GENI-based Network Experiments: Possibilities and Practice

Zongming Fei

University of Kentucky

Introduction

- GENI provides a virtual laboratory for exploring future networks and facilitates the testing of network architectures and protocols at scale.
  - Mainly used as network testbeds for research
- Educational use of GENI
  - Learning curves and extra work of using GENI
  - Experiments not well developed
  - Benefits not clearly shown

Introduction

- Some well-developed programming assignments to show
  - Feasibility of giving non-trivial assignment using GENI
  - Benefits of using the GENI environment
  - Flexibility of designing such experiments
- Goals
  - Providing hands-on experience to students
  - Experiments should start simple, but can be extended to a large scale
  - Leverage as many features of GENI (such as instrumentation and measurement features) as possible
  - Taking advantage of the virtualization and programmability

Outline

- Introduction
- Educational Use of GENI
- Network Experiments Using GENI
- Experiences and Lessons
- GENI Desktop
- Conclusion

Educational Uses for GENI

- GENI supports at-scale networks
  - The number of resources offered
  - The types of resources offered
  - The geographical scope of the resources offered
  - The speed/performance of the resources offered
- An incomplete list of projects (or types of projects)
  1. Conventional OS/networking assignments
     - Make modifications to existing OS and networking code to create their own protocols and network services
  2. Network monitoring assignments
     - Write active and passive monitoring code to measure the performance of the Internet

Educational Uses for GENI

3. Data center/cloud assignments
   - Use GENI high-performance clusters to implement data center services with custom or conventional data center software
4. Wireless and/or mobile assignments
   - Implement services using a variety of wireless and mobile resources and technologies supported by GENI
5. Home networking assignments
   - Write home services/applications utilizing the resources of opt-in home users in GENI

Educational Uses for GENI

6. High-performance networking assignments
   - Make use of GENI high-performance servers to test the scalability with regard to performance
7. Application-level monitoring assignments
   - Use low-power radar sensors and web cameras that are virtualized and accessible to users and the high-performance network links to move data off of these devices to network servers in real-time
8. Complete network assignment
   - Design and implement a complex/complete network ranging from mobile client nodes connected via wireless links to an optical backbone networks with advanced services built into the network structure, as well as data center computing power offering cloud services

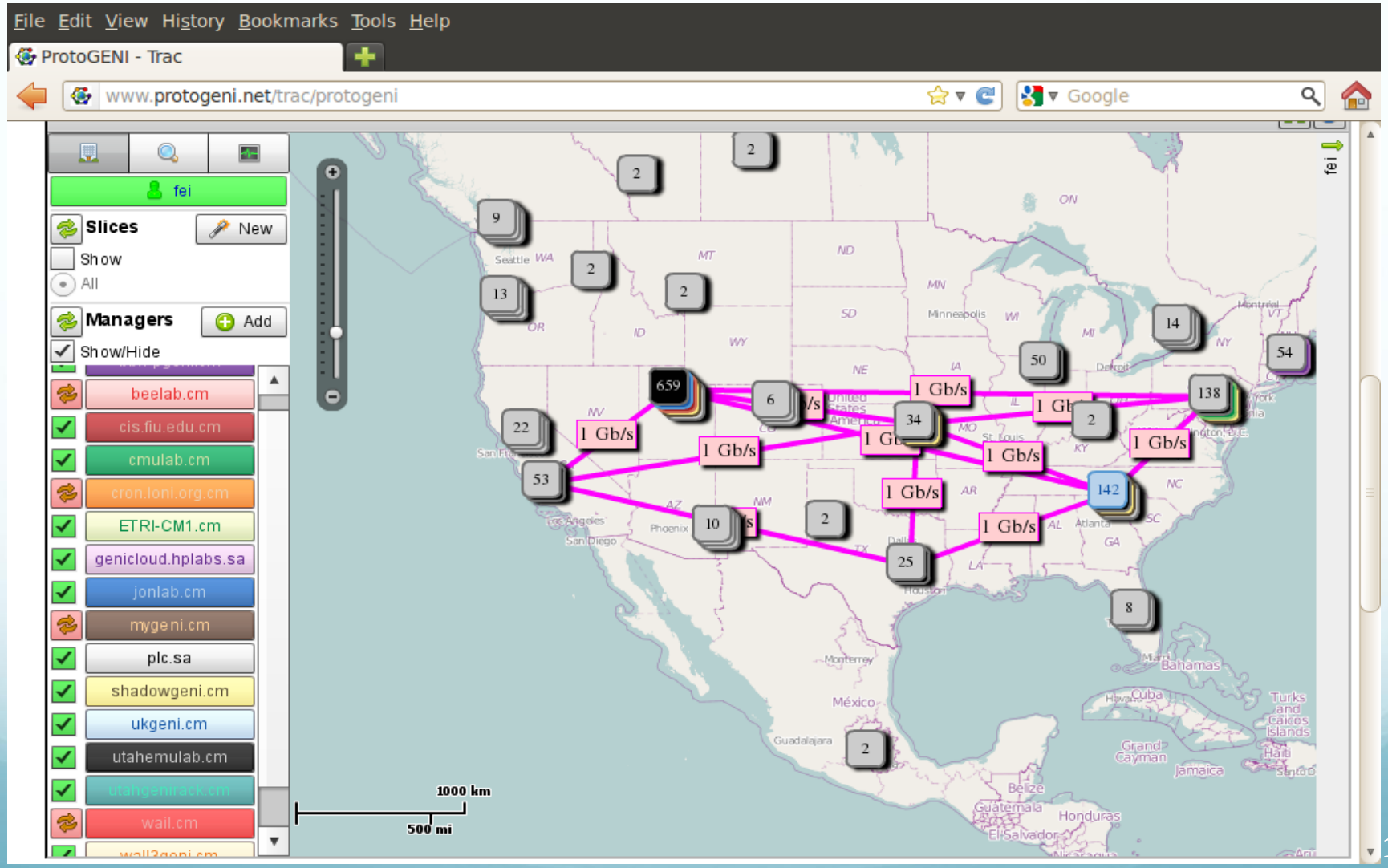
Outline

- Introduction
- Educational Use of GENI
- Network Experiments Using GENI
- Experiences and Lessons
- GENI Desktop
- Conclusion

Assignment 1 --Path Characteristics of the Internet

- Measure delay, bandwidth and loss rate of the links over the Internet and analyze their distributions
- Questions to be answered
  - What is the typical delay of a path from the east coast to the west coast?
  - How do delay, bandwidth and loss rate differ between a local link and a wide area link, or among different wide area links?
  - Do they change a lot over time? What is the distribution?
- GENI vs Guest accounts

