In the proceedings of the International Conference on Virtual Computing Initiative, May 7-8, 2007, IBM Corp., Research Triangle Park, NC, pp. 1-16.

Virtual Computing Laboratory (VCL)

Sam Averitt, Michael Bugaev, Aaron Peeler, Henry Shaffer, Eric Sills, Sarah Stein, Josh Thompson, Mladen Vouk,

North Carolina State University, Raleigh, NC 27695

{sfa, mabugaev, fapeeler, hes, edsills, sstein, jfthomps, vouk}@ncsu.edu

ABSTRACT

Virtual Computing Laboratory (VCL) – http://vcl.ncsu.edu, is an open source implementation of a secure production-level on-demand utility computing and services oriented technology for access to solutions based on virtualized resources, including computational, storage and software resources. It is serving the NC State student and faculty population, nominally 30,000+ users. Currently its primary focus is on "industrial strength" applications and environments, including individual (real or virtualized) desktops, servers and services, and high-performance computing resource, that otherwise may be difficult to install on end-user platforms due to memory, CPU, vendor licensing, or other issues. It was developed by the NC State College of Engineering and the NC State Information Technology Division in response to NC State mission needs of education, research and outreach. In this context VCL is an access, competitiveness and equity tool. VCL is also a prototype implementation in the Virtual Computing Initiative (VCI) launched jointly by IBM and NC State. Its key differentiators are, a) simplicity of implementation and maintenance, b) versatility, security and cost-effectiveness, c) its broad resource-based approach to "virtualization," d) and its very flexible way of delivering resource services through "images."

1. INTRODUCTION

A key differentiating element of the information technologies (IT) of the 21st century that will be/is a success is their ability to become true and valuable contributors to cyberinfrastructure. *Cyberinfrastructure*[1] *makes* ***[systems and] applications dramatically easier to [use,] develop and deploy****, thus expanding the feasible scope of applications possible within budget and organizational constraints, and* ***shifting the [educator's,] scientist's, engineer's effort away from information technology (development) and concentrating it on [knowledge transfer, and] 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."*

[1] From the Appendix of the Report of the National Science Foundation Blue-Ribbon Advisory Panel on Cyberinfrastructure, Jan 2003. Items in square brackets added by author.

Figure 1. VCL web portal.

This is especially true of any technologies that should be used in large-scale educational systems, solutions and environments. Educational environments are especially diverse, dynamic, and demanding. Global competitiveness of our workforce dictates a huge range of options – from the state-of-the-art technologies and research software to mundane commodity solutions, from single desktop solution to distributed high-performance clusters and supercomputing solutions, from trivial software to very complex and sophisticated software, to "precarious" software produced in research environments, from a synchronized group of classroom or laboratory 'seats" to a research or business multimode solution. From the perspective of end-users, information technology MUST be enabling and appliance-like. End-users should be able to use the technology to improve their productivity and reduce technology-driven overhead (e.g., less than 20% of their effort should have to do with IT issues). The IT overhead that the end-user sees, e.g., software installation or management issues, must be drastically reduced, while the overhead that underlying infrastructure installation and maintenance personnel sees MUST have the property of the economy-of-scale at all levels – hardware, software, provisioning, maintenance, etc.

2. ELEMENTS

2.1 Services and Virtualization

One such powerful enabling concept is utility-computing through service architectures. An environment where the end-user requests a service at the desired level, and receives it on-demand or through reservation. This can be achieved in many ways: through a true service oriented architecture, through more traditional web, through remote access solutions of various types, through GRID, through virtualization, and so on. The virtualization concept, of course, has been around since late 1960s (e.g., in IBM mainframe systems). Virtualization allows aggregation and sharing of physical resources in a way which is invisible to end-users, but which improves their utilization and reduces the overall cost of the solution by reducing the cost of some all of its components: hardware, software and licenses, management and administrative costs. There are currently numerous implementations of virtualization at many levels: service level, application level, desktop level, operating system level, and hardware level. In a broader sense, this includes virtual desktop infrastructure, more traditional server based solutions, web-based solutions, and terminal server solutions.

