eResearch – Paradigm Shift or Propaganda?

Bill Appelbe and David Bannon

Victorian Partnership for Advanced Computing (VPAC)
110 Victoria Street, PO Box 201, Carlton South, Victoria 3053, Australia
{bill, d.bannon}@vpac.org

eResearch is a concept or word which has come into vogue in academic research circles since 2000. Fundamentally, it is collaborative and interactive research made possible by the Internet and data and computational grids. But like any other new concept, it can and has been abused and misused by people keen to "jump on the bandwagon". eResearch presents many challenges, both technical and organizational – traditional academia rewards individual performance and specialization, not collective and interdisciplinary efforts that characterized eResearch. But eResearch is a paradigm shift that is changing the way that research is conducted and organized in many academic disciplines and research institutions.

Keywords: eReseach, Cyberinfrastructure, Grid Computing

ACM Classification: K.6.0 (Management of Computing and Information Systems), C.2.4 (Distributed Systems)

1. WHAT IS eRESEARCH?

eScience, or the more generic eResearch, has come into vogue recently, following on the heels of the more well-established term eCommerce. Like eCommerce, which can include anything from supply-chain integration to CRM (Customer Relationship Management), the definition of eResearch is very much dependent upon an individual or organization's perspective, and to confuse matters further it is called Cyberinfrastructure (Cyberinfrastructure, 2006) in the USA. So any group of researchers will have differing, and often vocal, opinions on what eResearch is. For example, for users of large data sets, such as climate modeling, it is all about having large data sets readily accessible, without them having to worry or waste time about sharing, data formats, backup, or security. To a big compute user, such as modeling cell membranes, its having massive compute capacity available on demand, without having to know anything about underlying details of the computers, operating systems, or file systems. Yet another group, such as the International Virtual Observatory in Astronomy, will tell you eResearch is a matter of breaking down the barriers between researchers, be they geographical, cultural or technical.

So there is no quantitative definition possible, or even desirable of what is, and what is not eResearch. Instead, there are useful characteristics of eResearch projects that can distinguish the degree to which a particular project might be promoted as eResearch, or "traditional" research as shown in Table 1.

A further point of confusion in the use of the term eResearch relates to whether eResearch is actually the "research" conducted this way, or the infrastructure that enables the research conducted

Manuscript received: 13 December 2006
Communicating Editor: Sidney A. Morris

Characteristic	eResearch	Traditional Research
Participants	Diversely skilled, distributed research team	Individual researcher or small local research team
Data	Generated, stored and accessible from distributed locations	Locally generated, stored and accessible
Computation and Instrumentation	Large scale, or on-demand computation or access to shared instruments	Batch compute jobs or jobs run on researcher's own computers or research instruments
Networking	Reliant on the internet and middeware	Not reliant on the internet
Dissemination of Research	Via websites and specialized web portals	Via print publications or conference presentations

Table 1 – Traditional vs. e-Research Paradigms

this way. We adopt the view that eResearch strictly means research conducted relying on supporting infrastructure that should properly be called either *eResearch Infrastructure* or *Cyberinfrastructure*. From the table above, it is clear that the supporting infrastructure can includes hardware, software, networking, and human resources.

But eResearch is not just about using new IT tools, such as teleconferencing or web publications, to *support* research projects. Use of such tools is a common mischaracterization of eResearch. eResearch projects do not just use IT technology, rather they are *reliant* on IT technology and organizational changes such as online collaboration to achieve the research outcomes.

It is also important to note that eResearch adoption is highly discipline dependent. Scientific disciplines such as observational Astronomy or High-Energy Physics have arguably been using eScience for close on decades. Such disciplines intrinsically have some of the characteristics of eResearch above: large, expensive shared instruments and the need to share data internationally using agreed standards (e.g., astronomical coordinates and reference frames). By contrast, some disciplines such as Pure Mathematics or Linguistics intrinsically have few of the characteristics of eResearch. Yet even here, the trend is towards eResearch. For example the case of Mathematics it is the growing use of computers for proofs and proof checking, and a repository of known theorems (Cruz-Filipe, 2004).

