

AC 2010-268: ON THE USE OF VIRTUALIZATION FOR ROUTER NETWORK SIMULATION

Tianyi Li, Northwestern University

Tianyi Li is currently a graduate student pursuing his Masters degree in the Department of Electrical Engineering and Computer Science at Northwestern University. He received his two B.S. degrees in Telecommunications Engineering Technology and Communications Engineering from Southern Polytechnic State University and North China University of Technology in 2009. He is now in the division of Signals & Systems concentrating on networks, communication and control. His research interests include resource allocation for wireless networks, congestion control for TCP/IP networks, and computer networking.

Walter E. Thain Jr., Southern Polytechnic State University

Walter E. Thain received his BS, MS, and Ph.D. degrees in Electrical Engineering from the Georgia Institute of Technology. He is an Associate Professor in Electrical and Computer Engineering Technology at Southern Polytechnic State University and teaches courses in voice and data networking, communications systems, and analog and RF electronics. Research interests include voice and data network design and management, network security, RF communication systems, and digital signal processing. He spent 12 years in industry, where he designed mixed analog-digital systems, including, short-pulse radars and antennas, low-noise analog circuits, RF circuits, pulse generators, frequency synthesizers, switching power supplies, and high-speed digital circuits. He is co-inventor on a patent for the design of electronic instrumentation used to steer oil wells while drilling.

Thomas Fallon, Southern Polytechnic State University

Thomas J. Fallon received his BS and MS degrees in Electrical Engineering from the Georgia Institute of Technology and his Ph.D. degree in Astronomy from Georgia State University. He is an Associate Professor of Electrical and Computer Engineering Technology at Southern Polytechnic State University and is coordinator of the BSTCET program. He has 20+ years of telecommunications industry experience, conducts networking workshops, and is author of the book *The Internet Today*. His astronomy Ph.D. research at Georgia State University involved astronomical instrumentation including the design and implementation of a remote operation system for a telescope array controlled via the Internet.

On the Use of Virtualization for Router Network Simulation

Abstract

The availability of powerful simulators has enhanced the ability to optimize communication network performance. The more complex ones can simulate many different types of network devices but with a limited degree of accuracy for each device. The intent of these simulators is to accurately represent the overall performance of the network in terms of metrics such as throughput, latency, etc. Still other simulators are specifically designed to emulate a particular manufacturer's equipment, primarily for training purposes. This type can simulate the responses of the devices to a subset of configuration commands and can simulate some types of interaction between the manufacturer's devices. This paper discusses the evaluation of the open-source Dynamips/GNS3 simulator for Cisco routers undertaken to determine its suitability as a teaching tool. This simulator is unique in that it is virtualization-based. It emulates the hardware of a number of key Cisco routers enabling the virtual router to run the actual Internetwork Operating System (IOS) software. Originally designed as an aid for those wishing to study for Cisco certifications it is extremely attractive as an instructional tool because no actual hardware is required and the simulations can include some esoteric capabilities of the routers. Indeed, fairly complex router networks can be simulated and virtual routers on one computer can interact with virtual routers on another. A number of the important capabilities of Dynamips/GNS3 were successfully executed, but with some difficulties at times. Nevertheless, the virtualization approach is extremely promising, if feasible, for a given type of simulation.

Introduction

Employing simulations as a teaching aid is very useful, perhaps even more so in a distance learning environment. Southern Polytechnic State University's (SPSU) Electrical and Computer Engineering Technology (ECET) program is engaged in converting its curriculum to a distance or hybrid distance format. Since all but two of the ECET courses have a laboratory component, this creates a significant challenge in providing a similar laboratory experience for the off-campus students. An obvious choice is to require distance students to visit the campus periodically to perform hands-on lab exercises. This can be difficult for some students to do and will limit enrollment to students who can.

