The Tradeoff Between Single Aggregate and Multiple Aggregates in Designing GENI Experiments

Zongming Fei[1(✉)], Ping Yi[1], and Jianjun Yang[2]

[1] Laboratory for Advanced Networking, Department of Computer Science, University of Kentucky, Lexington, KY 40506, USA
fei@netlab.uky.edu, yiping@netlab.uky.edu
[2] Department of Computer Science, University of North Georgia, Oakwood, GA 30566, USA
jianjun.yang@ung.edu

Abstract. The Global Environment for Network Innovations (GENI) provides a virtual laboratory for exploring future internets at scale. It consists of many geographically distributed aggregates for providing computing and networking resources for setting up network experiments. A key design question for GENI experimenters is where they should reserve the resources, and in particular whether they should reserve the resources from a single aggregate or from multiple aggregates. This not only depends on the nature of the experiment, but needs a better understanding of underlying GENI networks as well. This paper studies the performance of GENI networks, with a focus on the tradeoff between single aggregate and multiple aggregates in the design of GENI experiments from the performance perspective. The analysis of data collected will shed light on the decision process for designing GENI experiments.

Keywords: GENI · Network testbed · Network measurement · Experiment design

1 Introduction

The Global Environment for Network Innovations (GENI) is a project sponsored by National Science Foundation (NSF) with the aim to provide a collaborative environment to build a virtual laboratory for exploring future internets at scale [1,2]. It has attracted many universities and industrial partners to contribute their efforts towards developing a global federated network testbed for networking research and education. An experimenter can reserve both computing resources (such as PCs, virtual machines (VMs)), and networking resources (such as ION links, OpenFlow switches, VLANs, and GRE tunnels). GENI consists of many aggregates, each of which manages a set of resources [3]. Typically, a GENI aggregate is administrated and controlled by an institution which can impose its own policies about the allocation of the resources. As more GENI racks are

V.C.M. Leung et al. (Eds.): TridentCom 2014, LNICST 137, pp. 23–32, 2014.
DOI: 10.1007/978-3-319-13326-3_3

deployed on university campuses across the United States, GENI has grown to have tens of aggregates with resources available for network experiments [4].

One decision that needs to be made in designing a GENI experiment is whether to use resources from one aggregate or from multiple aggregates. It depends on the types of experiments to be performed. Some experiments such as multimedia applications may have a strict end-to-end delay requirement that cannot be satisfied by nodes distributed over a wide area. They may have to get resources from a single aggregate. On the other hand, there are experiments that need to test the behavior of protocols on how they react to the cross traffic from the real world. It may be preferable to have resources from multiple aggregates. There is also a question about which aggregates to choose to put the experimental nodes.

To make this decision, we need to have a good understanding of underlying networks. For example, what exactly can we get from links within an aggregate versus from multiple aggregates? How different are the bandwidths and latencies of links within an aggregate versus from multiple aggregates? What are their behaviors over a long period of time? We collect and analyze the measurement data and try to answer these questions. We expect that the analysis will provide helpful hints to the design of GENI experiments.

We understand that the distinction between single aggregate and multiple aggregates is not absolute. In a single aggregate experiment, the links generally have lower latencies and higher bandwidths. To make them suitable for an experiment that needs more realistic topology that has a wide variety of delays and bandwidths, we can add delay nodes in the middle of the topology to do traffic shaping, increasing the delay or reducing the bandwidth, or both. This added an element of simulations/emulations, instead of pure experimentations. The resulting topology will have some characteristics of multi-aggregate experiments. On the flip side of the coin are experiments using multiple aggregates. For large network experiments, the number of nodes usually exceeds the number of aggregates available. We have to allocate multiple nodes within an aggregate. Thus, even in a multi-aggregate experiment, we may still have links within an aggregate. In either case, we need to have an idea about delays and bandwidths of both single-aggregate links and cross-aggregate links.

In this paper, we present our study on performance of GENI networks, with a focus on the tradeoff between single aggregate and multiple aggregates in the design of GENI experiments from the performance perspective. We will analyze how the links behave differently over a period of time. The data collected will shed some light on the design process for choosing where the nodes in the experiment should be located.

The rest of the paper is structured as follows. Section 2 introduces related work and some background concepts. Section 3 describes the experiments we used to collect performance data. Section 4 presents the results about the latencies and bandwidths of the links within an aggregate and across aggregates. Section 5 concludes the paper.

