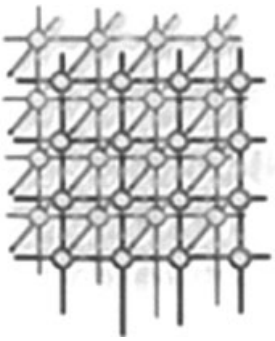
CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2010; **22**:2369–2380
Published online 28 August 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.1611



e-Science Central for CARMEN: science as a service

Paul Watson*,†, Hugo Hiden and Simon Woodman

School of Computing Science, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K.

SUMMARY

Scientists face many severe challenges in extracting value from the increasingly large volumes of data they generate. In this paper we describe the requirements we have derived from working across a wide range of e-science projects. In particular, the CARMEN neuroinformatics project has exposed a range of challenges due to a need to analyse and share large volumes of data. We have identified the four key activities required by scientists with whom we work, and designed an integrated system—e-Science Central—to provide them. This exploits three emerging technologies: software as a service to avoid the need for users to deploy and maintain any of their own software; social networking to allow users to collaborate by sharing data, services and workflows in a controlled manner and Cloud computing to provide scalable compute resources. The system can not only be used through any web browser, but also provides an API so that applications can build on the core functionality. We describe the requirements, and the design that flows from them. This includes data storage with in-built versioning and signing, an in-browser workflow editor and a job scheduling system that allows workflows to be run both on local 'private' clouds and the Microsoft Azure Cloud. Copyright © 2010 John Wiley & Sons, Ltd.

Received 9 April 2009; Accepted 30 March 2010

KEY WORDS: e-science; Cloud computing; workflow; neuroinformatics

INTRODUCTION

Over the past 10 years we have collaborated with scientists in many domains with the aim of assisting them to achieve more in their research. A key problem is to extract value from the increasingly large volumes of data that science now generates. For example, neuroscientists in the CARMEN project [1] are facing severe challenges. Understanding how the brain encodes, transmits

*Correspondence to: Paul Watson, School of Computing Science, Newcastle University, Newcastle-upon-Tyne NE1 7RU, U.K.
†E-mail: paul.watson@ncl.ac.uk

Contract/grant sponsor: EPSRC (CARMEN Project)
Contract/grant sponsor: OneNE (North East Software Services project)
Contract/grant sponsor: Microsoft (Junior Project)

and processes information is one of the major challenges in science, and significant progress in this area could revolutionize biology, medicine and computer science; it would help promote an understanding of brain development, assist in drug design and help to understand how to design systems that can carry out tasks, such as complex image recognition, which are beyond the current artificial computational systems.

Over 100 000 neuroscientists around the world are working on this problem, using various types of experimental data as evidence; these range across the molecular (genomic, proteomic and small molecule), neurophysiological (time-series activity), anatomical (spatial) and behavioural fields. Unfortunately, although this data is expensive to collect, it is usually locally stored and described. This makes it less likely that it will be shared with other researchers, and so often the full value is not extracted from it.

These issues are not restricted to Neuroscience. Much scientific work currently adheres to Bowker's 'Standard Scientific Model' [2]. This describes the process by which scientists first collect data; second they analyse the data and publish results and third they gradually lose the original data. This process causes a number of problems for science: papers often draw conclusions from data that is not published, and it is not possible to accurately reproduce experiments. In addition, where the original data is not published, it cannot be re-used for other experiments which may extract more value from it.

As scientific results are increasingly generated by computer programs, Bowker's model can also be translated to highlight a similar problem for software services and workflows used in e-science. If a paper includes results produced by running an analysis routine over some data, then how can we check how sensitive the results are to variations in parameter values, or re-use that same service on other datasets to see if similar results are produced?

Therefore, in the case of both data and software, lack of access means that experiments cannot be reproduced, while 'Standing of the Shoulders of Giants' is not possible when these potentially valuable resources cannot be used as a foundation for new research. These problems are by no means unique to neuroscience and hence we have been exploring how to address them for the range of researchers that we work with.

Several solutions have been proposed and implemented in the past, but we believe that none really solve this problem. There are moves by publishers to force authors to place the data used in papers in open repositories [3]. This is a step in the right direction, but it does not address the fact that there are many advantages in sharing data before the point of publication—which may come many years after the data is produced. Also, data that is never used in a paper might be very valuable to another scientist. Finally, results are often highly dependent on analysis services, and it does nothing to make those available.