The laboratory for the department's Telecommunications Engineering Technology (TCET) courses use primarily Cisco routers and switches that students connect in various configurations and program to give them the desired functionality for a given exercise or project. While remote access to a router via the Internet is possible, it must be done in a secure manner, such as through a VPN connection. Because the laboratory supports several courses, instructors change device connections and configurations as needed, sometimes on a daily basis. This instability leads to difficulty in managing distance students' remote access to the lab.

An attractive alternative is the use of realistic simulations that give distance students a learning experience as close as possible to that of the on-campus students. Some important general criteria for a simulator are listed below.

- Realistic simulations
- Cost
- Licensing restrictions
- Ease of use
- Stable operation

The general criteria have different facets to evaluate. Regarding licensing, the length of the license, whether a site license is available, and whether machines running the software must contact a license server are just some of the issues. This paper discusses an investigation of an open-source router network simulator considered for use for TCET courses.

Types of Network Simulators

There are two main types of network simulators. One is more general, intended to model large-network performance where equipment from multiple vendors may be in use. These simulators are often used for network design and optimization. They model some specific vendor equipment, but the modeling is done at a high level, where detailed functionality is sacrificed. This enables fast simulations of large networks, but the user is not required to understand how to program the individual network devices. The device models are given parameters that govern their performance in the simulation. An example of such a simulator is OPNET¹. A reduced-capability version is available for educational institutions.

The second type of simulator is vendor-specific and is designed as a training aid for users to learn how to program and interconnect a vendor's devices. These simulators allow users to interact with virtual devices using configuration commands just as they would with the actual equipment. Depending on their sophistication, the simulators permit multiple devices to be interconnected and the resulting network simulated for functionality. These simulators are designed by the equipment vendors as well as third parties. Since they are usually for training purposes, the simulators are often tailored for specific certification levels.

Cisco receives considerable attention when it comes to individuals earning their certifications and many simulator vendors design products for Cisco certifications. The Cisco certifications are described on their website⁵. Certifications mentioned in this paper in order from lowest level to highest are CCNA (Cisco Certified Network Associate), CCNP (Cisco Certified Network Professional), and CCIE (Cisco Certified Internetwork Expert). For example, commercial third-party Cisco simulator products are available from Boson², SemSim³, and RouterSim⁴. Different versions of each simulator are designed for the different Cisco certification levels such as CCNA and CCNP. This means that the simulators will simulate specific Cisco IOS (Internet Operating System) commands for the routers and switches making up networks of sufficient sophistication needed to give users practice learning principles needed for the certification level for which the simulator is tailored. Licensing is typically on a per-seat basis for these simulators with costs of about \$40 to \$500, depending on the level of sophistication. The higher level certifications require more sophisticated simulators.

Cisco has its own Packet Tracer⁶ simulator for use in its Cisco Networking Academy program, which is designed to prepare students for earning the CCNA certification. It is available free of charge for educational institutions that run Cisco Networking Academies as well as alumni of such programs. It is not available as a separate product.

Another Cisco-specific simulator is the open-source GNS3⁷ (Graphical Network Simulator 3), which is the main subject of this paper. It actually consists of three software programs, each developed at different times and by different teams. The programs can be run on Microsoft Windows and Linux operating systems. Dynamips⁸ is the core software program and main simulation engine. Dynamips is unique in that it simulates Cisco routers through emulation. It is designed as a virtualization hypervisor and can emulate multiple instances of router devices. Each router runs the actual router IOS software image for the router being emulated. Therefore, it can emulate most of the commands and functions available in the IOS image. Dynagen⁹ was developed as an overlay to Dynamips and provides simplified control of the simulation parameters for Dynamips through a management console. Although not a subject of this paper, Dynagen also has the capability of simulating Cisco PIX firewall devices as an overlay for a PIX emulator developed by another team. Finally, GNS3 was developed as a graphical interface to Dynagen and Dynamips, and can be considered as an overlay of Dynagen. For the purposes of this paper, reference to GNS3 is considered to refer to the combination of the capabilities of GNS3, Dynamips, and Dynagen.