2 Related Work

GENI has involved many universities and industry partners and grown significantly in recent years. It consists of multiple control frameworks [5,6] and has resources mainly on university campuses in the United States and several sites in other countries. It developed many tools supporting experimenters, such as Flack [7,8] of ProtoGENI [5].

Several early GENI projects investigated performance measurement [9–13] in the GENI environment. They have different focuses and generally emphasize on developing tools to enable users to collect performance data.

More recently, two major instrumentation and measurement efforts are under way in GENI. One is the Large-scale GENI Instrumentation and Measurement Infrastructure (GIMI) project [14], which makes use of OML library to instrument resources based on the ORBIT control framework. It can filter and process measurement flows, and consume measurement flows. The other is the GENI Measurement and Instrumentation Infrastructure (GEMINI) project [15]. It is based on earlier INSTOOLS system [9] and perfSONAR system [16]. It started with supporting ProtoGENI, but can now support nodes from other control frameworks as well. All these GENI measurement systems emphasize on building tools to support users to collect measurement data *after* their experiments have been set up. In contrast, this paper focuses on examining behaviors of different kinds of links in GENI networks and help users in the design process of their experiments.

3 Experiments for Data Collection

To measure the performance of links within an aggregate, we design a 11-node topology as shown in Fig. 1. In GENI, multiple virtual machines (VMs) can be allocated from a single raw physical machine/computer (PC). We want to measure both the links that connects two VMs from the same physical machine and the links that connects two VMs from two different physical machines. Theoretically, three VMs are enough because we can have two VMs from the same physical machine and the other one from a different physical machine. We can create both kinds of links with these three machines. However, if we create a topology with three VMs, most likely we will end up with three VMs from the same physical machine due to the allocation algorithm used in GENI aggregates. Even though we can bind a VM to a specific physical machine, the submission through the GENI Flack interface is not well supported. Our strategy is to specify a topology as shown in Fig. 1 with enough number of nodes so that they have to be allocated to different physical machines. We understand that we do not have to measure all the links. Rather we select four links as representatives.

We obtained the bandwidth and latency data for these four links using `iperf` [17] and `ping` over 10 days. One measurement (both bandwidth and latency) is taken for every hour, with 10 `ECHO_REQUEST`s for each ping.

To measure the performance of links from different aggregates, we select 10 aggregates and set up a mesh topology as shown in Fig. 2.