One positive recent move in this direction has been the exposure of code as accessible Web or REST services [4]. However, whereas this has worked well to encourage sharing and re-use [5], there are still problems in that it requires the author to ensure that the service remains available for the period of time in which others may wish to use it. Investigations in the myGrid [4] project showed that large workflows often could not be run because at least one service would be unavailable.

Another aspect of the problem is that while sharing is in the long-term interests of the scientific community as a whole, individual scientists gain recognition through being the first to publish a new idea, analysis or insight. Therefore any attempt to support fully open publication of data and

DOI: 10.1002/cpe

services at all stages of the scientific process are doomed to fail, for social reasons, even if all the technical problems could be overcome.

For all the above reasons, we felt that the only way to address the problem was to investigate ways to support the storage and *controlled* sharing of both services and data as part of the day-to-day process of scientific work. To this end we are currently exploring a new approach—e-Science Central. This is an integrated system for scientists that supports the storage and analysis of data, along with the sharing of both services and data. It is designed as a 'Science Cloud'—it offers a 'Software as a Service', browser-only interface to users, whereas internally it is structured as a Science Cloud Platform that offers e-science middleware to service and application writers.

The remainder of this paper describes e-Science Central, using the CARMEN neuroscience project to explain the requirements placed on it, and to give examples of its use. We describe the functionality of the system, its design and give an example of its use in CARMEN. Finally we draw the conclusions and point to future work.

REQUIREMENTS AND USER SCENARIO

When we examine the projects in which the North East Regional e-Science Centre has participated, there are four activities that emerge as being of prime importance for the scientists we have collaborated with:

- Storing the data collected from their experiments
- Interactively exploring and analysing the data
- Automating the analysis once a repetitive technique has been devised
- Sharing both the data and analysis services with selected colleagues and organizations.

Each of these activities presents its own challenges. Storage can be a problem as experiments often now generate vast quantities of data. Analysing the data can therefore require large processing capacity, and, even if the scientist has access to such resources, the cost of moving the data to them can be a bottleneck. Sharing data and services requires a way to provide controlled, remote access to colleagues irrespective of their location, at all stages of the scientific process, and demands the presence of metadata so that those re-using data understand its format and meaning.

We could find no existing, integrated 'off-the-shelf' way to meet these needs, although there are some separate components produced or exploited in e-science projects which address these issues separately. However, our experience is that deploying, integrating and harnessing the capabilities of these components is not straightforward, and is certainly beyond the capabilities of a typical application scientist.

As a result, we have been pursuing an alternative, which is to design and build e-Science Central—an integrated 'Science Cloud'. This has allowed us to explore the exploitation of three recent trends in computing to address the challenges described earlier:

Software as a Service: e-Science Central is provided as a website. Users can login to the e-Science Central website from any browser, and can then upload and analyse data, and share data and services. This allows scientists to work from anywhere, including be it on a PC or a mobile device.

Cloud computing to provide resources that scale with the number of users, size of data and the cost of performing the analysis.

DOI: 10.1002/cpe

Social networking: We adopt the approaches taken in social networking applications in order to support the sharing of data and services, and to facilitate user collaboration [5,6].

We now describe a typical scientific scenario and show how this is tackled using e-Science Central.

A neuroscientist performs an experiment in the lab, and the results are automatically recorded in a local file. The scientist logs-on to the e-Science Central Website and uploads the file. She also enters the associated metadata that will be used to locate and understand the data. At this stage, she can choose to share the data with selected colleagues (or everyone), but in this case she decides to wait until she has completed some preliminary analysis.

This analysis uses a workflow. She may pick an existing workflow created by herself or a colleague, or if this is the first time she has performed this type of analysis she may interactively