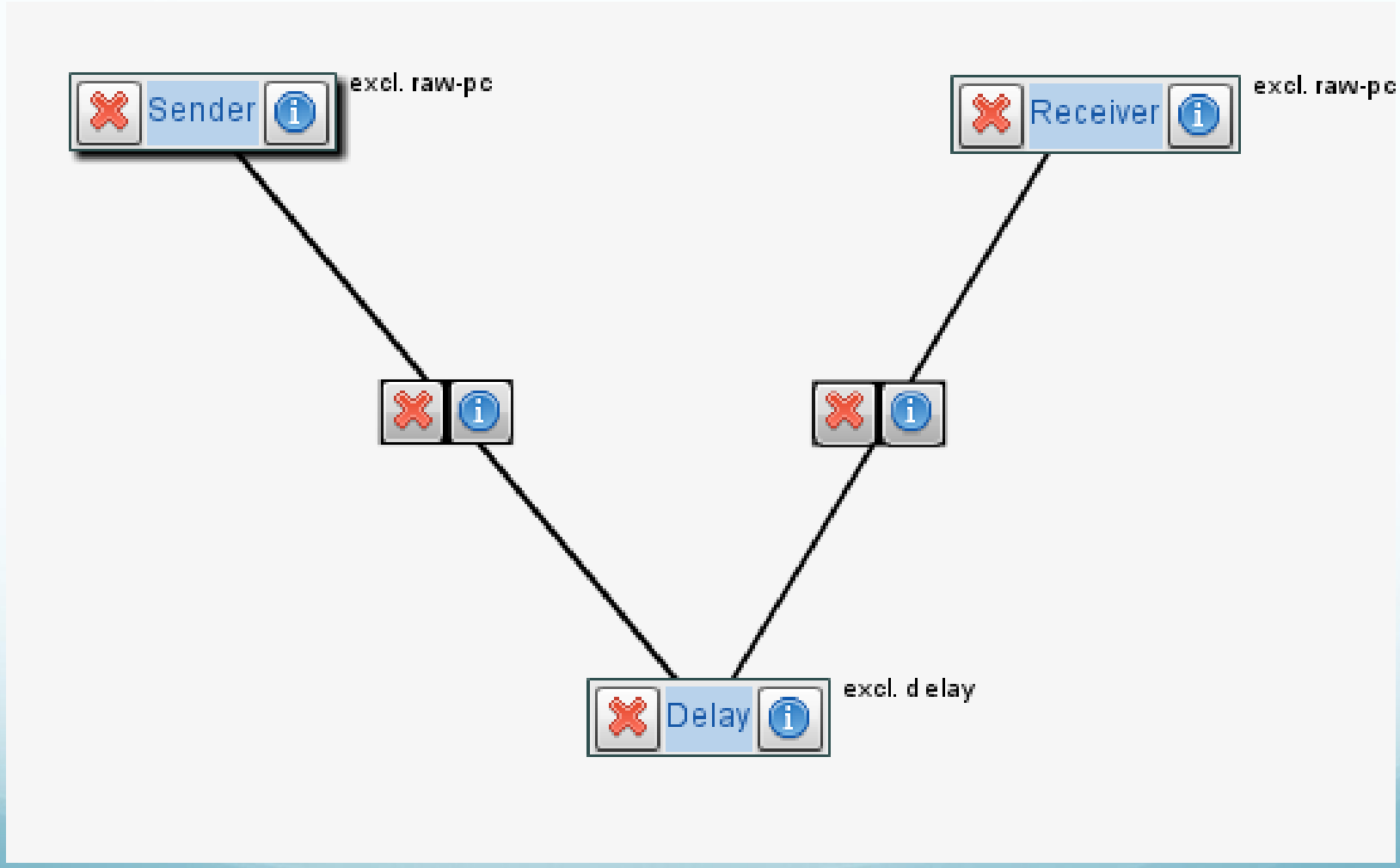
Assignment 1 --Path Characteristics of the Internet

