# Hybrid At-Scale Network Experimentation Needs Cloud-Enabled HPC and Virtualization

Jason Liu, Florida International University

***Abstract*—PrimoGENI is a network testbed for hybrid at-scale experiments on GENI resources with both simulated networks and real instances of applications running on virtual or physical machines. This paper underscores the importance of hybrid at-scale network experimentation and outlines a method for migrating PrimoGENI to the reconfigurable cloud environments provided by NSFCloud. Although such migration can be readily achieved, several challenges exist for fine-tuning its performance on cloud. As such, PrimoGENI will offer a unique perspective in using reconfigurable and programmable cloud services, such as the need to support high-performance computing with stringent performance guarantees and the need for reconfigurable multi-modal virtualization services.**

## I. Large-Scale Network Experimentation

Computer networking is an area driven by incessant technology innovations. The networking research community has a long history of inventing techniques and developing tools to evaluate new protocols, applications, and network designs. The ability to conduct high-fidelity, high-capacity, and large-scale network experiments is crucial for the healthy growth of the networking research, which has driven deeper and wider into the fundamental network architecture and protocol design.

Existing network testbeds offer different capabilities, in terms of realism, scalability, and flexibility. Physical testbeds (such as PlanetLab [1], EmuLab [2], and GENI [3]) provide realism, but may lack in scalability and flexibility. Adding emulation, such as dummynet [4] or Linux traffic control (tc), a network testbed can strike a good balance between flexibility and realism. On an emulation testbed, real applications can run in designated execution environments with traffic between them "shaped" according to specific network delay and bandwidth constraints. Emulation, however flexible, can still be restrictive in scalability in certain cases. For example, Mininet is a network emulator for OpenFlow applications, which can support experiments with hundreds and even thousands of virtual hosts on a single physical machine (by using Linux lightweight containers) [5]. The size of the experiment supported by Mininet is limited by the available bandwidth of the underlying system, which is no more than a few Gbps.

Simulation provides better flexibility and scalability, since it contains only software modules representing the network entities and protocols, regardless of real time. As such, a network simulator (e.g., [6]–[9]) can easily model much larger networks. Furthermore, using parallel simulation (e.g., [10]), one can substantially increase the scale of network models by harnessing the power of high-performance computing platforms. It is important to note, however, that simulation may lack realism—developing detailed models is both labor-intensive and error-prone. In particular, reproducing realistic network conditions in simulation is non-trivial.

All the above solutions offer different capabilities with respect to realism, scalability, and flexibility. A testbed that can simultaneously maintain accurate representation of the network protocols and applications, provide realistic and diverse network scenarios and conditions, and easily perform multitudes of large-scale experiments is not available.

## II. Cloud-Based At-Scale Network Experiments

Our research has been focused on real-time simulation, which allows simulation of large-scale networks in real time so that the simulated network can interact with real applications running on either physical or virtual machines [11]. We apply advanced parallel simulation and network traffic modeling techniques to increase the scale of the virtual network models and ensure real-time execution. We construct a hybrid environment consisted of a physical system and a simulation system: the former provides an execution environment for unmodified applications, and the latter represents a full-scale network model (with an arbitrary network topology and traffic). The two systems form a symbiotic relationship: the simulation system benefits from the physical system by incorporating real traffic generated by the applications; the physical system benefits from the simulation system by using the state of the full-scale virtual network to calibrate traffic from the real applications. Together, the symbiotic system allows one to test real applications in large-scale network settings.

We developed a network testbed, called PrimoGENI [12], for hybrid at-scale experiments on GENI resources. Using PrimoGENI, one can conduct experiments with a large-scale simulated network with a number of selected emulated hosts. Each emulated host is assigned to a separate virtual machine where real applications can run. Packets generated from the emulated hosts are modified to account for the delays and losses calculated by simulation as if they were on the virtual network. Currently, PrimoGENI can run on a variety of GENI resources, including ProtoGENI clusters [13], and ExoGENI or InstaGENI racks [14].

PrimoGENI can be ready migrated to the cloud environment offered by NSFCloud [15], [16]. Figure 1 depicts an overall architecture of the system running hybrid simulation and emulation experiments on the reconfigurable cloud. PrimoGENI consists of two elements: an integrated development environment (IDE), and a pre-allocated slice of resources for running the experiments. The PrimoGENI IDE is called

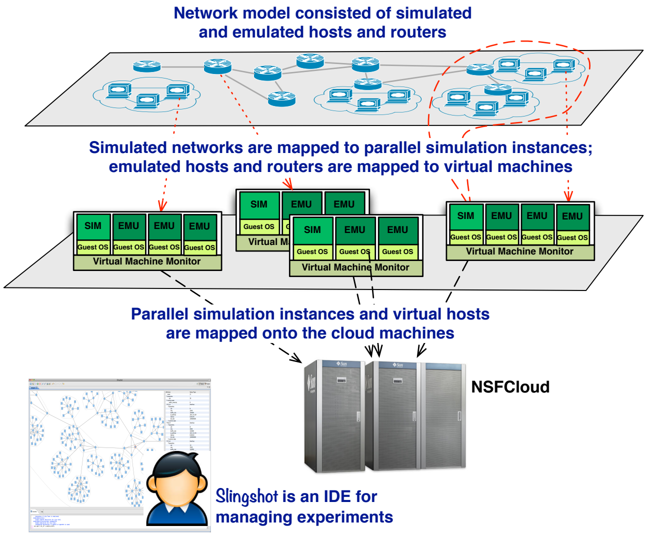


Fig. 1. PrimoGENI on cloud.