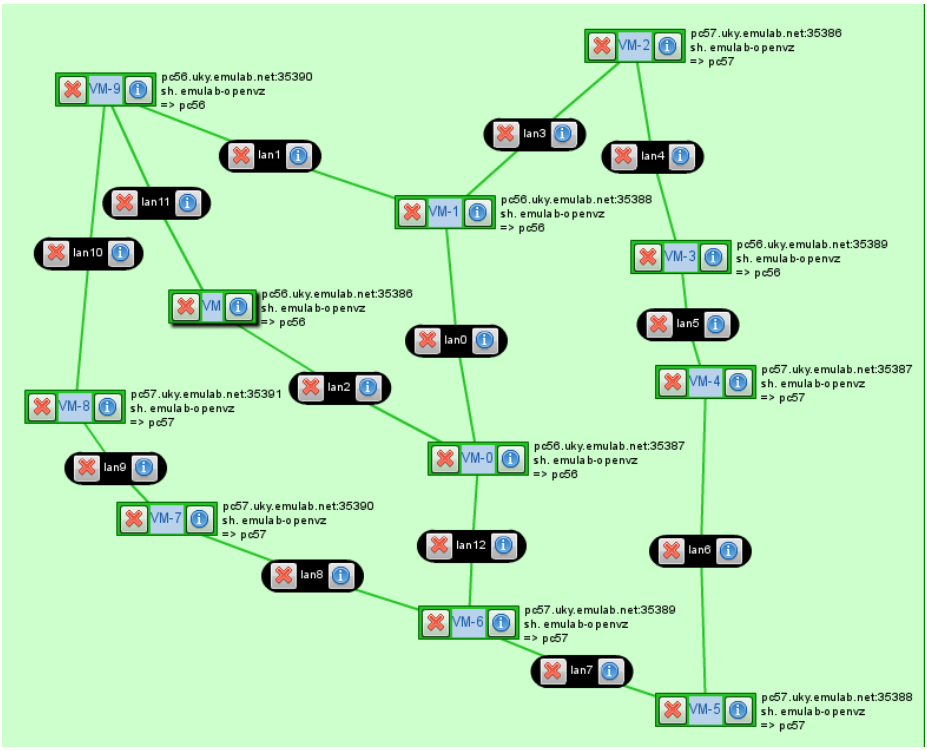


Fig. 1. The single-aggregate experiment

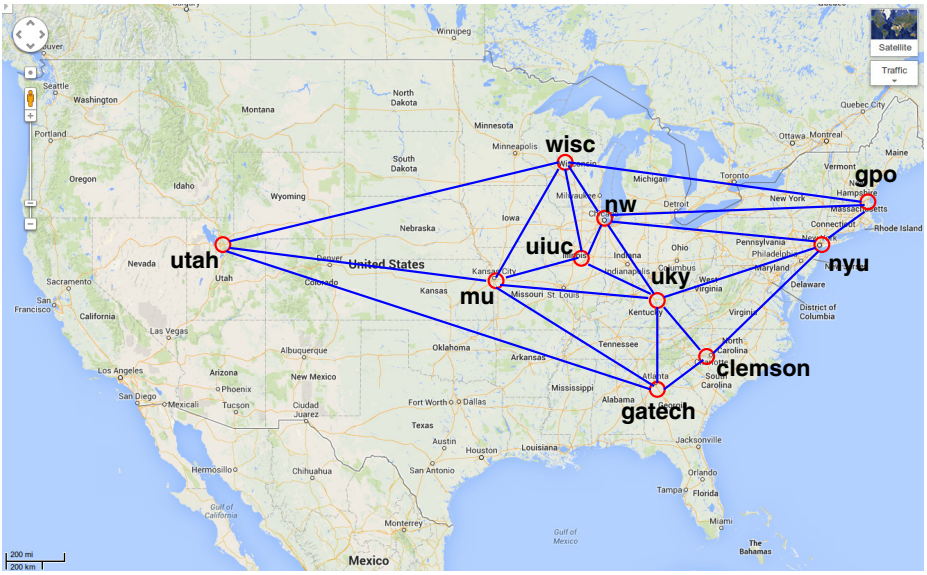


Fig. 2. The multi-aggregate experiment

4 Performance Results

We collected both latency and bandwidth information from these two experiments. Links in these two experiments can be divided into three categories:

Category 1 (**Same PC**): the links connecting two VMs that are allocated from the same physical machine;
Category 2 (**Same Aggregate**): the links connecting two VMs that are allocated from two different physical machines located in the same aggregate; and
Category 3 (**Different Aggregates**): the links connecting two VMs that are allocated from two different physical machines located in two different aggregates.

The first experiment covers the first two kinds of links, while the second experiment covers the third kind of links. We first calculate the averages of latencies and bandwidths over the 10 day period for each link. The results are summarized in Table 1.

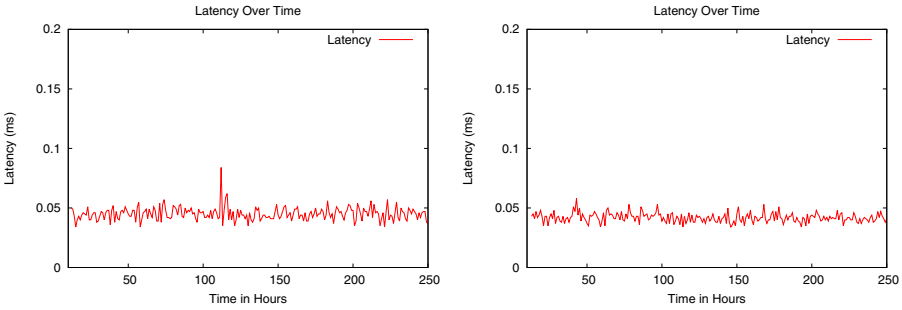
The links in the Same PC category have similar performance. So we only choose two links (from VM-0 to VM-1, and from VM-6 to VM-7) as representatives. For the same reason, we only choose two links (from VM-0 to VM-6, and from VM-3 to VM-4) as representatives for the Same Aggregate category. However, the performance of the links from the Different Aggregates category varies a lot. So we include the results for all the links in the second experiment in the table.

