The Implementation of Virtual Networking to Enhance Performance and Reduce Network Load in an Autonomous Systems Employing Virtual Computing

Corey Hemminger
Information Systems Department
Saint Cloud State University
Saint Cloud, MN 56301
heco0701@stcloudstate.edu

Martin Smith
Computer Networking Department
Saint Cloud State University
Saint Cloud, MN 56301
smma0901@stcloudstate.edu

Nicole Gillespie
English Department
Saint Cloud State University
Saint Cloud, MN 56301
gini0604@stcloudstate.edu

Dennis Guster
Information Systems Department
Saint Cloud State University
Saint Cloud, MN 56301
dcguster@stcloudstate.edu

Abstract

The concept of virtualized networking is nothing new in a world constantly advancing its technology. However, with a push for a greener environment and the advent of hardware architectures, such as grids and cluster, has made available substantial processing power typically in a single location. The virtualized hosts discussed in this paper are part of an autonomous system (AS) used to support research and instruction. It unquestionably can save energy resources, which can help the environment and improve efficiency in the data center. At the same time, virtualization reduces IT costs of hardware and personnel. For example, with an average 100 watts per hour, this would be a savings of 800 watts per hour or 19.2 kilowatts per day. At 10 cents a kilowatt per hour, that is over $700 dollars per year. While much of the attention is aimed at computing resources, this paper also explored expansion of virtualization concepts as well as leverage networking resources.

1 Introduction

Although the concept of virtualization within networking has been around for some time, it is the recent push toward green computing that has accelerated its use [2]. While virtualized hosts are useful for easing management cost, simplifying the enterprise, reducing hardware costs, reducing power consumption and cooling costs, the architecture of virtualized hosts also fits well into a companies' disaster recovery strategy [6]. However, often when virtualized hosts have been deployed, little thought was put into the networking strategy used. In fact, many times it was merely a logical replacement for a physical entity, and the majority of the time, the same networking scheme was used as before when all of the hosts resided in their own physical machine. Similar to virtual hosting, virtual networking has been available for quite some time [1]. Nonetheless, it is the recent promise of virtualization and the application of this logic within the framework of cloud computing, that have lead to rethinking how virtual networks might be used to increase network performance and security within an enterprise.

The advent of hardware architectures, such as grids and cluster, has made available substantial processing power typically in a single location. Thus, tuning a local area network to take advantage of that processing power might be viewed as a first step and supports the intensive virtualization of resources inherent in cloud computing. Additionally, grids and clusters can be linked together via wide-area links. In either case, the network logic can be tuned for performance and/or security purposes [8]. A simple way of looking at cloud computing is by combining all of a computer's resources into a pool, and then dynamically allocating those resources as needed. In this model, one expects that not all applications will be busy at the same time, hence, the ability to leverage resources. In other words, a processor and pool of memory being used intensely at one point by one application can be reallocated to another application later on, when the first application's intensity lessens. Further, this dynamic strategy can also add reliability because it is easy to invoke replication of hosts at various locations within the cloud. Obviously, this dynamic replication strategy requires inter-processor and across-node communication. Therefore, it is critical to use the available network resources prudently. Houidi, Wajdi, Djamal, Panagiotis, and Mathy [4] address this concern in the literature in an article related to network provisioning. To address this problem the authors suggest a distributed fault-tolerant embedding algorithm to optimize the performance and reliability profile of a virtualized enterprise. This concept is expanded upon in regard to network optimization in the work of Yu, Yi, Rexford, and Chuang [9]. Specifically, they advocate allowing multiple substrate paths and employing path migration to help deal with the dynamic workload nature one might expect in a "cloud" setting.

