Grid Services in the UK and Beyond

Mark Baker and Hong Ong,
Distributed Systems Group,
University of Portsmouth, Portsmouth, UK
E-mail: *Mark.Baker@computer.org*

Rob Allan
CCLRC e-Science Centre
Daresbury Laboratory, Warrington WA4 4AD, UK
E-mail: *R.J.Allan@dl.ac.uk*

June 2005

Abstract

This report summarises recent work, principally in the UK but also including examples from the rest of Europe and USA, using the emerging service-based Grid architecture. Many of the comments and conclusions arose during the UK OGSA Workshop that was held at the University of Westminster, March 22nd and 23rd 2005. Examples are based on presentations at this workshop. The agenda and presentations from the workshop, including the majority of presentation materials, are available on line at the OGSA Testbed [1]. Links to information about Grid services, projects, and related papers are included in the rest of this report. Since this meeting took place there have been a number of discussions to consider how active UK groups can best collaborate to consolidate and extend best practice and functionality of existing service-based Grids. Some suggestions are provided in the concluding section.

[This document is submitted as a UK e-Science Technical Report.]

1. Introduction and Background

This report is based principally on research and development undertaken during the Portsmouth-led OGSA Testbed project, which was funded by EPSRC, and ran from January 2003 to December 2004. This is a report based on the presentations, and related discussions, of the invited speakers about their projects, past experiences, and future goals, as well as other sources list below:

- The OGSA Testbed: the e-Science funded OGSA Testbed, led by Mark Baker (University of Portsmouth) [2];
- The OGSA Evaluation: another e-Science funded project to evaluate OGSA, led by Prof. Wolfgang Emmerich (University College London) [3];
- ETF GT4 Evaluation: an evaluation lead by Terry Harmer (Belfast e-Science Centre) on behalf of the UK Grid Engineering Task Force [5] This provides input to the e-Science Core Programme and OMII, the Open Middleware Infrastructure Institute;
- Grid Services Town Meeting: UK e-Science Core Programme meeting about the future use of Grid and Web services on 31st January 2005 [6]. This was entitled: *Defining the next Level of Services for e-Science.*

In this report we first document the details of the talks and discussions that took place at the UK OGSA Workshop in March 2005 at which the two EPSRC-funded projects were represented. An outline of the technologies used is provided where appropriate. We then review this alongside other work and provide conclusions including a recommendation for future work.

2. The OGSA Testbed Workshop

The UK OGSA workshop, whose theme was, *Experiences and Observations using the Emerging Grid Infrastructure and Standards for e-Science*, was held at The University of Westminster on 22nd-23rd March 2005. More than fifty researchers, developers, and users from the UK and EU community took part in this event. The keynote speakers were Prof. Peter Coveney from University College London and Mr. Franco Accordino from the Information Society and Media of the European Commission, Belgium. All participants considered the workshop to be a great success.

Day 1: Tuesday March 22nd 2005

Prof. Stephen Winter (University of Westminster) gave the initial welcome to the delegates and made some opening remarks. He noted that the Grid technologies are being taken up not only in academic but also in commercial areas. Hence, it is important that we report and share our experiences particularly in the use and development of Grid services extending the previous capabilities of Web services. He then gave a brief overview of the workshop agenda.

Mark Baker (University of Portsmouth) outlined the aims of the workshop and talked further about the agenda. He noted the Workshop is basically to conclude the e-Science funded OGSA Testbed project but was inclusive and aimed to give a summary of the current state of Grid services being used in the UK and the view of practitioners. The aims of the workshop were thus to report on practical experiences rather than give a "sales pitch" about individual projects. An objective of the workshop was to share common experiences and discuss what needs to be addressed, for instance in future work in the UK. The workshop agenda, which was outlined, was based around a discussion and analysis of Grid issues, recent experience, and potential UK and European solutions.

Session A: Issues - Chair: Prof. Peter Kacsuk (SZTAKI/ Westminster)

Real science on computational Grids
Prof. Peter Coveney (UCL) [Keynote speaker]

The high-profile RealityGrid project aims to do "real" science, not depend on "a promise without delivery". Where a typical Grid project said it would use the Grid to do science, it normally ended up with just a demo but not a real application. The RealityGrid promised and delivered the science that was originally proposed. The Grid definition emphasises that it is about distributed computing performed transparently across multiple administration domains. The key being transparency, the computing refers to numerical, symbolic and visualization parts. The problem was seen to be that no so-called "Grid" exists today that meets this definition. The Grid is not about launching isolated jobs onto medium-sized or “big iron” computers since there were no value-added benefits to this. It was observed for instance that there are a number of "informatics" projects not doing much to advance the Grid. The key to the success of a production Grid is to be "stable, persistent, and usable", whereas current often Grids last no more than a demonstration. Also, systems like the UK’s NGS and the US TeraGrid use the Globus Toolkit v2, which is complex to install and lacks flexibility. It is perceived that the e-Science core programme is run by, and for, computer scientists, and also that there is a cult of demonstration, where things work today (hopefully) but not tomorrow! This is stifling the uptake by “real” end-user scientists.

A description of RealityGrid followed to provide background. It was noted that there is funding for the computational sciences, but no computer resources for doing the actual science. In addition, there is a question over what can be seen as fast track versus deep track funding. Fast track funding is not generic and emphasises functionality, whereas deep track funding must address fundamental computer research where survival of fitness is key as it is in any other science domain. The question was asked as to which is most useful?

The speaker then went on to talk about the difficulties of building with GT2; it was observed that the Globus Toolkit is a moving target, with GT3 now available and GT4 following soon. Developers of complex applications find it hard to follow this evolution with constantly changing APIs and functionality and “unreliable” software. It was noted that a lightweight middleware alternative was needed. The headaches of administering Grid software were also discussed; where deployment takes effort and time, security lacks usability and is therefore not being taken seriously.

The RealityGrid project revolves around an application called LB3D, a 3D Lattice-Boltzmann simulation that is “embarrassingly” parallel, but steerable enabling the scientist to interact with the simulation and change parameters in real time. The RealityGrid partners organised the TeraGyroid experiment around the Super Computing 2003 Conference. The motivation of this was (and is) to combine resources from the USA and UK via the Grid to facilitate the study of defect dynamics in liquid crystal surfactant systems. The project created both demonstrations and achieved real scientific data, and 18 months later researchers are still looking at the outcomes. It was noted that, whilst the resources of the TeraGyroid had been extremely useful, they where ephemeral.

Other applications of LB3D are in calculating drug-binding affinities, which are also suitable for an HPC Grid – the RealityGrid is using the Grid for serious research with potential commercial interest. The project is working with TeraGrid and NGS to ensure compatibility for future, but efforts are currently held back by the state of existing middleware, e.g., non-uniform scheduling, policies, and complex Grid middleware.

It was asserted that there was a need for a lightweight Grid middleware, such as OGSI:Lite and WSRF:Lite which had been created by partners in the project at University of Manchester [21]. UCL has been developing WSRF-based Environment for Distributed Simulation (WEDS) [7]; a Web Services environment for distributed simulation, which may be supported by OMII.

The concepts inherent in WEDS are similar to the WS-I concept of a "Basic Profile". As such, WEDS compliance does not imply WS-I compliance; here it refers to WS-I’s concept of a Basic Profile.

A UK NGS Perspective
Stephen Pickles (University of Manchester)

The UK Grid Operations Support Centre, GOSC, is a distributed "virtual centre" providing deployment and operational support for the UK e-Science Programme. The NGS supports a production Grid. It was recognised in the Level-2 Grid work in 2003 that we cannot achieve a production Grid simply by connecting sites together in an *ad hoc* fashion. User requirements are important, for instance to gain access to the resources, the user might want to just sign-up via a Web browser. Actual login can be done in the conventional way or via digital certificates. The current NGS system uses GT2.4.3 including GRAM, GridFTP, GSI-SSH, MDS/GIIS/GRIS, and also SRB (Oracle) and OGSA-DAI currently as experimental services.