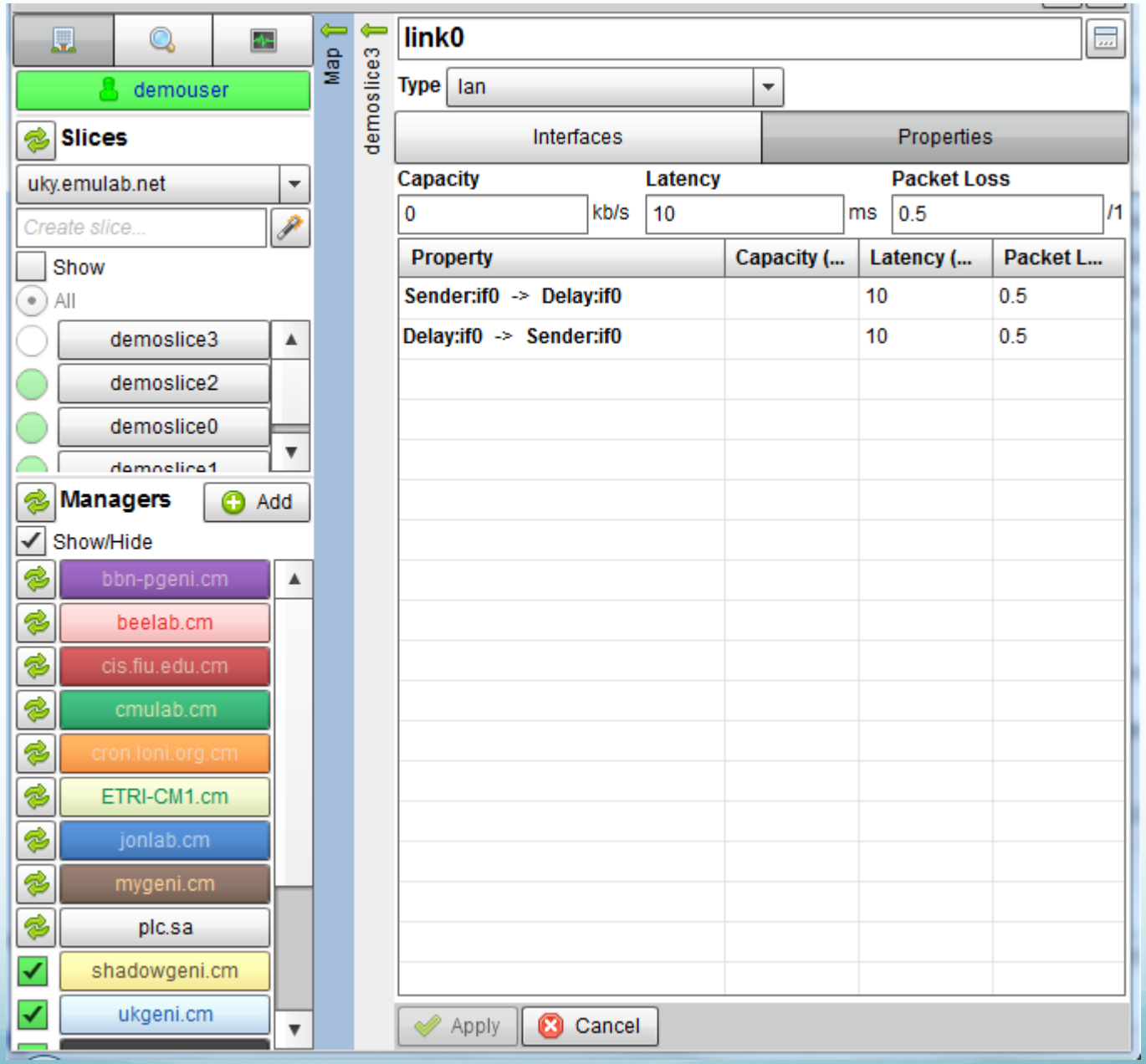


Assignment 2 --Reliability Protocols

- Implement reliability protocols at the application layer using UDP.
  - Stop and wait
  - Go back N
  - Selective repeat
  - TCP reliability
- Almost no loss in a traditional general purpose lab
- Need a repeatable, predictable and easy to control environment with a certain loss property
- GENI provides an environment to do that

Assignment 2 --Reliability Protocols

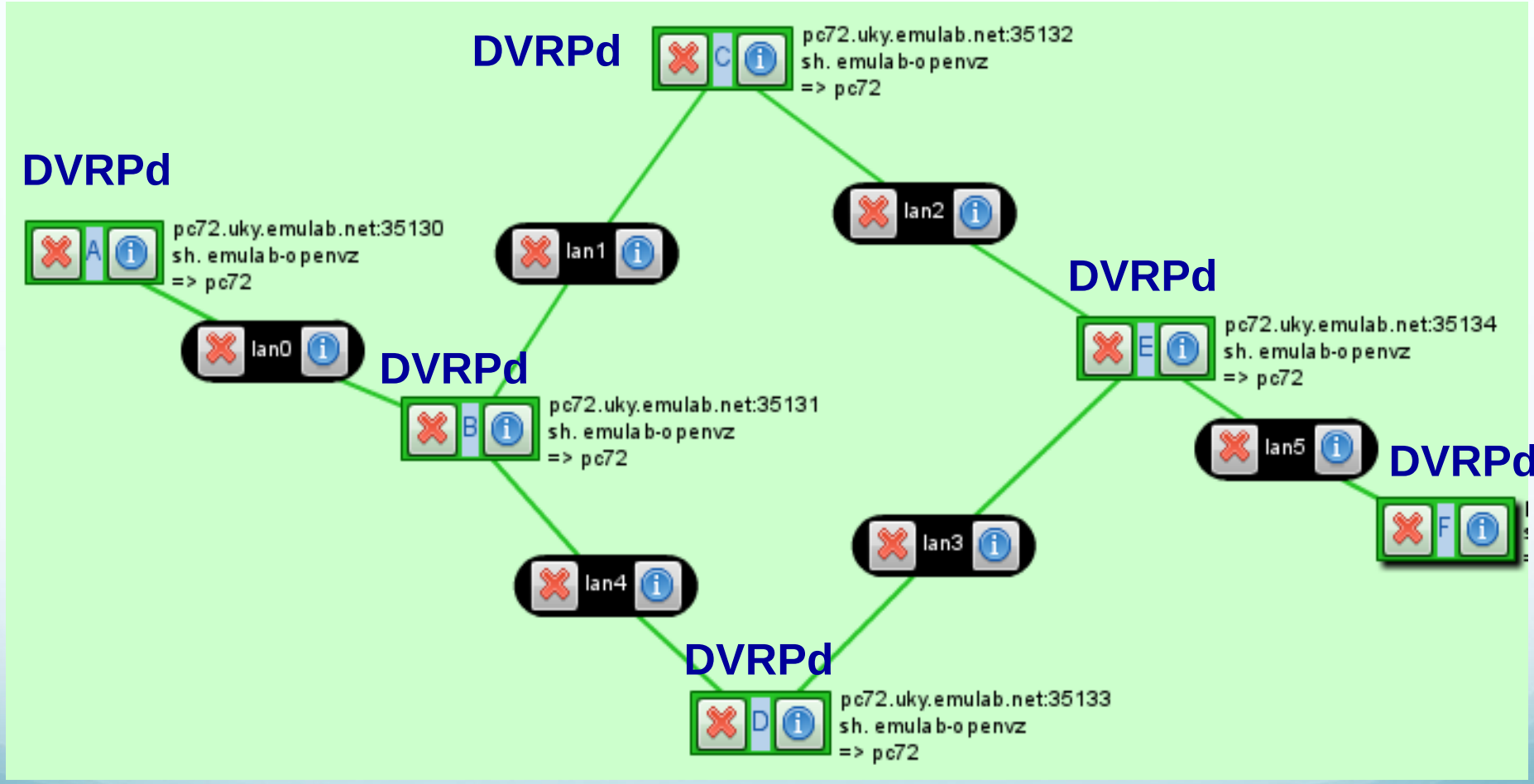



demouser
Slices
uky.emulab.net
Create slice...
Show
All
demoslice3
demoslice2
demoslice0
Managers
Add
Show/Hide
bbn-pgeni.cm
beelab.cm
cis.fiu.edu.cm
cmulab.cm
cron.loni.org.cm
ETRI-CM1.cm
jonlab.cm
mygeni.cm
plc.sa
shadowgeni.cm
ukgeni.cm
Map
demoslice3
link0
Type
lan
Interfaces
Properties
Capacity
Latency
Packet Loss
0
kb/s
10
ms
0.5
/1
Property
Capacity (...
Latency (...
Packet L...
Sender:if0 -> Delay:if0
10
0.5
Delay:if0 -> Sender:if0
10
0.5
Apply
Cancel