While the promise of cloud computing is immense, leveraging resources increases traffic and the diversity of that traffic providing at least theoretically additional attack scenarios. Therefore, it is critical to address the vulnerabilities from such scenarios. Keller, Lee, and Rexford [5] summarize virtualization as enabling multiple networks, each customized for a particular purpose to run concurrently over a shared substrate. The authors further state if security in such virtualized networks is to be maintained, policy must be devised and the traffic patterns monitored to access compliance. Furthermore, an automation strategy should be devised so as to drastically limit non-compliant activity in the first place.

It is the purpose of this paper to begin an analysis with one of the basic building blocks of a cloud: a virtualized host with multiple cores. The authors have used this technology for a number of years primarily to promote green computing, reduce personnel costs, and as an inexpensive means to achieve a viable disaster recovery plan [3]. However, when they virtualized their 10 physical production hosts into 10 virtual zones within a single physical computer, each zone was still treated the same way from a networking perspective. In other words, each zone was afforded a network connection to the switch, in spite of much of the communication that took place was among those 10 production hosts. Once virtualized, those 10 newly virtualized hosts all share the same motherboard and could communicate among themselves through a virtual network physically hosted on the motherboard. The analysis will begin with a description of the virtualization strategy, and a discussion of the network traffic profile will follow. Data was collected from the current switched environment, which will be analyzed in terms of intensity and distribution. The authors are currently converting the production host to a cloud configuration, and as part of that strategy the plan is to virtualize network traffic from the 10 logical hosts within the physical host.

2 Characteristics of the Virtualized Hosts

The virtualized hosts discussed in this paper are part of an autonomous system (AS) used to support research and instruction. However, because many of the clients are from other countries or spend significant time oversees, the system is designed to mirror the functionality of a corporation that has global presence. To support that global presence a distributed architecture and robust Internet connectivity is required. In this type of setting there are numerous physical production hosts that serve a variety of needs. If the traditional physical model is followed each service or application is housed in separate physical computers so isolation is ensured for security purposes.

The prime assumption in the design described below is that in all cases, the performance of the separate physical host model was acceptable, and unused computing cycles could be shared among all hosts. As a result, if this principle holds true, then one could expect the separate physical production hosts to be effectively converted to a virtualized host design. From a security perspective, the resulting design would ensure the virtual partitions would provide the required isolation while the performance would still be acceptable. Therefore, it is assumed that because all of the original physical hosts used only a fraction of their computing resources, they would function effectively in a shared virtual host [7]. Specifically, in this case study, 9 physical hosts are restructured into virtual zones within a single physical host. (Note that this requires an additional zone for the hypervisor, which in effect is the root management entity for the entire physical node, hence, resulting in 10 zones total). One might expect in this arrangement that each virtual host would be limited to about 1/9 of the available computing resources.

Nevertheless, modern operating systems can be configured to allocate resources dynamically, which means that the total available resources can be viewed as a pool. In effect, any unused resources could then be reallocated to any virtual partition. This dynamic allocation methodology can therefore be expected to work fairly efficiently as long as there are several lightly loaded virtual machines. In rare cases in which intense workload is distributed across all 9 virtual partitions, performance could fall rapidly. To combat this, a priority scheme can be used to control resource allocation. The wide spread use of multiple cored processors has also made it easy to alleviate this problem. The purpose of each of the 9 original physical hosts providing IT infrastructure in the authors' autonomous system is described below in Table 1. Each partition corresponds to a separate physical host.

Partition number	Description
Host	Time Server, Virtual Machine host (Running LINUX),
1	Main Client Access Server. SSH login server (Firewall will forward all port22 traffic to this partition)
2	Secondary Client Access Server, SSH login server as backup if partition 1 locks up (Firewall will forward alternate port to this partition, port22)
3	Global Authentication Server (OpenLDAP/Kerberos5-MIT)
4	Network Address Resolution Server (DNS/DHCP/LDAP)
5	E-Mail Server (LINUX Mail Server installed)
6	Web Server (Apache Tomcat)
7	Global File System Server for Home Folders (NFS Mounted)
8	Programming Application Development Server (java, C, C++ and etc.)
9	Data-Base Application Server (mySQL, SQL queries, DB tuning and etc.)

