Chapter 2
Clouds in Higher Education

Mladen A. Vouk

Abstract. North Carolina State University (NCSU) has been operating a production-level cloud, called Virtual Computing Laboratory (VCL) since 2004. VCL is an award-winning open source cloud technology that at NC State delivers services ranging from single desktops (VDI), to servers to, high-performance computing (HPC) and data (HPD). It serves over 40,000 users and delivers about 250,000 service reservations per year along with 14+ million HPC/HPD CPU hours per year. Through the lens of VCL, this paper discusses the needs, possible needs, and future of cloud computing in higher education – the range and complexity of the services, the needed capabilities of the cloud architecture and implementation, and possible future development directions embodied in the vCentennial campus model.

2.1 Introduction

We know how to construct clouds [2, 3, 6, 8, 11, 12, 13, 14, 19]. It is less clear that we know how to use them, particularly in research-intensive higher-education environments where teaching, research and innovation need to co-exist in a proactive and collaborative way, and need to be ahead of the curve.

Cloud computing is now sufficiently well understood that real [15, 19] and defacto [14] standards are beginning to emerge. Today a general cloud architecture is service-based and has at least three layers: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). IaaS is concerned with (seamless and elastic) provisioning and configuration of the basic real or virtual resources, such as computational hardware, networking, and storage infrastructure. PaaS is concerned with a higher level set of building blocks such as specific operating system platforms (e.g., Linux or Windows), middleware provisioning, and

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

R. Klempous and J. Nikodem (eds.), *Innovative Technologies in Management and Science*,
Topics in Intelligent Engineering and Informatics 10, DOI: 10.1007/978-3-319-12652-4_2

general mid-level cloud services such as accounting, authorization and authentication, provenance data collection, etc. SaaS is usually limited to the use and development of specific applications, and higher-level services. In general this model allows for X-as-a-Service, where X can be anything that is relevant to the user or customer community, e.g., Security-as-a-Service, Cloud-as-a-Service, etc. How much access a user has to a particular cloud layer and how much control (e.g., can a user be root on the platform provisioned for the user, can user define VLANs, can user modify the application or only input data and get results from it) depends on the nature of the cloud, institutional policies, user privileges, and so on.

In [4] authors observe that *"within an educational and/or research environment, cloud computing systems must be flexible, adaptable and serve a wide spectrum of users from among students, faculty, researchers, and staff. For students, educators, and researchers to use cloud computing as an integrated tool for their work, it must be designed to deliver services that support everyone from a novice user up to the most sophisticated expert researcher. As part of the mission of a higher-education institution cloud computing needs to reliably deliver:*

- *A wide range of teaching and other on- and off-campus academic IT support services to students, teachers, and staff.*
- *Research-level computational, storage and networking services in support of the research mission of the university.*
- *Any other information technology services that the institution needs. This may include outreach related IT services, continuing education IT services, and IT services needed to administer the institution, such as ERP services."*

This is not surprising. Cloud concepts were not invented overnight. They evolved over decades from research in virtualization [1], distributed computing, networking, web, utility computing, grids, software services, autonomic computing, and so on [12]. Universities were at the center of many of these developments which were in many situations driven by the actual needs of the education and research of the higher-education institutions.

2.2 Capabilities

There is a basic set of capabilities that a good modern information technology system needs to support. Most of these capabilities are essential for adequate support of a full cloud computing environment that is capable of serving the broad range of needs a top research intensive university may have. This is illustrated in Fig. 2.1. Each bubble represents an essential capability, tool or function needed to support cloud layers from Hardware-as-a-Service all the way to Cloud-as-a-Service and Facilities-as-a-Service (or virtualized brick and mortar) [7].

