RE-EVALUATING THE GRID: THE SOCIAL LIFE OF PROGRAMS

David De Roure
University of Southampton,
Electronics and Computer Science
Southampton, SO17 1BJ, UK
dder@ecs.soton.ac.uk

Carole Goble
The University of Manchester,
Manchester, UK
carole.goble@manchester.ac.uk

Abstract This paper discusses programming the Grid in the space between the Grid infrastructure and those using it to conduct scientific research. Rather than looking at any particular grid programming model, we consider the need to address 'usability' of programming solutions in this space. As a case study we consider a popular solution; i.e. scientific workflows, and we reflect on Web 2.0 approaches. We suggest that broad adoption of Grid infrastructure is dependent on ease of programming in this space.

Keywords: Grid, Scientific Workflow, Web 2.0.

1. Introduction

Grid computing is about bringing resources together in order to achieve something that was not possible before. In its early phase there was an emphasis on combining resources in pursuit of computational power and very large scale data processing, such as high speed wide area networking of supercomputers and clusters. This new power enabled researchers to address exciting problems that would previously have taken lifetimes, and it encouraged collaborative scientific endeavours. As it has evolved, Grid computing continues to be about providing an infrastructure which brings resources together, with an emphasis now on the notion of Virtual Organisations.

This emerging infrastructure is increasingly being considered for 'everyday science', enabling researchers in every discipline to make use of the new capabilities. However there is significant challenge in bringing the new infrastructure capabilities to broad communities of users, a problem which was perhaps masked previously by the focus on a more 'heroic' style of Grid project. Significantly, this is a programming challenge – how do we make it easy for people to assemble the services and resources they want in order to achieve the task at hand?

In this paper we look at programming in the space between the core infrastructure services and the users. This area has been the focus of attention for the Semantic Grid community for several years, initially using Semantic Web and more recently developing Web 2.0 techniques. In the next section we recap the Semantic Grid vision, then in Section 3 we take a look at scientific workflows as a case study in programming in this space, with a particular look at a system which emerged from one of the Semantic Grid projects. Section 4 reflects on everyday e-Science in the context of the principles of Web 2.0. We close by observing that success in programming the grid is not just about programming abstractions but also about ease of use and what we describe as the 'social life of programs'.

2. The Semantic Grid

The notion of the 'Semantic Grid' was introduced in 2001 by researchers working at the intersection of the Semantic Web, Grid and software agent communities [4]. Observing the gap between aspiration and practice in grid computing, the report 'The Semantic Grid: A Future e-Science Infrastructure' stated:

> e-Science offers a promising vision of how computer and communication technology can support and enhance the scientific process. It does this by enabling scientists to generate, analyse, share and discuss their insights, experiments and results in a more effective manner. The underlying computer infrastructure that provides these facilities is commonly referred to as the Grid. At this time, there are a number of Grid applications being developed and there is a whole raft

> of computer technologies that provide fragments of the necessary functionality. However there is currently a major gap between these endeavours and the vision of e-Science in which there is a high degree of easy-to-use and seamless automation and in which there are flexible collaborations and computations on a global scale.

We recognised that this emerging vision of the Grid was closely related to that of the Semantic Web – which is also, fundamentally, about joining things up. The Semantic Web is an initiative of the Worldwide Web Consortium (W3C) and at that time was defined by the W3C Activity Statement as "...an extension of the current Web in which information and services are given well-defined meaning, better enabling computers and people to work in cooperation".

To researchers aware of both worlds, the value of applying Semantic Web technologies to the information and knowledge in Grid applications was immediately apparent. At that time the service-oriented architecture of the Grid was also foreseen, and the need for machine-understandable metadata in order to facilitate automation was clear. Thus the vision of the Semantic Grid became established as the application of Semantic Web technologies both *on* and *in* the Grid [9]. Additionally, agent-based computing was proposed to achieve the necessary degree of flexibility and automation within the machinery of the Grid [8].

The dual aspects of *information* and *services* have been explored in various projects. Within the UK e-Science program, the Combechem project in particular focused on the 'Semantic Datagrid' [15], while ^{my}Grid focused on services [1]. Semantic Grid has been adopted in a range of grid projects across Europe and in 2006 the Next Generation Grids Experts Group articulated a vision for the future service-oriented Grid called the *Service Oriented Knowledge Utility*, which captured the Semantic Grid vision and identifying an agenda for future research [12].

