

# Proposed Framework of Private Cloud Setup in Lab for Teaching and Research

Rajender K Trivedi, Ph. D  
Graphic Era University  
Dehradun

Rajani Sharma  
Graphic Era University  
Dehradun

## ABSTRACT

Advancement in the field of information technology has enabled us to optimize the use of physical infrastructure with the help of virtualization, most of which is only used in specific applications like vitalizing a local area network on switch or vitalizing a compute resource to run more than one operating system on a desktop. The benefits of same has not been completely implemented and utilized in education and research environments where we require a large amount of IT resources. Cloud computing has emerged as one step further solution to deploy all virtualized IT resources as per need on self-service basis as a rental method for the users. Different cloud deployment methods have emerged according to the need of users, institutions and communities. This paper describes the framework of design and implementing a simple private cloud for education and research based on chip level virtualization with hypervisor to establish a simple model of Infrastructure as a service in cloud computing.

## General Terms

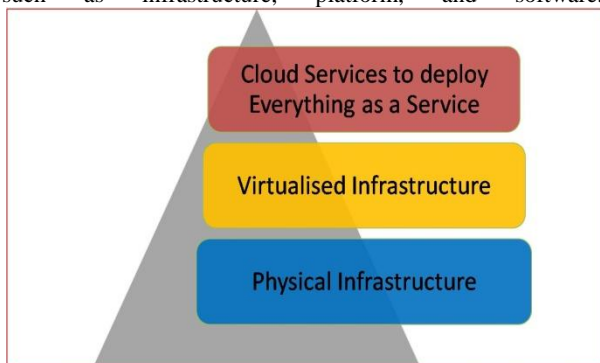
Private cloud framework

## Keywords

Cloud computing, hypervisor, virtualization, data center.

## 1. INTRODUCTION

In last decade virtualization has been a milestone in enabling optimal usage of information technology resources. Cloud computing has provided a platform for deploying everything in IT as a Service, which has changed the economics of IT based business and organizations from Capital Expenditure to Operational Expenditure. For enabling Cloud services we first need to understand our existing infrastructure. In second step we should be able to virtualize it for optimal usage. Then with the help of existing cloud applications we can deliver and deploy information technology as a resource at various levels such as infrastructure, platform, and software.



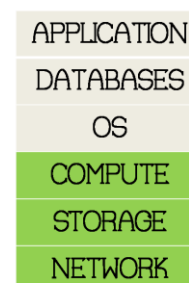
The advantage of using private cloud in education and research field is to enable complete utilization of Information technology resources, where most of the resources are underutilized. The cost involved in maintenance and

management of those resources is huge which can be saved due to centralized management of private cloud infrastructure. We create private cloud keeping in mind

- Availability and performance requirements for the cloud
- Infrastructure sizing, performance and continual availability requirements
- The need for data security or isolation
- The appropriate user interface

There are majorly three cloud service models existing. Infrastructure as a Service provides capability to the consumer to hire infrastructure components such as servers, storage, and network. It enables consumers to deploy and run software, including OS and applications. Consumer pays for infrastructure components usage, for example, storage capacity, CPU usage, etc.