Obviously computational, networking, storage and other physical resources are needed. Their utilization and management may be enhanced through virtualization and today most of the clouds do that. However, it is important to remember that

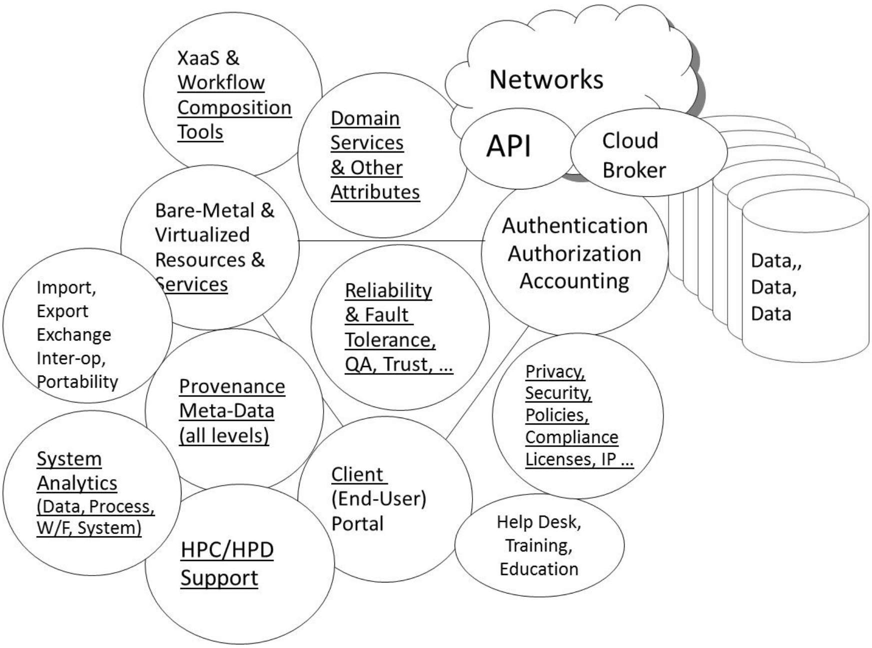

Fig. 2.1 Desired cloud subsytems, tools and capabilities

behind the scene there always are physical resources that need to be managed and that anchor any set of information technology (IT) services. At a more complex level, a cloud environment offers higher level service building blocks - compositions (or stacks) of software consisting of operating systems, middleware, and applications running on those resources. Those building blocks can be composed into more complex services and workflows. This is often a tedious, error prone and overhead inducing process. Therefore to construct cloud-based higher level services and workflows, especially in the self-service mode, it is necessary to augment cloud environment capabilities with PaaS and SaaS composition and workflow component-based construction tools [10, 9].

Naturally, a good environment needs to have an appropriate level of security, privacy, and policy compliance capabilities, as well as capabilities to honor licensing and intellectual property boundaries. It also needs to provide a human user access through some type of interface (e.g. web, client-side graphical user interface or GUI), but it also needs to be able to communicate (via application programming interfaces, API) with computer applications, with other clouds, and similar. The usual authentication, authorization, accounting (e.g. for chargeback), and access control services need to be there, as do customer-oriented help-desk, training services, and trouble-shooting services.

As the economic model of cloud-based services and resources (at all levels) evolves, it is reasonable to assume that some of those services may be more affordable if they came from a commodity provider rather than the university itself. A flexible cloud solution will have the capability to broker functionally equivalent services from external providers to mitigate situations such as peak demand load, disaster recovery, or continuity of service fail-over.

Central to trustworthy cloud-based services is their reliability and availability. This can be achieved in many ways. Our experience is that availability needs to be similar to what classical telephone services offer (five nines, 0.99999). Reliability of the cloud with respect to security failures should also be in the five nines range or better.

To achieve that pro-actively, as well as satisfactory performance, provenance support of higher-level composite functions, and similar, it is necessary to collect a considerable amount of information about the current and past cloud status, state, performance, processes, data, system, software, users, and so on. This amounts to having a robust and comprehensive provenance and meta-data collection subsystem as part of the cloud infrastructure. In order to achieve meaningful analysis of that information a well balance analytics and computational support needs to be implemented. This then, for instance, allows us to perhaps predict failure and security states of the cloud components, pro-actively manage them, and provide privacy protection, license management, and policy compliance. A particularly important function of a cloud, even a private cloud, that may have users from all over the world is the ability to import and export data, computational codes (e.g., as virtual machine appliances or OVF files), and at the same time understand local, state, country and regional policies, regulations, laws and similar in order to effect appropriate import/export controls, manage cost optimization in the case of service brokering and the like.