GNS3 is capable of emulating Cisco routers in the 7200, 3700, 3600, 2600, and 1700 families. As such, it has received attention from those seeking Cisco certifications, including CCNA, CCNP, and CCIE. Because GNS3 is an emulator and its router emulation capability means that it can simulate most of the functions of the particular IOS image, giving it greater flexibility than the other vendor-specific Cisco simulators mentioned above. It is this flexibility, and the graphical interface that make GNS3 attractive as a simulator for distance-learning applications. Exercises and projects can be readily tailored to illustrate many different concepts without restrictions imposed by simulators designed for particular certification levels.

GNS3 is very attractive, as the primary goal was to have students work with actual Cisco routers and switches. Therefore, students must be able to program the devices and observe their operation. Also, the ECET department does not teach Cisco certification courses and is not part of a Cisco Networking Academy, precluding access to Packet Tracer. Finally, there is no cost nor are there any complex licensing concerns with GNS3, Dynamips, and Dynagen.

GNS3 Capabilities

Since Dynamips is an emulator, it requires an actual IOS image which the user must provide. IOS images are the software running on Cisco router platforms, and therefore proprietary and subject to license restrictions even though GNS3, Dynagen, and Dynamips are open-source. IOS images are not available from the GNS3-related web sites. Because the TCET lab has Cisco 2621 and 2621XM routers, the IOS images for these routers were available for experimentation.

GNS3 only truly emulates routers, not switches. Generic LAN switches, frame relay switches, and ATM switches are available for network simulations, but their functionality is only basic.

However, as discussed below, Cisco routers can be configured as frame relay switches. The 2600 series router interfaces that can be emulated are the console (management interface), 10/100 Ethernet, and serial. Depending on the router platform, these interfaces are allocated as built-in or part of a network module or wide area network interface module. The serial interfaces provide the “wide area” network connections between the routers and switches.

The Dynamips hypervisor has some powerful capabilities designed into it. One is that the hypervisor on one computer can interconnect with the hypervisor on another through a TCP/IP connection, enabling a simulation to extend over multiple computers. This allows larger networks to be simulated than would be possible on a single computer due to processor and memory limitations. Also, computers external to the simulation computer can participate as hosts in the simulation through Ethernet connections to the simulated router network. Another very useful feature is the ability to examine traffic on the Ethernet and serial links using the open-source Wireshark¹⁰ protocol analyzer. Wireshark is already used in the TCET lab for monitoring Ethernet traffic. Note that examining traffic on actual hardware serial links is beyond the capability of equipment in many instructional laboratories, so the ability to observe simulated serial link traffic can be very useful.

In order to perform its emulation, the Dynamips hypervisor allocates RAM space on the host computer corresponding to each router’s RAM and ROM space and emulates a separate virtual machine for each router. As more routers are added to the simulation, more host RAM is required. A typical allocation for one 2600 router IOS is about 128MB for the router’s RAM, 16MB for each of the two PCMCIA flash memory slots, and 128kB for the NVRAM space. GNS3 will save the flash memory and NVRAM contents for each router on the computer’s hard disk. These values are user-selectable and should be at least the values specified for a given router’s IOS.

When the hypervisor initiates an instance of a router virtual machine, it will use as much of the simulation computer’s processor power as it can. Multiple virtual machines will compete for processor power and there is no priority scheme. This will result in the hypervisor using almost 100% of the simulation computer’s processor resource even to emulate a single router because it is emulating the router’s processor on the virtual machine. Most of the simulation computer’s processor resources are actually utilized by the virtual router’s idle routines. This idle process can be identified when it is entered by an IdlePC utility available in GNS3. After identification, the user can force the virtual machines to discontinue direct emulation of the idle process of each router, greatly reducing the simulation computer’s processor utilization. Still, adding more routers to a simulation, even when emulating the idle processes is eliminated, results in more and more processor utilization. Even though GNS3 is emulating a Cisco router, it cannot do so at real-time speeds. This means that network loading issues will not be accurately simulated.