Assignment 3 –Distance Vector Routing Protocol

- Goals:
  - Get hands-on experience with the distance vector routing protocol
  - Understand at-scale experiments
- Steps
  - Setting up the initial experiment
  - Distance vector routing protocol daemon (DVRPd) running on each node
  - Dynamic change of link costs
  - Generating traffic and monitoring the routing path

Setting Up the Initial Experiment



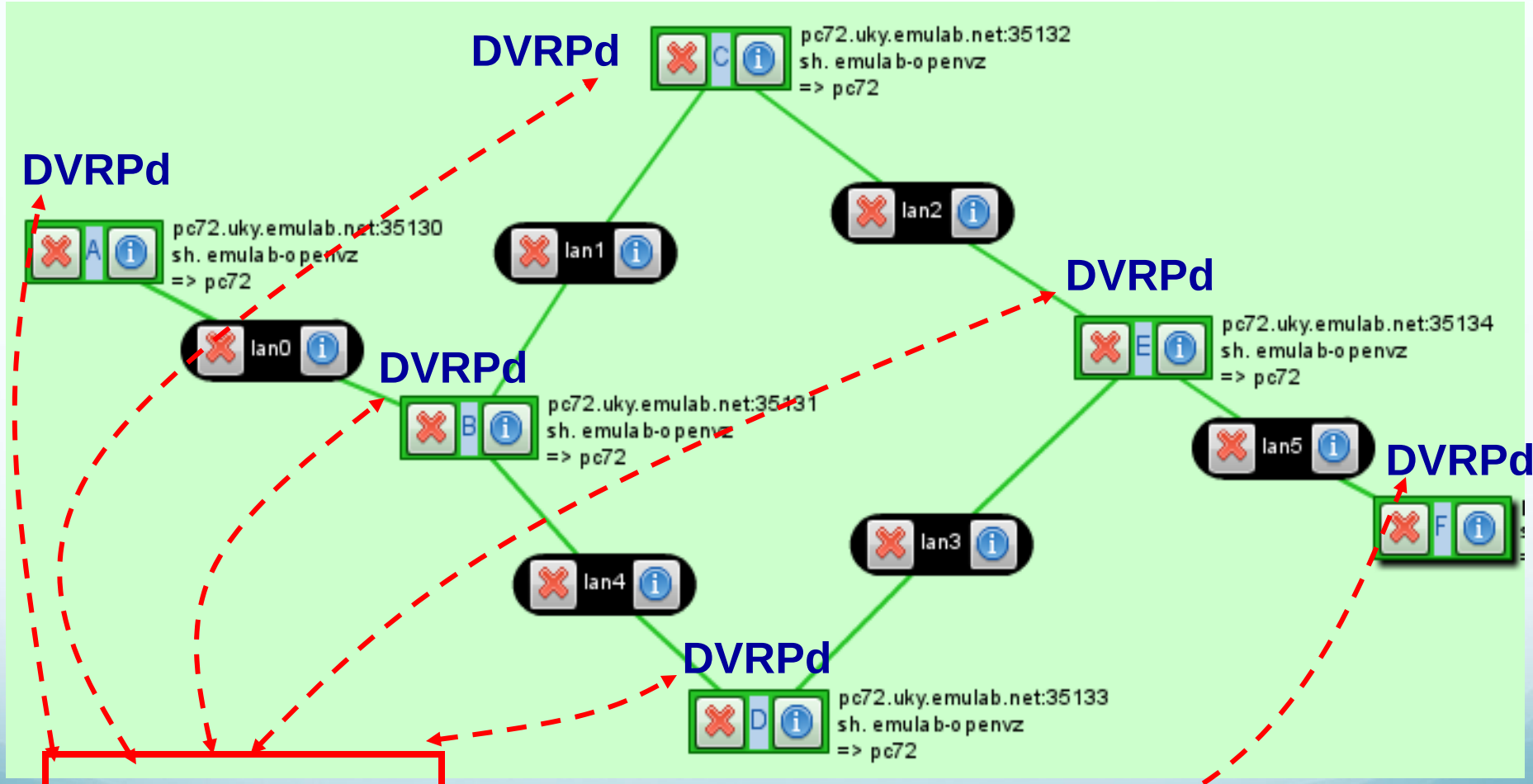
Distance Vector Routing Protocol

- DVRPd running on each node
  - To simplify the implementation using UDP, we replace the initial network discovery by providing a configuration file for each node

```
name_of_this_node
neighbor_1_name    link_cost    neighbor_1_IP_addr
neighbor_2_name    link_cost    neighbor_2_IP_addr
……
neighbor_n_name    link_cost    neighbor_n_IP_addr
```

  - The IP addresses are needed for UDP programs to know where to send packets.
  - Periodic updates and triggered updates
  - Make changes to forwarding tables