Table 1 Virtual Host Partitions (zones)

3 Characteristics of the Network Traffic

As one might expect on any network supporting an autonomous system, there are vast variances in workload. In the system studied, which is used primarily for instruction and research, there are times when the system is fairly loaded such as the last Friday of the semester before projects are due at 1pm. Also, when there is a deadline to get a paper submitted for a conference, one can expect a last minute surge in usage. Tables 2 and 3 below provide a snapshot of such traffic variation. Table 2 provides the statistics for interface 1/7 (card 1 port 7) and Table 3 provides the data for interface 2/11 (card 2 port 11). These are the two ports to which the virtual hosts connected. They are purposely separated on different cards within the open architecture switch for reliability and load balancing.

	Daily		Weekly		Monthly		Yearly	
	Max	Avg	Max	Avg	Max	Avg	Max	Avg
In	9.9	0.3	7.6	0.2	5.8	1.2	5.2	1.2
Out	106.0	2.9	85.2	1.4	35.8	3.0	9.5	2.9

Table 2: Traffic Analysis Interface 1 in KB

	Daily		Weekly		Monthly		Yearly	
	Max	Avg	Max	Avg	Max	Avg	Max	Avg
In	230.5	10.8	249.1	0.9	330.5	12.5	41.4	11.9
Out	7731.4	324.6	7799.7	158.3	3624.2	96.1	316.3	71.7

Table 3: Traffic Analysis Interface 2 in KB

There is clear variation across several levels within this traffic. First, each interface has dramatic differences in magnitude. Interface 2 has higher values particularly in the "out" and "max" categories. This in part can be explained by one of the services assigned to this interface, which is the back-up/replication service of the virtual host. Within the time trends, it is clear that the daily average is the high for both port sets, as a result of the autonomous system's continuous growth. This growth reverts back to the beginning of the semester, and is also supported by the weekly and monthly values depicted in Table 3. The yearly (projected) totals are significantly less, because they are influenced by summer and holiday periods when usage is at a bare minimum. Further, it is clear that the average load is minimal, but the max values are troubling at times. It is these values, coupled with the future expected growth trends, that make devising a plan to address network performance critical for the continuing success of the author' autonomous system.

To provide a better visual understanding of the variance, a series of figures are provided below. Once again, they are divided into two groups corresponding to either interface 1 or interface 2. The magnitude is a combination of both in and out traffic.