Money has been spent on "Informatics" projects, but it is not clear how it transcends to such a production Grid on which there are now a number of established projects and Virtual Organisation of users. The NGS currently comprises a set of four core clusters funded by JISC's Committee for the Support of Research. Two clusters are for compute-intensive work and two for data-intensive work, and there are plans for future expansion. The NGS also plans to use OGSA, which is important as it addresses the basic issues of how to build a Grid and also how to factor capabilities into services. OGSA is beginning to deliver the services as described in the specification. Service-based Grids are the trend in Europe, for in stance in EGEE the pan-European *Enabling Grids for e-Science in Europe* project with which the NGS aims to be compatible.

Maintaining compatibility between WS-* systems will be hard since we need to care about robustness and reliability over cutting-edge technologies. It is impossible to support everything but there is no clear implementation winner yet. NGS has to work with peer groups such GGF, OMII, Globus and EGEE, GridPP, and TeraGrid, as well as managing divergent forces responding to the needs of different groups. The NGS is managing middleware evolution via the ETF testbed, which has recently carried out an independent evaluation of OGSA. Grid deployability, reliability, capability, and usability are key criteria being using for such testbed evaluations.

Questions:

- Is the GGF doing its job? Yes and no - yes it needs more volunteers.
- How about the specifications?
- Are they coming up fast/soon enough? Yes and no!

OGSA-DAI - Responding to a Changing World
Mike Jackson (EPCC, Edinburgh University)

OGSA-DAI is OGSI-compliant, providing Data Access and Integration services, a Service Group Registry and Data Service Factory. The OGSI to WSRF transition is a problem as it impacts the OGSA-DAI project, which has leveraged GT3 (an OGSI implementation) at all levels. The solution is to maintain three flavours of OGSA-DAI as shown in Figure 1.

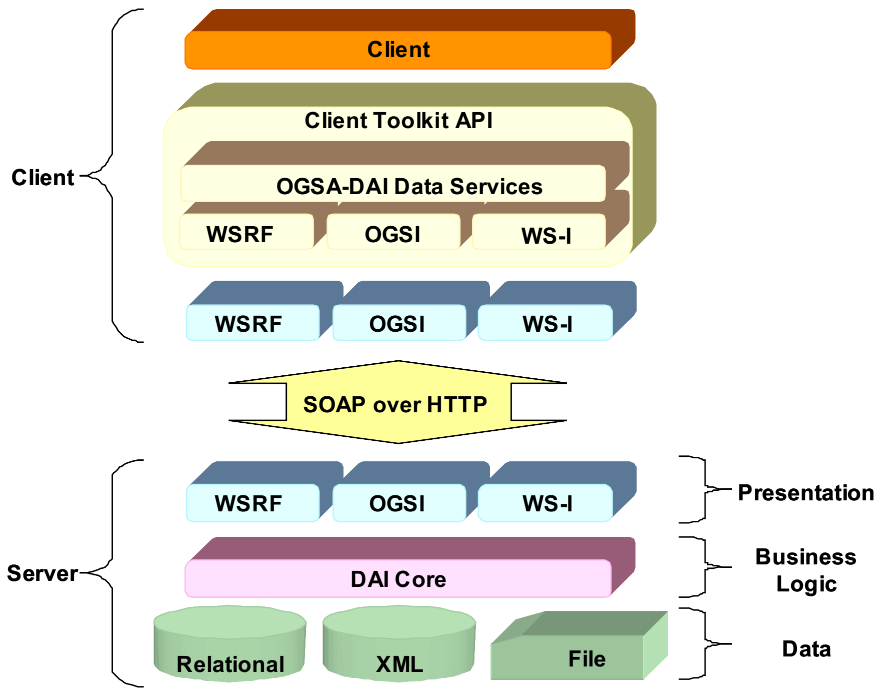

Figure 1: The OGSA-DAI Architecture Now

Within the project, there are currently various limitations, for example no GridFTP or other Grid data transport mechanisms. There is also a need to re-factor OGSA-DAI, restructure the CVS tree to incorporate the new source, and make a variety of changes to OGSA-DAI to support WSRF.

With respect to the service-based architecture, it was noted that there is a large overhead supporting three branches of OGSA-DAI in terms of development, testing, and support. The possibilities:

- OGSA-DAI/ OGSI (GT3);
- OGSA-DAI/ WSRF (GT4);
- OGSA-DAI/ WS-I (e.g. OMII stack).

The developers want to provide an extensible, pluggable architecture, which allows multiple specifications to be supported without having to implement the specifications, and are thus prepared to sacrifice functionality and performance at present.

Questions

- Do you have time to do any more development work? Not really.
- How many users OGSA-DAI support? Do not know.
- Support for three versions, is a waste of resources? Yes.

Session B: Recent Experience - Chair: Rob Allan (Daresbury)

The OGSA Testbed: Views and Opinions from the Trenches
Mark Baker (University of Portsmouth)

The UK OGSA Testbed was a partnership between the universities of Manchester, Portsmouth, Reading, and Westminster, and Daresbury Laboratory, with SZTAKI (Hungary) and Southampton as associate partners. The project received one year of funding from EPSRC, which is approximately half an FTE at each site. It was funded to be in the vanguard of testing and "experiencing" emerging OGSA-based Grid middleware with the aim of feeding back what was learned into the e-Science community. Each partner was responsible for porting applications to the OGSA Testbed set up between the sites.

The original Testbed strategy was tightly coupled with OGSI. This was revised in early February 2004 in light of the announcement of WSRF. The updated strategy was based on gaining as much knowledge and experience as possible in order to provide extensive feedback to the e-Science community. In particular:

- The Testbed platform followed the GT rollout, going from GT3 to GT4 (WSRF)
- With applications we followed multiple paths ending up with WSRF:
    - Follow rollout of GT3 → GT4,
    - Web Services + GT2.4 → GT4,
    - OGSI::Lite → WSRF::Lite.

Globus 3.2.1 was first installed, tested, and kept running at all sites. These cross-site logins were used for software installation and management. The project has provided feedback on bugs/ features of GT3 to the Globus developers. The consortium has had regular discussions and meetings, which have resulted in a number of joint papers and other project proposals. There is a project Web site [2] with private pages secured via Grid Site (access management software from GridPP). The project has produced and published various documents, including numerous reports and papers. All partners worked on porting their applications and deploying them across this testbed.

Various comments were made on experiences with Globus, including how documentation has improved over the course of the project, the time to compile each new release, setting up required security and certificates, the test programs and the GT3GITS scripts [9], aspects of reporting bugs/ features to the Globus developers, the implication of regular releases and updates to Globus, firewall issues with the need for many open IP ports, the stability of GT in the Tomcat container, and the implication of the January 2004 re-factoring exercise which led to WSRF. Details of these comments can be found in the associated talk [10].

It was noted in its defence that Globus is an ambitious effort that intends to produce middleware that satisfies the needs of wide-area distributed applications. Currently the software distributed was good for people who are familiar with GT, but a total disaster for a "newbie" as there is a very steep learning curve. Globus is a worthy effort, but it is still research software, with all the implications of such. It should be noted that many projects are staying with GT2.4, as this provides an understood and stable platform with interfaces more familiar to scientists. It was felt that the partners in the OGSA Testbed project had spent too much time chasing releases of software. However our current experience means that moving to GT4 will be straightforward and we can provide advice to others in the e-Science Programme. It is clear that the Programme had moved to GT2 too early (probably by around 12 months), GT3 was deprecated, and we are now (early 2005) waiting for GT4 to stabilise. The move from OGSI to WSRF was thought to have been done for the right reasons, but its announcement following a period of apparent secrecy had confounded the community and frustrated many developers. It is now suggested that we need alternative OGSA implementations, and emerging systems such as WSRF::Lite or UNICORE will help this diversity. Production Grids need "hardened" software; otherwise the Grid will encounter an "AI Winter" - hopefully OMII can address this area requirement. It was commented in this regard that there was a need for money to develop a robust middleware infrastructure, not just money to do further research in future infrastructure and applications.

In conclusion, there is currently much confusion as to which "standard" to follow. Many developers in the UK are developing Web services mainly just using SOAP and WSDL. The UDDI registry, which is part of the WS-I stack, however does not satisfy the complex needs of a Grid information service.