*slingshot*, and is intended for the users to manage the life cycle of a network experiment, including model construction, network configuration, deployment, execution, monitoring and control, data collection, visualization, and analysis.

The pre-allocated slice of resources consists of physical machines connected with sufficient bandwidth. Each physical machine is treated as a scaling unit: it runs a parallel simulation instance and possibly one or more virtual machines corresponding to the emulated hosts specified by the network model of the experiment. Each parallel simulation instance runs on a separate physical machine and models a portion of the virtual network. The simulation instances on different machines synchronize and communicate with one another using a parallel synchronization protocol. Using parallel computation, PrimoGENI is able to simulate a large network in real time.

A scaling unit may also run one or several virtual machines, each representing an emulated host in the target network. PrimoGENI provides an emulation infrastructure, which connects the virtual machines with the collocated simulation instances, so that packets generated by the real applications at the virtual machines can be captured and forwarded to the corresponding simulation instances. The simulator may then add delays or drop the packets as if they were on the simulated network and thus subjected to the simulated network conditions.

## III. Requirements for Reconfigurable Clouds

There are several challenges when we try to improve the performance of running hybrid at-scale network experiments in the cloud environment. These challenges will pose specific requirements for the reconfigurable clouds offered by NSFCloud. At the same time, they will underscore the importance of the open, large-scale, programmable cloud services. We list some of the main requirements below:

- PrimoGENI needs high-performance computing with stringent performance guarantees. An integral part of our system is the parallel simulator that can accurately capture the full-scale network behavior in real time. Parallel simulation is a tightly coupled system. Consequently, the cloud needs to provide strong resource isolation to ensure consistent performance.
- PrimoGENI uses multi-modal virtualization. Depending on the applications' workload and QoS demand, the emulated hosts may be assigned to different types of machines, ranging from "bare metals" to virtual machines. The latter may also have the choice of full-machine virtualization, para-virtualization, or simply containers with virtualized network resources. The cloud needs to be reconfigurable to accommodate different types of resources, possibly simultaneously, in order to address the requirements of different experiments.
- PrimoGENI requires high-speed connectivity between physical machines, and between virtual machines. Network capacity is crucial for large-scale network experiments. On the one hand, the cloud needs to provide resource management to guarantee sufficient network resources; on the other hand, the cloud needs to provide necessary tools for efficient network virtualization.
- PrimoGENI requires user configurable software stack with composable network services, such as virtual switches, OpenFlow, traffic shapers (or queuing disciplines), firewall filtering, packet inspection and modification, and other common network functions. The cloud needs to provide flexible mechanisms to easily integrate these services to achieve different network functions.

## References

[1] L. Peterson, T. Anderson, D. Culler, and T. Roscoe, "A blueprint for introducing disruptive technology into the Internet," in *Proceedings of the 1st Workshop on Hot Topics in Networking (HotNets-I)*, 2002.
[2] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed systems and networks," in *Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI'02)*, 2002, pp. 255–270.
[3] The GENI Project Office, http://groups.geni.net/.
[4] L. Rizzo, "Dummynet: a simple approach to the evaulation of network protocols," *ACM SIGCOMM Computer Communication Review*, vol. 27, no. 1, pp. 31–41, 1997.
[5] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks," in *Proceedings of the 9th ACM Workshop on Hot Topics in Networks*, 2010, pp. 19:1–19:6.
[6] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in network simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59–67, 2000.
[7] "The NS-3 Project," http://www.nsnam.org/.
[8] "OPNET Technologies, Inc." http://www.opnet.com/.
[9] A. Varga, "The OMNeT++ discrete event simulation system," in *Proceedings of the European Simulation Multiconference (ESM'01)*, 2001.
[10] J. Cowie, D. Nicol, and A. Ogielski, "Modeling the global Internet," *Computing in Science and Engineering*, vol. 1, no. 1, pp. 42–50, 1999.
[11] J. Liu, "A primer for real-time simulation of large-scale networks," in *Proceedings of the 41st Annual Simulation Symposium (ANSS)*, 2008, pp. 85–94.
[12] N. V. Vorst, M. A. Erazo, and J. Liu, "PrimoGENI for hybrid network simulation and emulation experiments in GENI," *Journal of Simulation*, vol. 6, no. 3, pp. 179–192, 2012.
[13] Flux Research Group at University of Utah, http://www.protogeni.net/.
[14] GENI Racks, http://groups.geni.net/geni/wiki/GENIRacksHome/.
[15] Chameleon Cloud, http://www.chameleoncloud.org/.
[16] CloudLab, http://www.cloudlab.us/.