So eResearch support is not a "one size fits all" – it is discipline and project dependent. There is no such thing as "eResearch Support In a Box". The very nature of research, which is constantly testing and pushing the boundaries of knowledge, means that eResearch support itself must be constantly pushing the boundaries of networking, data, computational, and collaboration support. eResearch infrastructure is a large system that is made up of a number of organic components software and hardware and organizational components, where each researcher should be able to readily find the components they want and need and not worry about the remainder. But they will have to be able to reach out and grab additional components when the need arises as it invariably will. For example, a scientist may find that they need access to a statistical analysis or visualization tool to interpret their data, or import data from a new instrument source that has just become available. Clearly, the individual components must work seamlessly together and the researcher who is widening his use must find the additional components working exactly as he expects, no surprises!

2. THE APAC GRID – AN eRESEARCH CASE STUDY

The Australian Partnership for Advanced Computing (APAC, www.apac.edu.au) started a program in 2004 to build a national grid, providing compute and data services to researchers across Australia. Computer grids had been introduced around 2000 (Foster, 2001), and by 2004 many such grids were appearing around the world at that time (e.g., the Teragrid in the USA), and APAC wished to be internationally competitive in eResearch Infrastructure. Approximately AU$6M was allocated to the project in direct funding over three years, and an equivalent amount of in-kind funding.

Some grid projects, such as Belle (Winton, 2004), support only a single research discipline or project, and thus can impose tight restrictions on connecting compute and data platforms. In the limit, such grids look like a distributed application. By contrast, the APAC Grid is build from a partnership between the quite diverse High Performance Computing (HPC) centres around Australia, one in each state, CSIRO and the ANU. APAC funded some manpower but each site was expected to contribute compute capacity and storage as part of their contribution to the partnership. So every ‘node’ in the grid is different, and there is thus a wide range of hardware platforms and an even wider range of access policies (e.g., how computer accounts and resources are allocated). The APAC grid needs to support interconnection and access between any pair of nodes, leading potentially to a very complex system to maintain and extend.

3. WHAT IS A COMPUTE GRID?

Traditional HPC was done from “the command line” – a user logs into an HPC cluster, submits a job and any data files it needs, then logs back in later to retrieve the data generated from the complete compute run. Generally large HPC systems are too expensive to be used interactively – instead a user requests a quantity CPUs, then is put into a batch queue until that number of CPUs is free. Then those CPUs are dedicated to the user’s job until it terminates.

Connecting together computers peer-to-peer into a simple grid using standard networking protocols is not difficult. Such ad hoc compute grids date back to BSD UNIX in the early 1980s. However, such ad-hoc grids are very difficult to use beyond a small number of nodes. The reason is simply that a user has to have separate accounts on each node, and manually remotely log in and submit jobs on a chosen node. Managing data and job transfers between heterogeneous nodes, and adapting the job to the peculiarities of each node (compilers, runtime environments, and file systems) quickly becomes impractical.

Ideally, any grid should have the property of *transparency* – a user should be unaware of which particular node any job is running on, and data transfers between nodes should occur automatically. In practice, such general-purpose transparency is only possible to achieve in a tightly coupled, homogenous local compute cluster. However, transparency can be provided for a particular application, typically by building a custom *web-portal* for managing the submission of jobs for that application.

Web portals are related to websites, in that both use the internet and HTTP protocols. However, a web portal for eResearch additionally acts as a gateway to remote computers (as well as processing some requests locally as a website does). Developing and deploying web portals across a large heterogeneous compute grid can be a complex task and combinatorially expensive (deploying portals for N applications across M machines requires N*M installs and updates).

Fortunately, there is a wide variety of software tools and packages that have been developed to help deploy and maintain data and computational grids and web portals. *Middleware* is the commonly used term for such software that provides an interface between end user applications and compute nodes operating systems. Middleware extends from the web portal down to the scheduling tools that actually launch the job on a particular supercomputer. Gridsphere, Globus, Condor, Nimrod/G and VDT are all examples of middleware. There is a huge and continuously evolving range of middleware available, most based on standards and most freely available (e.g., from www.sourceforge.net). The range of middleware available actually create quite a problem for grid maintainers, as each portal development effort or application may have chosen a different collection of middleware tools, all of which must be installed all over the grid before the application can be deployed.