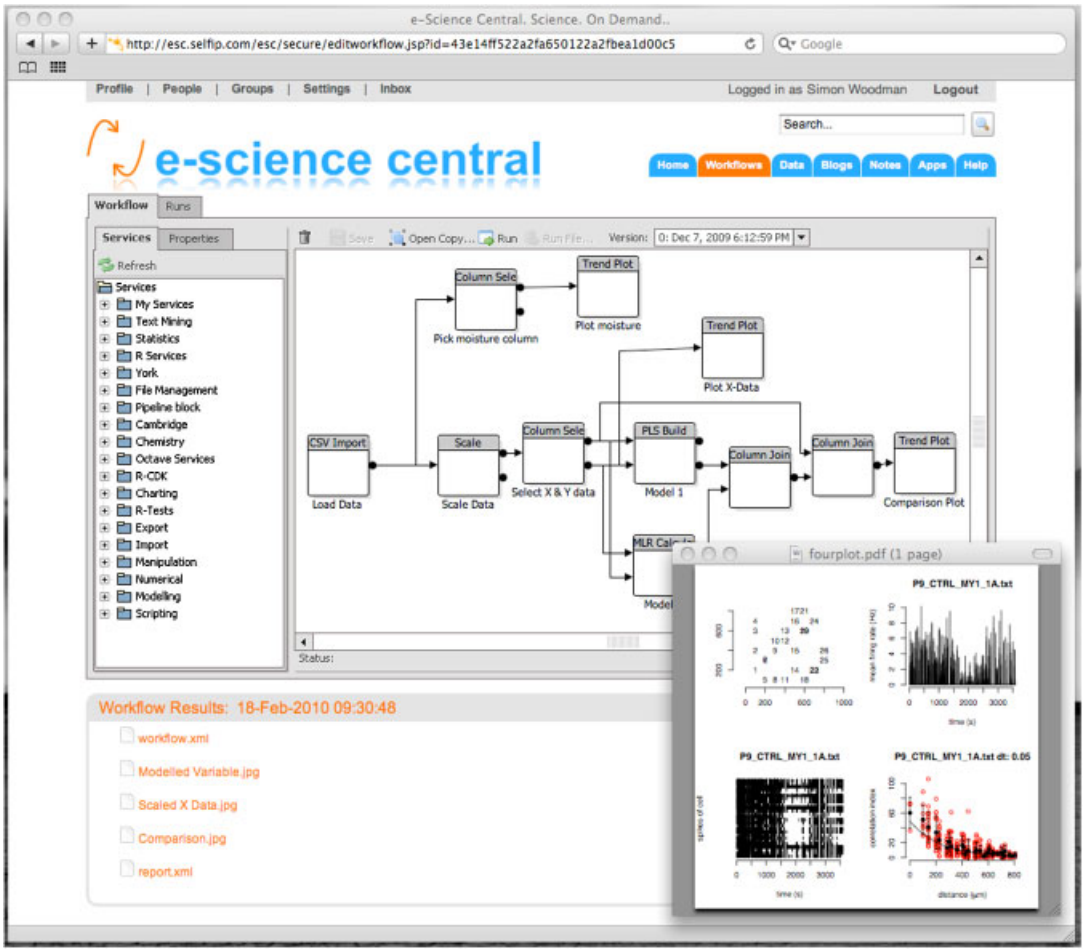


Figure 1. The Workflow Editor, with the result of an execution.

DOI: 10.1002/cpe

build a new workflow using the web-based editor (Figure 1). In doing this, she may use existing blocks (the term used in e-Science Central for services that have well-defined inputs and outputs and hence can be combined in workflows), or may need to write one or more new blocks that she can then combine into the workflow. The process of designing the workflow is likely to be iterative during the period when the scientist is exploring the data for the first time. When she is happy with a workflow, the scientist can run it on the new data (and any subsequent data she collects) at the press of a button. The workflows will execute within e-Science Central and hence the scientist need not concern herself about where the computations are being performed. In this scenario, the workflow produces some results in the form of statistics, graphs and tables which are automatically stored in e-Science Central, ready for her to review and possibly to use in publications.

During the process of collecting and analysing the data, the scientist may share the data, workflows, results and blocks with selected colleagues when she is ready. In the early stages, access is likely to be heavily restricted, perhaps only to close colleagues. Once the preliminary analysis has been completed, the scientist may allow a slightly wider audience of colleagues and collaborators to view the material. Later, while the paper that describes the results is in the publication process, the scientist may give the editors and reviewers access to the data. Finally, once the paper has been published, the scientist may allow full access to the data, workflows and results. This process of opening the data to wider audiences is likely to depend on the work practices of the scientist and the general attitude to open publication in their scientific community. In some extreme cases, patient confidentiality or other privacy concerns may prevent any sharing of the original data.

After reading the publication, another scientist may be interested in whether his own data might also lead to the same scientific conclusions. He uploads his new data into e-Science Central and analyses it using the workflow referenced in the paper. He then uploads a new, improved analysis service he has written and uses it to re-analyse the original data to see if any new, publishable results are produced.