Documentation is provided on the GNS3⁷ and Dynagen⁹ web sites in the form of static documents and video tutorials. Also, the GNS3 project page¹¹ on the SourceForge.net website has all present and past versions of the software and documentation. Nevertheless, the level of reliability of all three software packages, GNS3, Dynagen, and Dynamips is beta at best. Users will inevitably encounter interesting simulation runtime problems for which it is difficult to determine the cause. There is a forum and blog on the GNS3 websites as well as forums

elsewhere that can be readily found via an Internet search engine. These sites are a good resource for identifying and solving problems.

As a way of illustrating GNS3 operation, the following section describes an example of a small network simulation of 2600 series routers.

GNS3 Example – Frame Relay Network

Simulations were performed using GNS3 version 0.6 installed from an “all-in-one” package downloaded from the GNS3 website. The simulator was installed on two different laptop computers, each running the Microsoft Windows Vista operating system. One computer had 2GB RAM and a 2.0GHz Intel Core™2 Duo T7200 processor and the other had 4GB RAM and a 2.4GHz Intel Core™2 Duo T8300 processor. The compressed image file used for the Cisco 2600 routers was c2600-is-mz.122-8.T10.bin, which was a basic IOS image. Memory allocations for each router instance were 128MB RAM, 128kB NVRAM, and 32MB for each of the two flash memory cards.

The simulation example described here consisted of three 2621 routers connected to each other via a fourth router acting as a frame relay switch. The network was the same as one implemented with actual routers in the TCET lab as a lab exercise. Routers were connected to the frame switch using serial links operating with 1200 kHz clock speeds. The low clock rate was chosen to reduce the probability of simulation difficulties.

Fig. 1 shows a screenshot of the GNS3 interface with the frame relay network. The middle panel is the schematic window, the left “Nodes Types” panel has icons of devices available for the simulation, the right “Topology Summary” panel lists the devices in the schematic and their interconnections, and the bottom “Console” panel is a direct interface to the Dynagen management console for Dynamips.

Most details of the GNS3 configuration are omitted here, but are available in the software documentation. Configuration details for the routers are also omitted. A brief overview of the simulation follows.

After GNS3 is started, the user must specify the location of the working directory and IOS file locations on the simulation computer in the GNS3 preferences dialog. Project files will be stored in the working directory. When each router icon is dragged from the Nodes Types panel onto the schematic, the user must specify the RAM, NVRAM, and flash memory sizes, as well as which interfaces to include in the router. Interconnects are drawn using the “add a link” tool on the toolbar.

When the simulation is started, the virtual routers boot their IOS images, load their configuration files from the working directory, and begin operation. Users can interact and configure each router through its virtual console interface just as they would with real routers. The configuration files can be saved automatically or manually in the project working directory folder. As the simulation runs, users can issue the usual IOS show commands for monitoring router activity.

