network monitoring and fast fault diagnosis.

UK Work

GridMon

From inception, the GridMon project has aimed to create a basic, UK-wide network monitoring infrastructure. This was achieved by establishing a presence at each of the original e-Science Centres.

Monitoring was performed by a kit of tools installed on a suitable computing node at each centre, with regular tests performed between all centres. A mesh of monitoring was thus created, allowing each centre to build a picture of the quality of its links to all other centres.

Performance data was published to interested humans via the web interface shown in figure 6. Users were presented with a clickable UK map, leading to a form, which in turn lead to plots of performance data.

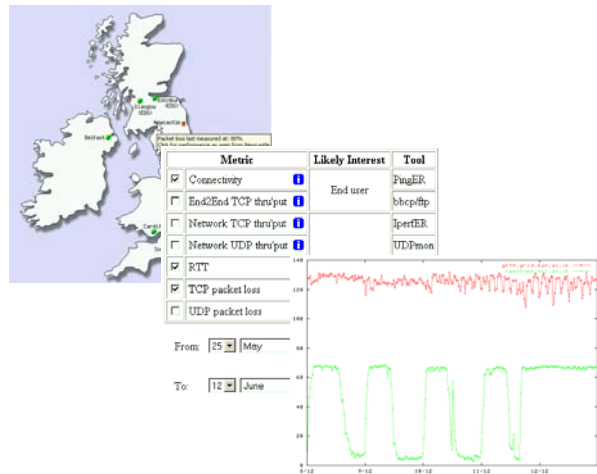


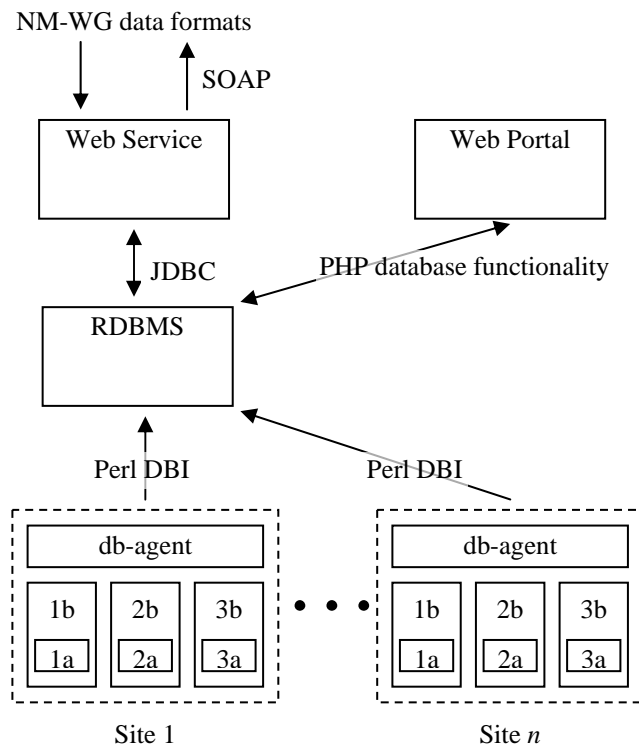
Figure 6: GridMon web interface

Each end site’s GridMon installation was self-sufficient. Monitoring tools ran regular tests, stored their results locally, and users accessed the data using the web interface served from the test machine’s web server.

The arguments for this approach were to allow local customisation of the installation, to provide redundancy and to keep reliance on other machines and services as low as possible. There were also historical reasons, such as the fact that

the monitoring tools inherited from previous projects used flat text files to store their data, and thus did not lend themselves to using a centralised data store. However, this flexibility and independence came at a price. The monitoring machines were by no means homogenous, utilising a multitude of Linux distributions and versions in use. Local site rules and conventions for machine configuration added to the mix, and the requirement to apply patches has shown that a distributed approach is inefficient and a frustrating experience.

Work is underway to revamp the infrastructure, and move to an architecture in which each site will store its test data in a central database at Daresbury. Web Services and human (web) access to the data will also be via services running at the laboratory. Storing and providing access to the data from a central location allows us to reduce the complexity of the individual monitoring nodes, and hopefully the problems associated with the individuality of those nodes. Critically, we can also move to a relational database model, speeding up access to the data and allowing considerably more advanced queries to be made, such as reporting the daily mean of TCP bandwidth over the last seven days. The new architecture is shown in figure 7.



