doi:10.2498/cit.1001391

Cloud Computing – Issues, Research and Implementations

Mladen A. Vouk

Department of Computer Science, North Carolina State University, Raleigh, North Carolina, USA

"Cloud" computing – a relatively recent term, builds on decades of research in virtualization, distributed computing, utility computing, and more recently networking, web and software services. It implies a service oriented architecture, reduced information technology overhead for the end-user, great flexibility, reduced total cost of ownership, on-demand services and many other things. This paper discusses the concept of "cloud" computing, some of the issues it tries to address, related research topics, and a "cloud" implementation available today.

Keywords: "cloud" computing, virtual computing lab, virtualization, utility computing, end-to-end quality of service

1. Introduction

"Cloud computing" is the next natural step in the evolution of on-demand information technology services and products. To a large extent, cloud computing will be based on virtualized resources.

Cloud computing predecessors have been around for some time now [1, 12, 15, 17, 18, 24, 29, 30, 35, 40], but the term became "popular" sometime in October 2007 when IBM and Google announced a collaboration in that domain [27, 22]. This was followed by IBM's announcement of the "Blue Cloud" effort [23]. Since then, everyone is talking about "Cloud Computing". Of course, there also is the inevitable Wikipedia entry [45].

This paper discusses the concept of "cloud" computing, some of the issues it tries to address, related research topics, and a "cloud" implementation available today. Section 2 discusses concepts and components of "cloud" computing. Section 3 describes an implementation based on Virtual Computing Laboratory (VCL) technology. VCL has been in production use at NC State University since 2004, and is a suitable vehicle for dynamic implementation of almost any current "cloud" computing solution. Section 4 discusses "cloud"-related research and engineering challenges. Section 5 summarizes and concludes the paper.

2. Cloud Computing

A key differentiating element of a successful information technology (IT) is its ability to become a true, valuable, and economical contributor to cyberinfrastructure [4]. "Cloud" computing embraces cyberinfrastructure, and builds upon decades of research in virtualization, distributed computing, "grid computing", utility computing, and, more recently, networking, web and software services. It implies a service-oriented architecture, reduced information technology overhead for the end-user, greater flexibility, reduced total cost of ownership, on-demand services and many other things.

2.1. Cyberinfrastructure

"Cyberinfrastructure makes applications dramatically easier to develop and deploy, thus expanding the feasible scope of applications possible within budget and organizational constraints, and shifting the scientist's and engineer's effort away from information technology development and concentrating it on scientific and engineering research. Cyberinfrastructure

also increases efficiency, quality, and reliability by capturing commonalities among application needs, and facilitates the efficient sharing of equipment and services."[5]

Today, almost any business or major activity uses, or relies in some form, on IT and IT services. These services need to be enabling and appliance-like, and there must be an economy-of-scale for the total-cost-of-ownership to be better than it would be without cyberinfrastructure. Technology needs to improve end-user productivity and reduce technology-driven overhead. For example, unless IT is the primary business of an organization, less than 20% of its efforts not directly connected to its primary business should have to do with IT overhead, even though 80% of its business might be conducted using electronic means.

2.2. Concepts

A powerful underlying and enabling concept is computing through service-oriented architectures (SOA) – delivery of an integrated and orchestrated suite of functions to an end-user through composition of both loosely and tightly coupled functions, or services – often network-based. Related concepts are component-based system engineering, orchestration of different services through workflows, and virtualization.

2.2.1. Service-oriented Architecture

SOA is not a new concept, although it again has been receiving considerable attention in recent years [9, 25, 38]. Examples of some of the first network-based service-oriented architectures are remote procedure calls (RPC), DCOM and Object Request Brokers (ORBs) based on the CORBA specifications [32, 33]. A more recent example are the so called "Grid Computing" architectures and solutions [15, 17, 18].

In an SOA environment, end-users request an IT service (or an integrated collection of such services) at the desired functional, quality and capacity level, and receive it either at the time requested or at a specified later time. Service discovery, brokering, and reliability are important, and services are usually designed to interoperate, as are the composites made of these services. It is expected that in the next 10 years, service-based solutions will be a major vehicle for delivery of information and other IT-assisted functions at both individual and organizational levels, e.g., software applications, web-based services, personal and business "desktop" computing, high-performance computing.

2.2.2. Components

The key to a SOA framework that supports workflows is componentization of its services, an ability to support a range of couplings among workflow building blocks, fault-tolerance in its data- and process-aware service-based delivery, and an ability to audit processes, data and results, i.e., collect and use provenance information.

Component-based approach is characterized by [13, 28] ***reusability*** (elements can be re-used in other workflows), ***substitutability*** (alternative implementations are easy to insert, very precisely specified interfaces are available, runtime component replacement mechanisms exist, there is ability to verify and validate substitutions, etc.), ***extensibility and scalability*** (ability to readily extend system component pool and to scale it, increase capabilities of individual components, have an extensible and scalable architecture that can automatically discover new functionalities and resources, etc.), ***customizability*** (ability to customize generic features to the needs of a particular scientific domain and problem), and ***composability*** (easy construction of more complex functional solutions using basic components, reasoning about such compositions, etc.). There are other characteristics that also are very important. Those include **reliability and availability** of the components and services, the cost of the services, **security**, total cost of ownership, economy of scale, and so on.

In the context of cloud computing we distinguish many categories of components: from differentiated and undifferentiated hardware, to general purpose and specialized software and applications, to real and virtual "images", to environments, to no-root differentiated resources, to workflow-based environments and collections of services, and so on. They are discussed later in the paper.