Keeping the original data available after publication may also encourage new avenues of research which were not considered at the time of the original paper. For example, the availability of the Enron Corpus has stimulated a diverse range of research into e-mail and Social Network Analysis [7].

ARCHITECTURE

Figure 2 shows the architecture of e-Science Central.

It utilizes a 'Science Cloud' which is accessed by users through a Web Browser, or by applications through an API. Traditionally, scientists have had to apply for funding to acquire the compute resources they needed for their science. This often resulted in a long delay between having an idea and having access to the resources necessary to realize it. Grid computing attempted to address this by providing easier access to shared facilities [8] and by allowing scientists to share and combine their own resources with those of collaborators where it was to their mutual advantage [9]. Cloud computing is a recent development offering the promise of access to compute resources as and when they are required [10]. What is significant about Cloud computing for scientists is not any particular novelty in the technology, but in the fact that commercial organizations are offering vast compute resources on demand. This allows scientists to gain immediate access to the resources they need (for a fee) and hence has the potential to revolutionize e-science by reducing the time

DOI: 10.1002/cpe

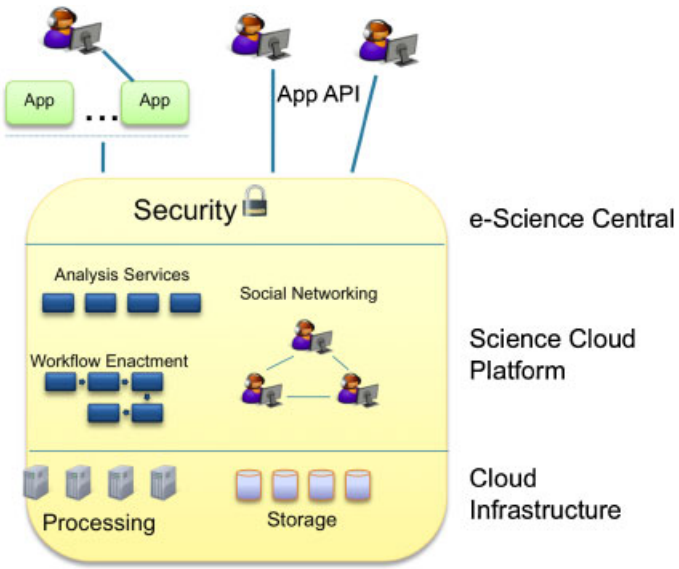


Figure 2. e-Science Central architecture.

from idea to realization. The key characteristics that differentiate clouds are: the illusion of infinite computing resources made available on-demand; no up-front commitment by users and pay-for-use of resources on a short-term basis, as needed [10]. These resources can be located in a private cloud [11], or externally in a public, commercial cloud [12,13].

We also chose to use a Cloud as we wanted users to interact with applications remotely over the internet (typically through a web browser). This approach has several advantages for both application providers and users. It prevents the application writer from having to buy and manage their own hardware; instead they can use highly scalable resources in the cloud to meet their needs. Owing to the commercial 'pay-as-you-go' payment regimes, they are only charged for resources as they need them and do not have to worry about over-provisioning (which wastes money on underused hardware) nor under-provisioning (which can result in disastrously poor performance for users). For users, having services delivered over the web removes the need to deploy, manage and maintain software on their own resources. With the growth of the mobile internet, it also opens up the possibility of being able to interact with a service from many locations—at work, at home and while travelling. From the point of view of the resource providers, it allows them to exploit centralized data storage and computation in large data centres which, due to economies of scale, reduces costs and energy consumption.

The Cloud computing approach was particularly attractive for meeting the CARMEN requirements because of the significant amount of data that will be stored and analysed by scientists. Current estimates put this in excess of 100TB by 2010 for the neuroscientists involved in the project, although if video capture of neuronal activity continues to supersede electrode-based recording this may be a serious underestimate. Where there are huge amounts of data to be processed, it is inefficient to move data over large distances [14]. This requires having computational resources

DOI: 10.1002/cpe

closely coupled to the servers holding the data. Cloud computing offers the chance to do this if the cloud is internally engineered with fast networking between the storage and compute servers. Once the data has been uploaded (once) into the cloud, it is then stored in the same data centre as the compute resources, hence allowing services to operate on vast amounts of data (e.g. the TBs collected in neuroscience) without them having to be transferred over the internet.

