Science Clouds: Early Experiences in Cloud Computing for Scientific Applications

Kate Keahey and Tim Freeman

About this document

The Science Clouds provide EC2-style cycles to scientific projects. This document is dated from 08/13/08 and contains an early summary of experiences of this project.

1 Summary

The Science Clouds project provides cycles to scientific projects using the Infrastructure-as-a-Service (IaaS) paradigm (similar to Amazon's EC2 service [1]). The project was formed with two objectives:

- Make it easy for scientific and educational projects to experiment with IaaS-style cloud computing, and
- Better understand the requirements of scientific communities relevant to this paradigm and what needs to be done to overcome them

The Science Clouds project allows members of the scientific community to lease resources for short amounts of time, in a manner similar to Amazon's EC2 service [1]: a client requests a resource lease for a few hours and, if the request is authorized, a virtual machine (VM) is deployed. The client can then use the VM as needed (e.g., ssh to it, move data to it, or run computations) for the requested time. The power of this model lies in the fact that the client is allowed to bring a VM configured to his/her exact specifications and is given an exclusive ownership of the leased resource (the VM) to be shared with others only at the client's discretion. Unlike the EC2 service, the Science Clouds do not require users to directly pay for usage. Instead, we loosely verify that the person asking for an allocation is indeed a member of the scientific community (through verifying an email account with the sponsoring institution, web pages, pointers to papers, etc.) and ask for a short writeup of the scientific project. Based on the project the individual is allocated a small (testing), middle (development), or large (science) hour credit on the Science Clouds.

Since the first cloud became operational in March 2008, the Science Clouds testbed attracted scientific users from high-energy physics, bioinformatics, computer science, economics, and others (see Section 3). It is also being evaluated for use in multiple educational and outreach projects. To facilitate movement between Science Clouds and commercial venues (right now Amazon's EC2 only) we built an IaaS gateway which allows scientific projects to move to commercial infrastructures for more cycles. The two original Science Clouds at University of Chicago and University of Florida have been joined by clouds configured by resource providers in Purdue and various European institutions. The following presentation describes the software that makes Science Clouds possible, the Science Cloud configuration, and summarizes our early experiences with the Science Cloud testbed.

2 The Nimbus "CloudKit"

The nimbus toolkit (formerly known as the "virtual workspace service" or simply "workspace service") was developed with the goal of providing an open source implementation of a service that allows a client to lease remote resources by mapping environments, or “workspaces” (e.g., implemented by VMs), onto those resources. Its primary objectives are to provide infrastructure semantics addressing the needs of the *scientific community*, in particular, through resource leases. Providing an extensible [2] open source implementation allows us to integrate community contributions of features required by scientists, as well as scientific resource providers to experiment with this new mode of resource provisioning. The first version of the workspace service was released in September 2005 after roughly two years of R&D. As its functionality grew, we decided to make the service available as a set of components (since version 1.3). In recognition of the fact that it was no longer just one service, we changed the name to “nimbus toolkit” (since version 2.0).

Today, the nimbus toolkit consists of the following components (Figure 1 shows their dependency graph):

- *Workspace service*, which allows a remote client to deploy and manage flexibly defined groups of VMs. The service is composed of a WS front-end to a VM-based resource manager deployed on a site. The workspace service supports two front-ends: one based on the Web Service Resource Framework (WSRF) [3], and one based on Amazon’s EC2 WSDL.
- *Workspace resource manager*, which implements deployment of VM leases with “immediate” semantics on a site infrastructure.
- *Workspace pilot*, which extends existing local resource managers (LRMs) such as Torque [4] or SGE [5] to deploy virtual machines which allows RPs to use virtualization without significantly altering the site configuration.