2.2.3. Workflows

An integrated view of service-based activities is provided by the concept of a workflow. An IT-assisted workflow represents a series of structured activities and computations that arise in information-assisted problem solving. Workflows have been drawing enormous attention in the database and information systems research and development communities [16, 20]. Similarly, the scientific community has developed a number of problem solving environments, most of them as integrated solutions [19]. Scientific workflows merge advances in these two areas to automate support for sophisticated scientific problem solving [28, 42].

A workflow can be represented by a directed graph of data flows that connect loosely and tightly coupled (and often asynchronous) processing components. One such graph is shown in Figure 1. It illustrates a Kepler-based implementation of a part of a fusion simulation workflow [2, 8].

In the context of "cloud computing", the key questions should be whether the underlying infrastructure is supportive of the workflow-oriented view of the world. This includes on-demand and advance-reservation-based access to individual and aggregated computational and other resources, autonomics, ability to group resources from potentially different "clouds" to deliver workflow results, appropriate level of security and privacy, etc.

2.2.4. Virtualization

Virtualization is another very useful concept. It allows abstraction and isolation of lower level functionalities and underlying hardware. This enables portability of higher level functions and sharing and/or aggregation of the physical resources.

The virtualization concept has been around in some form since 1960s (e.g., in IBM mainframe systems). Since then, the concept has matured considerably and it has been applied to all aspects of computing – memory, storage, processors, software, networks, as well as services that IT offers. It is the combination of the growing needs and the recent advances in the IT architectures and solutions that is now bringing the virtualization to the true commodity level. Virtualization, through its economy of scale, and its ability to offer very advanced and complex IT services at a reasonable cost, is poised to become, along with wireless and highly distributed and pervasive computing devices, such as sensors and personal cell-based access devices, the driving technology behind the next waive in IT growth [44].

Not surprisingly, there are dozens of virtualization products, and a number of small and large companies that make them. Some examples in the operating systems and software applications space are VMware[1], Xen – an open source

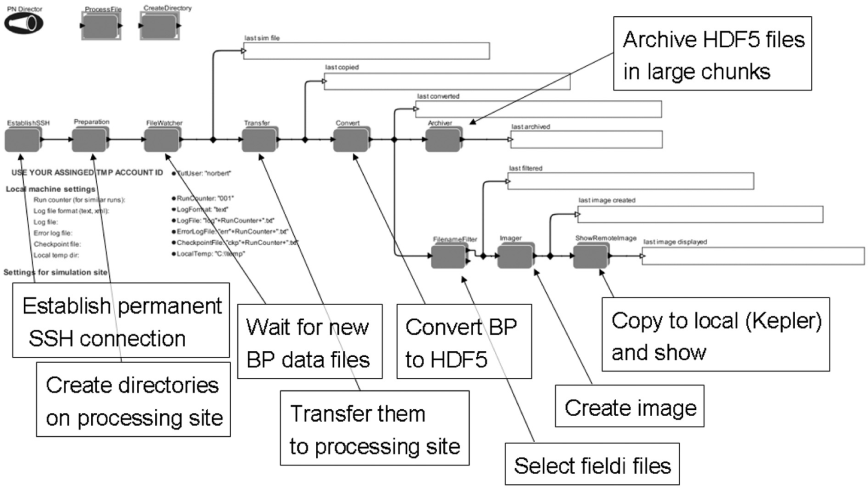

Figure 1. A Kepler-based workflow.

[1] http://www.vmware.com/

Linux-based product developed by XenSource[2], and Microsoft virtualization products[3], to mention a few. Major IT players have also shown a renewed interest in the technology (e.g., IBM, Hewllet-Packard, Intel, Sun, RedHat). Classical storage players such as EMC, NetApp, IBM and Hitachi have not been standing still either. In addition, the network virtualization market is teeming with activity.

2.3. Users

The most important Cloud entity, and the principal quality driver and constraining influence is, of course, the user. The value of a solution depends very much on the view it has of its end-user requirements and user categories.

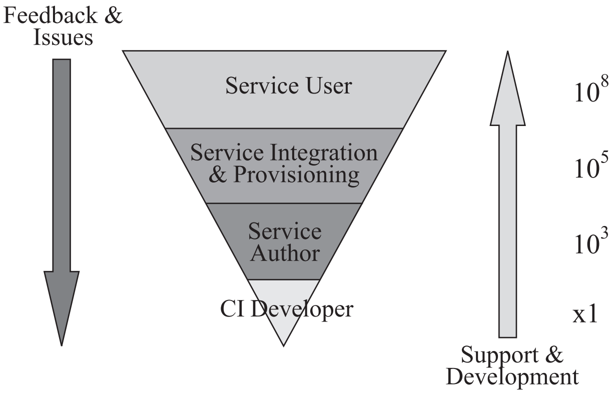

Figure 2. Cloud user hierarchy.