There are limits to what can be achieved with Cloud computing; highly interactive tasks requiring graphically rich interfaces may not work well as web applications. CARMEN utilizes one such application—the Signal Data Explorer [15]—that is deployed on the users' desktop, and so the project is taking the liberal approach of using web-based services where possible, but supporting desktop services where necessary. As will be described, an API is provided to allow such applications to programmatically access the data and functionality of the Cloud.

The basic aim of CARMEN is therefore to provide a cloud (which we name a CAIRN) that neuroscientists interact with through a web-based portal. Existing Cloud computing offerings focus on providing low-level compute and data storage services (e.g. Amazon S3 and EC2 [16], Microsoft Azure [13]). It would be possible to build services and applications to support these neuroscience requirements directly on this low-level platform, but for CARMEN we chose instead to identify and deploy a set of generic e-science services, and then build domain-specific neuroinformatics services and applications on top of these. In Anderson's three tier decomposition of clouds [17] the generic e-science services form the Cloud Platform, sitting on the lower level Cloud Infrastructure that provides dynamic access to storage and compute resources.

The selection and design of these services was made based on our experiences in a variety of e-science projects carried out since 2001, targeting a wide range of disciplines from bioinformatics, through transport, to artistic performance. Figure 3 shows the collection of e-Science services in the e-Science Central Cloud. These are now described in turn.

Storage

e-Science Central's storage system supports data versioning, signing and multiple storage systems. To support the widest possible deployment options, all data is persisted through a virtualized storage system that has multiple back-end drivers. This virtualized interface supports document versioning, and drivers are currently available for filesystems, relational databases and the Amazon S3 cloud storage service. We are currently working to provide drivers for the Storage Resource Broker (SRB) [18] and Microsoft Windows Azure [13].

All data is versioned so that scientists can work with old versions of data and workflows—this is important for reproducing experiments and seeing the effects of changes to data and the analysis process. For example, a scientist can retrieve the exact versions of both the data and workflow that were used in a publication some time ago, and see if running the latest version of the workflow on the same data makes any difference to the results.

All data versions are signed, hence it is possible to reliably validate data and detect any modifications. e-Science Central uses Public-Private key cryptography [19] and x509 certificates [20] to sign all data as it is uploaded to the system. A set of keys and certificates is created for each user during the registration process and these are subsequently used to sign data. Using the x509 certificates issued to users, any individual using data supplied by e-Science Central can verify its authenticity.