3. Scientific Workflows

The ^{my}Grid project produced a scientific workflow system, Taverna [13], which enables scientists to assemble services in order to conduct their research – it is a programming solution in the space between the infrastructure services and the research applications. This is our case study in programming the grid. We can think of workflows as scripts, and many of the lessons from workflows extrapolate to scripts in general.

Scientific workflows are attracting considerable attention in the research community. Increasingly they support scientists in advancing research through *in silico* experimentation, while the workflow systems themselves are the subject of ongoing research and development. The National Science Foundation Workshop on the Challenges of Scientific Workflows identified the potential for scientific advance as workflow systems address more sophisticated require-

ments and as workflows are created through collaborative design processes involving many scientists across disciplines [5]. Rather than looking at the application or machinery of workflow systems, it is the dimension of collaboration and sharing that is of particular interest to us here.

Understanding the whole lifecycle of the workflow design, prototyping, production, management, publication and discovery is fundamental to developing systems that support the scientists' work. Reuse is a particular challenge when scientists are outside a predefined Virtual Organisation or enterprise – where there are individuals or small groups, decoupled from each other and acting independently, who are seeking workflows that cover processes outside their expertise from a common pool of components. This latter point arises when workflows are shared across discipline boundaries and when inexperienced scientists need to leverage the expertise of others.

There are many workflow systems available — we found over 75 after conducting an informal search. These systems vary in many respects: e.g. who uses them, what resources they operate over, whether the systems are open or closed, how workflows are expressed (e.g. how control flow is handled), how interactive they are, when and how tasks are allocated to resources, and how exceptions are handled. Our focus here is on scientific workflows which are near the application level rather than those further down in the infrastructure; i.e. we are interested in composing scientific applications and components using workflows, over a service oriented infrastructure (which may include Grid services). These are the workflows which are close to the scientist, or indeed the researcher whatever their domain.

3.1 The workflow as a first class citizen

One immediate attraction of workflows which encourages their uptake is the easing of the burden of repetitive manual work. However, we suggest that the key feature for scientific advancement is reuse. Workflow descriptions are not simply digital data objects like many other assets of e-Science, but rather they actually capture pieces of scientific process – they are valuable knowledge assets in their own right, capturing valuable know-how that is otherwise often tacit. Reuse is effective at multiple levels: the scientist reuses a workflow with different parameters and data, and may modify the workflow, as part of the routine of their daily scientific work; workflows can be shared with other scientists conducting similar work, so they provide a means of codifying, sharing and thus spreading the workflow designer's practice; and workflows, workflow fragments and workflow patterns can be reused to support science outside their initial application.

The latter point illustrates the tremendous potential for new scientific advance. An example of this is a workflow used to help identify genes involved

in tolerance to Trypanosomiasis in east African cattle [7]. The same workflow was reused over a new dataset to identify the biological pathways implicated in the ability for mice to expel the Trichuris muris parasite (a parasite model of the human parasite Trichuris trichuria). This reuse was made easier by the explicit, high-level nature of the workflow that describes the analytical protocol.

Workflows bring challenges too. Realistic workflows require skill to produce so they can be difficult and expensive to develop. Consequently, workflow developers need development assistance, and prefer not to start from scratch. Furthermore it is easy for the reuse of a workflow to be confined to the project in which it was conceived. In the Trypanosomiasis example, the barrier to this reuse was how the knowledge about the workflow could be spread to the scientists with the potential need. In this case it was word of mouth within one institution; this barrier needs to be overcome. So, we have a situation of workflows as reusable knowledge commodities, but with potential barriers to the exchange and propagation of those scientific ideas that are captured as workflows.

Significantly, there is more to a workflow than a declaration of a process. An individual workflow description may take the form of an XML file, but these do not sit in isolation. We can identify a range of properties that are factors in guiding workflow reuse, including: descriptions of its function and purpose; documentation about the services with which it has been used, with example input and output data, and design explanations; provenance, including its version history and origins; reputation and use within the community; ownership and permissions constraints; quality, whether it is reviewed and still works; and dependencies on other workflows, components and data types. Workflows also enable us to record the provenance of the data resulting from their enactment, and logs of service invocations from workflow runs can inform later decisions about service use.

By binding workflows with this kind of information, we provide a basis for workflows to be trusted, interpreted unambiguously and reused accurately. But like the workflows themselves, the associated information is currently often confined to the system from which it originated and thus is not reusable as a useful commodity in its own right.