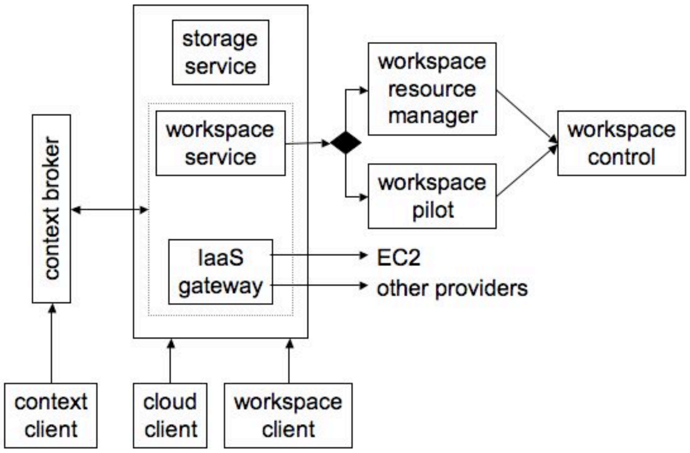

Figure 1: *Dependency graph of the nimbus toolkit*

- The *workspace control* tools, which are used to start, stop, and pause VMs; implement VM image reconstruction and management; connect the VMs to the network; and deliver contextualization information (currently work with Xen [6] and KVM [7]).
- *IaaS gateway*, which allows a client presenting a PKI credential to use another IaaS infrastructure (with different credentials). The IaaS gateway is currently used to map between PKI X509 credentials of individual users and EC2 accounts for specific projects and to enable scientific projects to run on Amazon EC2.
- *Context broker*, which allows a client to deploy a “one-click” functioning virtual cluster as opposed to a set of “unconnected” virtual machines as well as “personalize” VMs (i.e., seed them with secrets used to root a security context with the client).
- *Workspace client*, which provides full access to workspace service functionality (in particular, a rich set of networking options) but is relatively complex to use and thus typically wrapped by community-specific scripts.
- *Cloud client*, which provides access to only a select set of functions but is very easy to use and popular as an end-user tool.

- *Nimbus storage service*, which provides secure management of cloud disk space giving each user a “repository” view of VM images they own and images they can launch. It works in conjunction with globus GridFTP [8], which allows us to support any network file system, SAN, and so forth that GridFTP can interface to (this includes the possibility of drawing from parallel file sources).

The tools described above are small, lightweight, and self-contained so that they can be flexibly selected and composed in a variety of ways. For example, the original “nimbus configuration” was composed of the workspace service, workspace resource manager, workspace control, nimbus storage service, and cloud client. Adding the context broker on top of this configuration allowed users to deploy one-click virtual clusters. Replacing the workspace resource manager in the “nimbus configuration” with the workspace pilot results in a cloud with different leasing semantics (e.g. as installed at the University of Victoria). Replacing the cloud client in the “nimbus configuration” with the workspace client, allows the STAR community to create private networks and nodes with multiple NICs for their virtual clusters. By using the context broker with the IaaS gateway, we can deploy one-click virtual clusters on Amazon EC2.

3 Science Clouds: Experiences to Date

The first cloud, at the University of Chicago, became available on March 3, 2008, and was named “nimbus” [9] (the name was eventually adopted by our software project). It was deployed on a partition of 16 nodes of the TeraPort cluster [10] (each node composed of two 2.2 GHz AMD64 processors, 4 GB RAM, and 80 GB local disk). The Chicago cloud allocated 16 public IPs to provide for the VM leases and originally provided 100 GB of storage space (we recently purchased an additional disk to accommodate the raising traffic). The University of Florida cloud [11], made available on May 13, 2008, offers 16 Intel Xeon/Prestonia 2.40 GHz processors with up to 2.5 GB of memory. The UFL cloud configuration contains an innovation: instead of providing public IP addresses for leased VMs, it requires a client to get on a private network and provides private IP addresses to deployed VMs. Both Science Clouds were configured with the nimbus toolkit [12] to enable remote leasing of resources via VMs, and both were configured to support lease semantics that corresponded to EC2 “immediate leases”: a request either results in immediate VM deployment or is rejected.