Figure 2 illustrates four broad sets of non-exclusive user categories: System or cyberinfrastructure (CI) developers; developers (authors) of different component services and underlying applications; technology and domain personnel who integrate basic services into composite services and their orchestrations (workflows) and delivers those to end-users; and, finally, users of simple and composite services. User categories also include domain specific groups, and indirect users such as stakeholders, policy makers, and so on. Functional and usability requirements derive, in most part, directly from the user profiles. An example, and a discussion, of user categories appropriate in the educational domain can be found in [43]. Specifically, a successful "cloud" in that domain – the K-20 and continuing education – would be expected to:

a. Support large numbers of users that range from very naive to very sophisticated (millions of student contact hours per year).

b. Support construction and delivery of content and curricula for these users. For that, the system needs to provide support and tools for thousands of instructors, teachers, professors, and others that serve the students.

c. Generate adequate content diversity, quality, and range. This may require many hundreds of authors.

d. Be reliable and cost-effective to operate and maintain. The effort to maintain the system should be relatively small, although introduction of new paradigms and solutions may require a considerable start-up development effort.

2.3.1. Developers

Cyberinfrastructure developers who are responsible for development and maintenance of the Cloud framework. They develop and integrate system hardware, storage, networks, interfaces, administration and management software, communications and scheduling algorithms, services authoring tools, workflow generation and resource access algorithms and software, and so on. They must be experts in specialized areas such as networks, computational hardware, storage, low level middleware, operating systems imaging, and similar. In addition to innovation and development of new "cloud" functionalities, they also are responsible for keeping the complexity of the framework away from the higher level users through judicious abstraction, layering and middleware. One of the lessons learned from, for example, "grid" computing efforts is that the complexity of the underlying infrastructure and middleware can be daunting, and, if exposed, can impact wider adoption of a solution [46].

[2] http://www.xensource.com/
[3] http://www.microsoft.com/virtualization/default.mspx

2.3.2. Authors

Service authors are developers of individual base-line "images" and services that may be used directly, or may be integrated into more complex service aggregates and workflows by service provisioning and integration experts. In the context of the VCL technology, an **"image"** is a tangible abstraction of the software stack [6, 44]. It incorporates

a. any base-line operating system, and if virtualization is needed for scalability, a hypervisor layer,

b. any desired middleware or application that runs on that operating system, and

c. any end-user access solution that is appropriate (e.g., ssh, web, RDP, VNC, etc.).

Images can be loaded on "bare-metal", or into an operating system/application virtual environment of choice. When a user has the right to create an image, that user usually starts with a "NoApp" or a base-line image (e.g., Win XP or Linux) without any except most basic applications that come with the operating system, and extends it with his/her applications. Similarly, when an author constructs composite images (aggregates of two or more images we call environments that are loaded synchronously), the user extends service capabilities of VCL. An author can program an image for sole use on one or more hardware units, if that is desired, or for sharing of the resources with other users. Scalability is achieved through a combination of multi-user service hosting, application virtualization, and both time and CPU multiplexing and load balancing. Authors must be component (base-line image and applications) experts and must have good understanding of the needs of the user categories above them in the Figure 2 triangle. Some of the functionalities a cloud framework must provide for them are image creation tools, image and service management tools, service brokers, service registration and discovery tools, security tools, provenance collection tools, cloud component aggregations tools, resource mapping tools, license management tools, fault-tolerance and fail-over mechanisms, and so on [44].

It is important to note that the authors, for the most part, will not be cloud framework experts, and thus the authoring tools and interfaces must be appliances: easy-to-learn and easy-to-use and they must allow the authors to concentrate on the "image" and service development rather than struggle with the cloud infrastructure intricacies.

2.3.3. Service Composition

Similarly, services **integration and provisioning** experts should be able to focus on creation of composite and orchestrated solutions needed for an end-user. They sample and combine existing services and images, customize them, update existing services and images, and develop new composites. They may also be the front for

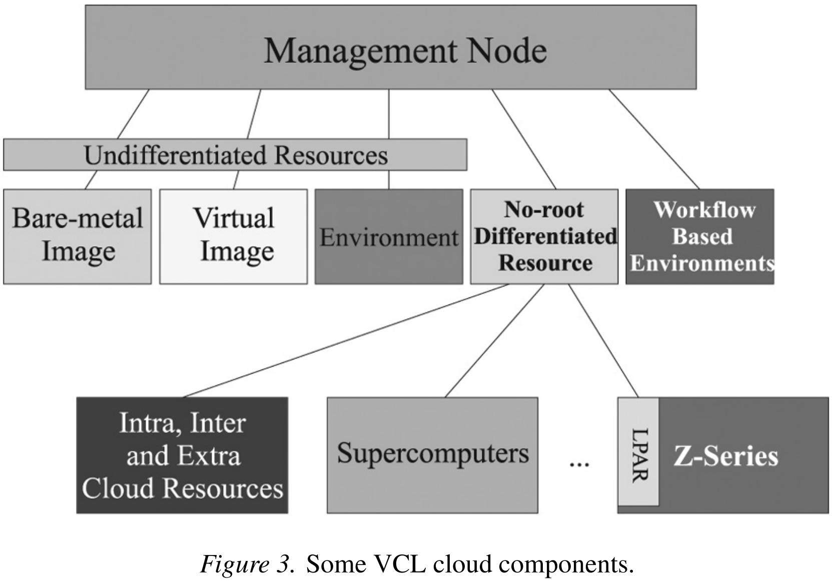
Management Node
Undifferentiated Resources
Bare-metal Image
Virtual Image
Environment
No-root Differentiated Resource
Workflow Based Environments
Intra, Inter and Extra Cloud Resources
Supercomputers
...
LPAR
Z-Series

Figure 3. Some VCL cloud components.

delivery of these new services (e.g., an instructor in an educational institution, with "images" being cloud-based in-lab virtual desktops), they may oversee the usage of the services, and may collect and manage service usage information, statistics, etc. This may require some expertise in the construction of images and services, but, for the most part, their work will focus on interfacing with end-users and on provisioning of what end-users need in their workflows.