Dynamic Change of Link Costs



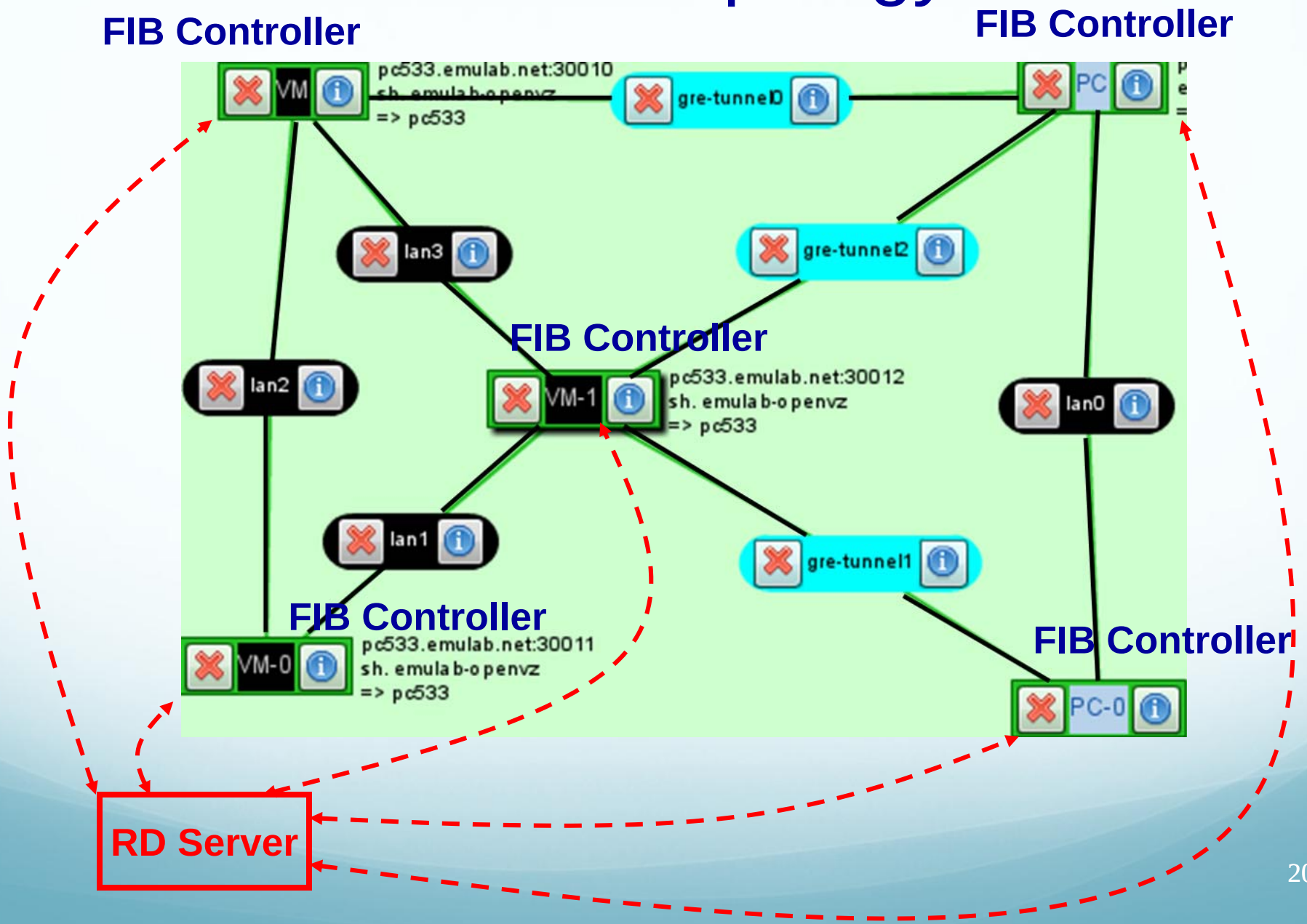
Generating Traffic and Monitoring Routing Path

- Traffic is generated from node A to node F using iperf
- Use GEMINI to monitor the TCP and IP traffic at node C and D
- As the link costs change, the routing path will be changed and the effect can be observed by the instrumentation and measurement tools
- Final note: The experiment can be expanded to an arbitrarily large scale.

Assignment 4 --Path Control and QoS Routing

- Goals:
  - Get hands-on experience with QoS routing
  - Understand the basic concept of software defined network: separation of forwarding and routing
- Forwarding will use the existing Linux functions
- Routing (codes written by students) is divided into two parts:
  - FIB Controller (Forwarding Information Base Controller)
    - runs on every node
    - Inserts and deletes entries in FIB (use Unix system call to call route to change the forwarding tables)
    - receives commands about from RD server about routing path
  - RD server (Routing Decision Server)
    - runs on an extra node to make routing decision
    - Communicates with FIB controllers via FIB controller Protocol

Initial Topology



Routing Decision (RD) Server Details Phase I: Simple RD server

- Command line interface for users to set/delete/show paths
  - **createpath** IPaddr_list DestAddr
  - **deletepath** IPaddr_list DestAddr
  - **showroute** IPaddr
- RD server program figures out what need to changed at each node
- RD server sends commands to relevant FIB controllers via FIB control protocol (FCB)
- FIB controllers make the actual changes

Routing Decision (RD) Server Details Phase II: QoS RD (QRD)server

- Link State Announcement (LSA) daemon at each node
  - Read a local lsa.config file periodically (content dependant on the node and can be changed over time via a text editor)

```
QRD=10.10.2.5                  # QRD server address
10.1.1.3 10.1.1.6   4   10     # link delay (4ms) and bandwidth (10Mbps)
10.1.2.1 10.1.2.2   6  100     # link delay (6ms) and bandwidth (100Mbps)
```

  - Send off an LSA announcement periodically (every 10s) with a sequence number to the QRD server
- QRD server records most recent LSA from each node.
- Node makes request path(src, dst, QoS) to QRD server to find a path from src to dst satisfying QoS specification, where QoS specify d(delay) or b(bandwidth) followed by a metric.

Routing Decision (RD) Server Details Phase II: QoS RD (QRD)server

- If the QRD server cannot find the path satisfying QoS, it will reply NO;
- If the QRD server finds a path satisfying QoS, it will reply YES, and set up the path along the way by communicating with FIB controllers and informing them the forwarding entries that need to be set up. The FIB controllers will set up the path.
- The requesting node can send the traffic to the destination using the path.
- Final Note: The experiment can be expanded to a much larger topology.

Outline

- Introduction
- Educational Use of GENI
- Network Experiments Using GENI
- Experiences and Lessons
- GENI Desktop
- Conclusion

Feedback from Students

- Use cool and most advanced technology
- Keep pace with most recent development in the real world
- Learn the distinction between experiment network and control network
- It is sometimes hard to figure out which part goes wrong? (not a production software)

- Lesson: Importance of step-by-step instructions for students to get started

Experience and Lessons

- Positive
  - Easy-to-use graphical interface
  - Easy setup and teardown of experiments
  - A wide range of available user-controllable resources
- Students have some difficulty in these aspects.
  - Login by ssh (especially scp) using private/public key pairs
  - Copy files from local desktop to experimental nodes
  - Keep track of open terminals and run commands on experimental nodes
- It is desirable to have a unified graphical interface/tool that can provide these functions.