3.2 Workflow Systems and Communities

Scientific workflow systems with significant deployment include the Taverna workflow workbench [13], Kepler [10], Triana [2] and Pegasus [6]. Taverna, which comes from the ^{my}Grid project, is used extensively across a range of Life Science problems: gene and protein annotation; proteomics, phylogeny and phenotypical studies; microarray data analysis and medical image analysis; high throughput screening of chemical compounds and clinical statistical analysis.

Significantly, Taverna has been designed to operate in the open wild world of bioinformatics. Rather than large scale, closed collaborations which own resources, Taverna is used to enable individual scientists to access the many open resources available in the cloud, i.e. out on the Web and not necessarily within their enterprise. Many of the services are expected to be owned by parties other than those using them in a workflow. In practice they are volatile, weakly described and there is no contract in place to ensure quality of service; they have not been designed to work together, and they adhere to no common type system. Consequently, they are highly heterogeneous. By compensating for these demands, Taverna has made, at the time of writing, over 3500 bioinformatics orientated operations available to its users. This has been a major incentive to adoption. This openness also means that Taverna is not tied exclusively to the bioinformatics domain – any services can be incorporated into its workflows.

By way of comparison, the lifecycle of workflows in the Pegasus system has also been the subject of study [6]. Pegasus has more of a computational and Grid emphasis. It maps from workflow instances to executable workflows, automatically identifying physical locations for workflow components and data and finding appropriate resources to execute the components; it reuses existing data products where applicable. Pegasus is used within large scale collaborations and big projects and is perhaps more typical of traditional e-Science and grid activities, while Taverna gives an interesting insight into another part of the scientific workflow ecosystem – it is being used by many scientists on their personal projects, constituting a distributed, disconnected community of users who are also the developers of the workflows. Taverna is very much about services – and scientists – 'in the cloud'.

3.3 Sharing workflows

It is apparent then that we can view workflows as potential commodities, as valuable first class assets in their own right, to be pooled and shared, traded and reused, within communities and across communities, to propagate like memes. Workflows themselves can be the subject of peer review. Furthermore we can conceive of packs of workflows for certain topics, and of workflow pattern books – new structures above the level of the individual workflow. We call this perspective of the interacting data, services, workflow and their metadata within a scientific environment the *work¯ow ecosystem* and we suggest that by understanding and enabling this we can unlock the broader scientific potential of workflow systems.

Workflow management systems already provide basic sharing mechanisms, through repository stores for workflows developed as part of projects or communities. For example, the Kepler Actor Repository is an LDAP-based directory for the remote storage, query and retrieval of actors (processes) and other work-

flow components and the SCEC/CME workflow system has component and workflow libraries annotated with ontologies [11]. These follow the tradition of cataloguing scripting libraries and codes.

In the ^{my}Experiment project we are taking a more social approach: we believe that the key to sharing is to recognise the use of workflows by a community of scientists [3]. This acknowledges a central fact, sometimes neglected, that the lifecycle of the workflows is coupled with the process of science that the human system of workflow use is coupled to the digital system of workflows. The more workflows, the more users and the more invocations then the more evidence there is to assist in selecting a workflow. The rise of harnessing the Collective Intelligence of the Web has dramatically reminded us that it is people who generate and share knowledge and resources, and people who create network effects in communities. Blogs and wikis, shared tagging services, instant messaging, social networks and semantic descriptions of data relationships are flourishing. Within the Scientific community we have examples: OpenWetWare, Connotea, PLoS on Facebook, etc. (see corresponding .org Web Sites and facebook.com).

By mining the sharing behaviour between users within such a community we can provide recommendations for use. By using the structure and interactions between users and workflow tools we can identify what is considered to be of greater value to users. Provenance information helps track down workflows through their use in content syndication and aggregation.

4. Web 2.0

While part of e-Science has focused on infrastructure provision, everyday scientific practice has continued to evolve, especially in use of the Web. Like workflows, the mashups which characterise Web 2.0 also enable scientists to bring together resources in new ways – they provide a means of coupling robust underlying services. Significantly, creating mashups is not such a specialist activity as working with Grid or Semantic Web, and this is illustrated by the many examples of mashups being used by researchers and by ICT experts within their research domains: the Web is increasingly seen as a distributed application platform in its own right. The simple interfaces based on REST, the content behind them such as the Google Maps API, and the sharing culture that characterises their development and evolution, is leading to uptake which is having immediate impact on everyday scientific practice in many domains – and can be contrasted with the uptake of Grid.