Their expertise may range from workflow automation through a variety of tools and languages, to domain expertise needed to understand what aggregates of services, if any, the end-user needs, to management of end-user accounting needs, and to worrying about inter-, intra- and extra-cloud service orchestration and engagement, to provenance data analysis.

Some of the components that an integration and provisioning expert may need are illustrated in Figure 3, based on the VCL implementation [6, 44]. The need may range from "bare metal" loaded images, images on virtual platforms (on hypervisors), to collections of image aggregates (environments, including high-performance computing clusters), images with some restrictions, and workflow-based services. A service management node may use resources that can be reloaded at will to differentiate them with images of choice. After they have been used, these resources are returned to an undifferentiated state for re-use. In an educational context, this could be, for example, a VMWare image of 10 lab-class desktops that may be needed between 2 and 3 pm on Monday. Then after 3pm another set of images can be loaded into those resources.

On the other hand, an "Environment" could be a collection of images loaded on one or more platforms. For example, a web server, a database server, and a visualization application server, or a high-performance cluster. Workflow image is typically a process control image that also has a temporal component. It can launch any number of the previous resources as needed and then manage their use and release based on an automated workflow.

Users of images that load onto undifferentiated resources can be given root or administrative access rights since those resources are "wiped clean" after their use. On the other hand, resources that provide access to only some of its virtual partitions, may allow non-root cloud users only: for example, a z-Series mainframe may offer one of its LPARS as a resource. Similarly an ESX-loaded platform may be non-root access, while its guest operating system images may be of root-access type.

2.3.4. End-users

End-users of services are the most important users. They require appropriately reliable and timely service delivery, easy-to-use interfaces, collaborative support, information about their services, etc. The distribution of services, across the network and across resources, will depend on the task complexity, desired schedules and resource constraints. Solutions should not rule out use of any network type (wire, optical, wireless) or access mode (high speed and low speed). However, VCL has set a lower bound on the end-to-end connectivity throughput, roughly at the level of DSL and cable modem speeds. At any point in time, users' work must be secure and protected from data losses and unauthorized access.

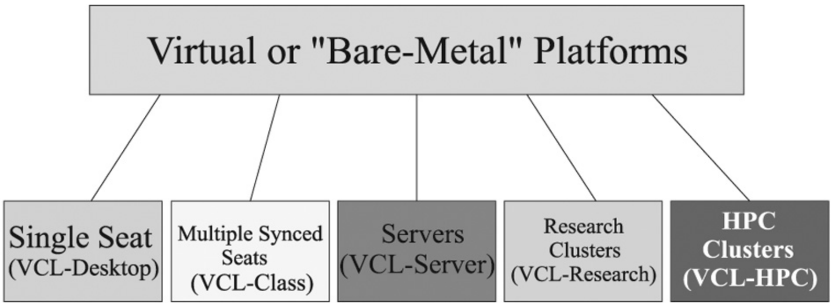
Virtual or "Bare-Metal" Platforms
Single Seat (VCL-Desktop)
Multiple Synced Seats (VCL-Class)
Servers (VCL-Server)
Research Clusters (VCL-Research)
HPC Clusters (VCL-HPC)

Figure 4. VCL "seats".

For example, the resource needs of educational end-users (Figure 4) may range from single-seat desktops ("computer images") that may deliver any operating system and application appropriate to the educational domain, to a group of lab or classroom seats for support of synchronous or asynchronous learning or hands-on sessions, one or more servers supporting different educational functions, groups of coupled servers (or environments), e.g., an Apache server, a database server, and a workflow management server all working together to support a particular class, or research clusters, and high-performance computing clusters. Figure 4 shows the current basic services (resources) delivered by VCL. The duration of resource ownership by the end-users may range from a few hours, to several weeks, a semester, or an open-ended period of time.

3. An Implementation

"Virtual Computing Laboratory (VCL) – `http://vcl.ncsu.edu` *is an award-winning open source implementation of a secure production-level on-demand utility computing and services oriented technology for wide-area access to solutions based on virtualized resources, including computational, storage and software resources. There are VCL pilots with a number of University of North Carolina campuses, North Carolina Community College System, as well as with a number of out-of-state universities – many of which are members of the IBM Virtual Computing Initiative".*

Figure 5 illustrates NC State Cloud based on VCL technology. Access to NC State Cloud reservations and management is either through a web portal, or through an API. Authentication, resource availability, image and other information are kept in a database. Resources (real and virtual) are controlled by one or more management nodes. These nodes can be within the same cloud, or among different clouds, and they allow extensive sharing of the resources provided licensing and other constraints are honored. NC State undifferentiated resources are currently about 1000 IBM BladeCenter blades. About 40% to 50% of them are serving high-performance computing needs, the rest are in the individual seat mode. Its differentiated services are teaching lab computers that are adopted into VCL when they are not in use (e.g., at night). In addition, VCL can attach other differentiated and undifferentiated resources such as Sun blades, Dell clusters, and similar. More detailed information about VCL user services, functions, security and concepts can be found in [6, 44].