In research universities it is also very cost-effective to have the cloud offer both services which are traditional (such as desktops, servers, clusters, etc.), and high-performance computing (HPC) and data (HPD) services (e.g., for simulations, big data analytics and similar).

There are, of course, many other fine-grained functionalities and specialized services that particular clouds may need to have. One may be very high grade security if the cloud is being used to host high-assurance applications and data, national security information, and the like. However, without at least the bubbles shown in Fig. 2.1, it may not be wise to get into the business of constructing and hosting a cloud.

2.3 VCL

2.3.1 Overview

Virtual Computing Laboratory (VCL, http://vcl.ncsu.edu) is an open-source (Apache, http://vcl.apache.org) cloud technology that was originally developed at North

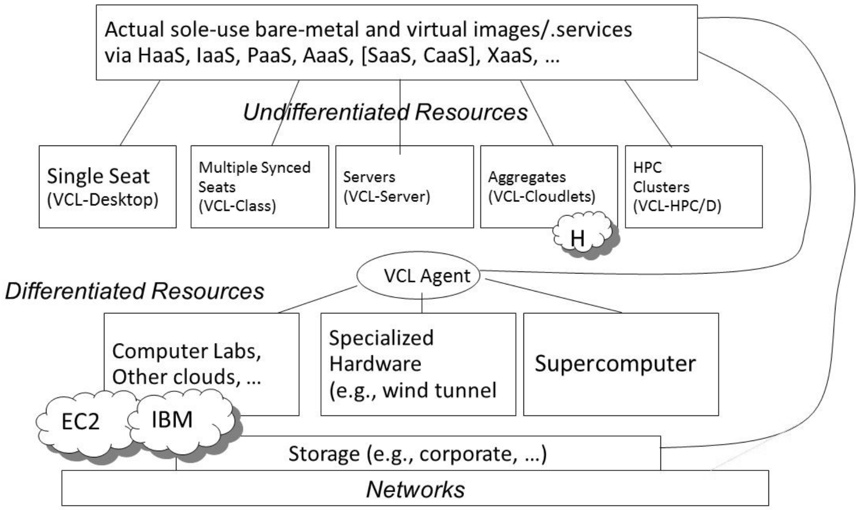

Fig. 2.2 NCSU VCL User services. Undifferentiated resources are malleable and can be used at IaaS or PaaS level, differentiated services may have a number of restrictions on how much a user can control the resource(s). They are typically SaaS, AaaS or higher, and may be attached to the cloud directly, or may be brokered by VCL from a non-VCL provider.

Carolina State University (NCSU), and has been in production use at NCSU since 2004. At NCSU VCL serves over 40,000 accounts and delivers about 250,000 service reservations per year, along with another 14+ million HPC/HPD CPU hours. The range of services offered by VCL is illustrated in Figure 2. At NCSU VCL is used to support individual students (typically augmenting their personal devices with software and services they may not have installed, e.g. specialized homework software), classes and labs, individual longer term servers and server clusters, storage, and of course HPC/HPD services. VCL capabilities cover almost all the bubbles shown in Figure 1, although for some of the services the user would need to have administrative privileges. Advanced users are allowed to create VCL images (software stacks based on NCSU licensed platforms such as Windows 7 or Linux) and deploy them to the user groups they are authorized to serve. VCL has a sophisticated role-based privilege management tree that can isolate individuals, groups and institutions, help ensure that licenses are honored, and in combination with other VCL and institutional services, can be used to implement a number of specialized policy, privacy, security, and topology related mappings.

For example, a user who wishes to construct a small virtual lab or lablet that consists of a web server, a data base server, and one or more computational nodes with specialized software (e.g. Hadoop-based), could do that in two ways. One would be to use VCL environment functionality associate with each image, and the other would be to use the VCL API. A combination of the two is also possible.