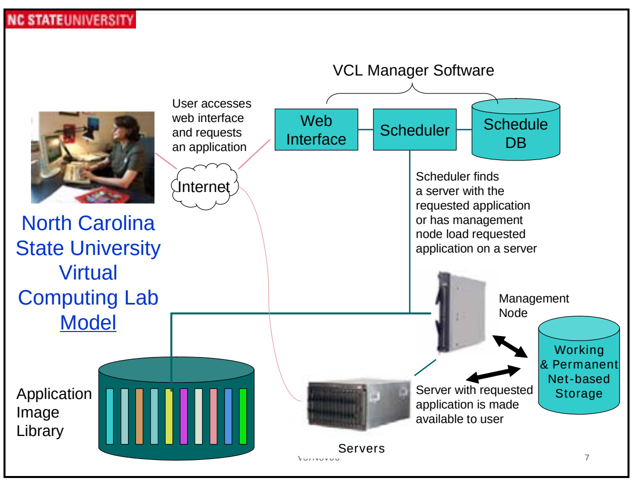


Figure 2. Illustration of VCL physical architecture

2.2 Model and Functionality

NCSU VCL model is a simple one (Figure 2), yet a very effective one. A user accesses VCL through a web interface (portal – see Figure 1) to select from a menu a combination of applications, operating systems and services she needs. If a specific combination is not already available as an "image", an authorized user can construct one's own from the VCL library components. This customization capability is very much in the spirit of what services engineering and management is all about. VCL Manager software then maps that request onto available software application images and available hardware resources and schedules it for either immediate use (on demand) or for later use. VCL Manager software was developed by NCSU using a combination of off-the-shelf IBM products (such as IBM xCAT and IBM Tivoli Monitor) and in-house developed "glue" or middleware software (about 30,000+ lines of code). All components of the VCL can be (and are) distributed.

From the NC State perspective, the network of VCL nodes and services is a **grid that works**. It is a secure, scalable, maintainable and sustainable service-oriented architecture (SOA) compliant solution for delivery of a variety of diverse service environments **anytime** and **anyplace on demand** or by **reservation.** It increases **utilization** of resources - it has **broadly variable** server resources matched to service environment requirements ranging from virtual servers, to single a physical server, to multiple physical servers for **dedicated** single user or **shared** multiple users.

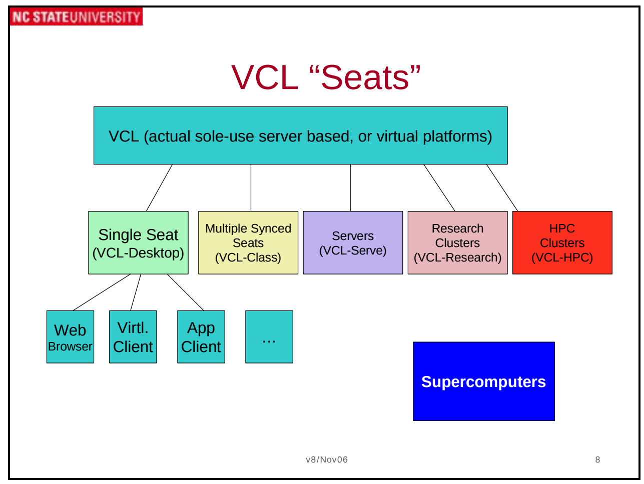


Figure 3. Illustration of the VCL service categories.

VCL delivers a range of options (Figure 3) – from single real or virtual computer laboratory "seats" or desktops, to single applications on-demand, to classroom size groups of seats, to enterprise server solutions, to research clusters, to high-performance computing services.

What differentiates VCL from the many **virtualization** and distributed computing solutions already available? VCL provides a uniquely simple, flexible, reliable, scalable and economical approach to mapping and managing user application and services needs to available local or distributed software (images) and hardware resources. It virtualizes the resources (the composite of the software and hardware stack), from the point of view of the end-user (Figure 4), and it is not only end-user access platform (e.g., laptop, PDA) agnostic, but to a very large extent, software service manufacturer agnostic.