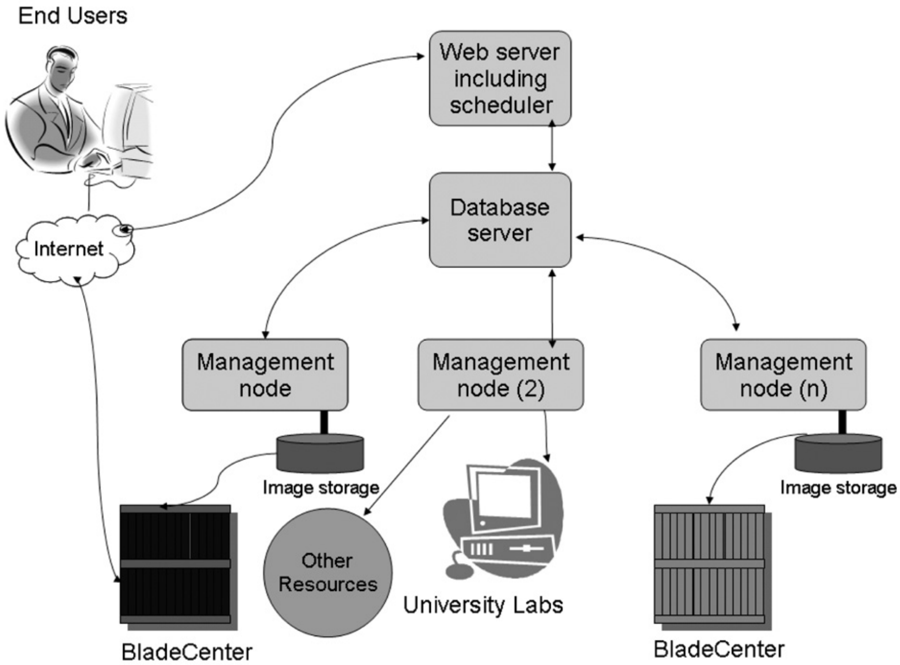

Figure 5. NC State "Cloud".

Currently, NC State VCL is serving a student and faculty population of more than 30,000. Delivery focus is augmentation of the student-owned computing with applications and platforms that students may otherwise have difficulty installing on their own machines because of licensing, application footprint, or similar. We serve about 60,000 to 100,000 "seat" reservation requests (mostly of the on-demand or "now" type) per semester. Typical single-seat user reservation is 1-2 hours.

We currently have about 150 production images and another 450 or so other images. Most of the images serve single user seats and HPC cycles, with a smaller number focused on environment- and workflow-based services.

The VCL implementation has most of the characteristics and functionalities discussed so far and considered desirable in a cloud. It can also morph into many things. Functionally it has a large intersection with Amazon Elastic Cloud [1], and by loading a number of blades with Hadoop-based images [18] one can implement a Google-like map/reduce environment, or by loading an environment or group composed of Globus-based images one can construct a subcloud for grid-based computing, and so on.

A typical NC State bare-metal blade serves about 25 student seats – a 25:1 ratio – considerably better than traditional labs at 5:1 to 10:1. Hypervisors and server-apps can increase utilization by another factor of 2 to 40, depending on the application and user profile. Our maintenance overhead is quite low – about 1 FTE in maintenance for about 1000 nodes, and with another 3 FTEs in development.

4. Research Issues

The general cloud computing approach discussed so far, as well as the specific VCL implementation of a cloud continues a number of research directions, and opens some new ones.

For example, economy-of-scale and economics of **image and service construction** depends to a large extent on the ease of construction and mobility of these images, not only within a cloud, but also among different clouds. Of special interest is construction of complex environments of resources and complex control images for those resources, including workflow-oriented images. Temporal and spatial feedback large-scale workflows may present is a valid research issue. Underlying that is a considerable amount of **meta-data**, some permanently attached to an image, some dynamically attached to an image, some kept in the cloud management databases.

Cloud **provenance data**, and in general meta-data management, is an open issue. The classification we use divides provenance information into

- Cloud Process provenance – dynamics of control flows and their progression, execution information, code performance tracking, etc.
- Cloud Data provenance – dynamics of data and data flows, file locations, application input/output information, etc.
- Cloud Workflow provenance – structure, form, evolution, . . ., of the workflow itself
- System (or Environment) provenance – system information, O/S, compiler versions, loaded libraries, environment variables, etc.

Open challenges include: How to collect provenance information in a standardized and seamless way and with minimal overhead – modularized design and integrated provenance recording; How to store this information in a permanent way so that one can come back to it at anytime, – standardized schema; and How to present this information to the user in a logical manner – an intuitive user web interface: Dashboard [7].

Some other image- and service-related practical issues involve finding optimal image and service composites and **optimization** of image and environment loading times.

There is also an issue of the image **portability** and by implication of the image **format**. Given the proliferation of different virtualization environments and the variety in the hardware, standardization of image formats is of considerable interest. Some open solutions exist or are under consideration, and a number of more proprietary solutions are available already [26, 41]. For example, VCL currently uses standard image snapshots [34] that may be an operating system, hypervisor and platform specific, and thus exchange of images requires relatively complex mapping and additional storage.

Another research and engineering challenge is **security**. For end-users to feel comfortable with a "cloud" solution that holds their software, data and processes, there should exist considerable assurances that services are highly reliable and available, as well as secure and safe, and that privacy is protected. This raises the issues of end-to-end service isolation through VPN and SSH tunnels and VLANs, and the guarantees one may have that the data and the images keep their integrity in the "cloud". Some of the work being done by the NC State Secure Open Systems Initiative [37] involves watermarking of the images and data to ensure verifiable integrity. While NC State experience with VCL is excellent and our security solution has been holding up beautifully over the last four years, security tends to be a moving target and a lot of challenges remain.

