

Tutorial: Advanced Topics in Networking Experiments using GENI

Niky Riga, Vic Thomas, Tim Upthegrove
GENI Project Office
24 October 2012

- Overview (tutorial, resources, Omni)

... prework ...

- Mesoscale GENI Infrastructure

... hands on ...

- Split into Groups

- Using Openflow to affect your topology
- Using software routers

- Wrap Up

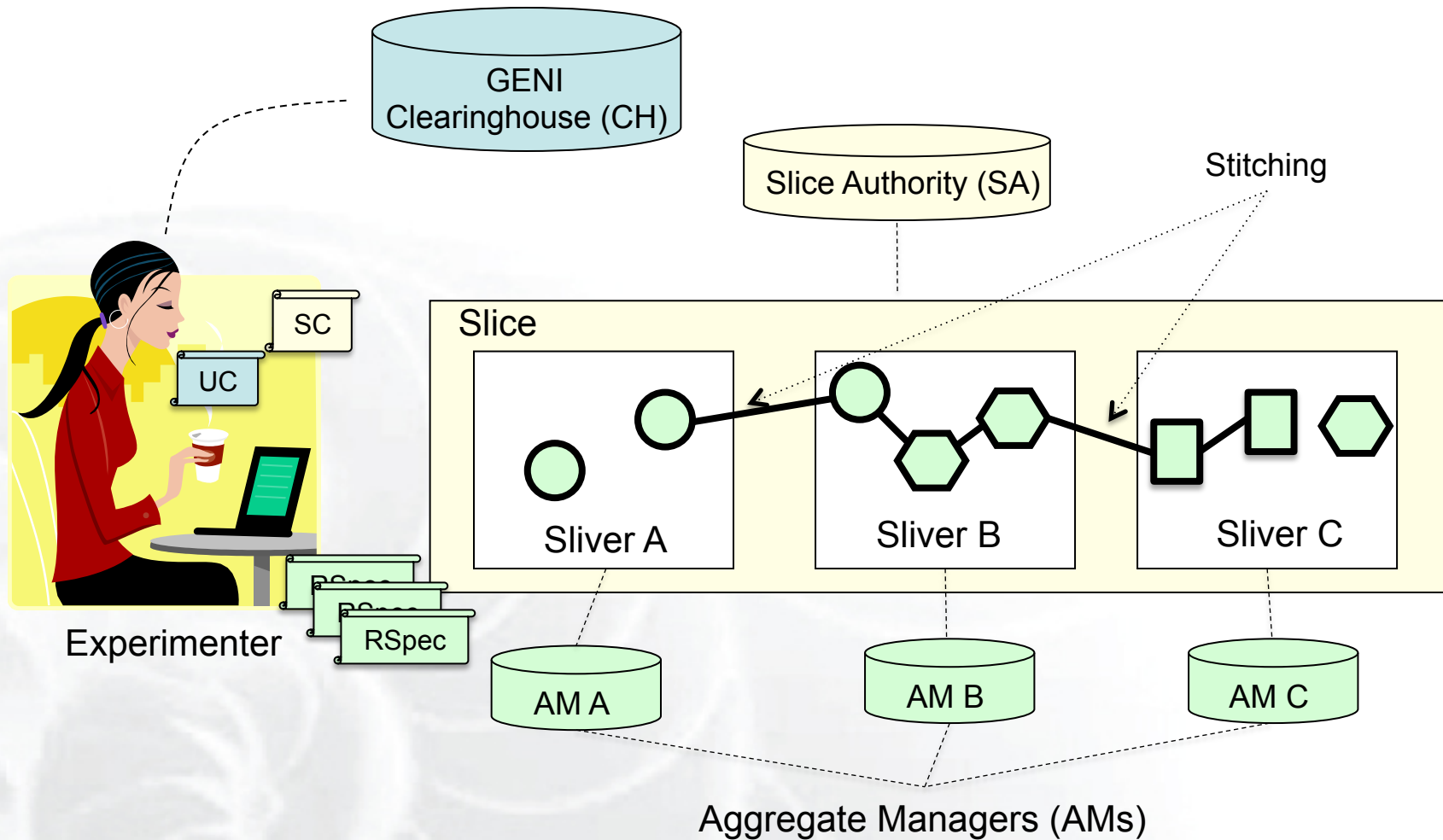
... hands on ...

