# CARMEN: a Scalable Science Cloud

Paul Watson, School of Computing Science, Newcastle University, UK (paul.watson@ncl.ac.uk)

Jim Austin, Department of Computer Science, University of York, UK (austin@cs.york.ac.uk)

## Introduction

Understanding how the brain works is a major scientific challenge which will benefit medicine, biology and computer science. Globally, over 100,000 neuro-scientists are working on this problem. However, the data that forms the basis for their work is rarely shared even though it is difficult and expensive to produce. Nor are the analysis tools that scientists use to make sense of the data. The CARMEN project (www.carmen.org.uk) is addressing these challenges by developing a scalable cloud architecture for data sharing, integration, and analysis. An expandable range of services are provided in the cloud to extract value from raw and transformed data. This promotes the sharing of analysis services as well as data, and allows services to execute close to the data on which they operate.

## Requirements

Detailed requirements capture from the neuroscientists in the project has shown that their main needs are to:

- Store experimental data. Neuroscience research utilises various types of experimental data, including the neurophysiological, time-series data that it the focus of CARMEN.
- Share data in a controlled (user-defined) way with collaborators.
- Search for data that meets some criteria (e.g. all data captured under particular experimental conditions, or all data containing a specific pattern)
- Analyse data. It is not possible to define a closed set of services that will meet all scientists' needs: they must be able to use new analysis services that they find or invent.

## The CARMEN CAIRN Architecture

We adopted a Cloud architecture (which we call the CARMEN CAIRN) for two main reasons. Firstly, it allows the co-location of data and computation, which this is essential to avoid having to ship vast quantities of data (TBs) over the Internet for analysis. Secondly, it allows users to carry out most of their science with only a web browser: they simply login to the CAIRN to upload new experimental data or search and analyse existing data placed there by themselves and their collaborators.

Existing cloud computing offerings tend to focus on providing low-level compute and data storage services (e.g. Amazon S3 and EC2). It would be possible to build applications to support the neuroscience requirements directly on this low-level platform, but for CARMEN we chose instead to deploy a set of generic e-science services, and then build domain specific neuroinformatics services and content on top of these (Figure 1). The selection and design of the e-science services was made based on our experiences in a wide variety of e-science projects carried out since 2001.

In the long-term we would like to utilise external "pay as you go" cloud services to provide the basic storage and processing services (Fig 1). However, for experimental reasons we currently host the services on our own cluster. In the next section we focus on two of the key e-Science Services that run on this platform.

## CARMEN e-Science Services

**Data repository**. The primary data consists mainly of files of time-series signal data. We use the Storage Resource Broker (SRB) to manage the files [1]. Each node of the cluster runs an instance of the SRB which stores a subset of the data files. As techniques and instruments improve, the quantities of data being collected are increasing, and single experiments are now generating TB of data; in the lifetime of the project we expect to capture over 100TB of data. Searching for patterns (such as neuronal spikes) in these files is a key requirement and a scalable solution is needed. CARMEN therefore uses a novel parallel search infrastructure to find patterns quickly, even in vast quantities of data [2]. Each instance of the SRB has an associated local Pattern Matching Controller (PMC) [3]. This manages high performance algorithms to find

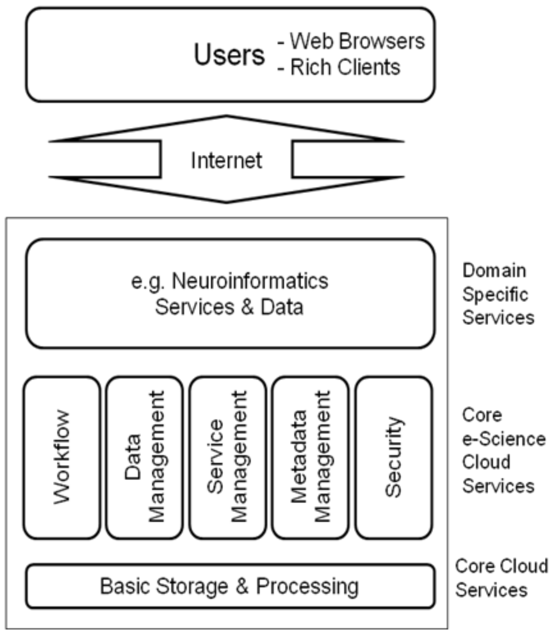


Figure 1. The Carmen E-Science Cloud

patterns in data. When a user wishes to locate a pattern, each PMC reads the relevant files from the local SRB and performs a pattern search on them. This parallelises the search across all the nodes and disks holding files in

 Google Seattle Conference on Scalability, June 14, 2008.

a SRB instance. The identifiers of any files that contain a match are returned to the user for further exploration.