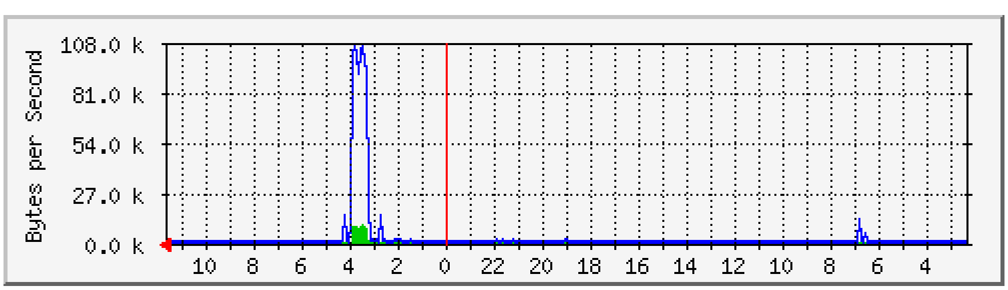

Figure 1: Interface 1 Hourly Traffic

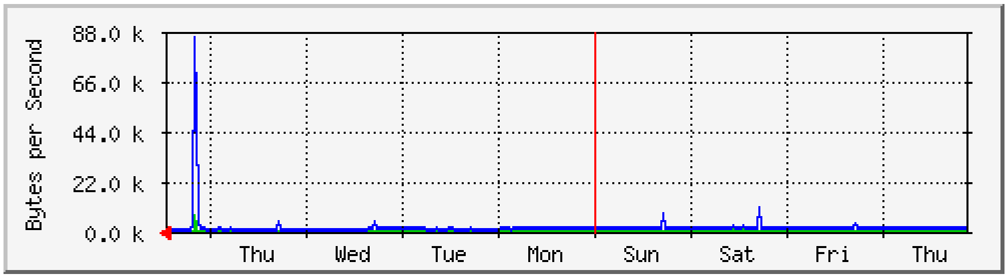

Figure 2: Interface 1 Daily Traffic

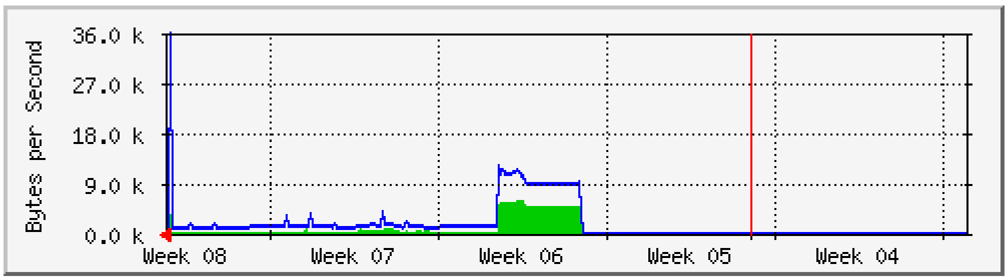

Figure 3: Interface 1 Weekly Traffic

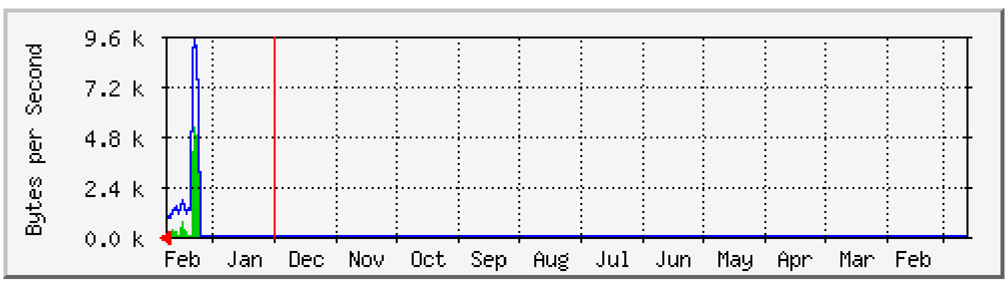

Figure 4: Interface 1 Yearly Traffic

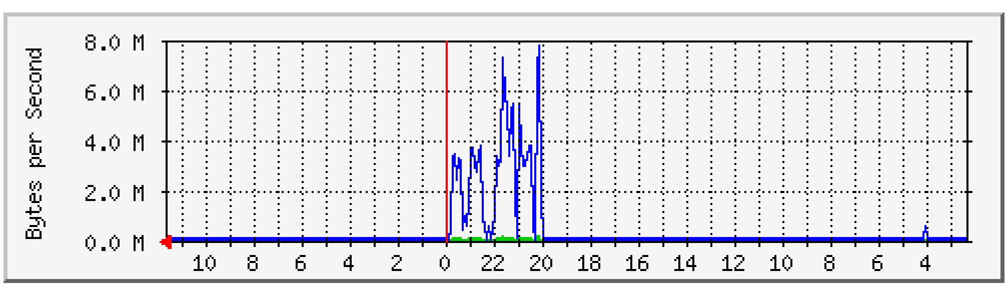

Figure 5: Interface 2 Hourly Traffic

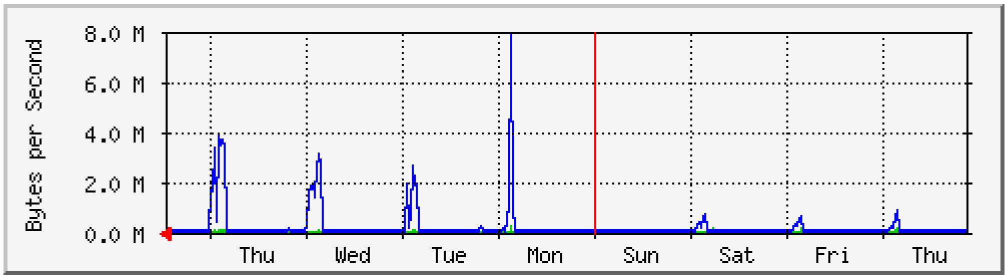

Figure 6: Interface 2 Daily Traffic

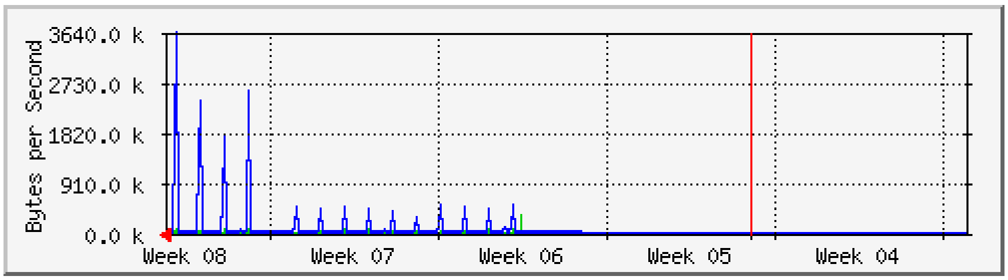

Figure 7: Interface 2 Weekly Traffic

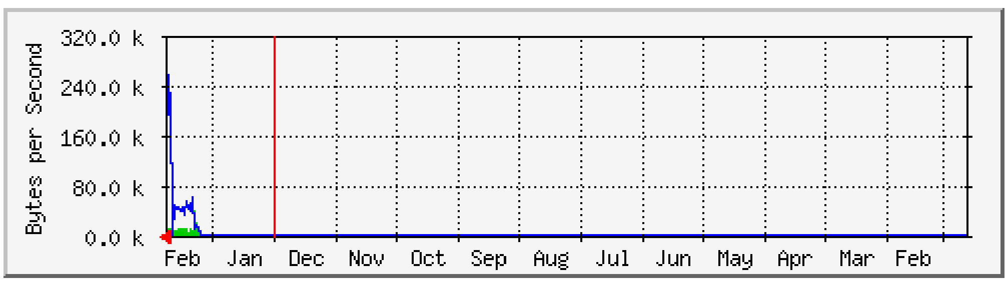

Figure 8: Interface 2 Yearly Traffic

One again, the traffic patterns in both cases reveal a great degree of variation. It appears to be more pronounced for “Interface 2 Port Traffic” (figures 5-8). Further, only about one month of data were collected. As a result, both cases display the yearly traffic as a projection based on that one month, so the validity of that project is a function of how representative that month is of the rest of the year.

4 Redesigning the Network to Take Advantage of Virtual Networking

The traffic data above was all collected on the switch level by utilizing a Force 10 switch with data analysis capabilities. From a topological perspective, each virtual host communicating with one another in the same physical host sent packets out to the switch, and the switch in-turn forwarded those packets back to the appropriate host. The data above indicates on average that this situation is currently acceptable from a performance perspective (all averages are less than 500KB/sec). However, periodically the spikes reach the 8MB/sec range, and the expected growth of the autonomous system—which triples annually—leads the authors to believe that they need to proactively plan for such growth in the future. Further, the autonomous system recently deployed a centralized logging facility in the form of a centralized SQL database. The plan is to create two uses of this database, one for system administration and one to support instruction for classes like Security Policy Analysis and Risk Assessment. Therefore, it is expected that these databases will place a greater load on the network infrastructure.

