# Grids and Web Services for e-Science

Tony Hey[1], Geoffrey Fox[2]

1. EPSRC, Polaris House, Swindon, UK
2. Computer Science Department, Indiana University, US

**Abstract**

This editorial describes four papers that summarize key Grid technology capabilities to support distributed e-Science applications. These papers discuss the Condor system supporting computing communities, the OGSA-DAI service interfaces for databases, the WS-I+ Grid Service profile and finally WS-GAF the Web Service Grid Application Framework. We discuss the confluence of mainstream IT Industry development and the very latest science and computer science research and urge the communities to reach consensus rapidly. Agreement on a set of core Web Service standards is essential to allow developers to build Grids and distributed business and science applications with some assurance that their investment will not be obviated by the changing Web Service frameworks.

**Keywords**

e-Science, Grids, Standards, Web Services, Grid Services, Scheduling, Databases

## 1. Web Service Grids Context

This special issue [1-4] covers some of the core capabilities of e-Science infrastructure. e-Science is about global collaboration in key areas of science, and the next generation of infrastructure that will enable it. Although e-Science is specifically a major UK Program, there are similar activities around the world including Cyberinfrastructure in the USA [7] and "e-Infrastructure" in Europe. e-Science reflects growing importance of international laboratories, satellites and sensors and their integrated analysis by distributed teams. e-Science is built using Grid technology which has been summarized recently in two books [5, 6]. This technology is still rapidly evolving and this issue covers some of the important issues – both established and those still emerging.

## 2. Special Issue Papers

The first paper describes the history and development of the well known Condor project [1] that enables one critical Grid capability; the scheduling and execution of jobs on a distributed network of diverse machines. Condor has proved critical in several application areas such as analysis of data from large physics accelerators (see chapter 39 of ref. [6]). However this article described how the system both created and supported computing communities and evolved as these changed with the Grid application, stressing the need to share resources across organizational boundaries. The present article describes the growing importance of data-intensive computing which leads into the second paper that describes the UK e-Science OGSA-DAI project. This, like the later papers in this issue, builds on the current consensus that Service Oriented Architectures are suitable for building a modern e-Science infrastructure. OGSA-DAI builds a Grid Service architecture for a wide range of database access functions. Issues discussed include federation, the critical role of metadata, security and the need to support multiple data delivery options. OGSA-DAI was one of the first Grid capabilities built in terms of Web services and it illustrates the need to agree on the Web Service specifications that one should use. There are over 60 proposed specifications presently in discussion at the different standards bodies and until the dust settles it is not clear which specifications will survive and in what precise form. The Web Service Grids paper [3] introduces a minimal set of specifications that appear necessary for Grid systems. The authors term this choice WS-I+ as it builds on Web Service profiles built by the Web Service Interoperability organization [8] that has broad industry support. The specification of appropriate interfaces for Grids to Web services is an area of active research with the Web Service Resource Framework [9] being an important proposal building on the Global Grid Forum OGSI activity [10]. The last paper in this issue [4] describes WS-GAF which looks at building Grids without new additional specifications. They examine different approaches to stateful interactions, logical resource naming, metadata and lifetime management.

Condor [1] is moving to a Web Service architecture and OGSA-DAI is already built on this framework. The last two papers directly address the question of which Web Services really are needed to support a wide spectrum of e-Science applications. Currently this process is in a state of flux, given the uncertainties as to which standards will get developed and adopted. Some confusion is probably inevitable since there are obvious technical issues that can only be resolved by further research and experience. However, some of the present roadblocks appear unnecessary and could be resolved if the communities involved were willing to negotiate and compromise. We discuss these important issues in the next section.

## 3. The Web Service Uncertainty

### 3.1 The Problem

The UK e-Science Initiative [11], which started in 2001, began by gaining some understanding of the problems of implementing distributed middleware services that crossed institutional boundaries by evaluating the then current NASA Information Power Grid software -- which primarily consisted of the Globus Toolkit [12], Condor [1] and the Storage Resource Broker [13] packages. But even in 2001, it was clear that any future distributed middleware that wished to have support and tooling from the IT industry would have to be based on Web services. However, it is an unfortunate fact that, although the Web services movement is supported by all the IT companies, even at the end of 2004, this "grand project" is still very much "work in progress." The presently accepted Web services certainly do not constitute a satisfactory basis to construct a robust, international Cyberinfrastructure on which to build novel and demanding e-Science and business applications [3, 4].