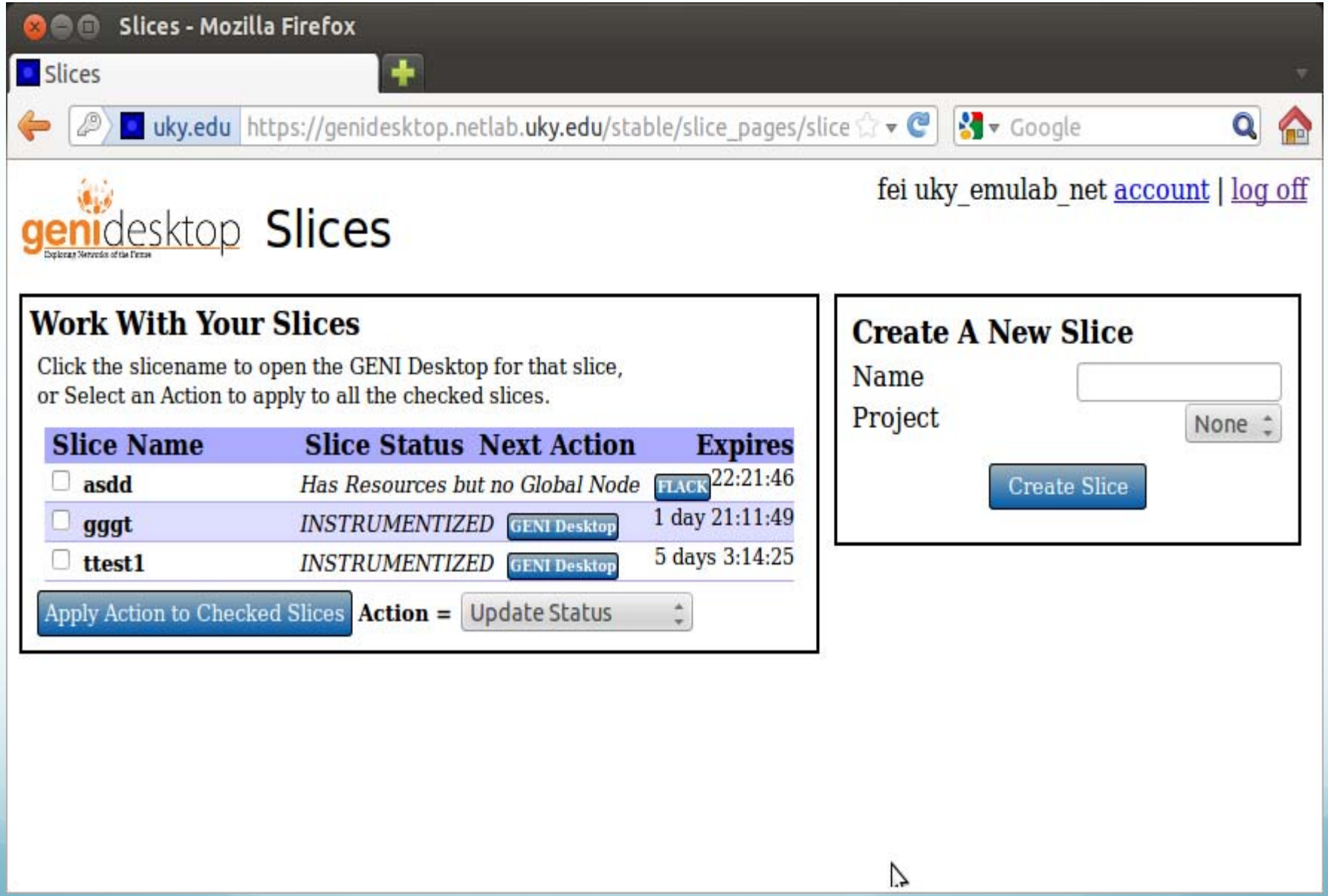
GENI Desktop

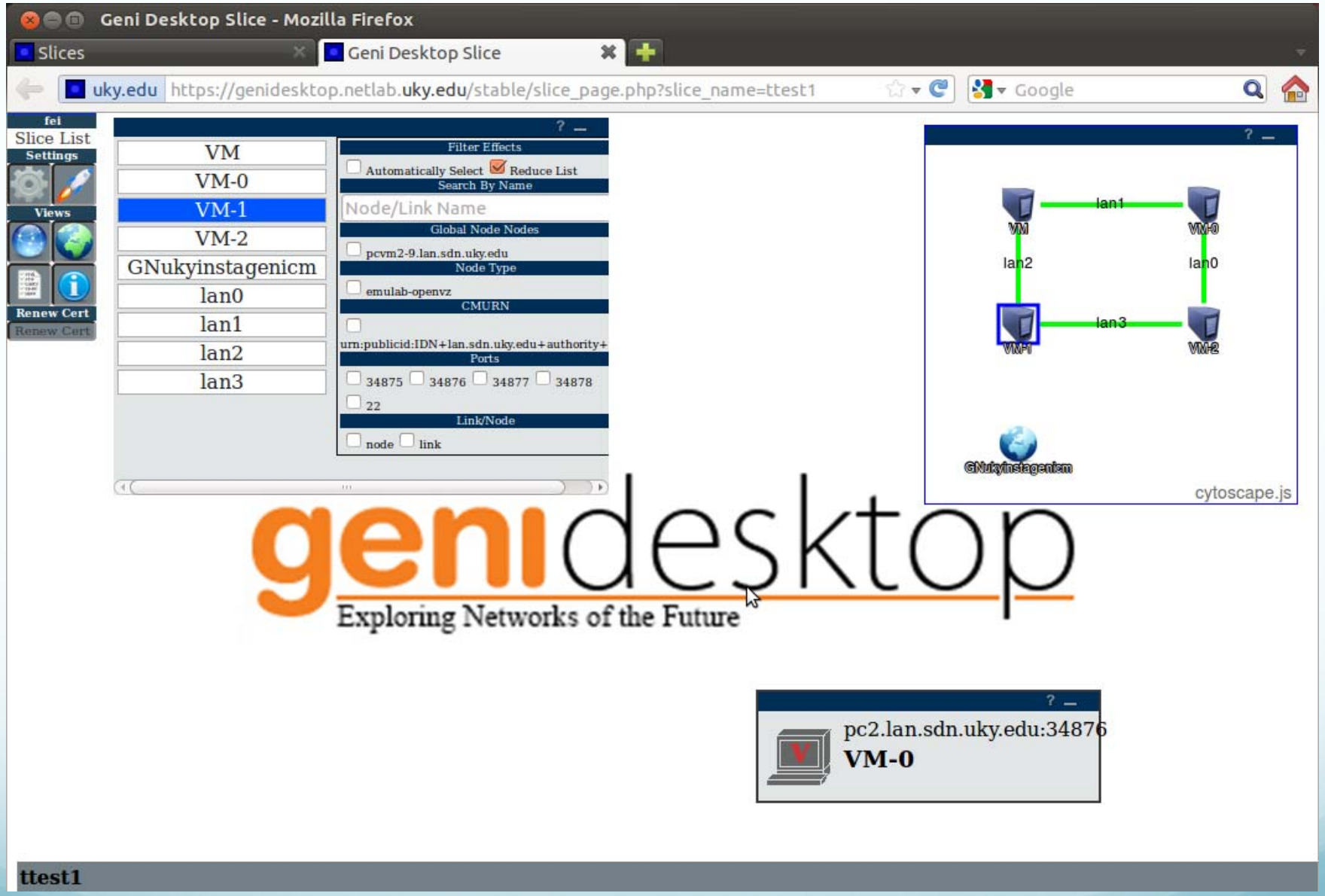
- Project team
  - James Griffioen, Zongming Fei, Hussamuddin Nasir, Xiongqi Wu, Jeremy Reed, Charles Carpenter
- Provide a single seamless web-based user interface to access many GENI tools
  - Based on a unifying abstraction – slice topology
    - List view, logical view, geographical view are provided
  - All resources are accessed through the web interface
    - Relieve the users of key management
  - A general and expandable framework
    - New functions/tools can be added as a plugin
- Some functions are motivated from our teaching experiences