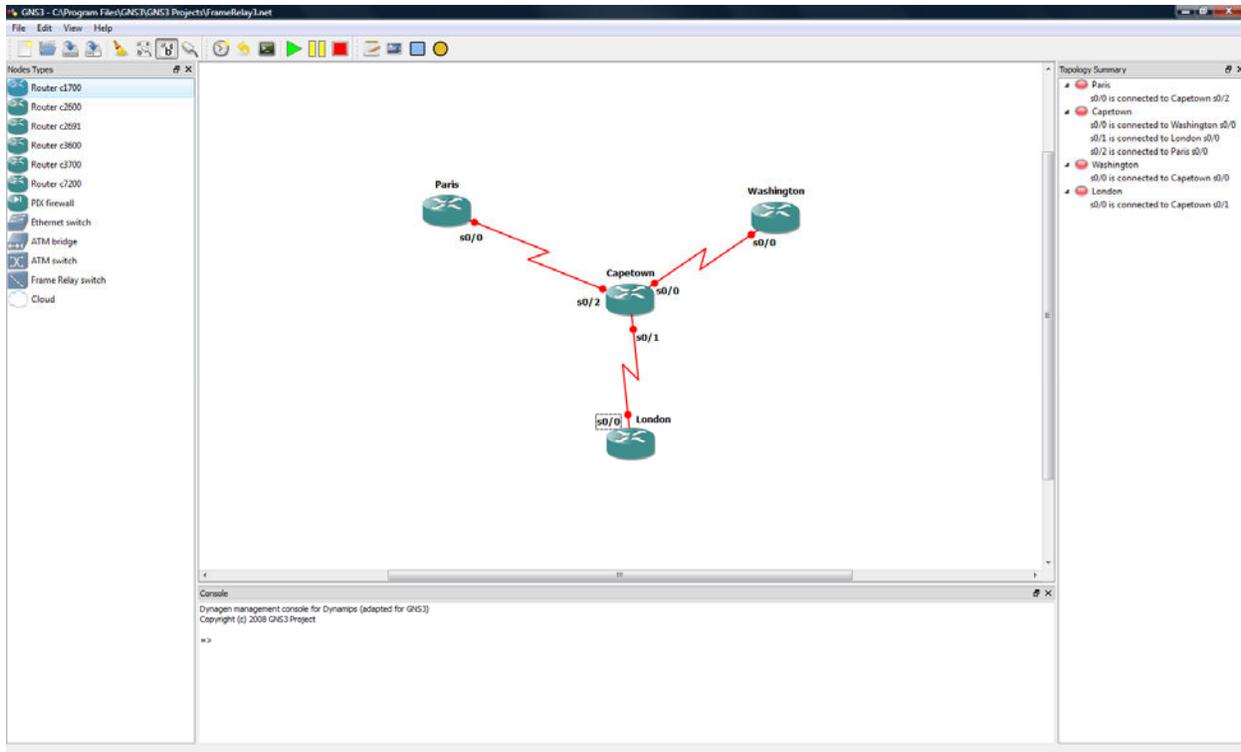


Fig. 1. GNS3 interface showing the frame relay network.

Fig. 2 shows captured traffic at the serial interface S0/0 of the Paris router. The router serial (frame relay WAN) interfaces are given 10.x.y.z IP addresses and the router Fast Ethernet addresses all have 172.a.b.c addresses. The captured traffic shows Cisco Discovery Protocol (CDP) packets, RIP routing protocol packets as well as packets generated by pinging a Fast Ethernet interface on the Paris router from the London router. The TCET instructional laboratory does not have equipment capable of capturing traffic on real router serial links, so an accurate simulator such as GNS3 provides the capability of observing this traffic.

Getting this simulation set up properly with respect to GNS3 and with respect to router configurations took more than a few iterations. It was known to operate properly in hardware, so the router configurations were basically correct. Modifications such as lowering the serial link clock rate to 1200 bps were done to ease the load on the emulator. Sometimes it took the simulation time to stabilize such that the routing tables were correct and the serial link line protocols all remained up. Ending all unnecessary programs and processes running on the host computer resulted in more consistent simulations.

This simulation used a maximum of about 60% of the processor resources on the CoreTM2 Duo T8300 host computer. The IdlePC values had been selected to reduce host computer processor utilization by the hypervisor due to emulated router idle routines. Otherwise, the simulation would have used 100% of the processor resources and still run slowly. Fig. 3 shows the effect of a properly selected IdlePC value on processor resources for a different simulation than the one discussed above. The CPU Usage History shows each of the CoreTM2 Duo processor cores under full utilization when the simulated routers are started. As the routers finish booting and much of