An "**image**" (Figure 4) is a software stack that 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.

A user can have either sole use of one or more hardware units, if that is desired, or the user can share the resources with other users. Scalability is achieved through a combination of multi-user service hosting and application virtualization, and load balancing.

By design, VCL subsumes and allows any of the "software as a service" solutions, virtualization solutions, and terminal services solutions available today – e.g., VMWare, XEN, MS Virtual Server, Virtuoso, and Citrix to mention a few. It also as allows any of the access/service delivery options that are suitable – e.g., from RDP or VNC desktop access, to X-Windows, to a web services or similar.

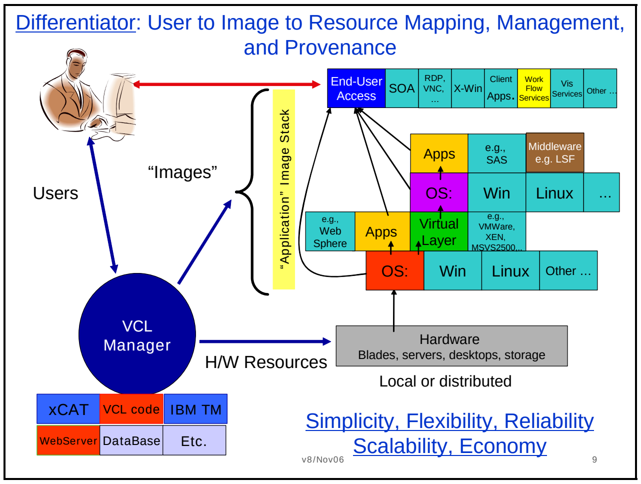


Figure 4. Illustration of the VCL logical architecture.

VCL Manager software is a combination of open software developed by NC State University, of-the-shelf open solutions such as the IBM xCAT, and open source web servers and data bases. System profiling and provenance information is used to optimize scheduling of the requests and problem resolution. Profiling and provenance is provided through a combination of system logs, image usage statistics and operational profiles. For example, IBM Tivoli Monitoring software can be used to assess image performance needs and thus make appropriate virtualization decisions.

Computational hardware and storage can be any appropriate platform – from a blade center, to a collection of diverse desktop units or workstations, to an enterprise server, to a high-performance computing engine. Storage, communications and other hardware us subsumed. VCL Manager provides appropriate virtualization (aggregation, dis-aggregation) of the available hardware resources before mapping the requested image onto that hardware.

3. PRODUCTION

NC State has been operating VCL for close to three years now (since 2004). Its current hardware base is a 600+ processor IBM blade cluster and over 200 classical laboratory desktops. It also incorporates over 20 TBytes of disk space and a tape-backup system. Laboratory facilities are used when they are closed for business. VCL serves both desktop users and high-performance computing users. Currently, there are three principal VCL nodes in operation at NCSU – one in our Data Center #1, one in our Data Center #2 and one in the Friday Institute.

3.1 Architecture

VCL is an SOA compliant environment delivery service (Figure 4), where in principle an "environment" can be any modern day host, operating system, application or service that is remotely accessible over the network. This includes standalone computers sitting in a public lab, blade or non-blade servers in a data center, Virtual machines running in a hosted environment, or even enterprise level services.

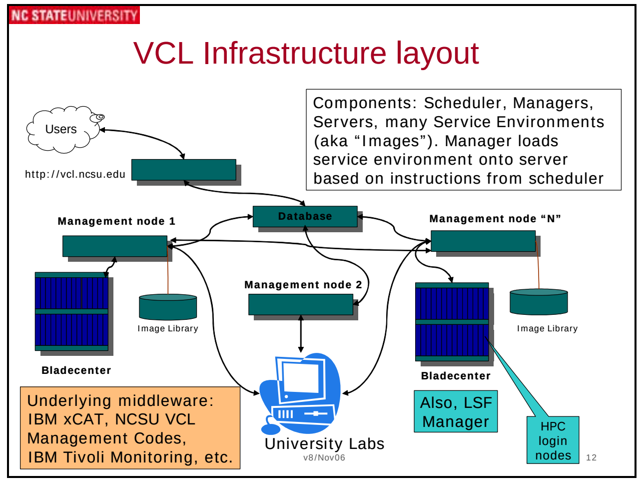


Figure 5. Infrastructure layout.