Naively, it might seem then to build a compute grid all you need is to select the appropriate middleware and get a number of supercomputer "owners" to install it. However, others and we have quickly found that it is not quite that easy. Hurdles in the way of a successful deployment include:

- *End users* – don't start without them, they provide the requirements and use cases that must be satisfied or the whole project is at risk. It is critical that they have active commitment and significant time contribution to the project. Commercial software development has long recognized the risks of not engaging with end users, and end-users who are researchers present an additional challenge insofar as they are individualistic and highly creative.
- *End user support staff* – most middleware is not "commercial strength", meaning it is difficult to install and maintain. So compute grids need full-time support staff that can talk to the end users, in their language, and install and use middleware. Its neither reasonable nor productive to expect most end user researchers to install or use middleware. Support staff need to be programmers and systems support staff with specialist skills.
- *Middleware* – much middleware is written just for specific applications or environments, and interoperability between versions is a major problem. Some middleware, such as globus (Foster, 1997), is intended to occupy a particular layer in a grid stack and depends on other tools to connect at either end. Worse, some middleware is designed to do everything and won't interoperate with anything else. Versioning is a tremendous problem, as middleware products generally come in several evolving versions, each of which can depend on various versions and configurations of other tools
- *Authentication, Authorisation and Accounting (AAA) infrastructure* – while strictly speaking middleware, AAA tools are distinct enough to need to be listed and considered separately and present major headaches for deploying large scale heterogeneous compute grids. Recall that the nodes in the grid are owned and operated by different organizations, each with their own existing usage rules that apply to their hundreds of local users. Thus it is not feasible or maintainable to give every user an account and password on every machine (for authentication), nor is it feasible to create a separate authorisation or group privilege for each project.

4. IMPLEMENTING THE APAC GRID

APAC provided leadership and some funding to build the APAC grid in a project that started in 2004. A few scientific fields of science were chosen as demonstration projects and funding was provided to build web based portals or workflow engines to cater to particular needs within those sciences. APAC funded systems staff at each APAC Partner site and charged them with preparing the middleware. The project was intended to run for three years with a goal of having usable grid infrastructure in place and operational by the end of 2007 (APAC, 2006).

Three distinct infrastructure groups were identified, portals (and workflow), data and compute. Staff from APAC Partners, which include state-based HPC support centres in all six Australian states, were appointed to lead each group. One of the authors, David Bannon, of VPAC was appointed to the lead the compute team. Initially it was determined that these leaders would devote about half their time to the APAC Grid project. An APAC Grid Manager was appointed, again, nominally half time.

Half way through the three year project, APAC changed this model somewhat, some project leaders were asked to increase their time commitment and the number of application projects was increased, more because of existing projects being divided rather than having new ones added.

Overall the project was closely managed with six monthly reports being required and good communications between the geographically separate sites. The only structural difficulties encountered were the occasional misconception that grid funding was for research purposes or for the day-to-day running of a partner site's operations.

5. OUTCOMES OF THE APAC GRID

Like many software projects, the effort required for the APAC Grid rollout was underestimated, and this has been confirmed by external expert reviews. In spite of under-resourcing (lack of full-time dedicated staff to support projects) many of the goals of the APAC Grid have been met. Significant achievements include:

- Standards based grid infrastructure installed at all partner sites allowing generic grid jobs to be run by members.
- Several application specific portals operational and in use, both compute based and data distribution.
- A working AAA infrastructure in place, on line reporting of usage (http://goc.apac.edu.au), a virtual origination management system operational and an internationally recognized PKI x509 based certificate authority.

6. APAC GRID – BENEFITS AND INNOVATIONS

Apart from the availability of the grid compute infrastructure itself the overall project has delivered a number of additional benefits to APAC partners and others.