We suggest that these two examples of programming above the service level – the scientific workflows of Taverna and mashups for everyday science – exemplify the way forward for e-Science and for Grid computing. We believe that the reason they work is that they thrive in the ecosystem between core

infrastructure services and the user: an ecosystem of scientists, domain ICT experts, companies, tools, workflow systems, and indeed computer scientists.

We can demonstrate the relationship between e-Science and Web 2.0 in this space by considering e-Science in the context of the eight design patterns of Web 2.0 [14]:

The Long Tail While e-Science has often focused on specialist early-adopter scientists and large scale collaborative projects, Taverna and mashups are used by the 'long tail' of researchers doing everyday science – by which we refer to the larger number of smaller-scale specialists who are now enabled by digital science. Rather than heroic science with heroic infrastructure, new communities are coming online and bring with them the power of community intelligence. They are often using services 'in the cloud' rather than in the enterprise.

Data is the Next Intel Inside e-Science has been motivated by the need to handle the data deluge brought about by new experimental methods, and this data is large, rich, complex and increasingly real-time. Significantly there is extra value in data through new digital artefacts (such as scientific workflows) and through metadata; e.g. capturing context for interpreting data, storing provenance in order to interpret and trust data.

Users Add Value This is already a principle of the scholarly knowledge life-cycle, now revisited in the digital age. e-Science increasingly focuses on publishing as well as consuming.

Network Effects by Default Brought about by working in more and more with shared digital artefacts, the actual usage of information brings new value – through explicit reviewing but also implicitly through the recommendations and advice that can be provided automatically based on usage patterns. For example, the choice of services to run a workflow can be based on the history of service usage and performance as well as sharing of community knowledge.

Some Rights Reserved Increasingly we see mechanisms for sharing scholarly outputs – data, workflow, mashups – which by default are open. This is exemplified by preprints servers and institutional repositories, open journals, movements such as Science Commons and technologies such as the Open Archives Initiative. Open source development, and the sharing of scripts used in mashups, exemplify the openness which accelerates the creation of programming solutions.

The Perpetual Beta The technologies that scientists are choosing to use are not perfect, but they are better than what went before. The solutions being adopted in the space we are discussing are often the result of extreme

programming rather than extensive software engineering, providing the essential agility in response to user needs.

Cooperate, Don't Control The success stories come from the researchers who have learned to use ICT – we are seeing an empowering of domain experts to deliver the solutions. Indeed, solutions which take away this autonomy may be resisted. This is achieved by making it as easy as possible to reuse services and code.

Software Above the Level of a Single Device e-Science is about the intersection of the digital and physical worlds. Sensor networks are responsible for the data deluge, but equally mobile handheld devices are increasingly the interface as opposed to portals in Web browsers on PCs.

5. Discussion

The Semantic Grid activities have demonstrated the value of Semantic Web technologies to meet some of the needs of e-Scientists, especially for information reuse and where automation is required. They have also demonstrated the need for ease of programming in the space above the robust services to enable agile provision of better solutions for the users.

Sometimes Web 2.0 is seen as a competitor to Grid, and criticised by the grid community for lack of robust engineering and the rigour needed to underpin scientific research. We have presented a different view: that a Web 2.0 approach is absolutely appropriate for use in the space between the robust grid infrastructure and the user. We note that the SOKU vision of robust services ('utilities') which are dependable and easy to use is entirely consistent with this.

The key point for those involved in programming the Grid is that ease of use – usability of programs – is just as important as well-designed programming models. It is necessary to think outside individual programs and think about their lifecycle, the interactions of users, developers and scientists with the programs – what we could call the 'social life of programs'. The ^{my}Experiment project adopts this approach for workflows.

One of the propositions of Grid computing has been a universal Grid achieved by a certain style of coupling of resources. The picture we have drawn is a little different: some robust services 'in the cloud', perhaps based on grid technologies, which are plumbed together towards the application level. We suggest that this latter view is more achievable and is actually what many users require. Aside from the distributed application platform, these technologies are clearly complementary within the research lifecycle; e.g. grid for capturing or generating data and Web 2.0 for working with it effectively.

e-Science is now enabling researchers to do some completely new research. As the individual pieces become easy to use, researchers can bring them together

in new ways and ask new questions. Hence usability of the programming tools – workflows, mashups, whatever new techniques may emerge – is what will enable new science. This should be on the agenda for the grid programming community.

Acknowledgments

Thanks to the ^{my}Grid, Comb*e*Chem and ^{my}Experiment teams and the Taverna user community, and also to our Semantic Grid colleagues, especially Geoffrey Fox and Marlon Pierce.