The question of the return-on-investment (ROI) and the total-cost-of-ownership (TCO) is complicated. Direct comparisons with existing solutions are lacking at this point. However, the cost of service construction, maintenance and commonality definitely plays a role. Our experience with VCL is that there are good returns from increased utilization of the resources (as noted earlier). In our case, we can, and do, move our cloud resources from single seat environment to HPC environments, and vice versa, as the interest in one wanes and in the other one increases on holiday and semester boundaries.

Figure 6 shows **utilization** of the VCL seat-oriented resources by day over the last 4 years. We see the growth in usage, but we also see seasonal and semestral variations in utilization that invite re-targeting of the resources. Not shown is the information about the actual number of resources available to VCL in each time period. Currently, the average number of blades participating on the single-seat side is over 200, however, initially it was in the 40-ies. The overall number of reservation transactions covered by the graph exceeds 200,000.

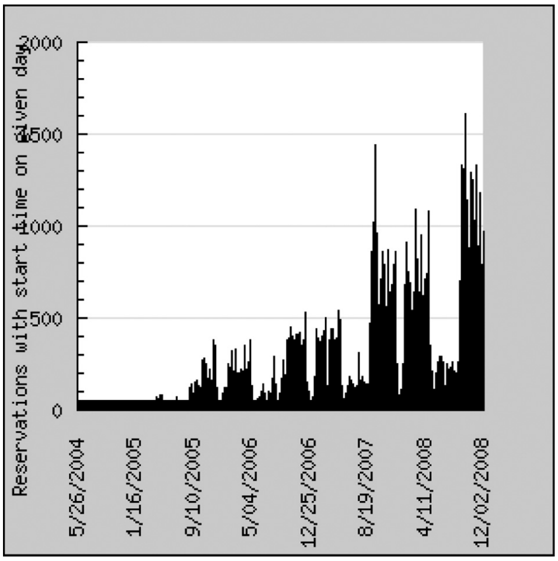

Figure 6. VCL resource utilization.

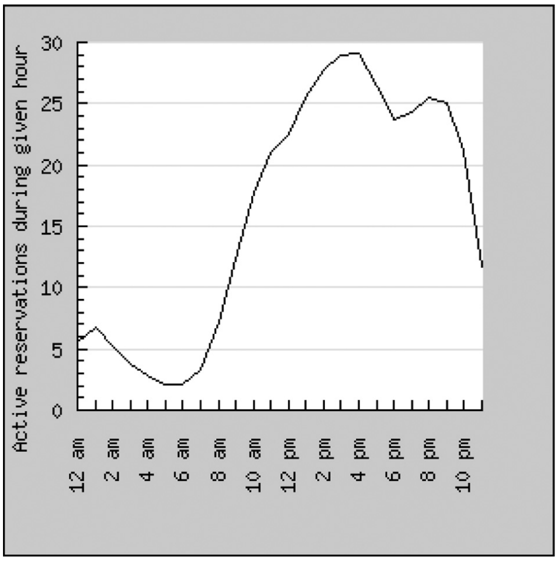

Figure 7. Daily variation in VCL single-seat utilization (averaged over four years).

A much more agile re-distribution of the resources (perhaps nightly) is possible since we have all the necessary meta-data, but we are not exercising that option right now. This is illustrated in Figure 7. It is interesting to see that there may be an opportunity to perhaps shift some of the resources to HPC, which is always short of CPU time, in the 12am to 7am time slot. It is not clear that this may be a cost-saving measure. Another option is to actually react to the rising issues with data-center energy costs and turn off some of the equipment during the low-usage hours. There are issues there too – how often would one do that, would that shorten lifetime of the equipment, and so on. Again, a possible applied research project.

5. Conclusions

“Cloud” computing builds on decades of research in virtualization, distributed computing, utility computing, and, more recently, networking, web and software services. It implies a service-oriented architecture, reduced information technology overhead for the end-user, great flexibility, reduced total cost of ownership, on-demand services and many other things. This paper discusses the concept of “cloud” computing, the issues it tries to address, related research topics, and a “cloud” implementation based on VCL technology. Our experience with VCL technology is excellent and we are working on additional functionalities and features that will make it even more suitable for cloud framework construction.

6. Acknowledgements

I would like to thank my colleagues on the NC State VCL team for their contributions, insights, and support.

This work is supported in part by the DOE SciDAC grants DE-FC02-01ER25484 and DE-FC02-07ER25809, IBM Corp., SAS Institute, NC State University, and State of North Carolina.

References

[1] Amazon Elastic Compute Cloud (EC2): `http://www.amazon.com/gp/browse.html?node=201590011`, accessed Dec 2008.

[2] Ilkay Altintas, Bertram Ludaescher, Scott Klasky, Mladen A. Vouk. “Introduction to scientific workflow management and the Kepler system”, Tutorial, in Proceedings of the 2006 ACM/IEEE Conference on Supercomputing, Tampa, Florida, TUTORIAL SESSION, Article No. 205, 2006, ISBN:0-7695-2700-0, also given at Supercomputing 2007 by Altintas, Vouk, Klasky, Podhorszki, and Crawl, tutorial session S07, 11 Nov 07.