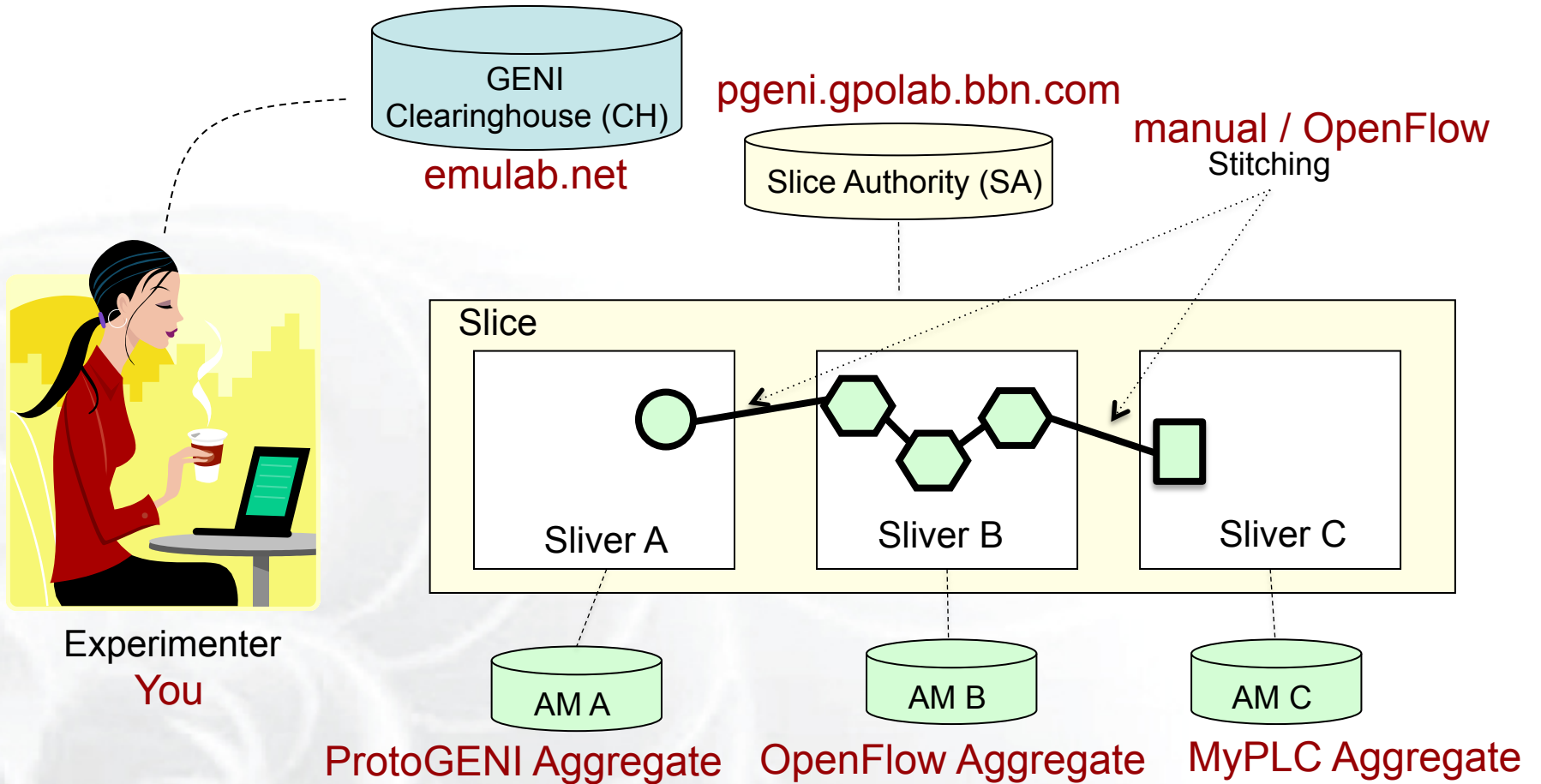
- Learn how to automate your experiment
 - Install scripts
 - What context are install scripts executed at?
 - Where are the logs for the execution?
 - When are they executed?
 - Logging in to nodes
 - How to access my nodes?
 - What is my username, password?
 - Use remote execution to orchestrate your experiment

- How can multiple user can access the same node?
- Use Mesoscale for your experiment
 - Run Layer 2 (and up) experiments across the country
 - Control your topology

Looking behind the scenes Disclaimer

- GENI is evolving fast
 - What is true today, might change tomorrow
- Use tutorial resources as a pointer
- Best place to get up-to-date info is the GENI wiki
<http://geni.net/experiment>
- Email us with questions (help@geni.net)