As expected, the average latencies for the links in the Same PC category are the smallest, measured at 0.042ms and 0.045ms. The latencies for the links in the Same Aggregate category are about 2.5 times as large, but still in the range of one tenth of a second. They are both much smaller than the links connecting VMs from two different aggregates. The lowest latency we got is the link connecting VMs from the Northwestern aggregate and the UIUC aggregate, measured at 3ms, which are 30 times as large as that of the links from the Same Aggregate category. We see a wide variety of latencies measured for different cross-aggregate links, ranging from 3ms to 60ms. When designing a GENI experiment, we may take the difference in latencies into consideration for reserving GENI resources.

Table 1. Average latency and bandwidth

Category	link	Avg. Latency (ms)	Avg. Bandwidth (Mbits/second)
1. Same PC	VM-0 to VM-1	0.045	97.3
	VM-6 to VM-7	0.042	97.4
2. Same Aggregate	VM-0 to VM-6	0.115	474
	VM-3 to VM-4	0.116	469
3. Different Aggregates	21 links	from 3 to 60	from 34 to 94

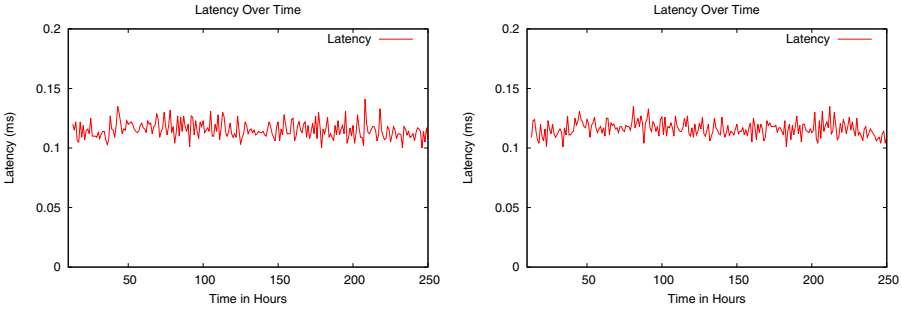
While the average latencies give a general idea about the tradeoff between using nodes from a single aggregate versus from multiple aggregates, it is more interesting to observe how they change over time. Fig. 3(a) shows how the latency of the link from VM-0 to VM-1 in the first experiment change over the 10 day period. We can see that it always hovers around 0.045ms, with the highest at 0.084ms at one time and with the lowest at 0.034ms three times. It is relatively stable and close to its average value. Fig. 3(b) shows that the link from VM-6 to VM-7 displays the similar pattern.



(a) from VM-0 to VM-1 (b) from VM-6 to VM-7

Fig. 3. Latency of the links connecting two VMs from the same PC

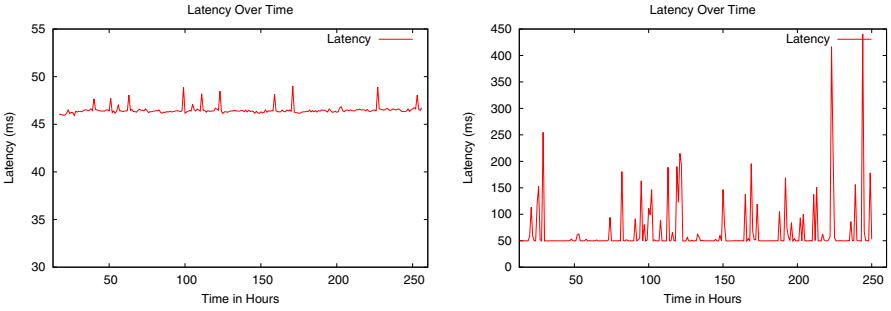
The latencies for the links connecting two VMs from two different PCs within an aggregate are larger than that of category 1 links as shown in Fig. 4. Also larger is the range these latencies change. However, we still see a very stable pattern in terms how they change over time.



(a) from VM-0 to VM-6 (b) from VM-3 to VM-4

Fig. 4. Latency of the links connecting two VMs from two PCs within an aggregate

The latencies for category 3 links demonstrate a wider variety of patterns. For lack of space, we cannot present all of them in this paper. Instead, we choose two as representatives here to show how they can be quite different. Fig. 5(a) shows how the latency of the link from Kentucky to Missouri [1] change over time. The absolute range of the change is larger than those links from categories 1 and 2. However, the percentage of the change is not large. It is a totally different story for the link from Utah to Georgia Tech (Gatech) as shown in Fig. 5(b). Notice that the scales on y-axis in the figures are different. The range of the change in this case is almost 10 times as large as the average value. We can end up with a much more unpredictable behavior if we have VMs allocated from different aggregates.