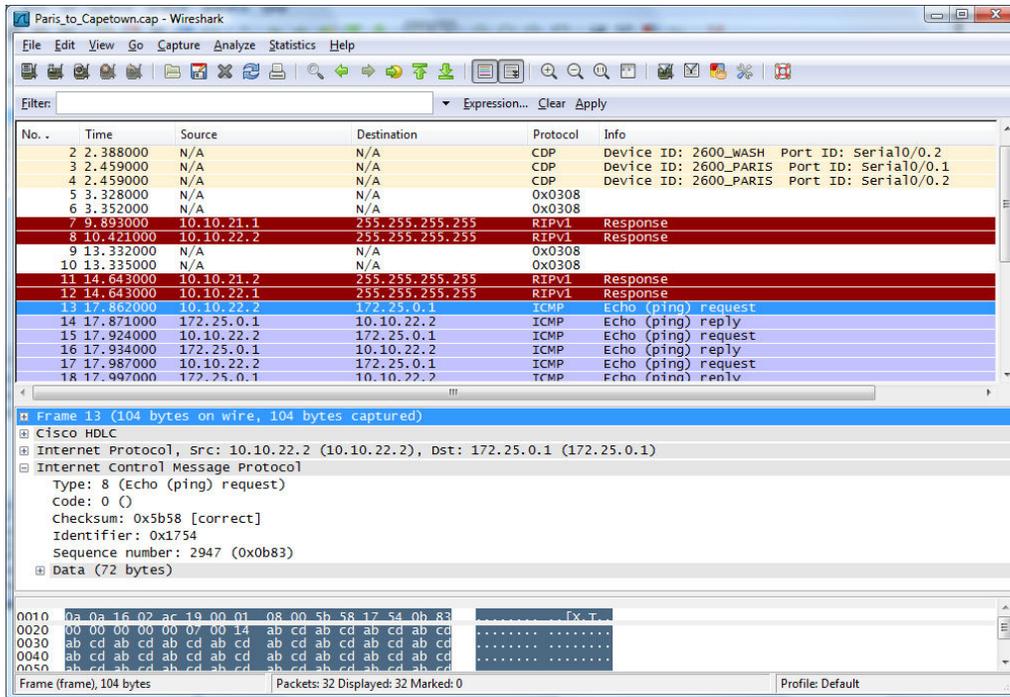


Fig. 2. Captured traffic on serial link between Paris and Capetown.

their activity is in the idle mode, the hypervisor stops emulating the idle routines, causing the processor utilization to drop.

Other simulations were performed to examine GNS3 capabilities. One was a modification of the frame relay simulation above that included another 2621 router that was connected via a frame relay NNI serial link to the first frame relay switch. Fig. 4 shows this arrangement with the frame relay DLCIs on each serial link labeled. This was a “what if” scenario since such a network

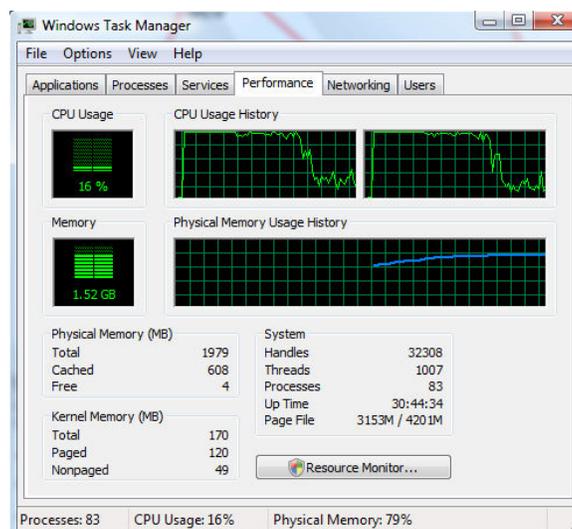


Fig. 3. Effect of properly selected IdlePC on processor resources.