References

[1] R. D. Stevens C. A. Goble, S. R. Pettifer and C. Greenhalgh. *Knowledge Integration: In silico Experiments in Bioinformatics*, pages 121–134. Morgan Kaufmann, May 2004.

[2] David Churches, Gabor Gombas, Andrew Harrison, Jason Maassen, Craig Robinson, Matthew Shields, Ian Taylor, and Ian Wang. Programming scientific and distributed workflow with triana services: Research articles. *Concurr. Comput. : Pract. Exper.*, 18(10):1021–1037, 2006.

[3] D. De Roure and C.A. Goble. myExperiment - a web 2.0 virtual research environment. In *International Workshop on Virtual Research Environments and Collaborative Work Environments*, May 2007.

[4] D. De Roure, N. R. Jennings, and N. R. Shadbolt. Research Agenda for the Semantic Grid: A future e-science infrastructure. Technical Report UK UKeS-2002-02, National e-Science Centre, Edinburgh, December 2001.

[5] E. Deelman and Y. Gil, editors. *NSF Workshop on the Challenges of Scienti® Work¯ows*. NSF, May 2006.

[6] Ewa Deelman, Gurmeet Singh, Mei-Hui Su, James Blythe, Yolanda Gil, Carl Kesselman, Gaurang Mehta, Karan Vahi, G. Bruce Berriman, John Good, Anastasia Laity, Joseph C. Jacob, and Daniel S. Katz. Pegasus: a framework for mapping complex scientific workflows onto distributed systems. *Scienti® Programming Journal*, 13(3):219–237, 2005.

[7] Paul Fisher, Cornelia Hedeler, Katherine Wolstencroft, Helen Hulme, Harry Noyes, Stephen Kemp, Robert Stevens, and Andrew Brass. A systematic strategy for the discovery of candidate genes responsible for phenotypic variation. In *Third International Society for Computational Biology (ISCB) Student Council Symposium at the Fifteenth Annual International Conference on Intelligent Systems for Molecular Biology (ISMB)*, July 2007.

[8] Ian Foster, Nicholas R. Jennings, and Carl Kesselman. Brain meets brawn: Why grid and agents need each other. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 8–15, Washington, DC, USA, 2004. IEEE Computer Society.

[9] C. A. Goble, D. De Roure, N. R. Shadbolt, and A. A. A. Fernandes. *Enhancing Services and Applications with Knowledge and Semantics*, pages 431–458. Morgan-Kaufmann, 2004.

[10] Bertram Ludäscher, Ilkay Altintas, Chad Berkley, Dan Higgins, Efrat Jaeger, Matthew Jones, Edward A. Lee, Jing Tao, and Yang Zhao. Scientific workflow management and

the Kepler system: Research articles. *Concurr. Comput. : Pract. Exper.*, 18(10):1039–1065, 2006.

[11] Philip Maechling, Hans Chalupsky, Maureen Dougherty, Ewa Deelman, Yolanda Gil, Sridhar Gullapalli, Vipin Gupta, Carl Kesselman, Jihic Kim, Gaurang Mehta, Brian Mendenhall, Thomas Russ, Gurmeet Singh, Marc Spraragen, Garrick Staples, and Karan Vahi. Simplifying construction of complex workflows for non-expert users of the southern california earthquake center community modeling environment. *SIGMOD Rec.*, 34(3):24–30, 2005.

[12] Next Generation Grids Experts Group. Future for european grids: Grids and Service Oriented Knowledge Utilities. Technical report, EU Grid Technologies, January 2006.

[13] Tom Oinn, Mark Greenwood, Matthew Addis, M. Nedim Alpdemir, Justin Ferris, Kevin Glover, Carole Goble, Antoon Goderis, Duncan Hull, Darren Marvin, Peter Li, Phillip Lord, Matthew R. Pocock, Martin Senger, Robert Stevens, Anil Wipat, and Chris Wroe. Taverna: lessons in creating a workflow environment for the life sciences: Research articles. *Concurr. Comput. : Pract. Exper.*, 18(10):1067–1100, 2006.

[14] T. O'Reilly. What is Web 2.0 - design patterns and business models for the next generation of software, 2005. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

[15] K. Taylor, R. Gledhill, J. W. Essex, J. G. Frey, S. W. Harris, and D. De Roure. A Semantic Datagrid for Combinatorial Chemistry. In *GRID '05: Proceedings of the 6th IEEE/ACM International Workshop on Grid Computing*, pages 148–155, Washington, DC, USA, 2005. IEEE Computer Society.