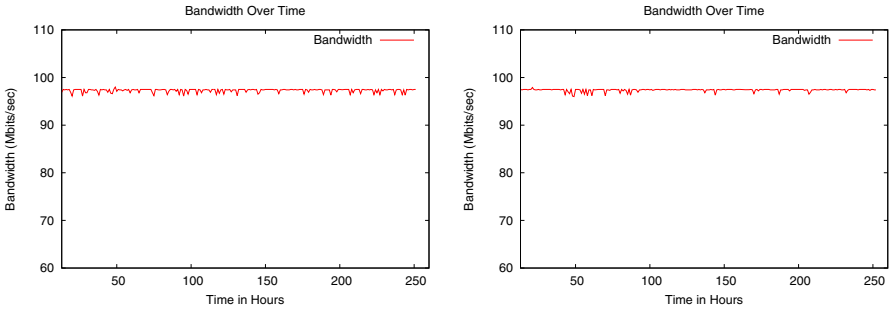
(a) from Kentucky to Missouri (b) from Utah to Gatech

Fig. 5. Latency of the links connecting two VMs from two different aggregates

The latency of the links is only one factor to consider in designing GENI experiments. The other factor is the bandwidth of the links. From Table 1, we can see that category 1 links have a measured bandwidth of 97.3 Mbps. It can be higher because the two VMs these links attached to are located within the same physical machine. However, due to rate limit of the VMs, they are most likely capped at 100 Mbps. Fig. 6 shows how the bandwidth of these links change over time. Similar to the latency case, it stays close to the average level, appearing almost like a straight line.

Category 2 links achieve higher bandwidth, having average values at 474 Mbps and 469 Mbps. VMs in this case are connected with a gigabit switch. Because of the traffic from other experiments or load on the shared physical machines, the measured bandwidth is smaller than the maximal possible value. For the similar reason, we can see in Fig. 7 that it oscillates quite a lot over

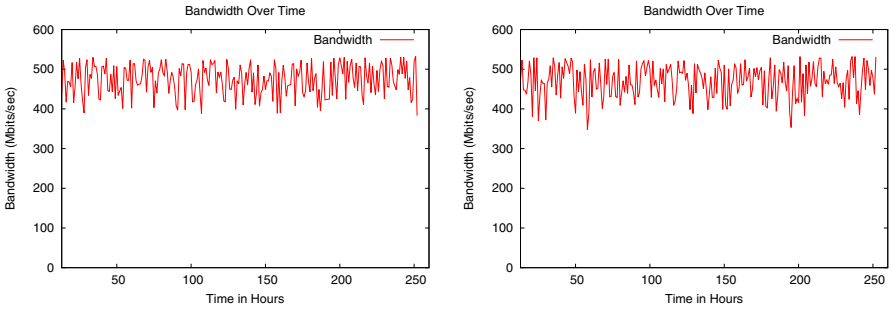
[1] We use abbreviations here to indicate the VMs from a certain aggregate. "Kentucky" means the VM allocated from the University of Kentucky GENI aggregate. Similarly, "Missouri" means the VM allocated from the University of Missouri GENI aggregate. We use this convention for naming other VMs, too.



(a) from VM-0 to VM-1 (b) from VM-6 to VM-7

Fig. 6. Bandwidth of the links connecting two VMs from the same PC

time, ranging from 347 Mbps to 533 Mbps. However, the bandwidth of category 2 links is still much large than that of both category 1 links and category 3 links.

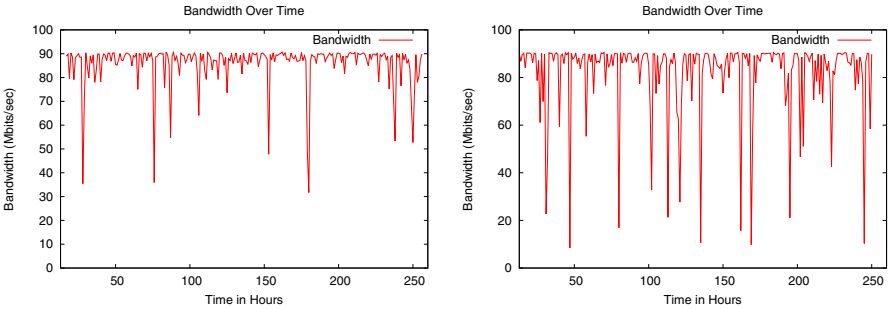


(a) from VM-0 to VM-6 (b) from VM-3 to VM-4

Fig. 7. Bandwidth of the links connecting two VMs from two PCs within an aggregate