DOI: 10.1002/cpe

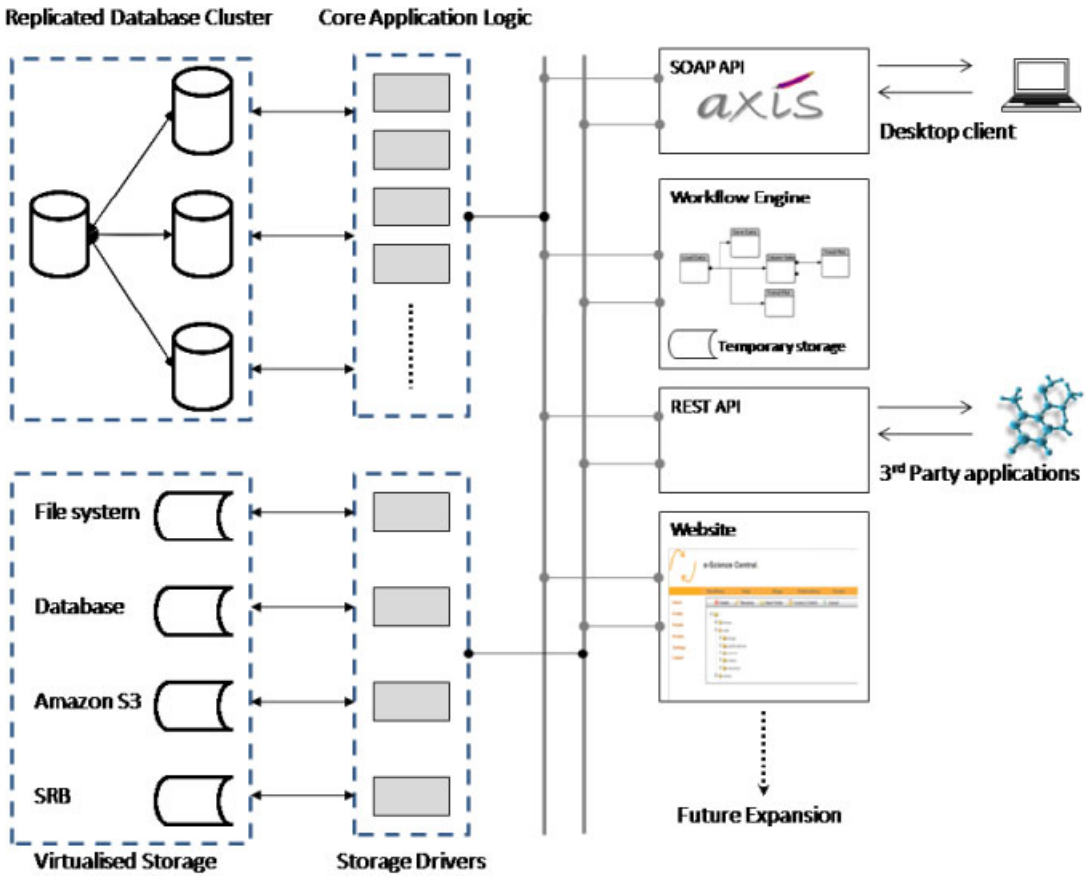


Figure 3. e-Science Central core architecture.

Social networking

e-Science Central provides a social networking implementation that allows users to form connections and create groups. It also allows the documentation of experiments through blogs and the controlled sharing of data and workflows by users to their contacts and groups. The social networking aspects of e-Science Central are modelled using the concept of links. These can connect users, files, workflows and external resources to such web pages. Links are similar to hyperlinks but are first class objects within e-Science Central and as such can have security and metadata attached to them. Security is enforced on links so that if the viewer of some data which contains links is not authorized to view the target of the link, they will be unaware of the link's existence. Another difference between these links and hyperlinks is that we enforce referential integrity on the links: if an object is removed, all incoming links to it are removed too.

Another aspect of the social networking system is the provision of blogs which allow a scientist to document their work and hence become an online lab book. There are several advantages in

DOI: 10.1002/cpe

documenting their work within e-Science Central when compared to a paper lab–book. First, they can allow selected others (colleagues, groups or everyone) to view their lab book so as to learn from their experiences. Second, as is normal for blogs, there is a facility for commenting, so that others can make suggestions, for example as to the interpretation of results. Finally, lab book entries can contain links to the data and workflows that they are discussing. This allows others not only to read about the work, but also to examine the data, re-produce the experiment, try other workflows on the same data, etc. Because of the versioning system described above, the links are guaranteed to point to the version of the data and workflows used at the time of the blog entry, even if they have subsequently been updated.

Security

A comprehensive security architecture is important for any data sharing platform, including one which will be used to exchange research data and results that have not yet been published. In the e-Science Central software, all entities—users, groups, relationships and stored data—are represented as subclasses of a single object. This allows all entities to be guarded by a security mechanism which controls access to a single point within the class hierarchy. Each object provides Access Control Lists (ACLs) [19] that are respected by this core object access code. These ACLs express security policies in terms of the actions that specific users and groups are allowed to perform on a given resource. ACLs are stored within a database as triples comprising the identity of the resource, the identity of the user or group and the action permitted. The set of actions that are allowed by e-Science Central are: Read, Write, Delete and Add (which allows the addition of data to a resource). Any attempt to access a secured object will evaluate the ACL associated with that object, with any rejected attempts being recorded by the logging service.

Workflow

Figure 4 shows the architecture of the workflow system (Figure 1 shows the workflow editor, and the result of a run). Scientists are able to design workflows using the drag-and-drop online workflow designer by selecting blocks (services) from a library (shown on the left of the editor pane in Figure 1). Generic blocks exist to provide file management, data manipulation, mathematical modelling and visualization. Libraries of domain-specific blocks are also provided—currently for neuroscience and chemical informatics. Blocks are connected by arcs, along which data flows. The input and output of each block is typed to prevent incompatible blocks being connected to each other—users are warned if they try to do this. In addition to the set of supplied blocks, users can write and upload their own blocks which they can use in workflows, and share with others (if they choose). Currently, Java and R [21] are supported, but other languages are being added, including Python and .net.