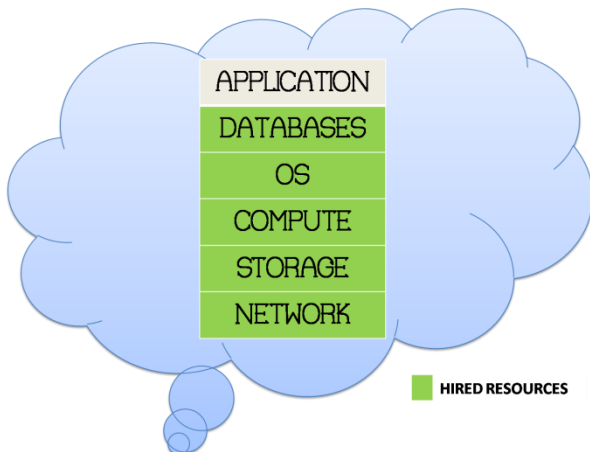
## Cloud---Infrastructure-as-a-service



HIRED RESOURCES

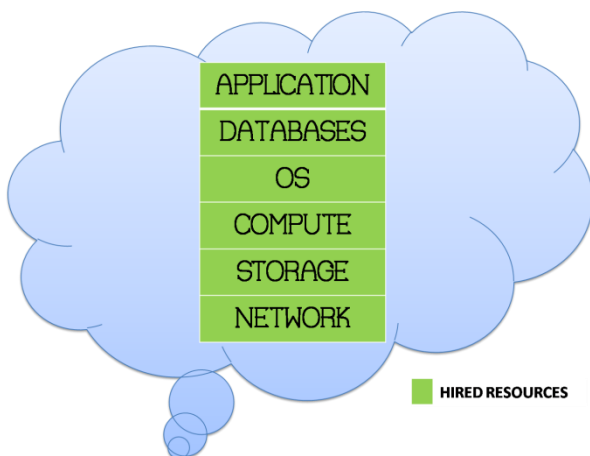
In platform as a Service consumer can deploy consumer-created or acquired applications on the cloud provider's infrastructure. Consumer has control over deployed applications and possible application hosting environment configurations. Consumer is billed for platform software components which include its associated infrastructure cost such as OS, Databases, and Middleware.

### Cloud---Platform-as-a-service



In Software as a Service cloud application consumer can use provider’s applications running in a cloud infrastructure. In this complete stack including application is provided as a service. Application is accessible from various client devices, for example, via a thin client interface such as a web browser.

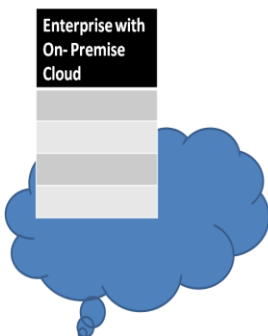
### Cloud---Software-as-a-service



## 2. PROPOSED DESIGN

Designing private cloud consists of following steps:

### Cloud Deployment Model – Private Cloud



The key steps in deploying a private cloud infrastructure are to:

- Deploy private cloud infrastructure from the different fabric resources in the data center.
- Deploy compute fabric through bare metal operating system deployments of Hypervisors.
- Discover, classify, and allocate our storage fabric for private cloud use.
- Abstract our networking fabric for use in private cloud.
- Pull these fabric resources together and create a cluster for use as the underlying infrastructure for the private cloud.

### 2.1 Cloud Infrastructure Framework

The cloud infrastructure consists of the following components:

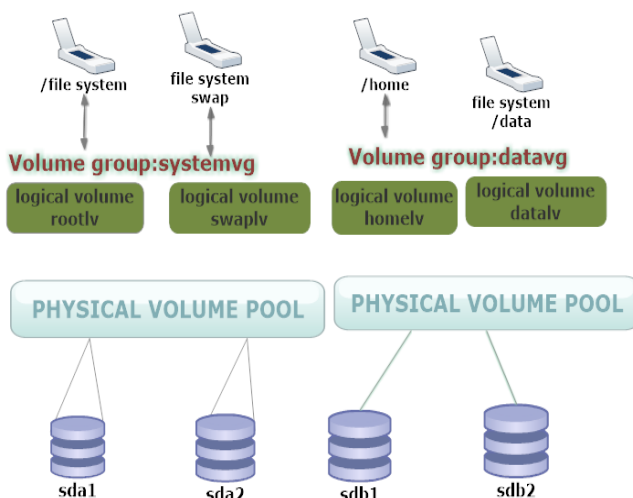
- Physical Infrastructure which includes physical IT resources such as physical servers, storage systems and physical network components. Physical servers are connected to each other, to the storage systems, and to clients via physical networks.
- Virtual infrastructure which includes Resource pools (CPU, memory, network bandwidth, storage), Identity pools (vlan ID, mac address) and Virtual IT resources such as virtual machines, virtual volumes and virtual networks such as virtual switches and virtual NIC’s. Virtual IT resources obtain capacity and identity from resource and identity pools.
- Application and platform software such as OS, Database. These are created to provide software-as-a-service and Platform-as-a-service.
- Cloud Infrastructure Management and service creation tools consists of following components
  - Virtual infrastructure management software – It creates pools of available physical infrastructure like creating storage pools, create vlan ID pools, CPU and memory pool. After creating pools this software help creating virtual infrastructure such as virtual machines, virtual LAN and virtual volumes.
  - Unified management software – It interacts with virtual infrastructure software and collects information on existing physical and virtual infrastructure configurations. Unified software uses this information and provides a consolidated view of IT resources scattered across VDC’s. This helps an administrator to monitor performance, capacity and availability of physical and virtual resources centrally. It also enables administrator to add capacity and identity to the existing pools. It passes configuration commands to the respective VDC management software and hence eliminates the administration of IT resources separately.
  - User access management software – Allows users to request cloud services. It interacts with unified management