Establishing an Inter-Organisational OGSA Grid: Lessons Learned
Wolfgang Emmerich (University College London)

The UK OGSA Evaluation was a one-year project that started in December 2003. It had partners from UCL, NeSC, NEReSC, and the LeSC. UCL deployed GT3.2 on a heterogeneous set of hardware and operating systems (Linux, Solaris, Windows/XP). The GT3.2 installation experiences are discussed in the accompanying slides [3]. The project created GTMark a configurable benchmark used to test results, reliability, and security. The security model was found not to scale to a large testbed, so project partners looked at deployment via HP's *SmartFrog* software, which worked well inside a node, but was impossible to use across organisations as the *SmartFrog* daemon needed to execute actions with root privileges, which some site administrators did not agree to. It was also found that *SmartFrog*'s security infrastructure was incompatible with that of GT 3.2.

It was noted that installation efforts need to be reduced significantly, such as with binary distributions for a few selected hardware and o/s platforms. Standards compliance is important, and there is a need to track standards by all means. It was thought that a "management console" was important to add or remove grid hosts. Also it was thought important to be able to monitor the status of Grid resources across organisational boundaries. Finally it was thought a more lightweight security model is needed.

GT4 Experiences
Terry Harmer (Queen's University Belfast)

This was a report on work carried out by the UK Grid Engineering Task Force in evaluating GT4. Previous Globus software releases have had the reputation of being error-prone to install and configure. GT3.9.x is, surprisingly, relatively easy to install. GT4 experiences have been positive and they found installation easy, especially on RedHat and SUSE versions of Linux.

With respect to security, authorisation can be applied at different levels of abstraction (service-level, resource-level, etc.). This is necessary for the structure of WSRF but allows more flexibility in security solutions. In general, security seems to be moving toward a more rigorous yet user-friendly implementation. Also the documentation is well developed and the standards seem to be more widely adopted allowing easier development of secure services.

Belfast have been developing *Gridcast*, which is a prototype Grid-based TV/ Radio broadcasting infrastructure developed with the BBC. It was an active GT3.0.2 deployment with 300k lines of Java code, which used various services including the Index services, RLS, and GridFTP. The exercise to move this application to GT9x was mostly successful, but required some changes to the WSDL

There were a number of conversion issues. There was an emphasis in the work to change as little as possible of the existing application. However, the application may contain operations and libraries that are no longer supported, although this has not proven to be much of a problem and most of the issues were overcome relatively easily.

In conclusion it was stated that GT4 software was now available on a wide variety of Unix operating systems, and in fact quite a bit also runs with Microsoft. Installations were straightforward with few problems. Installation documentation was of a high quality, well structured and accurate. GT4 has significant package capability improvements on GT3. GT4 simple services are relatively easy to develop and well documented in a programmers' tutorial. GT4 service development requires the management of a larger collection of files than GT3 but this not a significant effort. GT4 conversion from GT3 required effort in changes to service interfaces but required relatively few changes in the service code in the examples considered. GT4 services demonstrated significant improvements in performance and reliability over their GT3 versions.

However there are shortcomings with GT4. Currently there is the lack of documentation that will permit large-scale GT conversion, deployment, and GT4 development. The creation of simple services is well documented but the detail required for large-scale GT3 conversion or new development is not currently available. The need for documentation and examples is recognised by the Globus Alliance, so perhaps this is also an opportunity for the UK.

Questions

- Is it possible/desirable to move GT2 GRAM to GT4 GRAM? Not yet, many things still not clear e.g. SOAP.
- How about security? No security.

Session C: Evening Session

Panel: Rob Allan (Daresbury Laboratory), Stephen Pickles (University of Manchester), Savas Parastatidis (University of Newcastle), and Abdeslem Djaoui (RAL/EGEE), Moderator: Mark Baker (University of Portsmouth).

The question posed to all panel members was: *What standards and technologies will Grid use in five years time? Justify your answer.*

Savas Parastatidis (University of Newcastle)

The question is where did we go wrong? The Grid vision has been around for some time. So why are we still discussing Grid middleware? The GGF spent 18 mouths to move OGSI to WSRF. But, there has been no discussion about real high-level services.

What have we learned from campus Grid, multi-sites (intranet), multi-domains (VOs)? We have learned that standards need stability - no options no problem; what happens with more than one option? We need to standardise but typically standards bodies and commercial companies do not care about what other standard is around. The solution is to wait and see, depend on the community to bring stable software, even if it has to reinvent the wheel.

What does the future hold? Probably much of the following: protocol-based integration, declarative or intention-based programming, as well as automatic contract negotiation, auditing, non-repudiation, and virtual organisations. For the Grid there should be a focus on stable, production-quality deployments (finally!) rather than experimenting with crazy infrastructure-related ideas.

At "GridCosmos" 2010 the world will be given new versions of specifications, which will bring the distributed computing communities together again. The Grid community will finally come to its senses and go with what standards/ implementations the industry supports rather than trying to create their own. By 2030 the vision of the Grid being hidden from its users will still be far away because "interplanetary computing" will be attracting all the research money!

Question

- About letting the industry figure out the standard. P. Coveney: does this mean no more input from science to standards?

Stephen Pickles (University of Manchester)

Figure 2 shows a "view" of the probability of the various standards being around in five years time.

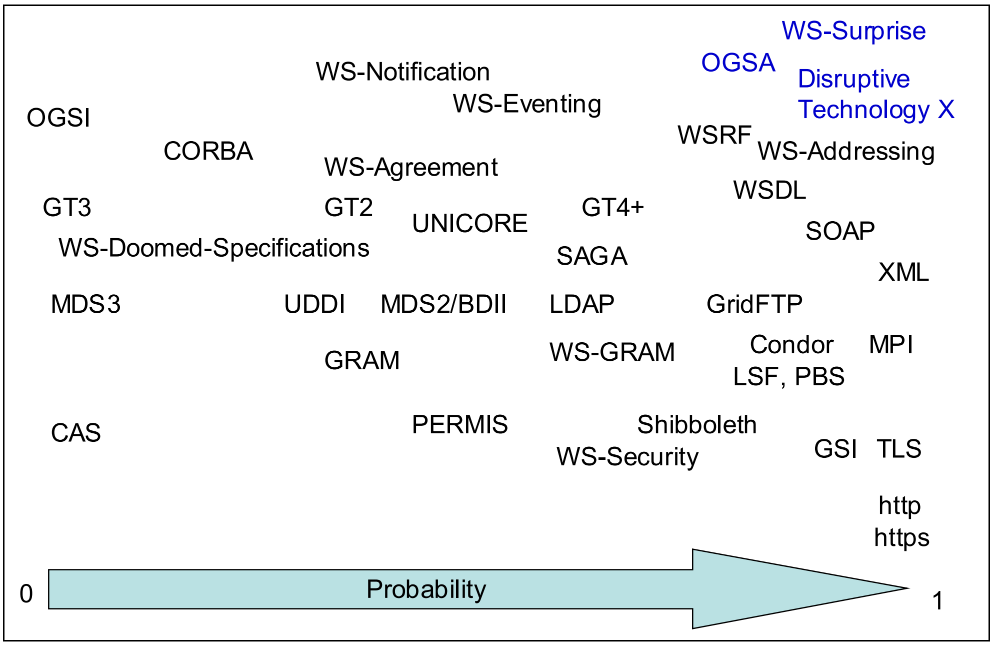

Figure 2: Five Years from Now

With regards to workflow and the Semantic Web, it is hard to see where these are going. Technologies, such as XML, SAOP, MPI, TLS, and WSRF will survive. It is assumed that OGSA will evolve and survive, i.e., SOA will survive. Some older technologies might still be used so people need to support them.

Question

- Do we spend too much time talking about standards? Depends who we are! Some people should take what works and move on - but someone has to talk about it.