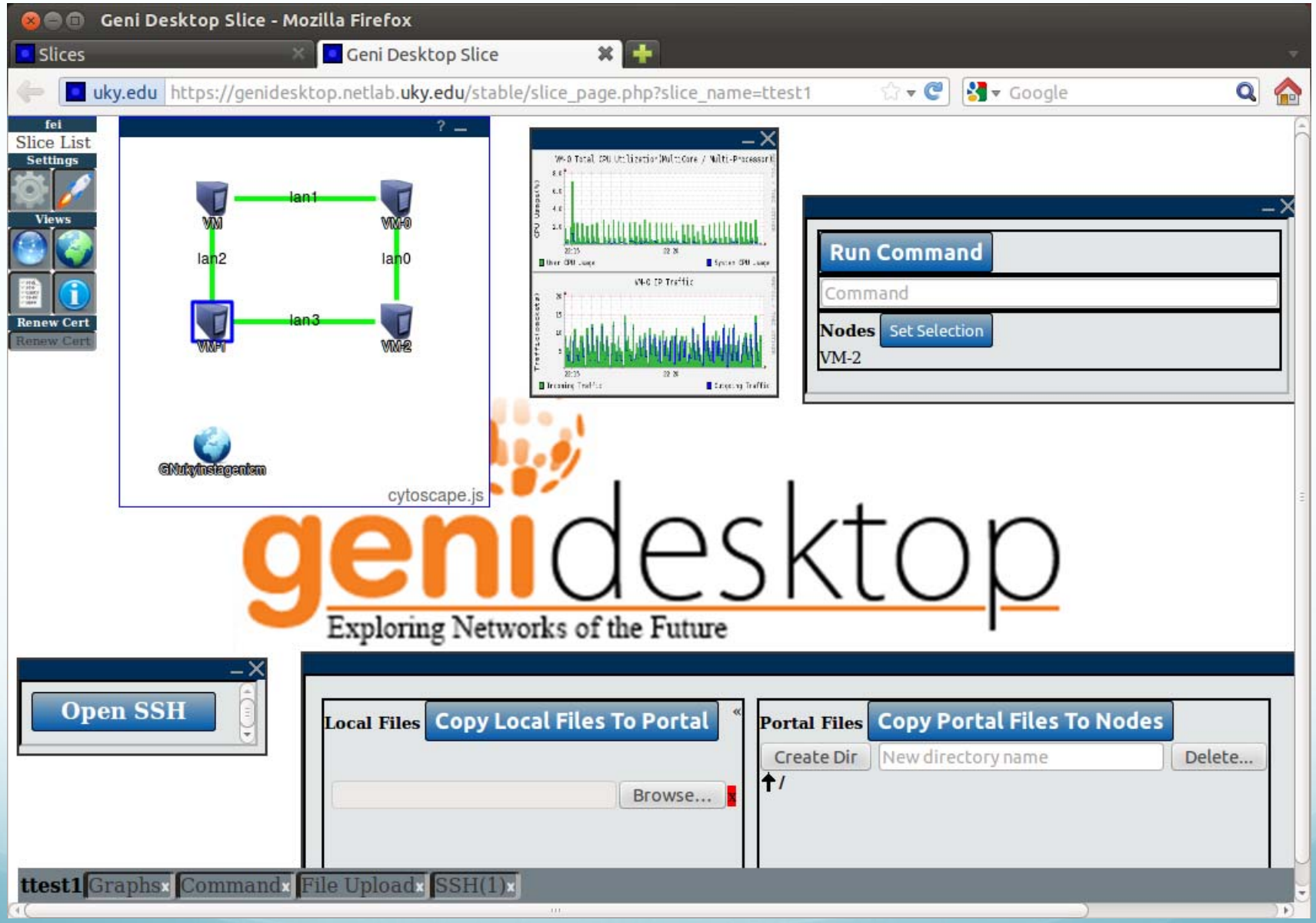
GENI Desktop

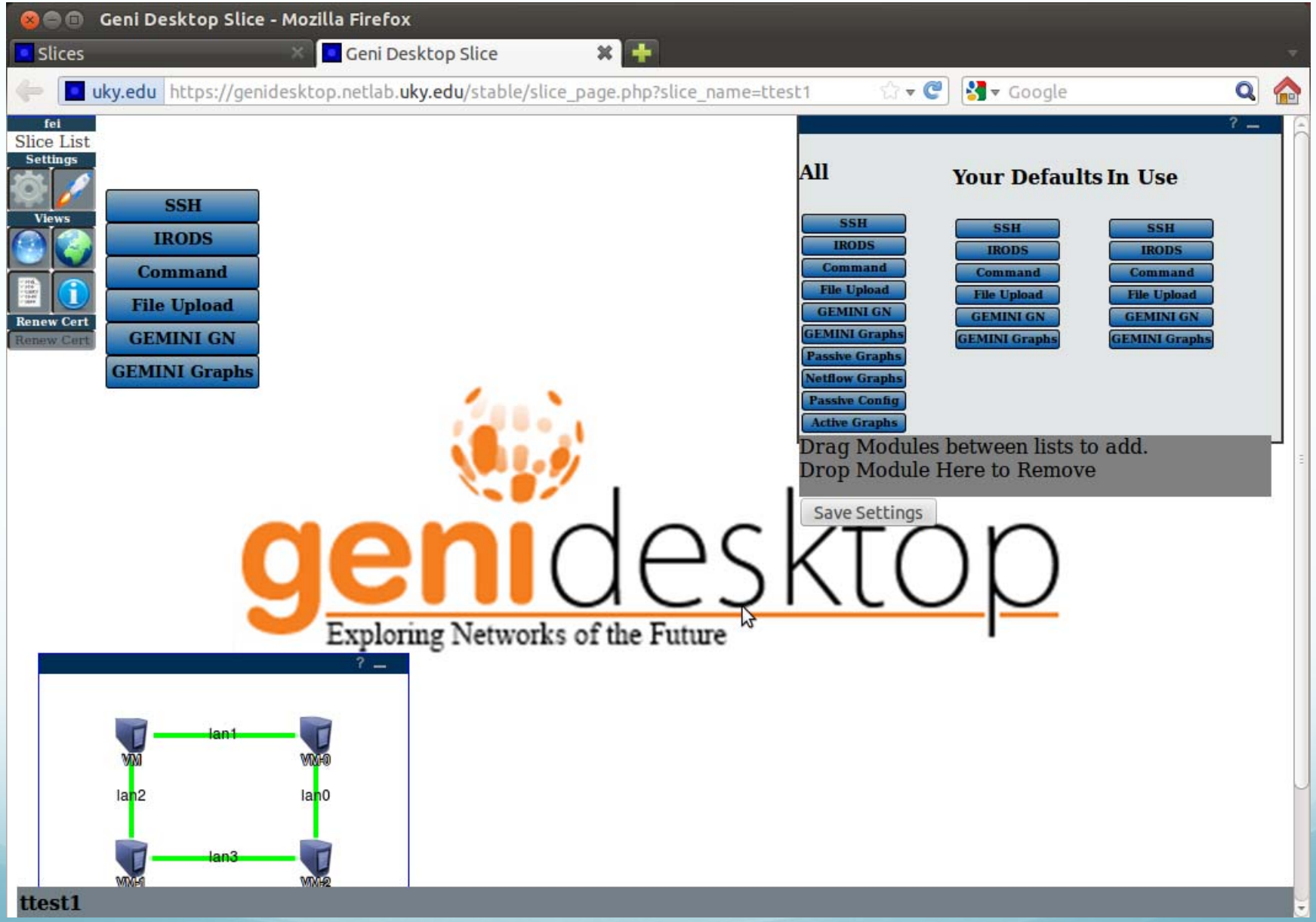
- Currently it supports (all through the web interface)
  - View slice topology
  - Login to nodes via ssh
  - Run commands on all/selected nodes
  - Copy files to all/selected nodes
  - Set up a routing path to a destination
  - Generate traffic between specified nodes
  - Invoke Flack to draw topology
  - View measurement information (GEMINI)


Genidesktop Login - Mozilla Firefox
Genidesktop Login
uky.edu
https://genidesktop.netlab.uky.edu/stable/logon/flash_logon/flash_logon_ad.ht
Google
genidesktop
Exploring Networks of the Future
Who issued your GENI account?
uky.emulab.net
Select


Slices - Mozilla Firefox
Slices
uky.edu
https://genidesktop.netlab.uky.edu/stable/slice_pages/slice
Google
genidesktop
Slices
fei uky_emulab_net account | log off
Work With Your Slices
Click the slicename to open the GENI Desktop for that slice,
or Select an Action to apply to all the checked slices.
Slice Name
Slice Status
Next Action
Expires
asdd
Has Resources but no Global Node
FLACK
22:21:46
gggt
INSTRUMENTIZED
GENI Desktop
1 day 21:11:49
ttest1
INSTRUMENTIZED
GENI Desktop
5 days 3:14:25
Apply Action to Checked Slices
Action =
Update Status
Create A New Slice
Name
Project
None
Create Slice


Geni Desktop Slice - Mozilla Firefox
Slices
Geni Desktop Slice
uky.edu
https://genidesktop.netlab.uky.edu/stable/slice_page.php?slice_name=ttest1
Google
fei
Slice List
Settings
Views
Renew Cert
Renew Cert
VM
VM-0
VM-1
VM-2
GNukyinstagenicm
lan0
lan1
lan2
lan3
Filter Effects
Automatically Select
Reduce List
Search By Name
Node/Link Name
Global Node Nodes
pcvm2-9.lan.sdn.uky.edu
Node Type
emulab-openvz
CMURN
urn:publicid:IDN+lan.sdn.uky.edu+authority+
Ports
34875
34876
34877
34878
22
Link/Node
node
link
lan1
lan2
lan0
lan3
VM
VM-0
VM-1
VM-2
GNukyinstagenicm
cytoscape.js
genidesktop
Exploring Networks of the Future