<u>Network Monitoring Tools</u>	<u>Tool Wrapper Scripts</u>
1a = ping	1b = PingER
2a = iperf	2b = IperfER
3a = UDPmon exe	3b = UDPmon

Figure 7: GridMon “version 2” architecture

The architecture is best described by considering the performance measurement workflow:

1. The primitive, or base, network monitoring tools ping, iperf and the UDPmon executable are run by the PingER, IperfER and UDPmon wrapper scripts.
2. The wrapper scripts output the tool data into a single file, as pre-formatted SQL INSERT statements.
3. The db-agent Perl script is executed, picking up the SQL statements generated by the wrapper scripts and using the Perl DBI (Data Base Interface) module to injects the data into the RDBMS.

The separation of wrapper and db-agent scripts serves two purposes:

1. The database related code is localised within one script, and hence requires minimal changes to the original wrappers scripts.
2. A single connection can push all the results from all three tools to the database, rather than one connection per tool.

The wrapper and db-agent scripts are all executed by cron jobs.

As a further aid to lowering operational overhead, plans are being drawn up to re-equip relevant monitoring sites with a homogeneous set of test systems. This will be coupled with standard package management techniques. Namely, that the monitoring node software is now available as a set of RPMs, and a Yum repository has been created to distribute initial installations and subsequent updates to the infrastructure.

To avoid creating a single point of failure, both the data and access software can be mirrored at another well known location. Our sister laboratory RAL would be an obvious candidate.

Networks for non-Networkers (NFNN)

Our collabatory work inevitably brings us into contact with other network performance researchers and developers, many of whom have

expert knowledge relating to analysing and improving network performance. While our main role is the provision of a UK network performance monitoring infrastructure, it is logical to make efforts to disseminate performance information to the scientists and researchers who it can benefit.

To this end, and building on the success of a first event run in 2004, the *Networks for non-Networkers 2* workshop [6] was organised at the NeSC. Aimed at people working at the technical level in high-bandwidth dependant science, the workshop gave attendees an introduction to computer networks and the performance issues surrounding them, with topics ranging from TCP, to LANs and end-user systems. Demonstrations of relevant tools and performance issues were also available during the breaks.

Feedback was very positive. Over half of the sixty attendees completed evaluation forms, and the results show that 97% rated the workshop at good or better, while almost 90% would recommend it to colleagues.

Conclusion

The last year of the project has gone well. GridMon has again been involved with the work of various international groups, and in some cases has lead relevant efforts. This not only ensures that the UK is well represented in such activities, but is also helping GridMon to evolve into a “best of breed” monitoring solution, building on the work of the GGF, EGEE, Internet2, SLAC and others, acknowledged leaders in their respective fields.

The next 12 months will see continued contributions to such Grid networking efforts, but with a strong push to re-establish a UK performance monitoring infrastructure with NM-WG compliant Web Services access, and a new Web portal to take advantage of the centralisation of data in a relational database.

Acknowledgements

The work described here is supported by the Gridpp2 project and the JISC. The initial GridMon infrastructure was closely coordinated with WP7 of the EDG project [7], and benefited from the IEPM work at SLAC[8].

References

1. GGF GHPN-
RG:<https://forge.gridforum.org/projects/ghpn-rg>
2. B. Lowekamp, B. Tierney, L. Cottrell, R. Hughes-Jones, T. Kielmann, and T. Swany. *A Hierarchy of Network Performance Characteristics for Grid Applications and Services*, Global Grid Forum, 19 June 2003: <http://www.didc.lbl.gov/NMWG/docs/draft-ggf-nmwg-hierarchy-00.pdf>
3. Network Weather Service: <http://nws.cs.ucsb.edu/>
4. piPEs: http://e2epi.internet2.edu/E2EpiPEs/e2epipe_index.html
5. *Network Performance Monitoring Prototype Architecture and Design*, EGEE JRA4, EGEE-MJRA4.3-575484-v1.2.doc, 18/04/2005: <https://edms.cern.ch/document/575484>
6. Networks for non-Networkers: <http://gridmon.dl.ac.uk/nfnn/>
7. EDG WP7, Network Services: <http://ccwp7.in2p3.fr/>
8. IEPM-BW: <http://www-iepm.slac.stanford.edu/bw/>