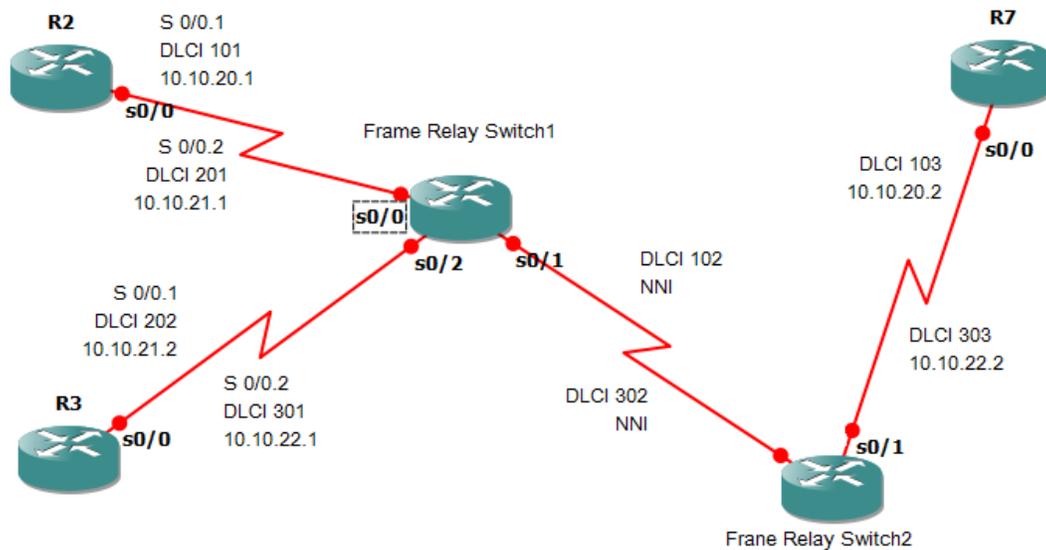


Fig. 4. Frame relay network with two frame switches and NNI connection.

configuration had not been used in hardware lab exercises before and was much faster in implementation since physical connections between routers did not have to be made. Also the GNS3 hypervisor capability of connecting a virtual router to an external real computer through the host computer's Ethernet interface was successfully tested.

Note that for instructional purposes, the GNS3 working directory files and router configuration files having various degrees of completeness can be made available to students running their own instance of GNS3 so that they can execute the proper steps necessary to complete the device simulations. Providing a basic configuration to students initially will allow them to avoid common and frustrating mistakes that are hard to isolate without an intimate knowledge of GNS3 operation.

Additional Remarks and Conclusions

Revisiting the five criteria for consideration when choosing a simulator, the unique GNS3 virtual router emulation capability provides a flexible and accurate simulation platform that can be used as an effective teaching tool for students, both on- and off-campus. It can be used to simulate more complex networks than those possible using the hardware in many instructional laboratories. It also allows students to spend more time working with a virtual network than they usually can in an instructional lab working with a hardware network.

The GNS3 software is open-source under the GNU General Public License and it has no cost. However, the user must provide an actual Cisco IOS image, which means that it must be obtained from Cisco directly or from a Cisco router. The particular utilization of a Cisco router image file by Dynamips in its emulation mode is an unusual application and may cause unforeseen legal complications. Some discussions of this issue have appeared on various Internet forums¹².

The GNS3 interface is simple and relatively easy to use, much easier than the text-based Dynagen interface. However, at times working with the GNS3 software can be frustrating. When using it for the first time, having a good background in configuring Cisco routers is important so that simulation difficulties can be more easily identified as being associated with the simulator or the router(s). Since all three components, GNS3, Dynagen, and Dynamips are essentially beta-quality software and interdependent, identifying simulation-software related problems is not easy. A number of simulations have simply terminated after starting successfully, the cause undetermined. Therefore, instructors must choose simulation topologies that are proven or based on proven simulations to avoid unnecessary hardship on the students' part.