- Development of the grid gateway concept (Bannon, 2006). The APAC Grid, confronted with a very diverse range and configuration of compute hardware could not follow the usual practice of installing grid middleware on head nodes of clusters. Instead, a grid gateway model was developed. A single box divided into a number of virtual machines (using Xen) and each machine providing a distinct function, perhaps a particular grid middleware stack, authentications etc. This has proved to be a very scalable and flexible architecture.
- Recognition of the importance and development of a standard "look and feel" between APAC Partners, even out of the grid field. This makes for easier transitions between partners for researchers making their applications more portable.
- A closer relationship between APAC Partners. During the APAC Grid project, APAC Partners came to realise that they have much in common and much to share and all partners stand to gain from every win by one partner.
- A sound skill base has been created in Australia in a technological field that is going to be a key component of research in the future.

7. APAC GRID – LESSONS LEARNED

While the APAC Grid Project was a great success (especially relative to many research software projects), there are several things that, with hindsight, could have been done better, or differently or not at all:

- *Pick the Right End Users* – the project targeted a number of high profile researchers, people with established track records and people who are already expert HPC users. These users, while initially keen to cooperate with the high-profile APAC Grid project soon found that the grid was cumbersome – in any technology field, expert users of traditional technology are often the least receptive to change. We would have been better targeting end users who were less comfortable with their existing HPC use and therefore more receptive to alternatives, and end user communities better suited to the pros of grid computing.
- *Web Portals are not the answer for everyone* – Initially it was thought that portals were the key to grid users (web-based solutions must be good). In fact, the restrictions that a portal places on a user can discourage many users, in particular those experts with the good HPC skills mentioned above. Workflows and launching scripts seem to be more attractive to many users. Perhaps they are coupling a number of processing stages together or maybe processing a large number of data stacks.
- *Wide area physical and organizational grid distribution is hard* – by its nature, a grid is distributed, typically by at least kilometers. But the APAC Grid is very widely and sparsely distributed by international standards (hundreds of kilometers between nodes). This has lead to many communication problems, both technical and organizational. The technical issues relate to the poor performance and reliability of distributed file systems over hundreds of kilometers. The organizational issues relate to a tendency for APAC Grid funded staff to find themselves torn between the APAC Grid's interests and their own site's. The project made extensive use of the Access Grid video conferencing system and conducted many face-to-face meetings but there is no substitute for being able to walk into someone's office to discuss a technical problem. Overall, wide area distribution is just a problem that has to be lived with.
- *Data Grids are less mature than compute grids* – while much has been done, the APAC Grid has not managed to get on top of the data issues quite as well as it did with portals, workflow and compute nodes. This is due to two reasons. Firstly, data and its middleware are a much harder and much less well-supported problem than compute grids (Hoschek, 2000) – due to complex issues such as distributed data access, replication, metadata management, security, provenance, and curation (what policy and funding will maintain the data, and for how long)? Secondly, there are many more agencies engaged in data curation, notably libraries and archives, which are increasingly digitizing their collections. Initially, one simplistic view within APAC was that tools such as SRB would be the under pinning of the Data Grid (an "install it and they will come" approach) but except for specific applications, this has not worked out.
- *Avoiding the Not Invented Here (NIH) Syndrome* – the APAC Grid has been very lucky in having the services of many technically excellent people as it has had. But one problem endlessly encountered is the tendency for good technical people to see a problem as something they need to create a solution for. The truth is that a good number of the problems we have had are problems that others have already experienced and perhaps even fixed. The need to use other people's tools, even if they are initially ill suited to the application cannot be over emphasized. There is no doubt that the other people's tool often need a great deal of work to be made suitable but once done, sometimes, its someone else's problem to maintain that tool.
- *Build On Alliances and Partners* – surprisingly, the APAC Grid found very few problems working with the various partners with their various practices and policies. The good will was a pleasant surprise and a credit to all involved. However there is a noticeable pattern within the current grid whereby users of the grid tend to use only resources made available within their own state. While there is nothing wrong with this it is an indication that the grid approach is not fully adopted and should be viewed with just a little concern.