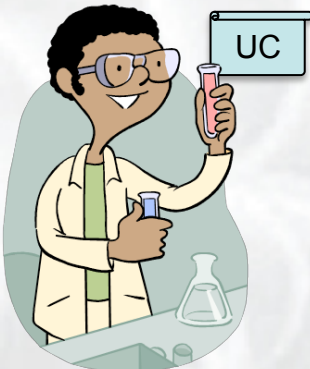
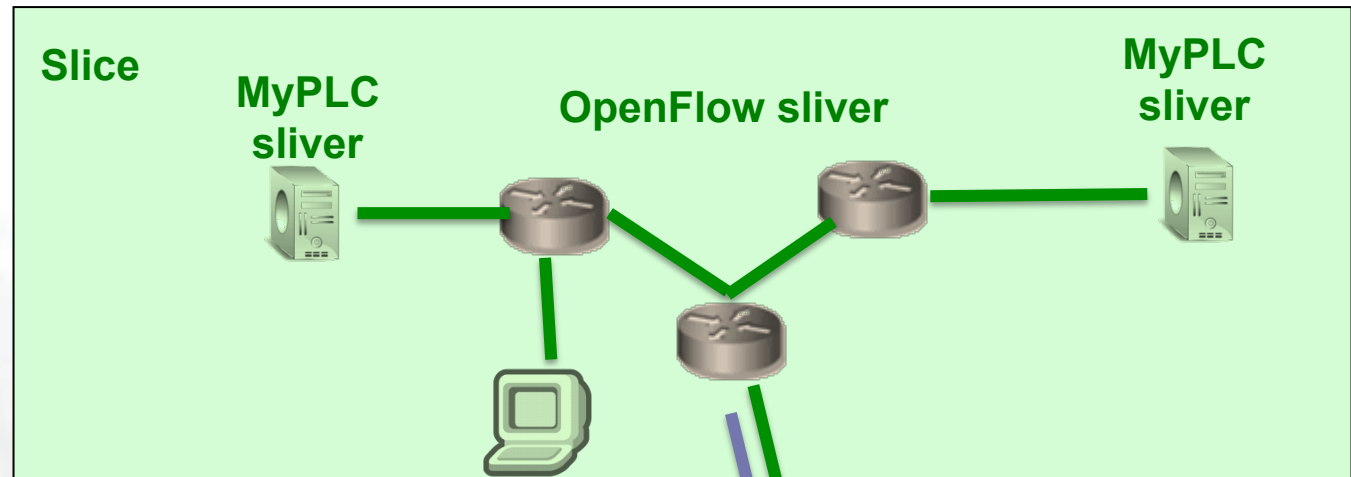