pc2.lan.sdn.uky.edu:34876
VM-0
ttest1


Geni Desktop Slice - Mozilla Firefox
Slices
Geni Desktop Slice
uky.edu
https://genidesktop.netlab.uky.edu/stable/slice_page.php?slice_name=ttest1
Google
fei
Slice List
Settings
Views
Renew Cert
Renew Cert
lan1
lan2
lan0
lan3
VM
VM-0
VM-1
VM-2
GNukyinstageniem
cytoscape.js
Run Command
Command
Nodes
Set Selection
VM-2
genidesktop
Exploring Networks of the Future
Open SSH
Local Files
Copy Local Files To Portal
Browse...
Portal Files
Copy Portal Files To Nodes
Create Dir
New directory name
Delete...
/
ttest1
Graphs
Command
File Upload
SSH(1)


Geni Desktop Slice - Mozilla Firefox
Slices
Geni Desktop Slice
uky.edu
https://genidesktop.netlab.uky.edu/stable/slice_page.php?slice_name=ttest1
Google
fei
Slice List
Settings
Views
Renew Cert
Renew Cert
SSH
IRODS
Command
File Upload
GEMINI GN
GEMINI Graphs
All
Your Defaults In Use
SSH
IRODS
Command
File Upload
GEMINI GN
GEMINI Graphs
Passive Graphs
Netflow Graphs
Passive Config
Active Graphs
Drag Modules between lists to add.
Drop Module Here to Remove
Save Settings
genidesktop
Exploring Networks of the Future
lan1
lan2
lan0
lan3
VM1
VM-0
VM-1
VM-2
ttest1

Conclusions

- It is feasible to give relatively complicated programming assignments using GENI.
- It is attractive to use GENI because of several unique features (such as easy setup, GUI, user-controlled distributed resources).
- It is helpful to students to understand the concepts of the project topics implemented because of real implementations rather than simulations.
- We can improve by providing a unified graphical user interface (such as GENI Desktop) for students to access all aspects of the experiment life cycle.

Thank You!

And

Questions?

This material is based upon work supported in part by the National Science Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of GPO Technologies, Corp, the GENI Project Office, or the National Science Foundation.