EGEE strategy on Standards and Web Services
Abdeslem Djaoui, EGEE UK representative Project Technical Forum

EGEE is about production, not R&D - EGEE has to deploy production quality middleware now. We believe that Web Services will be a key technology for gLite (EGEE Grid middleware), but need to convince users (for instance SOAP performance is perceived to be an issue, particularly in Java). Since standards have not solidified yet, EGEE is taking a cautious approach towards WS-*, no WSRF, not even WS-Addressing. This was not a problem in the LCG2 project, which has a closely-knit community. EGEE are committed to WS-I (Basic Profile) compliance to maximise interoperability. The benefit to users is apparent yet. Further WS-* standards will be used as their maturity is demonstrated.

With respect to standards for e-Science, they will be DIY. For top-level services such as GGF's OGSA, a number of profiles based on stable specifications will be available within 1-2 years. There is a general agreement to move towards standards for execution, data, and information management. The second level standards such as resource management; currently deployment, configuration and operations are very demanding. Automation is needed in order to reduce cost and complexity of managing the system. WSRF will be adopted (not necessarily as a result of technical merit). Many implementations (and backers) already exist

It was noted that Microsoft is expected to be a corporate member of GGF, the Global Grid Forum.

Web and Grid Services in Virtual Environments
Robert Allan, Daresbury Laboratory

The work currently funded by JISC was outline, much of which is summarised in the discussion section below. Services, tools and components are being developed in projects spanning e-Research, e-Learning and Information Management. It is recognised that many of these overlap. There are however issues of how best to deliver such services to end users, with a current emphasis on Web-based portals and a Service Oriented Architecture.

Each component must be investigated further. The underlying service it provides must be given a semantic description, interface specification, and implementation using middleware. This requires multi-domain agreement, particularly in the service description. Individual services might expose multiple methods implementing complex functionality.

Issues, which must be addressed, include:

- Functionality - does it match user requirements
- Granularity - re-usability vs. performance
- Security - secure, but accessible within existing mechanisms - Shibboleth?
- Standards and Specifications - internationally agreed, OKI?
- Middleware - deployable and resilient, but giving good performance for both services and clients - WS-I, WSRF, etc.
- Need for non-Web service protocols - e.g. GridFTP for binary data, streaming audio/ video etc. Peter Coveney mentioned W3C interest in binary data transfers.
- Legal - licensing, confidentiality and IPR issues are important
- Usable - pervasive, persistent and accessible

Work on these issues is only just beginning with the JISC working group for the e-Framework for Education and Research. For more details of ongoing work see [17].

The Grid and its associated middleware (e.g. Globus) links computers and data sources across institutional boundaries. Typically, end users do not want to be part of the Grid, they want to access its services from any device and from any here! This is what I refer to as the Client Problem. Institutions have firewalls to protect their services from being compromised. Grid middleware does not work well with firewalls (ports need to be open). We therefore proposed to use lightweight Web services to connect users in "polling" mode.

It is increasingly important to be able to re-use underlying services, partly for reasons of economy but also because they can be tested and will then be trusted. It is however important for the end users to be presented with services via a familiar environment, be that a Web-based one or a desktop toolset.

We are seeking the appropriate technology to realise this vision. We are currently focussing on standards such as JSR-168 and WSRP. Implementation of such a service-based framework is supported by programmes developing middleware, for instance for security. Extensions to the set of chosen standards will be documented and implemented in an "integration API". At some stage this will be compared to the OKI interface definitions being adopted in Sakai.

Web services technology is currently a strong contender for providing thin client interfaces to link a range of environments to remote services. Grid services and peer-to-peer technology are being considered. These middleware technologies give a middleware framework which complements the service component framework described above. JISC are interested in a report on our testbed and the conclusions of this workshop.

Question

- How can P2P work in the Grid? Using a P2P registry for discovery for instance.

Day 2: Wednesday March 22nd 2005

Session D: Solutions - Chair: Stephen Pickles (University of Manchester)

Legacy Applications on a Service-oriented Grid: GEMLCA
Peter Kacsuk (SZTAKI/Westminster)

The objective of the GEMCLA (Grid Execution Management for Legacy Code Architecture) project is to deploy legacy code applications as Grid services without re-engineering the original code and with minimal user effort. This will enable users to create complex Grid workflows where components are legacy code applications and the functionality is made available from a Grid portal (in this case the P-GRADE portal). The development roadmap of GEMLCA/ P-GRADE is shown in Figure 3.

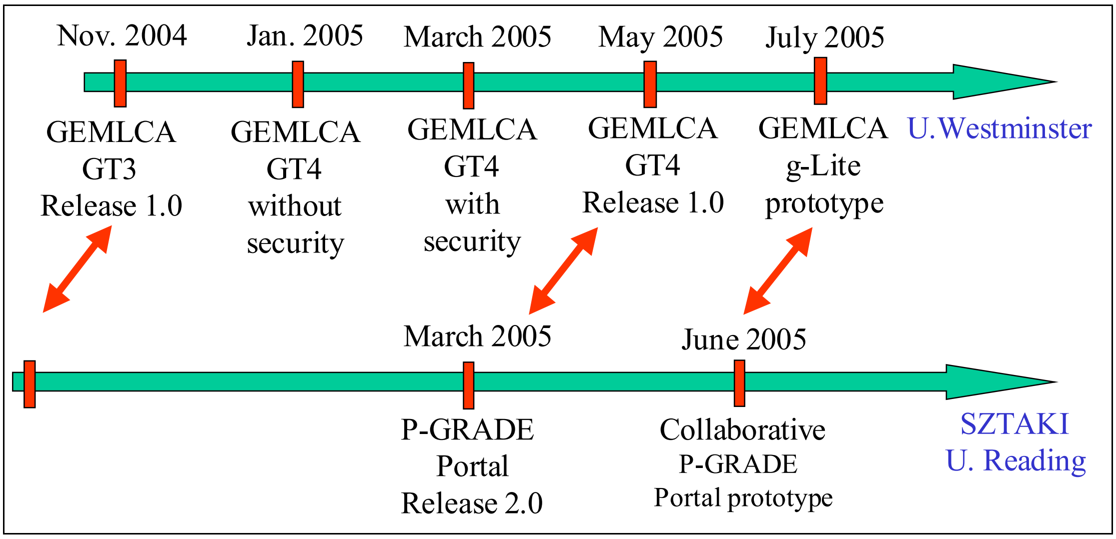

Figure 3: GEMLCA/ P-GRADE Portal Status and Development Roadmap

GEMLCA enables the deployment of legacy code applications as Grid services with minimal user effort. GEMLCA is integrated with the P-GRADE portal to offer user-friendly development and execution environment. The integrated GEMLCA P-GRADE solution has successfully been demonstrated using a traffic simulator and other legacy applications.

In an ideal OGSA-based Grid environment users are able to access predefined Grid services through a high-level user-friendly Web portal. More than that, users are not only capable of using such services but they can dynamically create and deploy new services in a convenient and efficient way. These services can be either specifically designed ones or legacy programs that are automatically deployed as Grid services when desired. In order to achieve this ideal scenario and to provide the high-level Grid application environment mentioned above the following tasks should be solved:

- A means to automatically wrap legacy codes and then deploy them as a Grid services.
- Methods to make these Grid services accessible and usable in workflows supported by a user portal.
- The new portal should be derived from an existing Globus Toolkit version (GT2) portal in order to minimise the users' learning curve moving from GT2 to OGSA-based Grids.

In GEMCLA, in order to access a legacy code, the user executes a client, which creates a legacy code instance with the help of the legacy code factory. Following this, a GEMLCA resource submits the job to the compute server through GT3 MMJFS using a particular job manager. A GEMLCA resource is composed of a set of Grid services that provides a number of interfaces in order to control the life cycle of the legacy code's execution. The GEMCLA system can be deployed in several user containers or Tomcat application contexts. GEMLCA has a three layer architecture: the first, layer offers a set of Grid Service interfaces that any authorised client can use in order to contact, execute, get the status and results from the legacy code. This layer hides the second core layer that deals with each legacy code environment and their instances as legacy code processes and jobs. The final layer, the backend is related to the Grid middleware where GEMCLA is being deployed. The current implementation is based on GT3 but this layer can be easily updated to any new standard, such as WSRF.

Questions

- Does wrapping cause overheads; what is the performance? Not really an issue, depending on how many users.
- Is it a generic approach? Yes.
- Can interactive legacy codes work? Not yet (currently, not supported).