The concept of using a virtual network via the motherboard is well established after years in the loopback concepts. The high speed of the motherboard, the short travel distance, reliability, and large packet size (16436 versus just 1500 bytes for Ethernet) make it an excellent choice for communication within a physical host. The authors have already taken advantage of the architecture when cloned virtual zones are used to support instruction for operation systems and security classes. For example, in an introductory UNIX class it is important for students to gain basic experience in adding, deleting, and managing users. In the past, each student was given his or her own physical UNIX host. This was problematic in a number of areas including security, green computing, and the number of hours staff had to spend configuring and managing such systems. Recently, that same capability was provided through virtualization.

A large-scale multi-cored physical host was allocated, and a virtual zone was cloned for each student. Further, the process was made more efficient by linking to services with a standard TCP connection. For example, rather than configure each student zone with authentication information needed for part of the autonomous system, a link was provided to the system's global LDAP (lightweight directory access protocol) authentication system, see Table 4 below.

The same logic is applied on a global file level through the NFS (network file system) link, which means that the student's home directory from his or her basic UNIX shell account follows the student to his or her own virtual zone (and of course any other host in the AS they might use). The ssh (secure shell) connection allows the student to get to his or her virtual zone via a saver internal private network, using a general purpose instructional host machine as a relay. However, the link to the mysql database is interesting in that it is entirely established across the local host (127.0.0.1), which means that physically, it is the motherboard of the physical host. While the production mysql database is housed on a separate physical host to reduce traffic, it is possible to provide students with an instance of the database (scrubbed and trimmed) that will still be instructionally useful. For example, the table below displays this instructional usage through the same physical host and is accessible via a local host in an effort to reduce traffic on the LAN and within the switch structure. This same kind of logic will be applied throughout both the production and instructional areas of the authors' AS. Once that is complete data will be collected and compared to the existing data described herein to obtain an idea of to what degree performance might be improved by such a design.

```
tcp        0      0 vubuntu-3.local:39591  10.1.3.17:ldap          ESTABLISHED
tcp        0      0 localhost:45208        localhost:mysql         ESTABLISHED
tcp        0      0 vubuntu-3.local:ssh    10.0.2.18:40402         ESTABLISHED
tcp        0      0 localhost:mysql        localhost:45208         ESTABLISHED
tcp        0      0 vubuntu-3.local:933    10.1.7.5:nfs            ESTABLISHED
```

Table 4: TCP Links to Common Services

5 Discussion and Conclusions

Virtualization is widely accepted in recent years as a means of increasing efficiency in the data center. A trend towards green computing requires the data center to adopt more responsible designs in an effort to spare the environment. The concept of virtualization offers enormous benefits in regard to saving resources in IT operations. In this example, the number of computers required to run a production domain are reduced from 10 down to 1. Assuming the average power consumption per box at a conservative 100 watts per hour, this would be a savings of 800 watts per hour or 19.2 kilowatts per day. At 10 cents a kilowatt per hour, that is over $700 dollars per year. Further, this value could possibly be doubled when one considers the associated reduction in air conditioning.

From a cost perspective, it is perhaps the reduction in complexity that makes it easier for system/network staff to configure and maintain the system. Given that personnel is the highest cost within IT, this is a most positive development. It is also obvious that less hardware needs to be purchased to perform the same tasks, which means less money is needed up-front to upgrade existing equipment. Specifically, Lee, Guster, Schmidt, and McCann [7] found that in a case study involving a six-host virtualization disaster recovery, the virtual architecture could save $39,000 in personnel costs and over $100,000 in hardware cost when compared to a traditional hot disaster recovery model.