software and forwards all service requests. It helps administrator to create and publish cloud service catalogue. It also authenticates users before fulfilling their service requests and monitors allocation or usage of resources associated with each Cloud service instance.

## 2.2 Lab set up for demonstration of Storage Virtualization

For teaching practical cloud computing, we can take approach of massive virtualization as one of the first step to enable cloud. We can demonstrate the abstraction of storage space with the help of storage virtualization by implementing Linux volume manager and creating logical volumes on the pool of physical volumes.

A Logical Volume Manager (LVM) abstracts disk devices into “pools” of storage space called Volume Groups (VGs). These Volume Groups may then be subdivided into virtual disks called Logical Volumes (LVs). These may be used just like “regular” disks, with file systems created on them, and mounted in the UNIX file system tree. There are many different implementations of the general concept of Logical Volume Management.

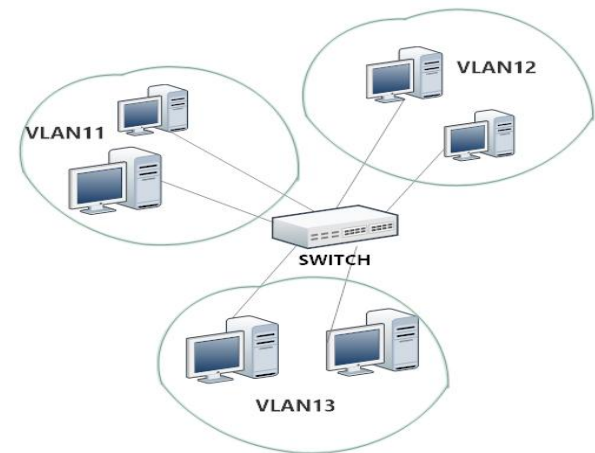


Some simple commands and steps to create storage virtualization with the help of Linux volume manager.

```
#fdisk /dev/sdb
#pvcreate /dev/sdb1
Physical volume “/dev/sdb1” successfully created
#vgcreate testvg /dev/sdb1
Volume group “testvg” successfully created
#lvcreate -L 20M -n testlv testvg
Logical volume “testlv” created
#mkfs.ext3 /dev/testvg/testlv
[... loads of superfluous text removed ...]
#mkdir /mnt/test
#mount /dev/testvg/testlv /mnt/test
```

## 2.3 Lab set up for demonstration of Network Virtualization

Like storage virtualization we can also demonstrate network virtualization concept with the help of VLAN implementation which is easily possible in a simple lab environment or on a network simulator like Boson or Packet Tracer.



**vtp mode [server | client | transparent]**- In global configuration mode, this command sets the operational VTP mode for the switch. The default is **server**.

**vtp domain name** - In global configuration mode, this command assigns a VTP domain name, which allows the switch to send VTP advertisements out trunk links. The default is NULL, which would allow a switch to join the first domain it received an update from.

**show vtp status** - Displays VTP status information including configuration revision number, domain name, and switch mode.

**Switchportmode [trunk | access] dynamic [desirable | auto | nonegotiate]** - In interface configuration mode, this configures the behavior of the interface. **Trunk** mode will force frame tagging. **Dynamic** mode can become a trunk if it negotiates with the other side of the link. **Access** mode is a no trunk port.