A typical VCL environment consists of an operating system with an application or a combination of applications, often with administrative or root access. The distributed nature of its architecture is further illustrated in Figure 5. Its components are a distributed resource scheduler and manager, a bank of computational, storage and "image" resources, and an end-user interface and authentication and security solution – typically web-based with multi-layer security elements.

A user can request these environments by going to the VCL portal where, based on the user's credentials (e.g., userid), the user is presented with a simple menu of VCL environments to choose from. This list may contain one application environment or many different environments. These environments can ranging from Windows, Linux and Solaris desktops to Windows2003 and Linux server-based images, to some application specific network-based service.

In our initial 2004 pilot we provided a Windows XP environment with Solidworks (a 3-D modeling application) for use by NC State's distance education engineering courses. Normally high-end applications are very costly for students to purchase directly and most vendors prohibit the distribution of the media for users to install on personal machines. The VCL model allows us to easily adhere to strict vendor licensing requirements, and in most cases provide greater licensing control. The model provides a cost effective way for students and faculty to gain access to applications that otherwise may not be available to them. It also provides a means for students, both on-campus and in distance education courses, to access high-end (industrial strength) applications running on powerful dedicated computer nodes. The only requirement is that a user has a reliable Internet connection and a computer which can run access software they have chosen, e.g., remote desktop connection, VNC, ssh with an X server, or an application-specific client.

The VCL service focuses on controlling the resource at the platform level. As such, it subsumes solutions such as Citrix and hyperviser-based implementations (e.g., VMWare, Xen and Microsoft's Virtual Server based solution). Since many of these solutions are simply just (possibly multi-server) environments, which can be dynamically loaded on the Blades, these solutions can become just another level in the VCL “image stack”.

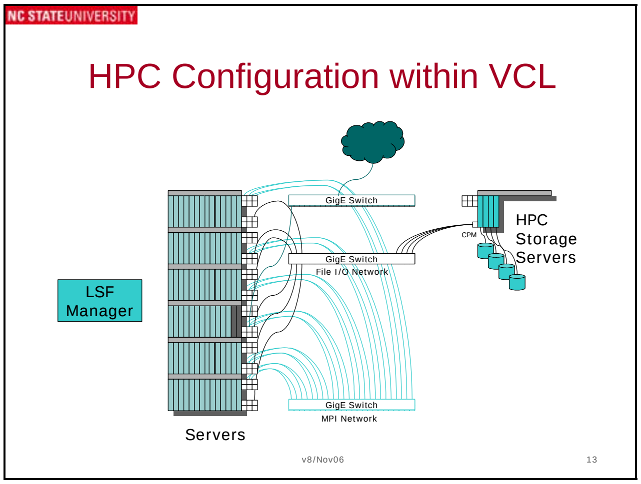


Figure 6. VCL integrates with our HPC structure.

The VCL idea is a direct derivative of the original NC Stae HPC activities. Indeed, ability to access HPC resources is integral to VCL (Figure 6). VCL development was initiated through, and augmented by, multiple IBM Shared University Research (SUR) equipment grants. Additionally, funding from the IBM RTP Center for Advanced Studies (CAS) has supported numerous research projects associated with the VCL. IBM software used in the lab (including the IBM Tivoli Monitoring and WebSphere Application Server middleware) was donated by the IBM Academic Initiative, a program that provides IBM software free for use in the classroom or in research, with an aim toward developing curricula that addresses critical industry skills.