We get a totally different picture for the links connecting two VMs from different aggregates. Depending on the links, we can get an average bandwidth as low as 34 Mbps and as high as 94 Mbps. They also change more wildly over time, as shown in Fig. 8. This is because these links are cross-Internet links that will compete with traffic from other applications. Their behaviors are much more unpredictable than those links within a single aggregate. For the same link from Utah to Gatech, we can get a bandwidth measure as low as 8.5 Mbps and as high as 90.5 Mbps. If we want to observe how a protocol performs and reacts to the real world traffic, this may be the link we should include in the experiment.

In summary, from the data we collected, we can see significant differences between single-aggregate links and cross-aggregate links in terms of latency and



(a) from Kentucky to Missouri (b) from Utah to Gatech

Fig. 8. Bandwidth of the links connecting two VMs from two different aggregates

bandwidth. Not only the average values are significantly different, but their behaviors over time can be quite different as well. When designing a GENI experiment, we can make use of performance data to decide where the nodes in the experiment should be located to meet the requirement.

5 Conclusion

Understanding the GENI networks is an important step in making a good design for GENI experiments. We focus on the performance aspect of the GENI networks by collecting latency and bandwidth data from two experiments. The results from this paper are only a snapshot of the GENI networks over a short period of time. However, the observed behaviors and the collected performance data of the links from different categories provide helpful information for GENI experimenters. As more researchers and educators use the GENI network testbed, there is a growing need to better understand all aspects of GENI.

Acknowledgments. We would like to thank Dr. Jim Griffioen for his comments on our earlier work on this topic. We also want to thank Mr. Hussamuddin Nasir and other members of the GEMINI project team for their help during the design and implementation of this project.

This material is based upon work supported in part by the National Science Foundation under Grant No. CNS-0834243 and CNS-1346688 Subcontracts 1925 and 1928. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of BBN Technologies Corp, the GENI Project Office, or the National Science Foundation.

References

1. The GENI Project Office, GENI System Overview. http://www.geni.net/docs/GENISysOvrvw092908.pdf

2. The GENI Project Office, GENI System Overview. http://groups.geni.net/geni/wiki/GENIConcepts
3. GENI glossary. http://groups.geni.net/geni/wiki/GENIGlossary
4. GENI aggregates. http://groups.geni.net/geni/wiki/GeniAggregate
5. ProtoGENI. http://www.protogeni.net
6. ORCA. https://geni-orca.renci.org/trac/
7. The Flack GUI (2012). http://www.protogeni.net
8. Duerig, J., Ricci, R., Stoller, L., Strum, M., Wong, G., Carpenter, C., Fei, Z., Griffioen, J., Nasir, H., Reed, J., Wu, X.: Getting started with GENI: A user tutorial. ACM SIGCOMM Computer Communication Review (CCR) **42**(1), 72–77 (2012)
9. Griffioen, J., Fei, Z., Nasir, H., Wu, X., Reed, J., Carpenter, C.: The design of an instrumentation system for federated and virtualized network testbeds. In: Proc. of the First IEEE Workshop on Algoirthms and Operating Procedures of Federated Virtualized Networks (FEDNET), Maui, Hawaii (April 2012)
10. GIMS: High-speed packet capture for GENI (2011). http://gims.wail.wisc.edu/docs/Tutorial.html
11. Leveraging and abstracting measurements with perfSONAR(LAMP) (2011). http://groups.geni.net/geni/wiki/LAMP
12. Calyam, P., Schopis, P.: OnTimeMeasure: Centralized and distributed measurement orchestration software (2012). http://groups.geni.net/geni/wiki/OnTimeMeasure
13. Fahmy, S., Sharma, P.: Scalable, extensible, and safe monitoring of GENI clusters (2010). http://groups.geni.net/geni/attachment/wiki/ScalableMonitoring/design.pdf
14. GIMI: Large-scale GENI instrumentation and measurement infrastructure. http://groups.geni.net/geni/wiki/GIMI
15. GEMINI: A GENI measurement and instrumentation infrastructure. http://groups.geni.net/geni/wiki/GEMINI
16. PerfSONAR. http://www.perfsonar.net/
17. iperf - TCP and UDP bandwidth performance measurement tool. http://code.google.com/p/iperf/