**switchport trunk encapsulation[isl | dot1q]** - Used in interface configuration mode to specify a trunking protocol. For some switches, before you can set an interface to **trunk** mode, you must first specify the encapsulation.

**show interface [type module/port]trunk** - Displays trunking information about the active or specified trunk links on the switch.

**show interface [type module/port]switchport** - Displays Layer 2 configuration and operational parameters of the switch. This includes VLAN membership and trunking status.

**vlan number** - In global configuration mode, this command defines a VLAN and puts the switch into VLAN configuration mode. In VLAN configuration mode, commands such as **name** can be used to further define the VLAN.

**show vlan [id vlan#]** - Displays VLAN information. The **id** option allows you to specify a particular VLAN.

**switchport access vlan [1-4096 |dynamic]** - In interface configuration mode, this command assigns an **access** port to a VLAN or makes it a dynamic port.

**show vlan brief** - Displays a brief table of the VLANs, including the port membership for each VLAN.

## 2.4 Lab set up for demonstration of Compute Virtualization for enabling a base cloud

We can set up a private cloud in lab for teaching purposes with the help of minimum two computer system connected with a physical switch.

The minimum required components of the test lab are following:

- The product disc or files for Windows Server 2012 Standard or Datacentre.
- Two physical computers that meet the minimum hardware requirements:
- 64 bit Intel Virtualization Technology (Intel VT) or AMD Virtualization (AMD-V) processor (2.8 Ghz dual core or better recommended)
- Dual 320 GB hard disks
- 8 GB RAM
- DVD drive optional
- Network adapter
- Two physical computers that meet the minimum software requirements:
- For client machine Windows 7/8 or Windows Server 2008R2/2012 are required.

Note: The two host machines selected for labs must use the same processor architecture. Both host machines must be

running either Intel VT or AMD-V processor.

Steps for private cloud set up in lab are:

1. Configure Hypervisor1
2. Configure Hypervisor2
3. Configure DomainController1
4. Configure Application Server1
5. Configure Client1
6. Join Hypervisor1 to the privatcloud.com
7. Join Hypervisor2 to the privatcloud.com

Step 1: Configure Hypervisor1

Hypervisor1 configuration consists of the following:

- Attach vhd disk with operating system included or install the new operating system
- Configure TCP/IP
- Install the Hyper-V role (Add roles and features)
- Configure a virtual switch

Step 2: Configure Hypervisor2

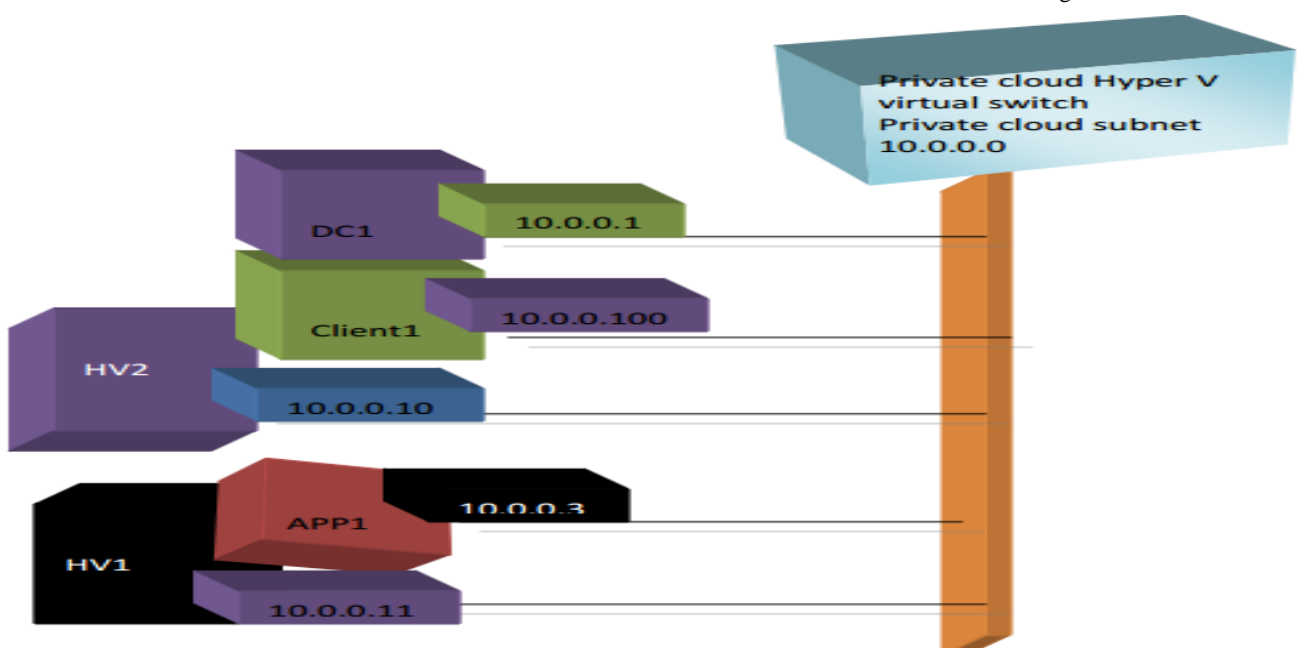
Hypervisor2 configuration consists of the following:

- Attach vhd disk with operating system included or install the new operating system
- Configure TCP/IP
- Install Hyper-V role
- Configure virtual switch

Step 3: Configure Domain Controller 1

Domain Controller 1 provides the following services:

- A domain controller for the privatecloud.com Active Directory Domain Services (AD DS) domain
- A DNS server
- Domain Controller 1 configuration consists of the



following:

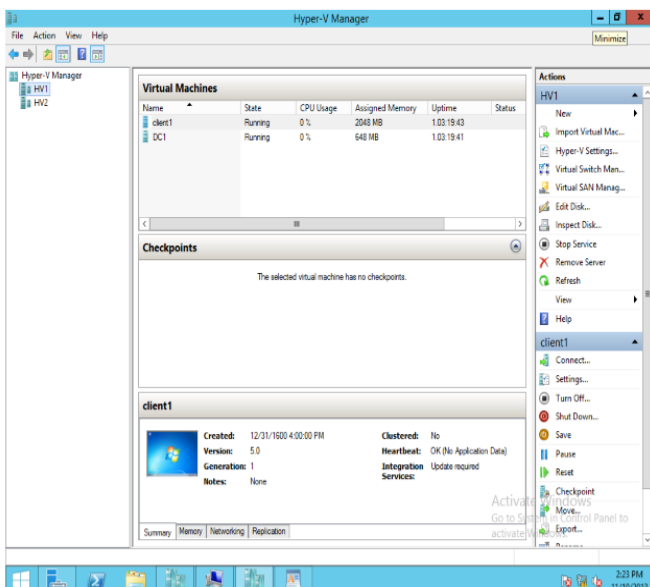
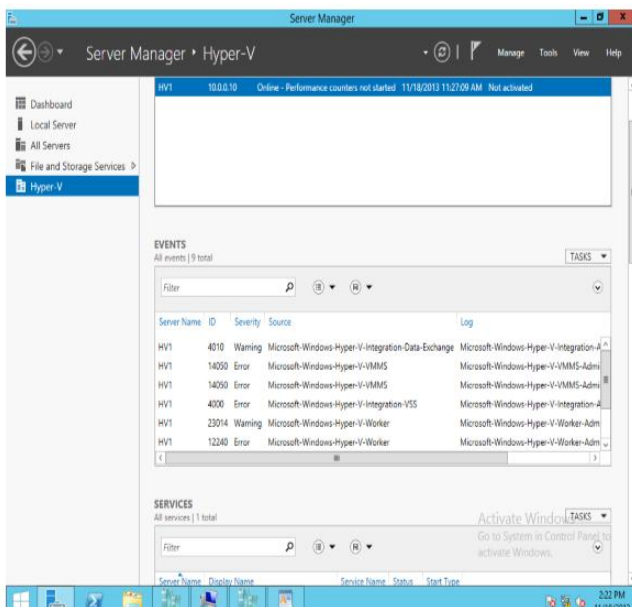
- Install the operating system on Domain Controller 1
- Configure TCP/IP
- Install Active Directory and DNS
- Create a user account in Active Directory

#### Step 4: Configure Application Server 1

Application Server 1 provides web and file sharing services.