This is a problem. Funding agencies in the United States, Europe and Asia are supporting many hundreds of e-Science or "Grid" projects, all of which involve one or more forms of distributed data, computation and collaboration. In the United States alone, even in the absence of a long-delayed Cyberinfrastructure initiative along the lines recommended by the Atkins Report [7], the NSF is funding over $400 million worth of "e-Science" projects per year. In the United Kingdom, with the present dollar exchange rate, the e-Science program amounts to some $500 million over five years. Germany and the Netherlands have just announced 90-million- and 50-million-Euro e-Science programs, respectively, and the European Commission have launched over 400 million Euros worth of new Grid projects. China and Japan also have ambitious and significant e-Science programs.

To underpin all of this activity we need a set of standard middleware services that enable the coordinated, collaborative use of distributed resources (computation, data sets, facilities). This set of middleware services -- determined by the application requirements -- is what we call the "Grid," as a shorthand for distributed middleware infrastructure or Cyberinfrastructure, according to our definition. There is clearly a whole community of scientists and engineers -- in both academia and industry -- all gearing up to make scientific and commercial e-Science and Cyberinfrastructure applications a reality. So what is the problem? The problem lies with the slow pace of the standards process and the ongoing Web services standards "wars."

### 3.2 Introduction to Web services

Web Services are the distributed computing technology that the IT industry is trying to define to be the building blocks for building loosely-coupled, distributed applications, based on Service Oriented Architecture principles [3]. Web services interact by exchanging messages in SOAP format while the contracts for the message exchanges that implement those interactions are described via WSDL and other metadata formats. When a SOAP message arrives at a Web service, it is first handled by the service's message processing logic which transforms network level SOAP messages into something more tangible for applications to deal with (such as domain-specific objects). Once the message has been consumed, its contents are then processed by the application logic, making use of the resources available to the service. Typically, some response is then generated which is fed back via one or more messages.

By encapsulating the internal resources within the service, and providing a layer of application logic between those resources and the consumers, the owners of the service are free to evolve its internal structure over time (for example to improve its performance or dependability), without making changes to the message exchange patterns that are used by existing service consumers. This encourages loose-coupling between consumers and service providers, which is important in inter-enterprise computing, as no one party is in complete control of all parts of the distributed application. However, loose-coupling does not mean that the functionality of applications is compromised, since the set of existing and emerging Web services specifications should allow distributed application builders to model complex interactions between services.

Web services specifications can be divided into two classes. Infrastructure specifications define generic aspects of services (or other specifications), e.g. WSDL, WS-Security and the proposed "Grid" service WSRF [9]. High-level specifications define domain specific aspects of services, e.g. a data access and integration service specification. Policy also plays a key role in a service oriented architecture. While WSDL describes the functional characteristics of a Web service -- such as operations supported, messages sent and consumed -- the non-functional requirements associated with service invocation are also a very important aspect of Web services and service oriented architectures in general. WS-Policy and WS-PolicyAttachment [14] describe a foundation policy framework within which the behaviors associated with a service -- such as security, transactionality, reliability and so on -- can be specified. Conceptually, WSDL and WS-Policy are peers in the Web services stack.

### 3.3 Web Services and Grids

By leveraging the developments in Web services technologies, Grid architects will be able to exploit the tools, documentation, educational materials, and experience from the Web services community when building applications, without having to create a parallel set of solutions. This will allow the Grid community to concentrate on building the higher-level services that are specific to the Grid application domain while the responsibility for the underlying infrastructure is left to IT industry. The software vendors will work on standardizing the Web services technologies, developing production-quality tooling, achieving wide adoption, testing for the interoperability of the implementations of those standards, educating developers, etc.

This all sounds very desirable and obvious. So, again, where is the problem? At this point in time there are a large number of industry-led standardization efforts, only some of which are being developed within an open standards organization. This makes it difficult for a user to identify those that have completed the standardization process, those that are proprietary standards or indeed those that may have little future in terms of broad acceptance. The sheer number of specifications and the mixed signals coming from industry due to competing specifications in similar areas can leave application architects with the impression that there is no single clear vision for Web services. Even where there is a clear need for a standard (e.g., workflow, security, transactions, notification), it is still taking a long time for a widely accepted one to emerge. Different sets of vendors are producing competing specifications and it will therefore take time to resolve the differences in a manner that is both technically and commercially acceptable.

The uncertainty that this range of specifications creates becomes a real problem for developers who must choose which specifications to use in their implementations. If a specification is chosen too early in its lifecycle, then developers may suffer from lack of tool support as well as instability due to changes incurred as the specification evolves through a standardization process. In the worst case, a specification may never be widely adopted, and so will over time wither and die, adversely impacting any services that chose to adopt it.