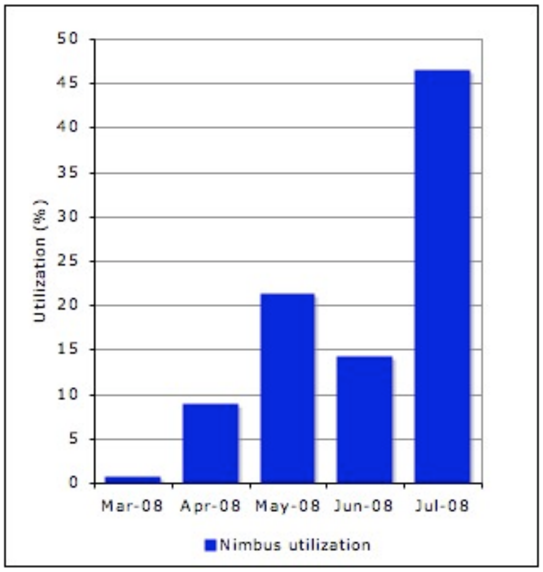

Figure 2: *Utilization of the Chicago cloud per month: the graph shows utilization from 3rd to 3rd of every month scaled to the number of days with the month.*

The Science Clouds have been in operation for five months. In the following, we present information about how the clouds were used, what applications they attracted, and the usage patterns we observed. The data discussed is based on utilization numbers from the University of Chicago cloud observed from March 3 to August 4, 2008.

Both the number of users and the time they have been spending on the cloud have risen significantly over the past five months. As of the time of this writing, we have 60+ users authorized to use the cloud, and new requests from scientific application projects worldwide come every week. The overall utilization of the clouds was around 20% (see Figure 2), with the peak per-week utilization of 86% reached in the second half of July (week of 07/14). This is remarkable utilization considering that immediate leases on a small resource do not lend themselves to a very efficient use of resources, an issue that we are working to resolve in our research [13]. One interesting measure of utilization is the number of lease requests that were rejected with the "cluster full" message: virtually no requests were rejected before 07/14. In the period after 07/14, a total of 65 requests were rejected. It is no coincidence that utilization increased significantly in mid-July. On 07/09 the nimbus team released and integrated into the cloud deployment the context broker that allows users to create one-click virtual clusters. This enabled new applications (such as the Alice high-energy physics experiment [14] and Montage workflow testing [15]) to run and old ones to run in new configurations.

The cloud proved popular among projects as diverse as high-energy physics, computer science, bioinformatics, and more recently economics. It is also being explored for use in educational projects [16]. This diversity is remarkable considering it still is relatively difficult for scientific applications to use cloud computing. Moreover, we are seeing many diverse applications co-existing, for example, interactive sessions with scientific runs. To date, two papers have been written about work using the cloud [15, 17].

Figure 3 shows a per-project breakdown of the overall cloud utilization by various projects in the defined period: the most time has been used by the projects that have been using the cloud the longest: computer science project studying the behavior of hadoop [18] over distributed resources and the STAR high-energy physics runs [19, 20]. Many new projects came onboard in July, but not all of them have resulted yet in significant usage.

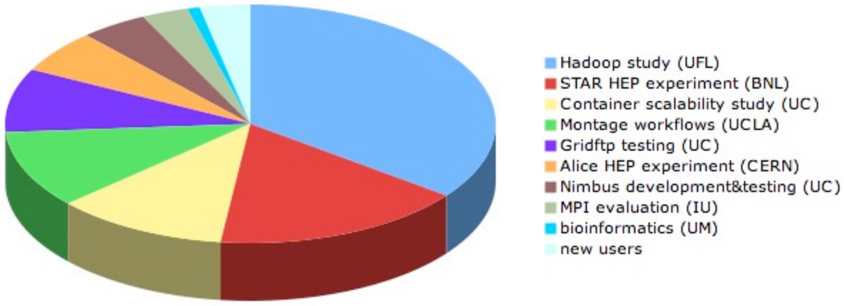

Figure 3: *Per-project utilization of the nimbus science cloud at UC (only projects that spent more than 5000 minutes are called out)*