Workflows are saved into the e-Science Central storage and hence are versioned and signed. This provides some protection against malicious code being uploaded. When running a workflow, jobs corresponding to the workflow are placed on a job scheduler and executed by dedicated workflow servers. In the server, a workflow is ‘sandboxed’ so that it can only access data for which it has permission—its input and output files. Each workflow also has access to local storage for temporary files and for passing data within the workflow. Workflows run asynchronously so that the user

DOI: 10.1002/cpe

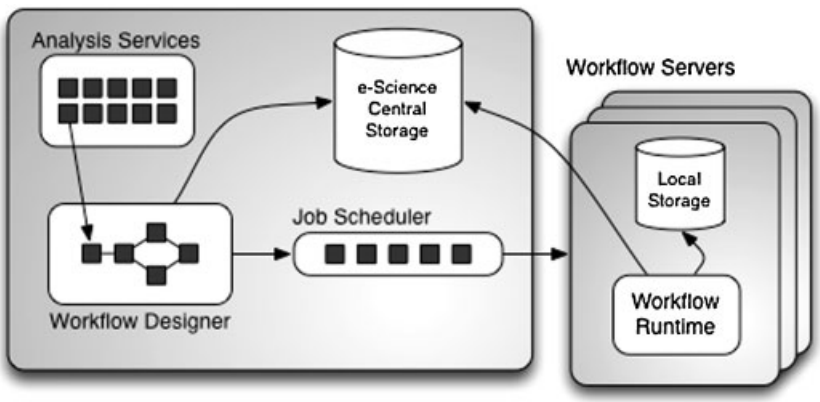


Figure 4. Workflow architecture.

can carry out other activities while they are running. The results of each run are stored in folders categorized by date, time and workflow name ready for the user to examine them. Every run also results in provenance traces that are stored in a database, and made available to the user so that she can, for example, see by which workflows a piece of data has been analysed, or on which data a workflow has already been run. This is useful if, for example, a bug is found in a workflow as it allows all data derived from the workflow to be identified.

e-Science Central has a simple workflow engine, which we believe will be sufficient for most scientific workflows. Its main advantages are: the browser-based editor which obviates the need to deploy any software on the desktop and its efficiency when operating on large datasets—no data need enter or leave the e-Science Central Cloud when a workflow is being enacted. However, to support more complex workflows, we are investigating integration with other systems, including Taverna [22] and Trident [23].

Application API

e-Science Central allows integration with third-party applications through providing a REST-based API. This provides operations to authenticate, retrieve and modify data, files and links, and to control the execution of workflows. Applications can be used to provide domain-specific functionality that is not present in e-Science Central or provide novel ways of viewing data (e.g. rich Silverlight applications or mashups using Google Maps).

e-Science Central users must give their explicit permission for an application to be able to access their data (and are able to restrict the privileges of the application). All API requests must be signed with a combination of a unique application key and a token that identifies the current user of the application. A hashing mechanism, similar to that adopted by sites, such as Facebook [6], is used to sign REST requests and any XML object representations posted to the system. Once the validity of the requests has been verified, the access granted to external applications is again controlled by the internal ACL mechanism.

DOI: 10.1002/cpe

CONCLUSIONS AND FUTURE WORK

e-Science Central has been designed to explore whether an integrated e-Science platform, accessible over the Web, can be used to meet the needs of the majority of application scientists with whom we work. It exploits three emerging technologies: software as a service to avoid the need for users to deploy and maintain any of their own software; social networking to allow users to collaborate by sharing data, services, workflows and blogs in a controlled manner and Cloud computing to provide scalable compute resources.

e-Science Central currently runs on our own servers which operate as a private cloud—resources are acquired and released as required. However, the main attraction of Cloud computing to science is, we believe, the access it provides to 'infinite' resources provided on a pay-as-you-go basis. We have therefore been experimenting with the Microsoft Azure Cloud. Early experiments running a QSAR drug discovery application have shown the benefits of having scalable resources available. This can be used both to speed up the generation of results, and to do more fine grained 'parameter sweeps', so building better chemical models. This is implemented by having the workflow scheduler (Figure 4) send jobs to a number of workflow servers located in the cloud. The number of servers is varied over time depending on the workload and the desired response time for users.

In the long term, using an external cloud will prevent us from having to raise new capital every three to four years to refresh the hardware on which e-Science Central runs. It will also allow us to grow the system organically as new users join. However, there still needs to be a business model to pay for the resources consumed in commercial clouds, and hence we are exploring charging models that would allow the organization running e-Science Central to recover costs from users depending on the resources they have used.