[3] Ilkay Altintas, George Chin, Daniel Crawl, Terence Critchlow, David Koop, Jeff Ligon, Bertram Ludaescher, Pierre Mouallem1, Meiyappan Nagappan, Norbert Podhorszki, Claudio Silva, Mladen Vouk, “Provenance in Kepler-based Scientific Workflow Systems”, Poster # 41, at Microsoft eScience Workshop Friday Center, University of North Carolina, Chapell Hill, NC, October 13 – 15, 2007, pp. 82.

[4] D.E. Atkins et al., “Revolutionizing Science and Engineering Through Cyberinfrastructure: Report of the National Science Foundation Blue-ribbon Advisory Panel on Cyberinfrastructure”, NSF, Report of the National Science Foundation Blue-ribbon Advisory Panel on Cyberinfrastructure, January 2003, `http://www.nsf.gov/od/oci/reports/atkins.pdf`

[5] Ditto, Appendix A (`http://www.nsf.gov/od/oci/reports/APXA.pdf`).

[6] Sam Averitt, Michale Bugaev, Aaron Peeler, Henry Schaffer, Eric Sills, Sarah Stein, Josh Thompson, Mladen Vouk, “The Virtual Computing Laboratory”, *Proceedings of the International Conference on Virtual Computing Initiative*, May 7-8, 2007, IBM Corp., Research Triangle Park, NC, pp. 1–16.

[7] Roselyne Barreto, Terence Critchlow, Ayla Khan, Scott Klasky, Leena Kora, Jeffrey Ligon, Pierre Mouallem, Meiyappan Nagappan, Norbert Podhorszki, Mladen Vouk, “Managing and Monitoring Scientific Workflows through Dashboards”, Poster # 93, at Microsoft eScience Workshop Friday Center, University of North Carolina, Chapell Hill, NC, October 13 – 15, 2007, pp. 108.

[8] D.A. Batchelor, M. Beck, A. Becoulet, R.V. Budny, C. S. Chang, P. H. Diamond, J. Q. Dong, G. Y. Fu, A. Fukuyama, T. S. Hahm, D. E. Keyes, Y. Kishimoto, S. Klasky, L. L. Lao1, K. Li1, Z. Lin⁄1, B. Ludaescher, J. Manickam, N. Nakajima1, T. Ozeki1, N. Podhorszki, W. M. Tang, M. A. Vouk, R. E. Waltz, S. J. Wang, H. R. Wilson, X. Q. Xu, M. Yagi, F. Zonca, “Simulation of Fusion Plasmas: Current Status and Future Direction”, Plasma Science and Technology, Vol. 9, No. 3, Jun. 2007, pp. 312–387, doi:10.1088/1009-0630/9/3/13.

[9] Michael Bell, “Introduction to Service-oriented Modeling”, Service-oriented Modeling: Service Analysis, Design, and Architecture. Wiley & Sons, 3. ISBN 978-0-470-14111-3, 2008.

[10] W. M. Bulkeley, “IBM, Google, Universities Combine ‘Cloud’ Foces”, *Wall Street Journal*, October 8, 2007, `http://online.wsj.com/public/article_print/SB119180611310551864.html`

[11] `http://www.cca-forum.org/`, accessed February 2006.

[12] Condor: `http://www.cs.wisc.edu/condor/`, accessed May 2008.

[13] I. Crnkovic and M. Larsson (editors), Building Reliable Component-based Software Systems, Artech House Publishers, ISBN 1-58053-327-2, 2002, `http://www.idt.mdh.se/cbse-book/`

[14] R. L. Dennis , D. W. Byun, J. H. Novak, K. J. Galluppi, C. C. Coats, M. A. Vouk, “The Next Generation of Integrated Air Quality Modeling: EPA’s Models-3”, Atmospheric Environment, Vol. 30 (12), pp. 1925–1938, 1996.

[15] The Grid: Blueprint for a New Computing Infrastructure, 2nd Edition, Morgan Kaufmann, 2004. ISBN: 1-55860-933-4.

[16] D. Georgakopoulos, M. Hornick, and A. Sheth, "An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure", Distributed and Parallel Databases, Vol. 3(2), April 1995.

[17] Globus: http://www.globus.org/, accessed May 2008.

[18] Hadoop: http://hadoop.apache.org/core/, accessed May 2008.

[19] Elias N. Houstis, John R. Rice, Efstratios Gallopoulos, Randall Bramley (editors), Enabling Technologies for Computational Science Frameworks, Middleware and Environments, Kluwer-Academic Publishers, Hardbound, ISBN 0-7923-7809-1, 2000.

[20] M. Hsu (ed.), "Special Issue on Workflow and Extended Transaction Systems", *IEEE Data Engineering*, Vol. 16(2), June 1993.

[21] IBM, "IBM Launches New System x Servers and Software Targeting Large Scale x86 Virtualization, http://www-03.ibm.com/press/us/en/pressrelease/19545.wss, 21 Apr 2006.

[22] IBM, "Google and IBM Announced University Initiative to Address Internet-Scale Computing Challenges", October 8, 2007, http://www-03.ibm.com/press/us/en/pressrelease/22414.wss

[23] IBM, "IBM Introduces Ready-to-Use Cloud Computing", http://www-03.ibm.com/press/us/en/pressrelease/22613.wss, November 15, 2007.