WS-GAF - The story so far and a brief look into the future
Savas Parastatidis (NEReSC, University of Newcastle)

The first question is "what is a Grid?" There is a plethora of definitions, which all reflect the viewpoints of stakeholders. The Newcastle team define it as an *Internet-scale, service-oriented distributed computing infrastructure (using Web Services).*

The Grid promises to build applications that span organisations. Using this we can create virtual organisations, with seamless integration and hide (virtualise) or share use of resources, network, and infrastructure. Web Services promise to be the glue for heterogeneous platforms, applications and systems. We aim to provide cross- and intra-organisation integration with standards-based distributed computing giving interoperability and service composeability; all based on the concepts of Service Orientation.

Service Orientation is built around the concepts of "service" and "message". A service is the logical manifestation of some physical or logical resource (such as a databases, program, device, or even human user) and/ or some application logic that is exposed to the network. A message is a unit of communication for exchanging information. The sending and receiving of messages facilitates all communication between services. It should be noted that service orientation is not simply equal to Web Services.

The Web Services Grid Application Framework (WS-GAF) project and ideas were motivated by the GT3 OGSI release. Thus followed a WS-GAF paper, the imminent WS-RF release and the community's concerns over WS specification instability. The focus for the WS-GAF project was on creating applications and demonstrating ideas for Internet-scale applications. It makes a distinction between production and experimental deployments. The WS-GAF ideas and approach became part of the UK e-Science Strategy white paper for a Web Services-based National Grid infrastructure [22].

The WS-GAF strategy is about simplicity and minimalism, along with composeability. There should be no changes in the semantics/ characteristics of Web Services, with a synergy with existing WS specifications and practices. There was a policy of avoiding the effort involved in influencing the WS community and the risk associated with not being able to influence the WS community. The focus was away from the low-level "networking" to the more important high-level services (application-domain specific) thus filling the gap between infrastructure and application. Overall, we did not need anything other than existing industry standards and open-source WS tools and platforms (e.g., ASP.NET, Axis, etc.). We aimed at producing educational material, having industry investment, having good performance and scalability, and encouraging performance through large granularity with loose coupling (richer, fewer message exchanges).

With WS-GAF, we wanted to build Grid applications using Web Services in order to demonstrate our claims. Therefore our aims were to define the characteristics of a "typical" Grid application, demonstrate the applicability of the WS-GAF approach in building Grid applications, and finally learn from the challenges of constructing a truly global, distributed, scalable, loosely-coupled application. The tools used were .NET 2.0 Beta 1 and .NET 1.1, VS.NET 2005 Beta 1 and VS.NET 2003, Web Services Enhancements 2.0 SP1. The main test application involved searching for astronomical White Dwarfs. It involved Jim Gray's SkyServer and Edinburgh's SuperCOSMOS archive and utilised computational resources and visualisation.

The output from a year of work on WS-GAF includes a number of journal, conference, and other publications. Future research will include investigating contracts and service description. We want to try to capture behaviour through models and make it available as part of the service description. We also wish to create tools and apply ideas throughout NEReSC projects. In addition we wish to build interesting applications, look at P2P for Internet-scale, service-oriented applications and develop SOAP Service Description Language (SSDL).

It is clear that service-orientation is a good paradigm for Internet-scale computing when used appropriately and Web Services are the preferred implementation technology. It should be emphasised that Web Services technologies do not implicitly mean automatic scalability and loose coupling and that they need to be accompanied by good architecture design. There should be a focus on a message-based SOA and protocol-based integration. We recommend that the Grid community fully embrace

Web Services technologies (without changing them) and build concentrate Grid applications using them.

Linking Portlets to Services - Creating a Virtual Research Environment.
Rob Allan (Daresbury Laboratory)

The point was made that users are interested in application-level functionality, not the underlying Grid services. Portals give us a familiar Web-based interface. It is pervasive and hopefully persistent! Requests "pushed" from client through port 80 or 8080 (Web cache). A four-layer approach is used for development at Daresbury, this comprises: client interface; Web server front end; Web server back end (could be separated by a firewall for additional security); Grid service. Examples of portals include: GridSphere, OGCE-2, JISC Sakai VRE, P-Grade GEMLCA. All using are portlet standard JSR-168 and potentially WSRP to ensure re-usability of the client-side tools.

Developers want a "transparent", highly functional way to link remote resources (services?) into their existing or new applications. "Heritage" apps are written in Fortran or C, not Java. We are therefore developing a lightweight C library using Web services. Need to "push" requests through port 80 or via Web cache proxy. Callbacks must therefore be done in polling mode.

The back-end services we are using are the same ones as in the portal, so both interfaces can be used, e.g. we are wrapping GT2, SRB etc. There are possible issues of performance (multiple hops) and functionality if non-http protocols are required (e.g. using Webstart for GridFTP onto the desktop...) Examples of lightweight programming interfaces include: JISC VRE GROWL project, OGSI::Lite, WSRF::Lite, and WEDS?

Whatever happens to the Grid, the user interfaces must be acceptable. Not many scientists are working in an object-oriented, let alone service-based way!

Question

- Service framework concepts appear to be in parallel with OGSA? Yes, if OGSA is successful, people will use it.

Building Middleware to meet Evolving Requirements
Steven Newhouse (OMII, Southampton)

Where are we (the UK) now? Actually in the middle of a five year and £250M programme, where there are 100+ projects applying e-Science technology across all branches of scientific research. Most of the initial projects are however coming to an end, so there is a question of how their outputs are going to be taken forward? There are now a bunch of new projects starting up, so there is another question as to what technologies they should use? Finally there is an issue of how to grow from research to production and what technologies will provide the best robust and stable infrastructure?

The “trailblazing” experimental projects are concluding. During their time the middleware has grown and changed. They are developing or delivering tools against evolving middleware. The users and uses have grown and changed. We are moving from experimentation into production, which was the reason for the Core Programme establishing the OMII (Open Middleware Infrastructure Institute) and GOSC (Grid Operations and Support Centre).

There are multiple diverse requirements for a production Grid. We want to do simple things very easily, for example execute applications remotely with input and output, access data (for example files and databases). There is a need to develop with the latest standards. We also want robust services and stable infrastructure and initiate software engineering processes, releases, and provide support as well as having interfaces (APIs) that will last for several years. We want these aspects across all platforms at once, and ideally we do not want to change what we do now.

In the UK, OMII has been established as a source of high-quality software for the Grid. The Grid Engineering Task Force is now undertaking trial deployments of prospective Grid software, which will be deployed on the production National Grid Service. The UK e-Science community will provide demonstrations of real science on real Grids.

OMII will deliver its mission by providing a software repository of Grid components from e-Science projects. It will re-engineer software, hardening it and providing support for components sourced from the community. OMII has a managed programme to contract out the development of "missing" software components necessary in grid middleware. OMII's first output is OMII 1.2.0, which provides middleware for a basic file and compute Grid.

The OMII managed programme includes:

- GridSAM (Job Submission and Monitoring service),
- BPEL (Workflow service),
- Grimoires (Registry service based on UDDI),
- FIRMS (Reliable messaging),
- FINS (Notification),
- GeodiseLab (Matlab and Jython environments),
- WSRF::Lite integration,
- OGSA-DAI (Database service),
- WSeSS (Using SSH to tunnel requests to resources).

Question:

- About the authorization process, have you received comments from DataGrid about how things should be done? Yes Medical Grid as well.

Web Services for Grid Computing
Mark McKeown (University of Manchester)

Mark started his talk with various clarifications on the WS-* stack. WS-Addressing allows the message to be routed to its final destination, offers transport independence. WS-ReliableMessaging insures messages are delivered in the order they were sent. WS-CAF provides a mechanism to support more complex interactions than simple request-reply. WS-Notification provides support for asynchronous communication. There are a few others basic things such as SOAP, WSDL, UDDI, BPEL, and now WSRF.