One significant obstacle that prevents projects from considering cloud computing is scarcity of resources: while 16 nodes is sufficient to build proof-of-concept solutions, it is not enough for a typical scientific production run where hundreds of nodes are required. We circumvented this barrier by developing the IaaS gateway that (since June 2007) allows us to run scientific codes on EC2. The gateway enabled the first production run of applications on a virtual STAR cluster on 100 nodes of EC2 in September 2007 [21] (STAR was an alpha tester of the context broker technology for nearly a year now). This led to the development of a pattern where we use the UC cloud for small runs and move sponsored large runs to EC2. Based on our experiences with scientific users, we believe this pattern of resource usage where scientific communities needing more resources for specific runs can be seamlessly migrated to commercial infrastructure holds significant promise for future use of resources in the scientific domain.

Nimbus also has proved popular among resource providers. Several sites either have already installed nimbus (UFL, Clemson University, University of Victoria (Canada), Vrije University (Amsterdam), ForschungsZentrum Karlsruhe and Masaryk University (Brno)) or have expressed the intention of doing so. Many of those installations were inspired by the nimbus cloud. In fact, the GridFTP and container scalability tests at UC proved so popular that two new nimbus private clouds were configured on newly purchased infrastructure at Argonne National Laboratory to support this mode of usage for internal projects.

References

1. *Amazon Elastic Compute Cloud (Amazon EC2)*: http://www.amazon.com/ec2.
2. *Workspace Extensibility Plugins*: http://workspace.globus.org/vm/TP1.3.3/plugins/index.html.
3. Czajkowski, K., D. Ferguson, I. Foster, J. Frey, S. Graham, I. Sedukhin, D. Snelling, S. Tuecke, and W. Vambenepe, *The WS-Resource Framework*. 2004: www.globus.org/wsrf.
4. *Torque*: http://www.clusterresources.com/pages/products/torque-resource-manager.php.
5. *Sun Microsystems Grid Engine*.
6. Barham, P., B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebar, I. Pratt, and A. Warfield. *Xen and the Art of Virtualization*. in *ACM Symposium on Operating Systems Principles (SOSP)*.
7. *KVM: Kernel-based Virtual Machine.*
8. Allcock, W., *GridFTP: Protocol Extensions to FTP for the Grid*. 2003, Global Grid Forum.
9. *The Nimbus Cloud*: http://workspace.globus.org/clouds/nimbus.html.
10. *The TeraPort Cluster*: http://www.ci.uchicago.edu/research/detail_teraport.php.
11. *The Florida Cloud*: http://www.acis.ufl.edu/vws/.
12. *The Nimbus Toolkit*: http://workspace.globus.org/.
13. Sotomayor, B., K. Keahey, and I. Foster. *Combining Batch Execution and Leasing Using Virtual Machines*. in *HPDC 2008*. 2008. Boston, MA.
14. *ALICE: A Large Ion Collider Experiment*: http://aliceinfo.cern.ch/Public/Welcome.html.
15. Hoffa, C., T. Freeman, G. Metha, E. Deelman, and K. Keahey, *Exploration of the Applicability of Cloud Computing to Large-Scale Scientific Workflows.* to be submitted to SWBES08: Challenging Issues in Workflow Applications, 2008.
16. Medaris, K., *New specialization will focus on supercomputing*: http://www.purdue.edu/uns/x/2007b/071217HackerHPC.html.
17. Matsunaga, A., M. Tsugawa, and J. Fortes, *CloudBLAST: Combining MapReduce and Virtualization on Distributed Resources for Bioinformatics Applications.* submitted to eScience 2008, 2008.
18. *Hadoop*: http://hadoop.apache.org/.
19. *The STAR Experiment*. 2007: www.star.bnl.gov.
20. Keahey, K., T. Freeman, J. Lauret, and D. Olson. *Virtual Workspaces for Scientific Applications*. in *SciDAC Conference*. 2007. Boston, MA.
21. *The Nimbus RSS News Feed*: http://workspace.globus.org/news.html.