In the first case, the user would first make a reservation for the base-image (e.g. just Linux with no applications on it). This would be used to construct the anchoring or parent or control image. That image may need to have more than just an application on it (e.g. web server). It may need to have some control intelligence to know which other nodes (images running on either bare-metal or virtual machines) are in the environment. Then the user would construct a separate data base image (basically make a reservation of a base-line Linux image, and then install on it a data-base), and a separate computational image (with whatever application is of interest). At this point the user would have constructed and saved three images web + controller, data-base, and application. Now the user would (through VCL web interface) configure the controller image to be a parent, and the other two images to be children of this particular parent (one could add as many children as wanted e.g. five computational node images). From now on, when the user makes a reservation of the parent lablet image, all three or more images are loaded. Depending on how the mapping of the images onto hardware (real or virtual) is done, this small lab cluster could be tightly coupled (low latency network) or loosely coupled. In any case, when the reservation is granted, each image would find in its $/etc/cluser\_info$ file primary IP number of all its siblings and of its parent, that is the parent and the children know about each other. Following illustrates the content of the file

```
[vouk@bn19-36 etc]$ more cluster_info
child= 152.46.19.36
parent= 152.46.19.5
child= 152.46.20.78
child= 152.46.20.86
[vouk@bn19-36 etc]$
```

This enables the controller to find all its children, and the children to find their parent without fixed IP addresses (latter is of course possible but in a cloud it could be a limiting factor). Now a student who makes a lablet reservation gets, on demand, an independent personalized set of resources. Typical VCL reservations are about couple of hours (e.g. homeworks, class), but longer term and open ended resource reservations are possible and are being used for different types of projects.

If a dynamic cluster is needed, for example we would like the lablet above to dynamically extend its computational resources, the controller could use VCL API at run-time (after being deployed) to make (elastically) additional reservation of the computational node and deploy a larger computational sub-cluster that would serve the lablet.

In order to support applications as the one above, as well as services that range from individual desktops (or seats), to classroom or lab clusters, to server farms, to specialized sub-clouds and HPC or HPD services, VCL has a hierarchical set of built in functions that allow sophisticated manipulation of VCL resources, their scheduling, their monitoring, and their management. VCL architecture supports a distributed cloud and therefore its resources can be located in one data center, or over a number of data centers in different states and countries. Out-of-the-box VCLs very strong and sophisticated Hardware-as-a-Service (HaaS) and Infrastructure-as-a-Service (IaaS) capabilities are augmented with basic Platform-as-a-Service (PaaS) in the form of base-line images (operating systems, e.g. Linux, Windows). At NC State we also then provide centrally a certain number of Software-as-a-Service (SaaS) images (e.g. MatLab images or Maple software images). However, for most part it is up to users to extend the basic PaaS and SaaS offerings with their own services. When permitted, they reserve a basic PaaS image and install on it (provided they have appropriate licenses) software they wish, save the image(s), and then make that service available to their user community. Centrally, NC State maintains perhaps 30 or so PaaS and SaaS services, the rest of the 2,000 or so available service images have been constructed by authorized advanced users (e.g. professors, research assistants, students).

Of course, not all of the images are available to all users. Role based access control and image visibility is maintained using a variety of data sources for example class roles, research group membership, etc. This allows very fine grained mapping of individual services (individual images, or image groups) onto specific resource groups, specific user groups, specific schedule groups, specific management nodes and data centers, and specific privilege tree branches and leaves. Through this one can control many things computing power, licensing, service isolation, security, and so on.

Security typically involves use of VPNs to initially access VCL private cloud authentication and authorization, and later to access activated resources. IP address locking is used to make the resources visible to authorized user only, and watchdog time-out is used to manage inactivity, resource hogging, and possible denial of service situations. Also, traffic monitoring is used to identify potentially compromised edge-resources and trigger resource isolation if needed. VLAN-ing can be used to isolate resources provided appropriate VLAN-ing hardware equipment, or similar virtual solution, is available. Real-time and asynchronous dashboards and other tools are available to analyze and mitigate VCL internal performance and issues. Interested reader should consult some of the following articles and web-sites to obtain more information about VCL [13, 11, 6, 7, 20, 21].

2.3.2 VCL as an Operating System

At this point, almost 10 years after we have started working on a cloud solution which has arguably become the first academic production level cloud in the world, we believe we know how to build a secure and efficient cloud that is functionally suitable for the broad range of needs that arise in research universities. A natural question is do we stop there? What is next?