We note that 85% of Amazon Web Service developers using the Amazon API chose to use REST, partly because Amazon REST requests are processed 6 times faster than equivalent SOAP requests. Roy Fielding first described REST, Representational State Transfer, in his PhD thesis [13]. Roy categorised various types of distributed systems and then identified the things that made the Web such a scalable system. He defined a set of constraints, which, if followed, should lead to a massively scalable Grid system. The question is: *is this true?*

Principles of REST are that a resource is anything that has an identity. Every resource has a URI; here a URI is "opaque" and exposes no details of implementation. GET operations are "idempotent", i.e. free of side effects. Any operation that does not have side effects should use GET. All Interactions are stateless. Data and metadata formats are documented and data is available in multiple formats. The representations include links to other resources. It is necessary to document and advertise your service API (possibly using WSDL). It uses available standards and technology.

The "pros" of REST are that it forces the developer to think carefully about how they design a service. REST is established and unlikely to disappear. The focus is on XML and message exchanges not on APIs so the focus moves to the problem rather than choosing WS-* components. Binary data is not a problem. The "cons" of REST are that the architecture may look simple but there is still a lot of infrastructure and technology involved: caches, proxies and lots of RFCs. Also there is at present limited tooling support. There are things you cannot do with REST – but what are those things, and should we be doing them anyway?

It is clear that Grid Computing is distributed computing on an Internet scale and crossing administrative boundaries. Distributed Computing is hard – Grid computing is going to be *very* hard. We do distributed computing, for improved performance, increased reliability and for sharing information. We note that Web Services do not inherently make it easy to build high performance or highly available systems.

Session E: The European Solution - Chair: Stephen Winter (Westminster)

Ongoing Activities and Future Challenges of EU Grid Research
Franco Accordino (Grid Technologies, IST, Brussels)

Three main themes were described:

Focus 1 – Grid Foundations: This theme is about the architecture, design, and development of technologies and systems for building which should be invisible to the Grid. This needs a scale-independent, adaptive and dependable Grid architecture allowing the management of large networked distributed resources characterised by evolutionary, non-functional behaviour. This theme may include agent-based approaches and peer-to-peer technologies. The systems designed must be self-organising, fault-tolerant autonomous systems and virtualised. This theme will include new models and environments for programming the Grid and include semantic and agent technologies for resource brokering/ management

Focus 2 – Grid-enabled Applications and Services for Business and Society: This theme is about the research, development, validation, and take-up of generic environments/ tools. It includes Grid-based environments for dynamic service creation and provision supporting collaborations spanning multiple administrative domains. The theme deals with Grid business models and economics, intelligent tools and interfaces supporting ubiquitous Grid access, as well as Grid-enabled decision support services.

Focus 3 – Network-centric Grid Operating Systems: This theme involves the research and development of new or enhanced fabrics layer for future distributed systems and services. This includes new approaches to system management/ operation, which involves factors such as global state, resource lifetime, replication control, scheduling, synchronization, service response time, security, scalability, and interoperability. The theme has a dual scope; development, testing validation of enhanced fabrics based on existing operating systems (Grid built-in functions or modules leading to a meta-OS) and the research and conceptualisation of new fabrics replacing existing OSs. The aim of this theme is to simplify the use of Grid systems and services, supporting mobility and pervasiveness, enhancing performance and build on EU strengths with existing and emerging embedded and mobile OSs

The Information and Communication Technologies strategy is based on the Lisbon agenda and the Wim Kok report [23]. It is a key component to improving EU productivity and central to mastering innovation, as well as necessary for modernising public services. For R&D in ICT, it is essential to intensify effort and expand strengths, as well as extend scope and seize opportunities, and shape to fit the needs of businesses and citizens. This will necessarily involve all stakeholders and reach out beyond Europe. With respect to Grids, the Web and knowledge technologies, these are drivers for the evolution of the Internet and business IT infrastructures. It is clear that Web services will be a key technology but will not be a panacea (EU has a role to play). The FP7-ICT draft paper discusses software, Grids, security, and dependability.

3. e-Science Town Meeting

A Town Meeting was held by the UK e-Science Core Programme on 31st January 2005 to discuss the topic of defining the *Next Level of Services for e-Science*. The primary goal of this meeting was to explore the present situation with respect to the OGSA and the move of Grid Middleware to a Web Services-based architecture, and to see if there could be some coherent UK position on these issues. The secondary goal of the meeting was to examine the role of GGF, the Global Grid Forum and the importance of the OGSA architecture. In particular, since the UK has funded over forty people to attend GGF and actively participate in its various groups on a regular basis, the meeting could evaluate the value or otherwise of attempting to write down an "OGSA-UK" document to assist GGF in formulating its future strategy.

The meeting was organised in two sessions to address the above two goals. It was introduced by Tony Hey (Director of the UK e-Science Core Programme) who highlighted some of the issues that we are facing as a community in the move towards Web Service Grids. The discussion was guided by a number of invited talks from well-known speakers, including Mark Linesch the new Chair of GGF and

Andrew Grimshaw, GGF Architecture Area Coordinator and leader of the OGSA effort, and panel sessions to consider and debate the issues.

Andrew Grimshaw provided information on the OGSA process in terms of use cases, profiles, design teams, and working groups. He explained their use of “profiles" which define a usage pattern and include appropriate specifications. There are three profiles currently in discussion – Basic (essentially at the level of WSRF), Data, and Execution Management Services (EMS). Andrew emphasised his view that naming is the essential foundation on which distributed systems need to be built and explained that this is also a problem for the Web Services community. He briefly discussed OGSA security where he perceived there to be only slow progress. He emphasised that the OGSA Data and InfoD teams are use case driven. He thought that there was a strong consensus emerging with some issues around metadata and information dissemination still to be resolved. He concluded with an overview of EMS, which is intended to provision, execute, and manage services on the Grid.

The subsequent discussions pointed to three very clear conclusions relating to the UK situation:

1. The UK OGSA activity should focus on a small set of useful services: these should be spelt out in an OGSA-UK document;
2. There is clear disagreement between the “IBM" (WSRF) and “Microsoft" (WS-I+) approaches to how Web Services should handle state and other issues. Furthermore, this division is reflected in both the distributed computing community at large and in the portfolio of UK e-Science projects;
3. There was however clear agreement that the users should be insulated from these low-level technical implementation disagreements as far as possible. One possible route could be through APIs and increased emphasis on activities such as the SAGA (Simple API for Grid Applications) working group of GGF.

4. Discussion

Whilst the OGSA Testbed Workshop and this report focus on Grid services, it is worth pointing out that there is a large body of other work going on in the UK, which has identified a SOA as the best way to define a re-usable framework of components. This has also suggested Web Services as the preferred way to expose the framework’s functionality. Framework definitions and descriptions are now being formalised by JISC, the Joint Information Systems Committee in the form of e-FER, the e-Framework for Education and Research. This takes input from the three primary areas of JISC development: the Information Environment [14], Learning and Teaching [15] and e-Research [16].

In the Information Environment, service providers in multiple institutions wish to make their resources available on-line for both learning and research processes. Resources are of many types. They include databases of survey results or other research material such as experimental and simulation data. Other resources may contain textual and pictorial data related to social or historical events or arts including literature. Music and speech are also special cases. Tools for annotation, markup, publication, cataloguing, provenance tracing, format conversion and cross searching are all-important. In some cases additional protocols are required, such as in the delivery of streamed audio or visual material, so Web services alone may not be adequate.

The JIE (JISC Committee for the Information Environment) has very well established activities and has developed the architecture shown in Figure 4.