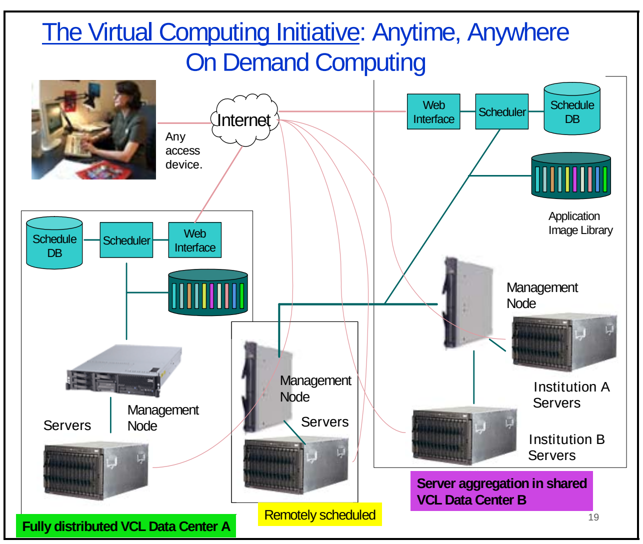


Figure 7. Virtual Computing Initiative

The physical heart of the VCL is IBM BladeCenter which integrates servers, networks, storage and applications into highly efficient systems that sit in a rack like books on a shelf. The solution can also tap into classical computer laboratory desktops when they are not in use (Figure 5, Figure 7). The brain of the solution is the open source VCL resource management software developed at NC State.

VCL is deployed as management, image, and portal nodes (Figure 5, Figure 7). VCL manager allocates the appropriate number of server resources for the task at hand, automatically installing the desired operating system, applications, and other software “images,” and distributing them across those resources to optimize performance. VCL allows students and faculty to tap into the power of high-end laboratories directly from their laptops or computers from anywhere on campus - or from anywhere in the world.

3.2 VCI Partners

The VCL community has been expanding and now includes a number of institutions within and out of state, as well as some other industrial partners such as IBM Corp. and the SAS Institute.

To expand this model across the state and beyond, IBM and NC State University have started the "Virtual Computing Initiative" (VCI). In addition to the founding partners, NC State University and IBM Corp., initial participants in this joint endeavor include North Carolina Central University (NCCU), University of North Carolina at Chapel (UNC-CH), University of North Carolina Greensboro (UNCG), East Carolina University (ECU), Duke University, some K-12 schools in Granville, Franklin, Halifax and Northampton counties.

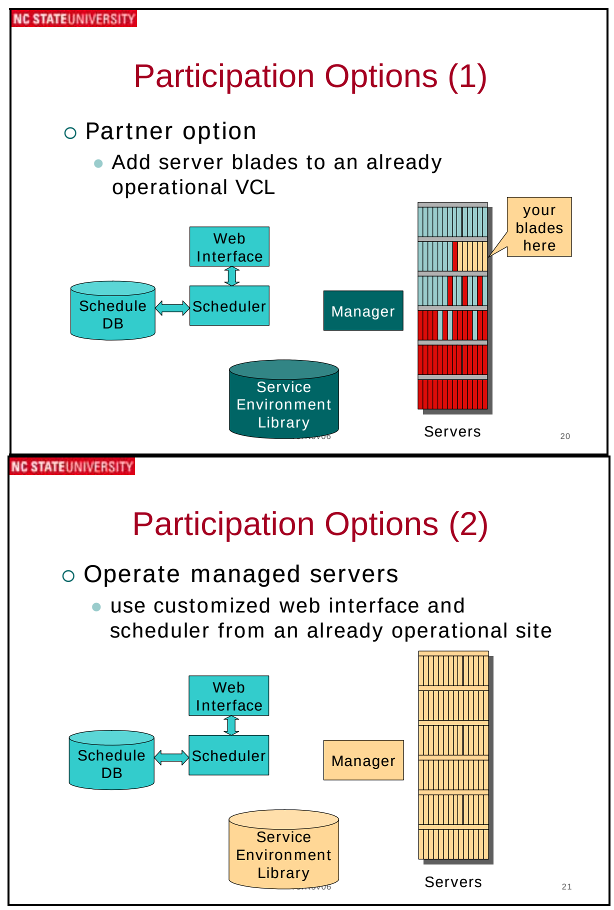
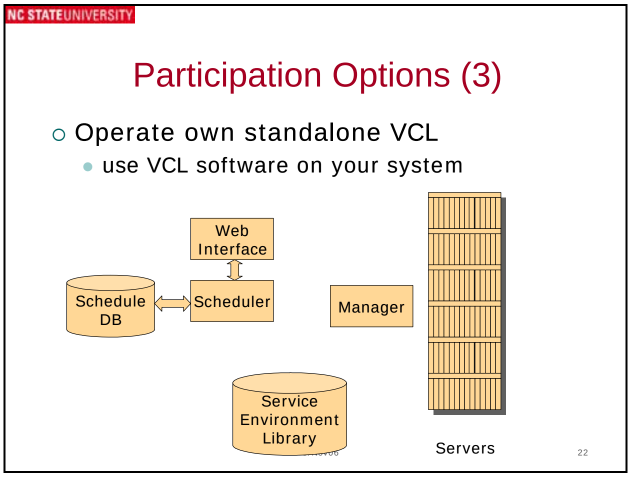