We have decided that a natural extension to cloud services of the classical type is the ability to plug into that cloud almost any device that may be useful in the university from a microscope (particularly an expensive one), to a genetic sequencer, to a networking laboratory, to a classroom. The intent is to either increase utilization of the hardware by making it available 24/7, or virtualize it and thus increase utilization through multi-tenancy, or emulate, or simulate normal part or all of the physical device, room, and even building. This would, in theory, enable us to allow authorized external users (students, faculty, researchers) to access these resources in some form on a 24/7 basis. This helps extend access options and turn resources and services that are often idle at night, or out of reach to distance education students, or unavailable to remote researchers and industrial partners for some other reason, into easy to access utilities. It would be like walking onto the campus with appropriate permissions of course and making use of sometimes unique university research and teaching resources. If done right, this could level the education field, and provide unprecedented advances to higher-education resources and sophisticated research partnerships.

This of course, is not a new idea. Remote controlled equipment has been around for a long time. NC State has been experimenting with this for years [5, 4, 16]. For example, Fig. 2.3 illustrates the 1999 instrumented backhoe that was installed in the Department of Civil Engineering at NCSU and was network-enabled in collaboration with Computer Science for remote manipulation. In addition to joy-stick operation of the equipment, remote access also involved stereoscopic cameras and sound so that operators would have good visual and audio awareness of the remote environment. It was demonstrated at an Internet2 meeting in Atlanta. At about the same time much larger facilities were also being turned into virtual labs over internet at other places for example the University of Hawaii and the Association of Universities for Research in Astronomy connected eleven leading observatories to Internet2 networks via the Mauna Kea Observatories Communication Network [17].

Today, almost any significant equipment does come with processors, IP numbers, and ability to access and control it from a distance. However, most of the large equipment is part of stand-alone solutions, and not necessarily part of an AaaS cloud offering, and this is understandable. Open questions are many. For instance, how safe and secure this is, how to schedule this, what are the pedagogical and other obstacles, what are user interface issues, how to ensure that such as service is very utility-like in both ease of use and reliability and availability (probably a few other trust related parameters), and so on.

To learn more we have decided to pilot several more complex options to see how they can be smoothly (ideally plug-and-play) integrated into an environment such as VCL as Application-as-a-Service or Instrument-as-a-Service. In this context, VCL serves as a campus-area PaaS and SaaS container environment essentially an operating system for services that are composed of physical resources, information technology resources, and remote and local users and provides scheduling, access, accounting control, monitoring and similar for higher level services that need those functions.

2.3.3 Virtual Campus

As part of its business continuity plans NCSU can launch 100+ virtual classrooms on short notice from its cloud computing solution [21]. This option currently uses

Fig. 2.3 Turn of the century NCSU experimental virtual laboratory featuring remotely manipulated teaching and research backhoe (Civil Engineering, Computer Science)

Blackboard Collaborate (formerly Elluminate) to provide classroom spaces in the event of a physical campus shutdown. We also have Internet Reactor Laboratories that are available to external academic institutions who wish to utilize the NCSU PULSTAR reactor to demonstrate nuclear reactor operations and kinetics for their students. This capability enriches academic programs at universities without research reactors of their own, and may be used to expand the educational opportunities for nuclear engineering students throughout the United States and internationally. [16]. A considerable number of our networking degree and cloud computing training is done using VCL hosted virtual lablets, and so on. However, we also have a number of facilities which could benefit from remote access option, but for variety of reasons do not have that capability yet.

This has prompted us to initiate a project we call vCentennial. It is an ambitious vision of change in education and research paradigm that would offer NCSU the ability to replicate its Centennial Campus services and functionality of this physical environment and its virtual avatars anywhere, anytime in the world using a cloud of clouds platform. The plan is to appropriately virtualize its award-winning Centennial Campus [18]. Centennial Campus is a small city made up of NCSU research, teaching and outreach facilities, entrepreneurs, academic entities, private firms, and government agencies. The path that we are taking is to systematically build on the state-of-the-art infrastructure and facilities of the Centennial Campus. This is illustrated in Fig. 2.4.