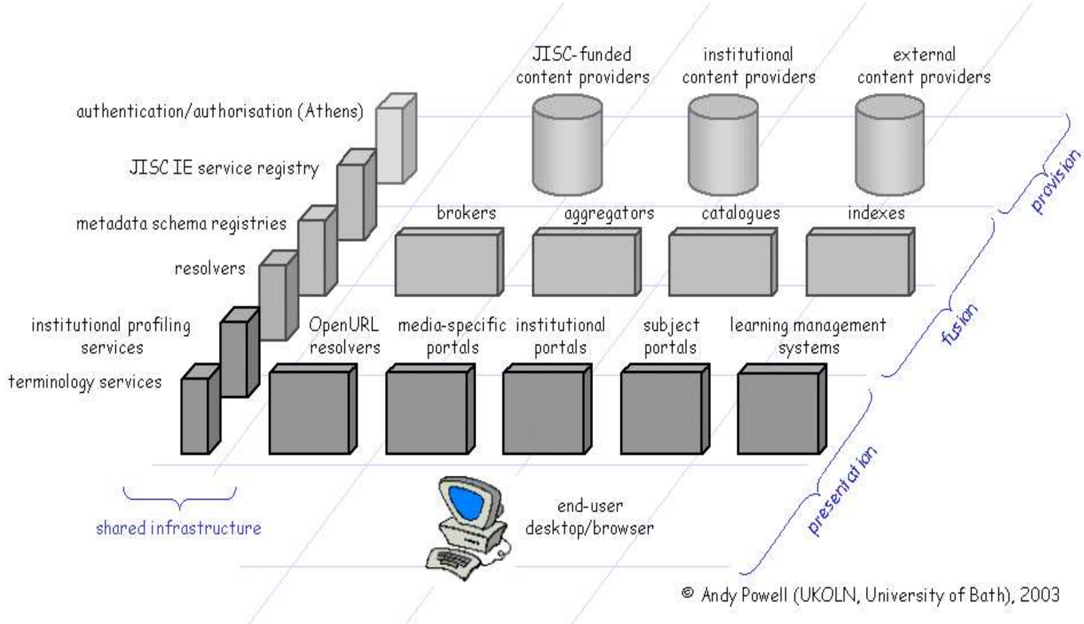

Figure 4: JISC Information Environment Architecture

E-Learning is characterised by providing electronic access to learning materials and well-defined activities typically within a single institution. This involves large numbers of learners either working as peers or in hierarchies with teachers and tutors. Assessment and grading is also included in a Managed Learning Environment as part of the administrative process. This is not required in a personal learning environment where it is assumed that the learners are self-motivated. Other institutional services might be included, e.g. administrative, or financial services.

The JLT (JISC Committee for Learning and Teaching) has had a strong portfolio of projects developing tools and standards for many years. JISC also has services such as CETIS (Centre for Educational Technology Interoperability Standards), which provides input on standards and strategy at an international level. The e-Learning Programme has four main areas of focus, reflected in four strands:

- e-Learning and Pedagogy;
- Frameworks and Tools (ELF);
- Distributed e-Learning;
- Innovations.

A recent high-profile discussion focussed on the definition of the E-Learning Framework, ELF, which is a joint initiative by between JISC, Australia's Department of Education, Science, and Training (DEST), and the Carnegie Mellon Learning Services Architecture Lab (LSAL), as well as others to build a common approach to Service Oriented Architectures for education. The ELF is the result of a shared conviction that exposing networked functions such as user and group data or learning content as simple services rather than as features locked up inside monolithic systems offers institutions more flexibility, more scope for pedagogic innovation and better return on present and future investment.

ELF is not an architecture in itself, rather, it captures a common way of factoring functionality across different components and provides their scope and definition. This is a vision related to that of a SOA. It is implied that Web services are to be used to connect the components and that agreed specifications will enable the inter-operating of open source and commercial tools. Another aim is to provide specifications enabling existing services to be exposed with useful functionality and granularity. ELF will form part of the SOFER.

E-Research is characterised by small communities of collaborating researchers in multiple institutions wanting to carry out complex processes to create new knowledge. This is also often linked to personal learning activities and electronic publication. e-Researchers are using the Grid and the JISC Virtual

Research Environments (VRE) Programme was recently launched to demonstrate the delivery of a wide variety of services to diverse user communities in various forms.

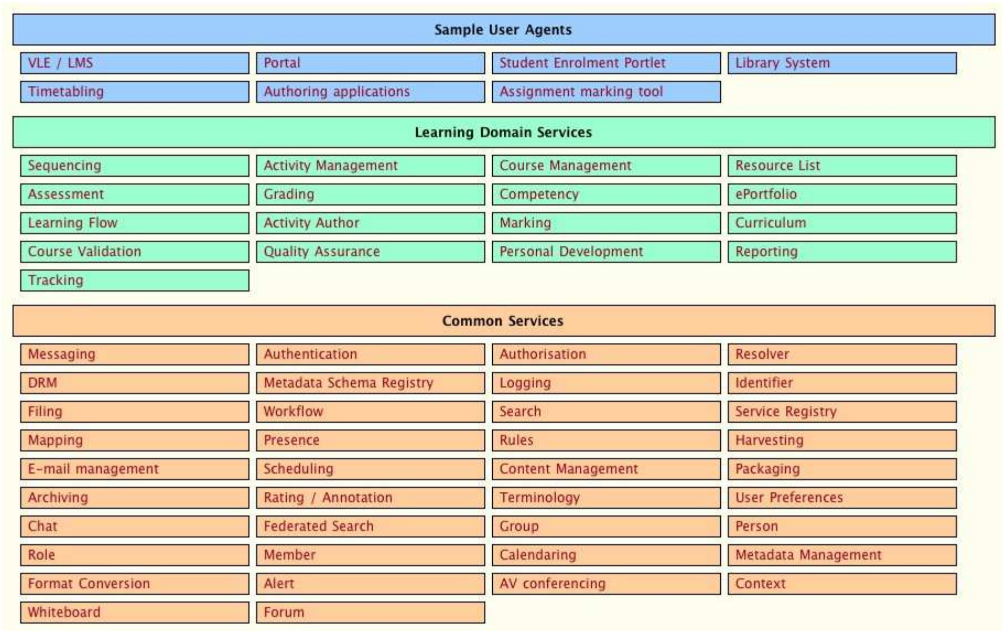

Figure 5: E-Learning Framework

There are a number of areas of overlap between the different pillars; Figure 6 shows examples including some e-Research subject domains. These are of course for illustration only and not specific.

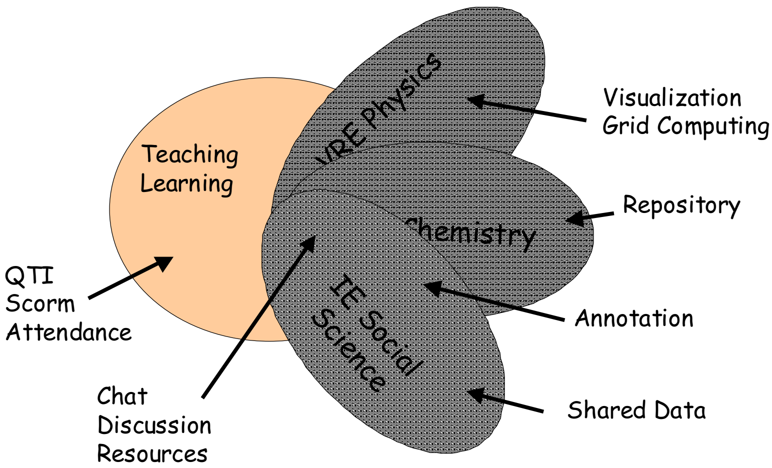

Figure 6: Overlapping Areas

Figure 7 shows the components, or services, identified as being required for a VRE. Similar service classification has been done for the other pillars and all will be included in the e-FER. Input from the UK Grid Engineering Task Force and others is summarised with more detailed information in a separate document [17].

Other activities of JISC (the Core Middleware Programme) are evaluating and developing security services including authentication using Shibboleth [18] and policy-based authorisation using PERMIS [19]. Of particular interest is the SPIE project [20] in which Shibboleth is being investigated in a Grid environment. Clearly with all these related activities the extensions of Web services to Grid services will have an important role.