These lessons learned have been broadly, although not universally, recognized within the wide Australian academic research community that uses large-scale compute and data infrastructure. The Australian government is funding a "successor" to APAC (which was funded through two 3-year grants expiring in 06/2007), through a program called the NCRIS Platforms for Collaboration (PfC). There is every indication that PfC's organization and programs build on the APAC Grid, while addressing its weaknesses and lessons learned.

CONCLUSION

e-Research is "here to stay". Both in Australia and overseas, there is an evolving but sustained push from academic research funding agencies towards collaboration, inter-discipinlinary research, and effective use of IT to support research (e-Research or Cyberinfrastructure) across an increasing range of disciplines. This push, combined with the motivation for researchers to tackle ever more complex multidisciplinary problems, will drive the increasing adoption of e-Research. However, this can create a "cultural divide" between traditional academia, rewarding relatively short-term and independent and individual research efforts, and e-Research, which encourages long-term, large-scale collaboration. e-Research offers diverse incentive, opportunities, and challenges for different research communities. For example, the global climate modeling research community is highly dependent on, and committed to, an e-Research agenda. By contrast, there are many researchers who still have no incentive, need or desire to engage in the complex web that is e-Research.

ACKNOWLEDGEMENTS

We are grateful for the support of APAC, the Australian Partnership for Advanced Computing, and VPAC, the Victorian Partnership for Advanced Computing, through a range of grants and support programs from the Commonwealth and Victorian governments.

REFERENCES

APAC (2006): APAC Grid, http://www.vpac.org/twiki/bin/view/APACgrid/GeneralInformation (accessed 15 Feb, 2007).

BANNON, D., CHHABRA, R., CODDINGTON, D.C., CRAWFORD, C., FRANCIS, R., GALANG, G., JENKINS, G., ROSA, M.L., McMAHON, S., RANKINE, T., WOODCOCK, R. and WRIGHT, A. (2006): Experiences with a Grid gateway architecture using virtual machines, *Proc. of First International Workshop on Virtualization Technology in Distributed Computing* (VTDC 2006), Tampa, USA, November.

CRUZ-FILIPE, L., GEUVERS, H., WIEDIJK, F., ASPERTI, A., BANCEREK, G. and TRYBULEE, A.: C-CoRN (2004): The constructive coq repository at Nijmegen, *Lecture Notes in Computer Science*, 3119: 88–103, Springer, Berlin.

CYBERINFRASTRUCTURE (2006): Cyberinfrastructure, http://en.wikipedia.org/ (accessed 15 February, 2007).

FOSTER, I. and KESSELMAN, C. (1997): Globus: A metacomputing infrastructure toolkit, *International Journal of High Performance Computing Applications*, 11(2): 115–128.

FOSTER, I., KESSELMAN, C. and TUECKE, S. (2001): The anatomy of the grid: Enabling scalable virtual organizations, *International Journal of High Performance Computing Applications*, 15(3): 200–222.

HOSCHEK, W., JAEN-MARTINEZ, J., SAMAR, A., STOCKINGER, H. and STOCKINGER, K. (2000): Data management in an international data grid project, *Lecture Notes in Computer Science*, 1971: 77, Springer, Berlin.

WINTON, L. (2004): Data grids and high energy physics, a Melbourne perspective, *Space Science Reviews*, 107(1–2): 523–540, Springer, Netherlands.

BIOGRAPHICAL NOTES

Bill Appelbe

Bill Appelbe is the founding CEO and Chief Scientist of VPAC, the Victorian Partnership for Advanced Computing. He is also an Adjunct Professor at Monash University and RMIT, on the Board of Directors of the AutoCRC, and the Executive Board of the NSF funded Center for Computational Infrastructure in Geodynamics based at Caltech. Bill has a PhD in Computer Science and Electrical Engineering from the University of British Columbia

(1978). His areas of expertise include software engineering and parallel programming, and he has published more than 60 papers in refereed journals and conferences.

David Bannon is the manager of the Systems group at VPAC, the Victorian Partnership of Advanced Computing, which supports a range of HPC systems for universities and industry. He is also the manager of the APAC Grid project, that links together the major HPC centres in each state in Australia.

David Bannon