Figure 8. End-user only option (1) and end-user and equipment management options (2).

The goal of the Virtual Computing Initiative is to create a multi-institutional shared computing **services community** based upon the VCL model that will include universities, community colleges, K-12 schools and business partners. Through this initiative, IBM provides support for research into the service, applications middleware (e.g. Tivoli performance monitoring solutions) and infrastructure innovations that wide deployment of the architecture requires. Many of the supported activities are in the form of open source community projects.

Participation in VCI can take a number of forms – from just being an end-user, to actually owning VCL nodes and sharing VCL resources with others. This is illustrated in the figures 8 and 9.



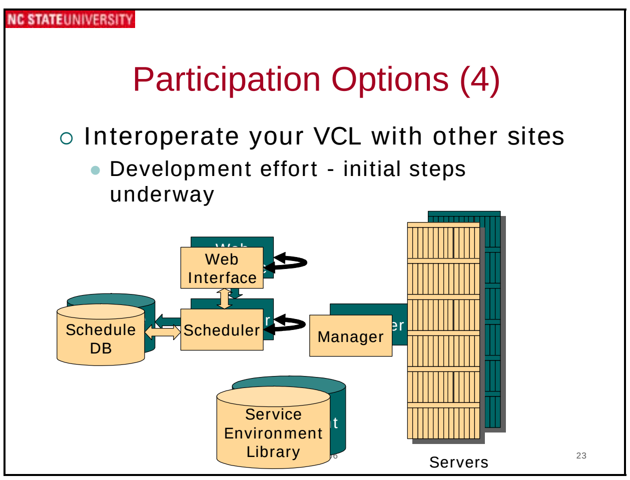


Figure 9. Full ownership (3) and shared resources (4) options.

4. SUMMARY

Virtual Computing Laboratory (VCL) – http://vcl.ncsu.edu, is an open source implementation of a secure production-level on-demand utility computing and services oriented technology for access to solutions based on virtualized resources. Currently, it is serving the NC State student and faculty population, nominally 30,000+ users.

The benefit of this scalable and flexible shared open source solution, to which we hope all Virtual Computing Initiative (VCI) participants who wish to do so will also contribute as developers as well, are numerous. They include management and provisioning that can be as centralized or as distributed as one wishes, considerably increased utilization of the resource and thus reduced cost of service delivery, reduced information technology overhead on institutions that may not have appropriate resident personnel for construction and maintenance of very complex information technology services, provision to end-users of a very broad palette of computational services on-demand - thus leveling the field at an affordable cost, individualization and customization that each institution and user group needs, and provision of a viable disaster recovery and business continuity solutions across all participating institutions.