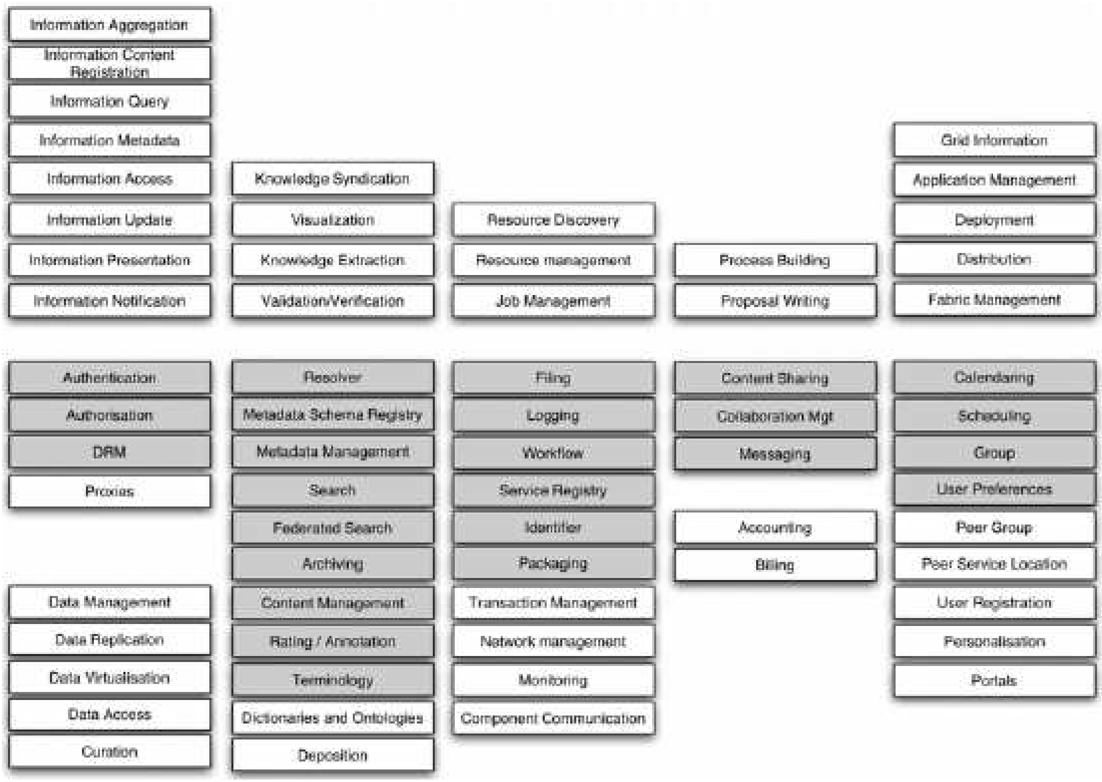

Figure 7: e-Science Framework (JISC JCSR)[1]

5. Conclusions.

The talks given at the OGSA Testbed Workshop and the associated discussion between the presenters and delegates highlighted many of the issues, pitfalls, problems, evolutionary changes and solutions being used, as well as the possible ways forward for providing production-level Grid services to application developers.

It is clear that so far there is little production quality Grid middleware available to support wide-area distributed applications. This situation seems to be changing if the experiences of those investigating GT4 are to be believed. However, a single Grid middleware solution is seen as less than ideal and it was asserted that there was a pressing need for lightweight alternative solutions to GT4, such as WSRF:Lite. An issue that was repeatedly brought up and debated during the workshop was that of emerging Grid standards [24] and what was likely to be the most appropriate ones to follow.

The Portsmouth-led OGSA Testbed project has been a success at many levels. First, and perhaps most importantly, from its vanguard position, it was able to feedback significant information to the e-Science community about the stability and reliability of GT3. This feedback delayed the take up of GT3 by the community, as it was determined as being of insufficient quality for widespread use, and in particular by those who were not experts in computer software. Secondly, the project successfully investigated the most appropriate means to implement and deploy applications onto a GT3-based platform; here one particular success has been the development of GEMCLA from Westminster. Related to this has been a strong emphasis on developing application specific portals to hide the underlying complexity from developers and users. In addition to these achievements, the Testbed has established a strong collaborative relationship between the partner sites, which has been rewarded for instance by further funding (by JISC under the VRE programme), and should help future bids for EU funding.

[1] Shading indicates Services that are present in both the e-Learning and e-Science Frameworks.

There are a number of areas where further research and evaluation work is required, and where the UK already has the right expertise to make rapid progress, Indeed work has already started in these areas and partners of the OGSA Testbed are involved in taking it forward. Briefly we note:

- Portals and Portlets – Daresbury, Portsmouth, and Westminster are developing Web-based user interfaces using the JSR-168 (Java) and WSRP (Web Services) standards to link into Grid services; The JISC-funded Sakai VRE project is an example as is the PGrade Portal.
- Workflow – Daresbury is involved in *WOSE: Workflow Optimisation Services for E-Science*, an EPSRC-funded project with Cardiff and Imperial College. Workflow technology can underpin much of Grid service deployment and usage, and issues of discovery, aggregation, and dynamic scheduling are investigated. The GEMLCA system also involves workflow.
- Semantic Grid – Daresbury is working with others to solve problems of describing scientific applications and their associated data using XML-based technology such as RDF and OWL. Semantically rich data models support the use of Grid services, which wrap large-scale applications, such as in GEMLCA. Registries such as UDDI also need associated meta-data objects to enable semantic service discovery.

It is our belief that in a service-oriented architecture, such as OGSA, the above areas hold the keys to successful deployment and wide uptake for production systems.

References

[1] UK OGSA Workshop, http://dsg.port.ac.uk/projects/ogsa-testbed/workshop/schedule.html

[2] OGSA Testbed project, http://dsg.port.ac.uk/projects/ogsa-testbed

[3] OGSA Evaluation project, http:// http://www.sse.ucl.ac.uk/UK-OGSA

[4] P. Brebner, J. Wu and O. Malham, *Evaluating OGSA Across Organisational Boundaries.* UK OGSA Evaluation Project Report 2.0 (Final) February 2005 http://sse.cs.ucl.ac.uk/UK-OGSA-Report2.doc

[5] ETF Evaluation of GT4, http://www.grids.ac.uk/ETF/

[6] e-Science Town Meeting, http://www.rcuk.ac.uk/escience/

[7] WEDS, http://www.nesc.ac.uk/technical_papers/UKeS-2004-07.pdf

[8] OGSA Testbed, http://dsg.port.ac.uk/projects/ogsa-testbed/

[9] GT3GITS, http://holly.dsg.port.ac.uk:8889/gridDaiMiner/start.jsp

[10] Portsmouth, http://dsg.port.ac.uk/projects/ogsa-testbed/workshop/Talks/Mark%20Baker.ppt

[11] UCL, http://dsg.port.ac.uk/projects/ogsa-testbed/workshop/Talks/Wolfgang%20Emmerich.ppt

[12] Newcastle, http://dsg.port.ac.uk/projects/ogsa-testbed/workshop/Panel/Savas%20Parastatidis.ppt

[13] REST, http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[14] JISC, http://www.jisc.ac.uk/index.cfm?name=ie_home

[15] e-Learning, http://www.elearning.ac.uk

[16] VRE, http://www.jisc.ac.uk/index.cfm?name=programme_vre

[17] R.J. Allan, A. Hardisty, S. Wilson and A. Powell *A Service Classification,* http://www.grids.ac.uk/Papers/Classes/classes.html and http://www.grids.ac.uk/Papers/Classes/classes.pdf

[18] Shibboleth, http://shibboleth.internet2.edu/

[19] PERMIS, http://www.permis.org/

[20] SPIE, http://www.jisc.ac.uk/index.cfm?name=project_spie

[21] M. McKeown *OGSI::Lite, WSRF::Lite* (University of Manchester), http://www.sve.man.ac.uk/Research/AtoZ/ILCT

[22] M. Atkinson, D. De Roure, A. Dunlop, G. Fox, P. Henderson, T. Hey, N. Paton, S. Newhouse, S. Parastatidis, A. Trefethen and P. Watson. *Web Service Grids: An Evolutionary Approach.* Technical Report UKeS-2004-02 (NeSC, 2004) http://www.nesc.ac.uk/technical_papers/uk.html

[23] Wim Kok, *Making a success of Enlargement*, http://europa.eu.int/comm/enlargement/communication/#kok_report

[24] Mark Baker, Amy Apon, Clayton Ferner, and Jeff Brown, *Emerging Grid Standards*, IEEE Computer, April 2005 (Vol. 38, No. 4), pp. 43-50, ISSN 0018-9162