Whilst CARMEN exploits the cloud computing approach where possible, we recognise that there are limits to what can be achieved in the Cloud; interactive tasks requiring graphically rich interfaces may not work well as web applications. One such application is the Signal Data Explorer [2] which the neuroscientists can deploy on their desktop to explore the time-series data files returned by the matching process. To support this and other applications, CARMEN provides APIs that allow external applications to access its facilities.

**Service Management**. Users can apply the services in the CAIRN to analysing their data. A novel feature of CARMEN is to allow users to write and upload their own services into the CAIRN. Services can be written in a variety of languages (MATLAB, C, C++ and Java are currently supported) and must be packaged as WS-I compliant Web Services before being uploaded into a Service Repository.

It is a challenge to provide a high-performance, scalable system when the services being invoked are changing as users upload and use new services. Traditional static, manual deployment of services onto available compute nodes is not feasible. Therefore, the Dynasoar [4] dynamic service deployment infrastructure is used to deploy the services on demand from the repository onto the available compute resources. The handling of each message sent to a service is split between the *Web Service Provider* and *Host Provider* components:

The *Web Service Provider* accepts the incoming SOAP message and forwards it to a Host Provider, along with a pointer to the service repository holding a deployable version of the service.

The *Host Provider* controls the computational resources in the CAIRN. When the message reaches the Host Provider, there are two possibilities, depending on whether or not the service is already deployed on the node that is to process the message. If the service is already deployed on the node then the Host Provider simply routes the message to the service for processing.

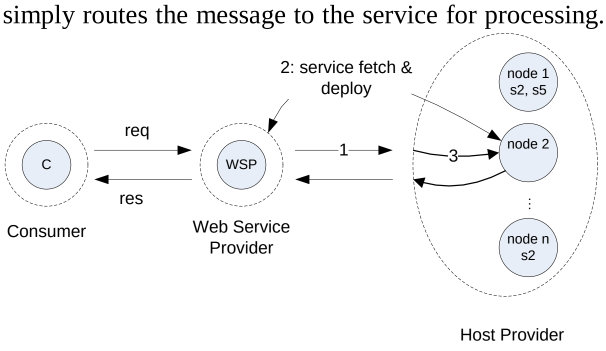


Figure 2. Dynamic Service Deployment

Figure 2 shows an example of the other possibility, when dynamic service deployment is required. A request for a service (say s3) is sent by the client to the endpoint at the Web Service Provider which passes it on to a Host Provider (step 1 in the Figure). As s3 is not deployed on any of the nodes it controls, based on loading information it chooses a node (node 2 in this case), fetches the service code from the Web Service Provider and installs the service on that node (step 2). It then routes the request to it for processing (step 3). The response is then routed back to the consumer.

To provide scalable performance, as the arrival rate of requests for a service increase, causing corresponding response time increases, the Host Provider will deploy the service on multiple nodes and load-balance requests across them (e.g. Service s2 in Fig. 2). Once a service is installed on a node it can process requests until the Host Provider decides to reclaim it. This can prove much more efficient than traditional job-based scheduling systems in which each job execution requires the program to be moved and installed. As a result, Dynasoar can efficiently support services wrapped in Virtual machines. Although the deployment cost may be high due to the time it takes to move a multi-GB VM from the Service Repository, once that is done there is no further cost for subsequent requests.

## Conclusions

A full prototype of the CARMEN CAIRN has been running for a year, and tested on a set of use-cases derived from a detailed study of the requirements of the project neuroscientists. The components of the CAIRN build heavily on software developed in previous e-Science projects (e.g. SRB, myGrid, Dynasoar, Symba, Dame, Gold). However the project is the first to integrate the core set of services that we believe are important for supporting e-science applications, and make them available to users “in the Cloud”. The CARMEN CAIRN is also novel in providing users with the ability not just to store and share both data and code, but also to perform analyses.

## Acknowledgements

CARMEN is funded by the UK Engineering and Physical Science Research Council (EPSRC). The project is a multi-site collaboration between neuroscientists and computer scientists, and we wish to acknowledge the contributions of all project members.

## References

[1] Storage Resource Broker, www.sdsc.edu/srb/
[2] The Signal Data Explorer, M. Fletcher, T. Jackson, M. Jessop, B. Liang, J. Austin, Proc. of 6$^{th}$ IEEE Int. Symposium on Cluster Computing and the Grid
[3] A Scalable Service Architecture for Distributed Search, M. Jessop, A. Pasley, J. Austin, Proc. UK e-Science All Hands Meeting, Sept. 2005.
[4] Dynamically Deploying Web Services on a Grid using Dynasoar, P. Watson, C. Fowler et al, Proc. Ninth IEEE ISORC, IEEE Computer Society 2006.