[24] IBM, "North Carolina State University and IBM help bridge digital divide in North Carolina and beyond", May 7, 2007, http://www-03.ibm.com/industries/education/doc/content/news/pressrelease/2494970110.html

[25] IBM, "Service-oriented Architecture – SOA", http://www-306.ibm.com/software/solutions/soa/, accessed May 2008.

[26] IBM, "Mirage: Virtual Machine Images as Data", http://domino.research.ibm.com/comm/research_projects.nsf/pages/mirage.index.html, 6-Feb-08.

[27] S. Lohr, "Google and I.B.M. Join in 'Cloud Computing' Research", October 8, 2007, http://www.nytimes.com/2007/10/08/technology/08cloud.html?_r=1&ei=5088&en=92a8c77c354521ba&ex=1349582400&oref=slogin&partner=rssnyt&emc=rss&pagewanted=print

[28] B. Ludscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger-Frank, M. Jones, E. Lee, J. Tao, Y. Zhao, "Scientific Workflow Management and the Kepler System", Concurrency and Computation: Practice & Experience, Special Issue on Workflow in Grid Systems, Volume 18, Issue 10 (August 2006), pp. 1039–1065, 2006.

[29] E. Naone, "Computer in the Cloud", Technology, Review, MIT, Sept 18, 2007, http://www.technologyreview.com/printer_friendly_article.aspx?id=19397

[30] The NetApp Kilo-Client: http://partners.netapp.com/go/techontap/tot-march2006/0306tot_kilo.html, March 2006.

[31] Newcomer, Eric; Lomow, Greg, "Understanding SOA with Web Services", Addison Wesley, ISBN 0-321-18086-0, 2005.Ö

[32] CORBA, http://www.omg.org/corba/

[33] http://www.service-architecture.com/web-services/articles/object_management_group_omg.html

[34] Partimage, http://www.partimage.org/Main_Page, accessed Dec 2008.

[35] J. Reimer, "Dreaming in the "Cloud" with the XIOS web operating system", April, 8, 2007, http://arstechnica.com/news.ars/post/20070408-dreaming-in-the-cloud-with-the-xios-web-operating-system.html

[36] M. P. Singh and M. A. Vouk, "Network Computing", in John G. Webster (editor), Encyclopedia of Electrical and Electronics Engineering, John Wiley & Sons, New York, Vol. 14, pp. 114–132, 1999.

[37] Secure Open Systems Initiative, http://www.sosi.ncsu.edu/, accessed Dec 2008.

[38] Thomas Erl, "Service-oriented Architecture: Concepts, Technology, and Design", Upper Saddle River: Prentice Hall PTR. ISBN 0-13-185858-0, 2005.

[39] Mark VanderWiele, "The IBM Research Cloud Computing Initiative", Keynote talk at ICVCI 2008, RTP, NC, USA, 15–16 May 2008.

[40] Virtual Computing Laboratory, VCL, http://vcl.ncsu.edu, on-line since Summer 2004.

[41] VMWare, "The Open Virtual Machine Format – Whitepaper for OVF Specification", v0.9, 2007, http://www.vmware.com/pdf/ovf_whitepaper_specification.pdf

[42] M. A. Vouk and M. P. Singh, "Quality of Service and Scientific Workflows", in The Quality of Numerical Software: Assessment and Enhancements, editor: R. Boisvert, Chapman & Hall, pp.77–89, 1997.

[43] M. Vouk, R. L. Klevans and D. L. Bitzer, "Workflow and End-user Quality of Service Issues in Web-based Education", *IEEE Trans*. On Knowledge Engineering, to Vol 11(4), July/August 1999, pp. 673–687.

[44] MLADEN VOUK, SAM AVERITT, MICHAEL BUGAEV, ANDY KURTH, AARON PEELER, ANDY RINDOS*, HENRY SHAFFER, ERIC SILLS, SARAH STEIN, JOSH THOMPSON "'Powered by VCL' – Using Virtual Computing Laboratory (VCL) Technology to Power Cloud Computing". *Proceedings of the 2nd International Conference on Virtual Computing (ICVCI)*, 15–16 May, 2008, RTP, NC, pp 1–10.

[45] WIKIPEDIA, "Cloud Computing", `http://en.wikipedia.org/wiki/Cloud_computing`, May 2008.

[46] ANN ZIMMERMAN, THOMAS FINHOLT, "Report from the TeraGrid Evaluation Study, Part 1: Project Findings", August 2008, School of Information, University of Michigan, `http://www.nsf.gov/od/oci/XD_Program/TeraGrid_evaluation_report_09_02_08.pdf`

Received: June, 2008
Accepted: September, 2008

Contact address:
Mladen A. Vouk
Department of Computer Science
North Carolina State University
Raleigh, NC 27695, USA
e-mail: vouk@ncsu.edu

MLADEN A. VOUK received Ph.D. from the King's College, University of London, U.K. He is Department Head and Professor of Computer Science, and Associate Vice Provost for Information Technology at N.C. State University, Raleigh, N.C., U.S.A. Dr. Vouk has extensive experience in both commercial software production and academic computing. He is the author/co-author of over 250 publications. His research and development interests include software engineering, scientific computing, information technology (IT) assisted education, and high-performance computing and networks. Dr. Vouk has extensive professional visibility through organization of professional meetings, membership on professional journal editorial boards, and professional consulting. Dr. Vouk is a member of the IFIP Working Group 2.5 on Numerical Software, and a recipient of the IFIP Silver Core award. He is an IEEE Fellow, and a member of several IEEE societies, ASQ, ACM, and Sigma Xi.