Centennial Campus facilities and buildings are new, but not all of them are yet smart. The process of adding sensors and an intelligent campus operations capability is in progress. Centennial networking infrastructure is first class fiber in the building risers and in the ground, 1 Gbps to the desktop capability, 4 to 10 Gbps among the buildings, latest wireless technology in all buildings and classrooms, and an experimental digital wireless canopy called CentMesh. What we are in the process of doing is extending VCL capabilities so that it can act as cloud operating system for Centennial facilities and equipment that we wish to plug in. This involves

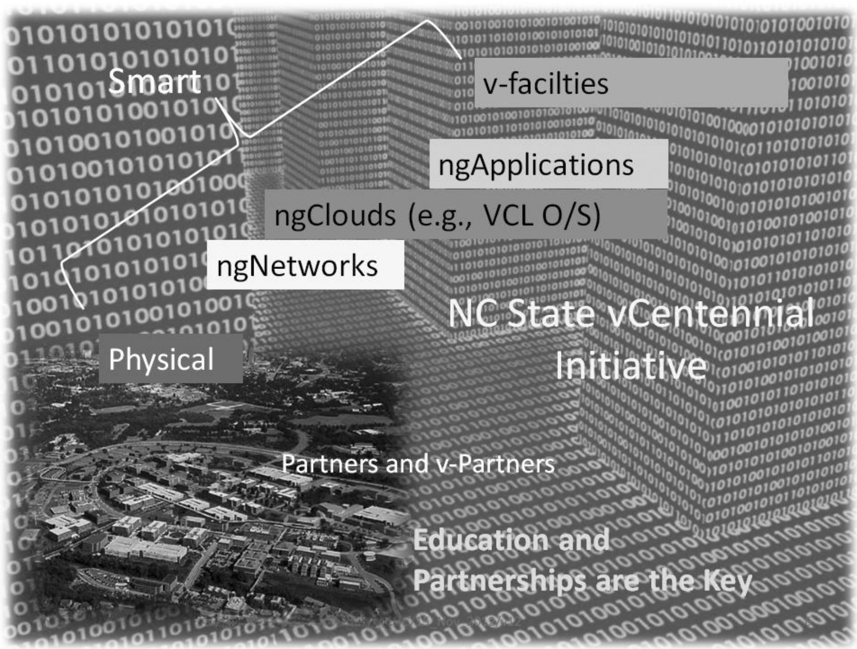

Fig. 2.4 The path to vCentennial digital campus

development of complex equipment interfaces, which may consist of one or more VCL images, in addition to the computational equipment attached to devices such as advanced electron microscopes or biomanufacturing laboratory and pilot production facilities, and interfaces to other campus laboratories and classrooms. The current challenge is that most of the complex applications (including some otherwise very sophisticated analytics software) are not fully cloud-ready. Next generation cloud applications will be aware of the applicable cloud services such as scheduling, authentication and authorization, provenance collection, policy management, and similar subsystems and would leverage such services (see Fig. 2.1).

Ultimately, new physical facilities (including buildings), research equipment, laboratories, and so on, would routinely become vFacilities. We hope this will amplify mission capabilities of the campus a thousand fold. While currently Centennial Campus has about 10,000 residents (faculty, researchers, students, staff), about 3 million square feet of space and some 60 on-campus industrial and government partners, without vCentennial capabilities when Centennials 1,300 acres are is fully built out, this might grow to some 30,000 residents, 200+ partners and perhaps 9 million square feet of partner space. With vCentennial we envision thousands of vPartners and capabilities that can reach a student, researcher and faculty populations world-wide.

2.4 Summary

North Carolina State University (NCSU) is embarked on an ambitious vision to change the paradigm for higher education and research by virtualizing its award-winning Centennial Campus. Centennial Campus is a small city made up of NCSU research, teaching and outreach facilities, entrepreneurs, academic entities, private firms, and government agencies. NCSU wants the ability to replicate services and functionality of this physical environment and its virtual avatars anywhere, anytime in the world using a cloud of clouds computing platform. The initial operating systems for this platform is NCSUs Virtual Computing Laboratory (VCL) technology, a powerful but low cost, and easy to use, private cloud developed over the last 8 years in collaboration with IBM. VCL now powers a considerable number of information technology offerings to students and researchers both at NCSU and world-wide.