Application Server 1 configuration consists of the following:

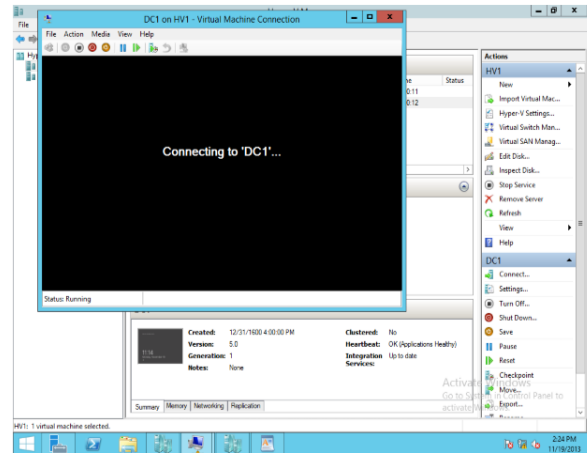
- Install the operating system.
- Configure TCP/IP.
- Join the computer to the domain.
- Install the Web Server (IIS) role.
- Create a shared folder.



#### Step 4: Configure CLIENT1

CLIENT1 configuration consists of the following:

- Install the operating system
- Join CLIENT1 to the PRIVATECLOUD domain



#### Step 5: Join HV1 to the privatecloud.com domain

#### Step 6: Join HV2 to the privatecloud.com domain

### 2.5 Delivering IT as a Service

All the processes involved in delivering IT as a service, can really be categorized into three areas:

- Application management: Deploying and operating our business applications.
- Service delivery and automation: Standardizing and automating service and resource provisioning, and managing change and access controls.
- Infrastructure management: Deploying and operating the entire underlying infrastructure on which our business applications and services run.

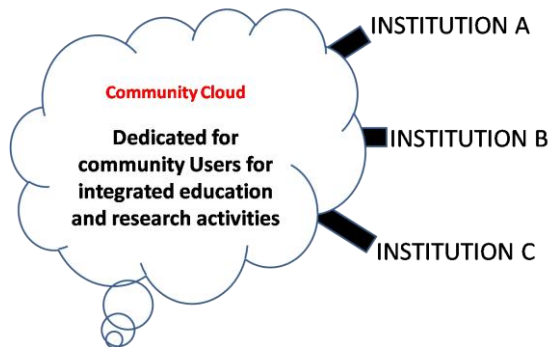
Windows system center 2012 following components are capable to deliver IT as a Service like App Controller would offer that self-service experience that allows your application owners to manage their apps across private and public environments. Service Manager offers the standardized self-service catalog that defines “templates” for your applications and infrastructure. App Controller, Virtual Machine Manager, Service Manager, and Operations Manager work together to maintain the service model through the application service life cycle. Orchestrator and Service Manager offer orchestrated automation for the process workflows required to drive your provisioning and monitoring tools. Virtual Machine Manager and Configuration Manager can provision physical, virtual, and cloud environments. Operations Manager (AVI code capabilities will be built into Operations Manager) monitors your application services end to end and offers deep app insight to help you deliver predictable SLA.

Open Stack and Open Nebula are prominent open source cloud applications for providing infrastructure as a service. Cloud Foundry is an open platform as a service, providing a choice of clouds, developer frameworks and application services. Cloud Foundry makes it faster and easier to build, test, deploy and scale applications. Cloud Platform is the

application-centric, cloud orchestration platform that enables you to quickly and efficiently build a cloud.

### 3. ROLE OF COMMUNITY CLOUD COMPUTING IN EDUCATION AND RESEARCH

#### Cloud Deployment Model – Community Cloud



To make computing resources availability more economic for education and research activities, community cloud framework is proposed. In this system massive virtualization of distributed infrastructure of all the educational institutes can lead to optimal utilization of resources.