The development teams for GNS3, Dynagen, and Dynamips are small, and therefore unable to respond quickly with software updates. There is no guarantee that future updates will be available. Indeed, Dynamips emulates the particular processor hardware that is common between the 1700, 2600, 3600, 3700, and 7200 series Cisco routers. As they become obsolete and no longer used in Cisco certification training courses, the existing Dynamips emulator and hypervisor will also be obsolete. If Dynamips is not updated to support the newer Cisco routers, the lifetime of the GNS3 will be limited to not much more than a few years from now.

Experimentation showed that having sufficient host computer resources is important for successful simulations. Therefore, having a fast processor and sufficient memory are essential. GNS3, Dynagen, and Dynamips have versions compatible with Linux, which may prove to be a better host operating system than Microsoft Windows. However a Linux implementation was not attempted during this research but will be if it is decided to continue working with GNS3. Also, it is possible in Microsoft Windows XP and Vista to change the priority of a process through Windows Task Manager. Elevating the priority of the Dynamips process once it is running may be helpful. Certainly terminating any unnecessary programs and processes is beneficial.

The value of a simulator based on emulation of router hardware enabling the use of actual router software cannot be underestimated. This means that the emulated routers will correctly respond to commands and not be subject to bugs that can affect the router simulators discussed earlier. Furthermore, emulation enables router virtualization such that multiple virtual machines can be implemented on a single host. The benefits of virtualization in server farms and for users desiring to run multiple operating systems on their computers are resulting in the growth of virtualization applications such as VirtualBox¹³ and VMware¹⁴.

The development of Dynamips, Dynagen, and GNS3 is an important step in the evolution of network simulators and deserves a try by instructors in a similar situation as those in SPSU's ECET department regarding distance learning course development. Even if Dynamips and GNS3 software development cannot be sustained by its open-source teams, perhaps Cisco Systems itself can use the emulation paradigm as a model for their own future simulator development. It would greatly lower the costs of teaching telecommunications laboratories using their equipment.

Bibliography

1. OPNET web site. <http://www.opnet.com/> (Last accessed January 2010).
2. Boson Software LLC. *NetSim Network Simulator*. <http://www.boson.com/AboutNetSim.html> (Last accessed January 2010).
3. SemSim. *Router Simulator Features*. <http://www.semsim.com/ccna-router-simulator-features.html> (Last accessed January 2010).
4. RouterSim. <http://www.routersim.com/default.aspx> (Last accessed January 2010).
5. Cisco Systems, Inc. *IT Certifications and Career Paths*. http://www.cisco.com/web/learning/le3/learning_career_certifications_and_learning_paths_home.html (Last accessed January 2010).
6. Cisco Systems, Inc. *Cisco Packet Tracer Data Sheet*. http://www.cisco.com/web/learning/netacad/course_catalog/docs/Cisco_PacketTracer_DS.pdf (Last accessed January 2010).
7. GNS3 web site. <http://www.gns3.net/> (Last accessed January 2010).
8. Dynamips web site. *Cisco 7200 Simulator*. http://www.ipflow.utc.fr/index.php/Cisco_7200_Simulator (Last accessed January 2010).
9. Dynagen web site. <http://www.dynagen.org/> (Last accessed January 2010).
10. Wireshark Foundation. *Wireshark User's Guide*. http://www.wireshark.org/docs/wsug_html_chunked/ (Last accessed January 2010).
11. SourceForge web site. *GNS3*. <http://sourceforge.net/projects/gns-3/files/> (Last accessed January 2010).
12. CCIE Pursuit Blog. *Cisco's Official Response Concerning Dynamips*. <http://cciepursuit.wordpress.com/2008/01/17/ciscos-official-response-concerning-dynamips/> (Last accessed January 2010).
13. Sun Microsystems, Inc. *Sun VirtualBox® User Manual*. <http://www.virtualbox.org/manual/UserManual.html> (Last accessed January 2010).
14. VMware website. <http://www.vmware.com/> (Last accessed January 2010).