Over the coming months our aim is to attract more users, from a range of scientific domains, who will add content in the form of data, workflows and services. We are seeding this process by pre-loading content for particular areas such as neuroscience through the CARMEN project. However, as with most websites, success will be dependent on reaching a tipping point where a virtuous circle is in place—users are attracted to the system because of its content, which encourages them to add more content, so attracting more users.

ACKNOWLEDGEMENTS

We are grateful to the following for funding this work: EPSRC (CARMEN Project), OneNE (North East Software Services project), Microsoft (Junior Project). We acknowledge the contribution to CARMEN by our collaborators in Prof. Jim Austin's group at York University. The Microsoft Azure Cloud QSAR prototype was developed with Paul Appleby & team at the MS Technology Centre (Reading), Christophe Poulain (Microsoft External Research, Seattle) and Professor David Leahy (Newcastle). We are also pleased to acknowledge the contribution Dr Savas Parastatidis (Microsoft) has made to our approach to e-science over the past years.

REFERENCES

1. CARMEN: Code Analysis, Repository and Modelling for Neuroscience. Available at: http://www.carmen.org.uk/ [9 April 2009].
2. Bowker G. The New Knowledge Economy and Science and Technology Policy. *UCSD Technical Report*, Department of Communication, UCSD, 2002.

DOI: 10.1002/cpe

3. Engineering and Physical Sciences Research Council. Policy on Access to Research Outputs. Available at: http://www.epsrc.ac.uk/AboutEPSRC/AccessInfo/ROAccess.htm [9 April 2009].
4. myGrid. Available at: http://www.mygrid.org.uk/ [9 April 2009].
5. De Roure D, Goble C, Stevens R. The design and realisation of the myExperiment virtual research environment for social sharing of workflows. *Future Generation Computer Systems* 2008; **25**:561–567.
6. Facebook. Available at: http://www.facebook.com [9 April 2009].
7. Klimt B, Yang Y. The Enron corpus: A new dataset for email classification research. *Proceedings of Machine Learning: ECML 2004* (*Lecture Notes in Computer Science*). Springer: Berlin, 2004.
8. National Grid Service. Available at: http://www.grid-support.ac.uk/ [9 April 2009].
9. Catlett C. The philosophy of TeraGrid: Building an open, extensible, distributed TeraScale facility. *Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*. IEEE Computer Society: New York, May 2002; 8.
10. Armbrust M *et al.* Above the Clouds: A Berkeley View of Cloud Computing. EECS Department, University of California, Berkeley, 2009.
11. Arjuna Technologies Ltd. Available at: http://arjuna.com [9 April 2009].
12. Amazon Elastic Compute Cloud (EC2). Available at: http://aws.amazon.com/ec2/ [9 April 2009].
13. Windows Azure. Available at: http://www.microsoft.com/azure [9 April 2009].
14. Watson P. Databases in grid applications: Locality and distribution. *British National Conference on Databases*. Springer: Berlin, 2005.
15. Signal Data Explorer. Available at: http://www.cybula.com/flyers/SignalData.pdf [June 2010].
16. Amazon Simple Storage Service. Available at: http://aws.amazon.com/s3/ [9 April 2009].
17. Anderson RW. The Cloud Services Stack. Available at: http://et.cairene.net/2008/07/28/the-cloud-services-stack-infrastructure/ [9 April 2009].
18. Storage Resource Broker. Available at: http://www.sdsc.edu/srb/ [9 April 2009].
19. Anderson R. *Security Engineering*: *A Guide to Building Dependable Distributed Systems*. Wiley: New York, 2001.
20. IETF, Internet X.509 Public Key Infrastructure. Available at: http://www.ietf.org/rfc/rfc2459.txt [9 April 2009].
21. The R Project. Available at: http://www.r-project.org/ [9 April 2009].
22. Hull D, Wolstencroft K, Stevens R, Goble C, Pocock M, Li P, Oinn T. Taverna: A tool for building and running workflows of services. *Nucleic Acids Research* 2006; **34**(Web Server issue):729–732.
23. Barga RS, Jackson J, Araujo N, Guo D, Gautam N, Grochow K, Lazowska E. Trident: Scientific Workflow Workbench for Oceanography. *IEEE Congress on Services*. IEEE Computer Society: New York, 2008; 465–466.

DOI: 10.1002/cpe