References

1. Adair, R.J., Bayles, R.U., Comeau, L.W., Creasy, R.J.: A Virtual Machine System for the 360/40, IBM Corporation, Cambridge Scientific Center Report No. 320 (May 1966)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., et al.: Above the Clouds: A Berkeley View of Cloud Computing, Electrical Engineering and Computer Sciences, University of California at Berkeley, Technical Report No. UCB/EECS-2009-28 (February 10, 2009), http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html
3. Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: A view of cloud computing. Communications of the ACM 53(4), 50–58 (2010)
4. Bernold, L.E., Lloyd, J., Vouk, M.: Equipment Operator Training in the Age of Internet2. In: NIST SP 989, 19th International Symposium on Automation and Robotics in Construction (ISARC), Gaithersburg, Maryland, pp. 505–510 (2002)
5. Case, G.N., Vouk, M., Mackenzie, J.: Real-Time Full Motion Stereo Microscopy Over Internet. Microscopy and Microanalysis 5(suppl. 2), 368 (1999)
6. Dreher, P., Vouk, M., Sills, E., Averitt, S.: Evidence for a Cost Effective Cloud Computing Implementation Based Upon the NC State Virtual Computing Laboratory Model. In: Gentzsch, W., Grandinetti, L., Joubert, G. (eds.) Advances in Parallel Computing, High Speed and Large Scale Scientific Computing, vol. 18, pp. 236–250 (2009) ISBN 978-1-60750-073-5
7. Dreher, P., Vouk, M.: Utilizing Open Source Cloud Computing Environments to Provide Cost Effective Support for University Education and Research. In: Chao, L. (ed.) Cloud Computing for Teaching and Learning: Strategies for Design and Implementation, pp. 32–49. IGI Global (2012)
8. Garrison, G., Sanghyun, K., Wakefield, R.: Success factors for deploying cloud computing. CACM 55(9), 62–68 (2012)
9. Crnković, I., Sentilles, S., Vulgarakis, A., Chaud, M.: A Classification Framework for Software Component Models. IEEE Transactions on Software Engineering, 593–615 (September 2011)

10. Ludaescher, B., Altintas, I., Bowers, S., et al.: Scientific Process Automation and Workflow Management. In: Shoshani, A., Rotem, D. (eds.) Scientific Data management – Challenges, Technology and Deployment, ch. 13, pp. 467–507. CRC Press (2010)
11. Schaffer, H.E., Averitt, S., Hoit, M., Peeler, A., Sills, E., Vouk, M.: NCSUs Virtual Computing Lab: A Cloud Computing Solution. IEEE Computer, 94–97 (July 2009)
12. Vouk, M.: Cloud Computing - Issues, Research and Implementations. Journal of Computing and Information Technology 16(4), 235–246 (2008)
13. Vouk, M., Rindos, A., Averitt, S., Bass, J., Bugaev, M., Peeler, A., Schaffer, H., Sills, E., Stein, S., Thompson, J., Valenzisi, M.: Using VCL Technology to Implement Distributed Reconfigurable Data Centers and Computational Services for Educational Institutions. IBM Journal of Research and Development 53(4), 1–18 (2009)
14. Amazon Elastic Compute Cloud (Amazon EC2), http://aws.amazon.com/ec2/ (last accessed November 3, 2014)
15. IEEE, IEEE Launches Pioneering Cloud Computing Initiative (April 4, 2011), http://www.computer.org/portal/web/pressroom/20110404cloud
16. Internet Reactor Labs, North Carolina State University, Nunclear Engineering, http://www.ne.ncsu.edu/nrp/irl.html (last accessed November 3, 2014)
17. Mauna Kea Observatories, Hawaii, http://www.ifa.hawaii.edu/mko/
18. North Carolina State University Centennial Campus, http://centennial.ncsu.edu/history.php (last accessed November 3, 2014)
19. NIST, NIST Cloud Computing Program, http://www.nist.gov/itl/cloud/ (last accessed November 3, 2014)
20. VCL-Apache, http://vcl.apache.org (last accessed November 3, 2014)
21. Virtual Computing Laboratory, http://vcl.ncsu.edu (last accessed November 3, 2014)