- Cloud infrastructure will be shared by several educational organisations and will support the concern of higher education and research
- Shared concerns could be mission, information technology resources requirements, integrated research resources, and technology enabled teaching and instructional methodologies.
- It will reduce the overhead expenditure of managing the IT infrastructure.
- This is economic than private cloud and offer a higher level of privacy, security and policy compliance.

Community cloud computing for education and research is next economic platform for delivering Information Technology as a service for the growth of research activities. In future work we can go into details of details and designing community cloud services and its challenges. This work will help supporting and enhancing quality of education and research especially in developing countries. It will change the capital expenditure requirement of various newly established institutes to operational expenditure for simply hiring the resources from community cloud services and still providing all the Information Technology facilities to students for research and education.

### 4. REFERENCES

[1] An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.* 3 (Mar. 2003), 1289-1305.

[2] Brown, L. D., Hua, H., and Gao, C. 2003. A widget framework for augmented interaction in SCAPE.

[3] Y.T. Yu, M.F. Lau, "A comparison of MC/DC, MUMCUT and several other coverage criteria for logical

decisions", *Journal of Systems and Software*, 2005, in press.

[4] <http://social.technet.microsoft.com/wiki/contents/articles/7807.windows-server-2012-test-lab-guides.aspx>.

[5] Spector, A. Z. 1989. Achieving application requirements. In *Distributed Systems*, S. Mullender.

[6] Khajeh-Hosseini, Ali, et al. "The cloud adoption toolkit: supporting cloud adoption decisions in the enterprise." *Software: Practice and Experience* 42.4 (2012): 447-465.

[7] Frey, Sören, Wilhelm Hasselbring, and Benjamin Schnoor. "Automatic conformance checking for migrating software systems to cloud infrastructures and platforms." *Journal of Software: Evolution and Process* 25.10 (2013): 1089-1115.

[8] David Chappell, "THE MICROSOFT PRIVATE CLOUD A TECHNOLOGY OVERVIEW" 2011.

[9] Akintomide, O. A. "Cloud computing: The third revolution in IT." *Library Progress (International)* 33.1 (2013): 77-94.

[10] Mell, Peter, and Timothy Grance. "The NIST definition of cloud computing (draft)." *NIST special publication* 800.145 (2011): 7.

[11] Pearson, Siani, Yun Shen, and Miranda Mowbray. "A privacy manager for cloud computing." *Cloud Computing*. Springer Berlin Heidelberg, 2009. 90-106.

[12] T OGRAPH, B., and Y. RICHARD MORGENS. "Cloud computing." *Communications of the ACM* 51.7 (2008).

[13] Velte, Toby, Anthony Velte, and Robert Elsenpeter. *Cloud computing, a practical approach*. McGraw-Hill, Inc., 2009.

[14] Marinos, Alexandros, and Gerard Briscoe. "Community cloud computing." *Cloud Computing*. Springer Berlin Heidelberg, 2009. 472-484.

[15] Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." *Journal of internet services and applications* 1.1 (2010): 7-18.

[16] <http://technet.microsoft.com/en-us/cloud/hh147296.aspx>.

[17] <http://technet.microsoft.com/en-us/edge/ff945081>.

[18] Zhang, Liang-Jie, and Qun Zhou. "CCOA: Cloud computing open architecture." *Web Services, 2009. ICWS 2009. IEEE International Conference on. Ieee*, 2009.

[19] Iosup, Alexandru, et al. "Performance analysis of cloud computing services for many-tasks scientific computing." *Parallel and Distributed Systems, IEEE Transactions on* 22.6 (2011): 931-945.

[http://blogs.technet.com/b/keithmayer/archive/2013/04/01/build-your-private-cloud-in-a-month-new-article-series.aspx#.UxUgP\\_ldXso](http://blogs.technet.com/b/keithmayer/archive/2013/04/01/build-your-private-cloud-in-a-month-new-article-series.aspx#.UxUgP_ldXso).