### 3.4 A Challenge to the Industry Leaders

What can be done about this and who should take the lead? The "Men in Black" theory of standards suggests that a Web Service specification that is supported by both Microsoft and IBM is most likely to achieve widespread acceptance. Indeed, it was not so long ago that Bill Gates from Microsoft and Steve Mills from IBM shared a stage and gave guarantees that their implementations of Web services would interoperate with each other. The problem is that this agreement apparently does not extend to Web services for Grids. Examples of competing or overlapping specifications relevant to Grids are WS-Eventing, WS-Transfer, WS-Enumeration and WS-Management, supported by Microsoft and others; and WS-Notification, WS-ResourceFramework and WS-DistributedManagement, supported by IBM and their friends. (see Ref. [3] for a detailed discussion of these specifications.) The resolution is not so obvious -- since both companies approach Web services from very different commercial perspectives. Microsoft is concerned with keeping Web Services as simple as possible and easy to implement efficiently on Intel architectures. By contrast, IBM is concerned with defining more sophisticated Web services that can be used to create robust applications for commercial data centers.

However, by not reaching some compromise, both Microsoft and IBM risk confusing and antagonizing their major commercial customers. There are dozens of major companies involved in the latest set of European Grid projects -- Atos Origin, DataMat, Telenor, EADS, ESI, MSC, BAESystems, Boeing, SAP, T-Systems, Daimler-Chrysler, Audi, GlaxoSmithKlein and BT, among others, as well as Microsoft and IBM. All of these are multi-national companies with multi-vendor IT systems and it makes no sense for Microsoft and IBM to continue to talk past each other about Grids. Agreement on these low level standards is a matter of urgency for the world-wide e-Science and Grid research and business community and needs some resolution as soon as possible. Only Microsoft and IBM can provide the necessary leadership and this is what they need to show now. Only when these Web services standards have been stabilized can the Global Grid Forum [15] concentrate on defining and standardizing, where appropriate, the higher level services that will constitute the Open Grid Services Architecture.

## 4 Conclusions

e-Science is an important paradigm for science and engineering research and Grids are an important approach. There are still many unsolved research issues and there is no consensus as to “the best stable infrastructure”. However this is to be expected at this stage of the ‘Grid project’. There are already some clear successes emerging and these will motivate exciting research and more progress. These papers are a snapshot of “significant work in progress” and how we see an usual confluence of mainstream industry and leading edge research.

## References


1) Douglas Thain, Todd Tannenbaum, and Miron Livny, Distributed Computing in Practice: The Condor Experience, Concurrency And Computation: Practice & Experience, this issue.
2) Mario Antonioletti, Malcolm Atkinson, Rob Baxter, Andrew Borley, Neil P Chue Hong, Brian Collins, Neil Hardman, Alastair C. Hume, Alan Knox, Mike Jackson, Amy Krause, Simon Laws, James Magowan, Norman W Paton, Dave Pearson, Tom Sugden, Paul Watson and Martin Westhead, *The Design and Implementation of Grid*

*Database Services in OGSA-DAI*, Concurrency And Computation: Practice & Experience, this issue.

3) Malcolm Atkinson, David DeRoure, Alistair Dunlop, Geoffrey Fox, Peter Henderson, Tony Hey, Norman Paton, Steven Newhouse, Savas Parastatidis, Anne Trefethen and Paul Watson, *Web Service Grids: An Evolutionary Approach*, Concurrency And Computation: Practice & Experience, this issue.
4) Savas Parastatidis, Jim Webber, Paul Watson, Thomas Rischbeck, *WS-GAF: A Framework for Building Grid Applications using Web Services*, Concurrency And Computation: Practice & Experience, this issue.
5) *The Grid 2: Blueprint for a new Computing Infrastructure*, edited by Ian Foster and Carl Kesselman, Morgan Kaufmann 2004.
6) *Grid Computing: Making the Global Infrastructure a Reality* edited by Fran Berman, Geoffrey Fox and Tony Hey, John Wiley & Sons, Chichester, England, ISBN 0-470-85319-0, March 2003. http://www.grid2002.org.
7) Report of the National Science Foundation Blue-Ribbon Advisory Panel, *Revolutionizing Science and Engineering Through Cyberinfrastructure*, http://www.communitytechnology.org/nsf_ci_report/report.pdf
8) "Web Services Interoperability (WS-I)." http://www.ws-i.org.
9) WSRF Web Service Resource Framework. http://www.oasis-open.org/committees/tc_home.php?wgabbrev=wsrf
10) [OGSI] OGSI Open Grid Service Infrastructure Working Group of Global Grid Forum https://forge.gridforum.org/projects/ogsi-wg
11) UK e-Science Program http://www.escience-grid.org.uk/
12) Globus Project http://www.globus.org
13) Storage Resource Broker (SRB) http://www.npaci.edu/DICE/SRB/index.html
14) Web Services Policy Framework, WS-Policy. http://www-106.ibm.com/developerworks/library/ws-polfram/