The concern as to whether the virtualized model would be able to offer the same performance as the separate physical hosts model is well founded. One can expect that the performance will be related to the workload distribution on each of the virtual machines. If they all are experiencing intense workloads at the same time, then inevitably performance will suffer drastically. However, if the analysis of the existing physical hosts' model reveals intense workload occurs when limited to a single host, then that would probably be acceptable. The same concern is also applicable to network traffic. If that traffic is evenly distributed, then the virtual zones should be able to communicate effectively. On the other hand, the density of the virtualization can become a concern because the number of hosts can grow extensively. The data herein show that on an average intensity level there is little concern, but the peaks are getting to the point where they are causing concern. That is why we are also suggesting a network virtualization. Determining the most efficient density for virtual zones and their related virtual networks is certainly an emerging topic. Accordingly, data collection and analysis offered in this paper will become crucial in evaluating future designs for specific application of virtualization. A simple example of determining the optimum number of zones that can be safely loaded into a physical host is related to the processing power required and the workload distribution each virtual partition will generate. This requires detailed analysis, but a general observation is that multi-cored 64 bit processors with large memory capacity can support 50 instances in our instructional setting. Obviously, in a production setting that value might be reduced; ours is currently around 10.

Our observations thus far have verified that the isolation provided by the partitioning strategy is effective for security purposes. While the process of converting all physical hosts in our AS to a virtual server solution has not yet been completed, the experience to date is entirely positive. Over the next five years, the plan is to continue to utilize and optimize the virtual model from both a CPU and network perspective. Because our available physical space is maxed out and we are limited to 400-amp electrical service, we must rely on virtualization for continual growth without having to request additional physical space and electrical power. This is important in today's economic times as such a request would be most difficult to have fulfilled.

In summary, it appears that virtualization predominantly fulfills much of its promise. It unquestionably can save energy resources, which can help the environment and improve efficiency in the data center. At the same time, virtualization reduces IT costs of hardware and personnel. While much of the attention is aimed at computing resources, this paper also explored expansion of virtualization concepts as well as leverage networking resources. Virtualization is thus, a topic worth investigating.

References

[1] Bush, S. (1999). Active virtual network management protocol. *Proceedings of the 13th Workshop on Parallel and Distributed Simulation*. IEEE Computer Society, Washington, DC.

[2] Guster, D. C., Hemminger, C., & Krzenski, S. (2009). Using virtualization to reduce data center infrastructure and promote green computing. *International Journal of Business Research, 9*(6), 133-139.

[3] Guster, Dennis C. & Olivia F. Lee. (2011). Effective infrastructure protection through virtualization. *ICT Ethics and Security in the 21st Century: New Developments and Applications*. IGI Global Publisher, Hersey, PA.

[4] Houidi, I., Wajdi, L., Djamal, Z., Panagiotis, P. & Mathy, L. (2010). Adaptive virtual network provisioning. *Proceedings of the 2nd ACM SIGCOMM Workshop on Virtualized Infrastructure, Systems and Architectures*, ACM, New York.

[5] Keller, E., Lee, R. & Rexford, J. (2009). Accountability in hosted virtual networks. *VISA, ACM Sigcomm workshop*, Barcelona, Spain, August 17.

[6] Lee, O. F. & Guster, D. C. (2010). Virtualized disaster recovery model for large-scale hospital and healthcare systems. *International Journal of Healthcare Information Systems and Informatics*, 5(3): 69-81.

[7] Lee, O. F., Guster, D. C., Schmidt, M. B., & McCann, B. (2009). Applying the scale-free degree distribution algorithm to assess communication complexity and failure points in disaster recovery models. *Journal of Information Technology Management*, 20(2).

[8] Sundararaj, A. & Dinda, P. (2004). Towards virtual networks for virtual machine grid computing. *Proceedings of the third USENIX Conference on Virtual Machine Technology* (VM 04), May.

[9] Yu, M., Yi, Y., Rexford, J & Chuang, M. (2008). Rethinking virtual network embedding: substrate support for path splitting and migration. *ACM SIGCOMM Computer Communication Review*, 38(2). http://www.cs.princeton.edu/~jrex/virtual.html