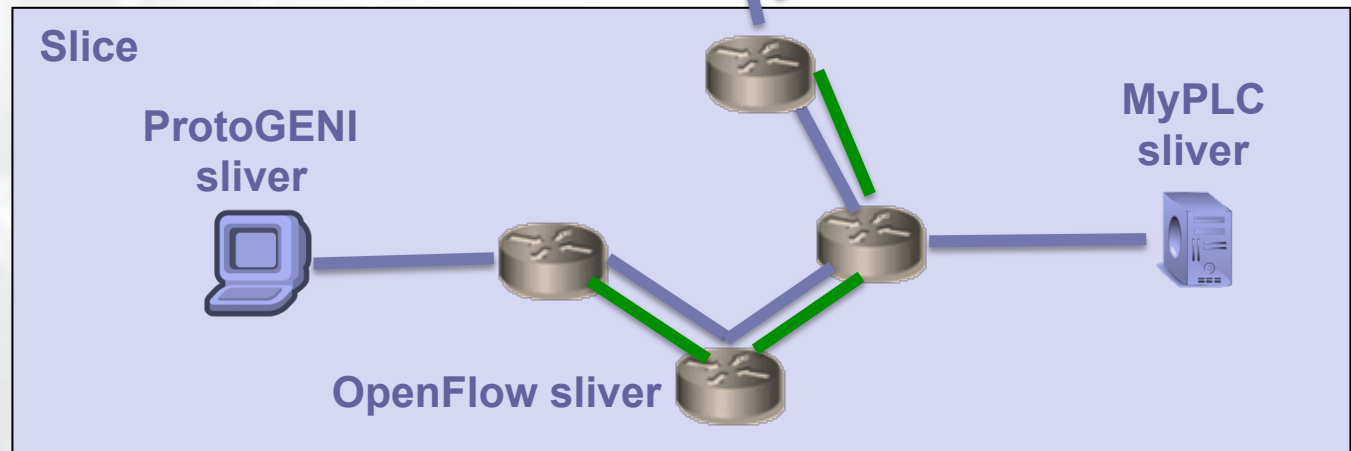
Today's GENI Experiments



Omni

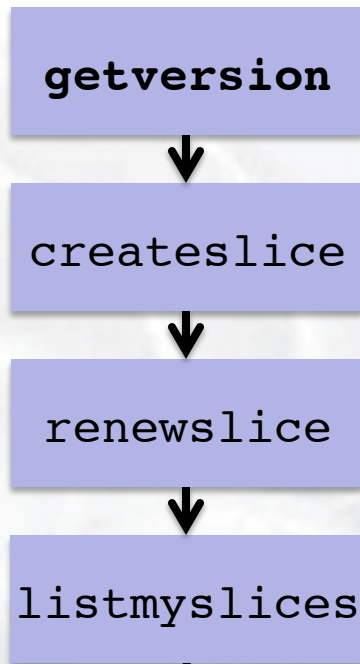


Omni

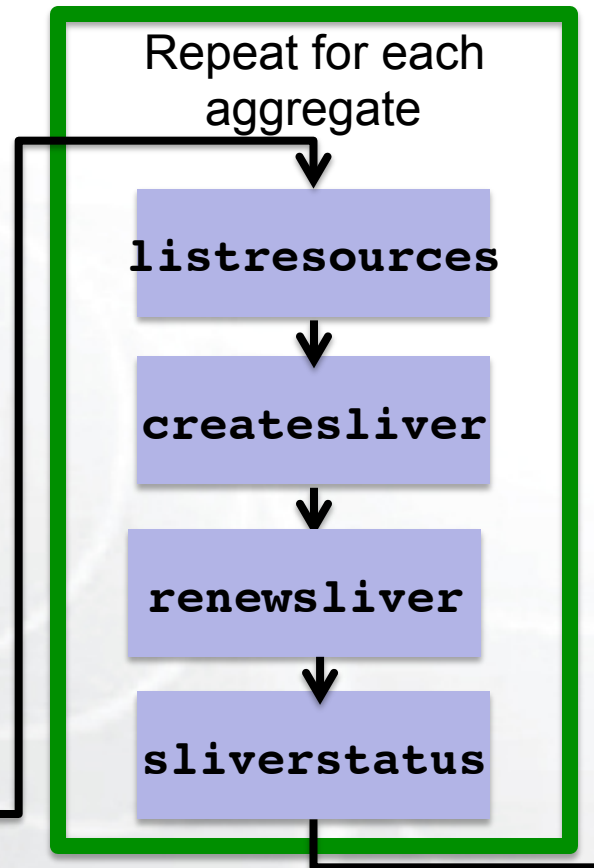


Omni Command Workflow

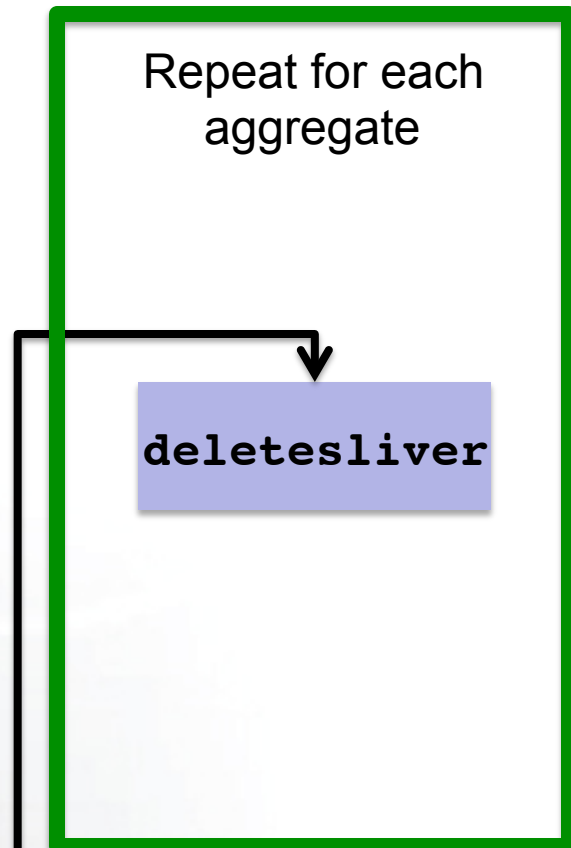
Create Slice



Create Sliver



Cleanup



Legend: **AM API command**

- `omni.py -h`
 - Lists all commands and their arguments
 - Lists all command line options
 - Lists Omni version
 - Lists url to find out more information about Omni
- Omni Troubleshooting page:
<http://trac.gpolab.bbn.com/gcf/wiki/OmniTroubleShoot>

- You can write custom Python scripts
 - Call existing Omni functions
 - Parse the Output
- Example: readyToLogin.py
 - Calls sliverstatus
 - Parses output of sliverstatus
 - Determines ssh command to log into node
- More examples distributed with Omni

- Overview (tutorial, resources, Omni)

... prework ...

- Mesoscale GENI Infrastructure

... hands on ...

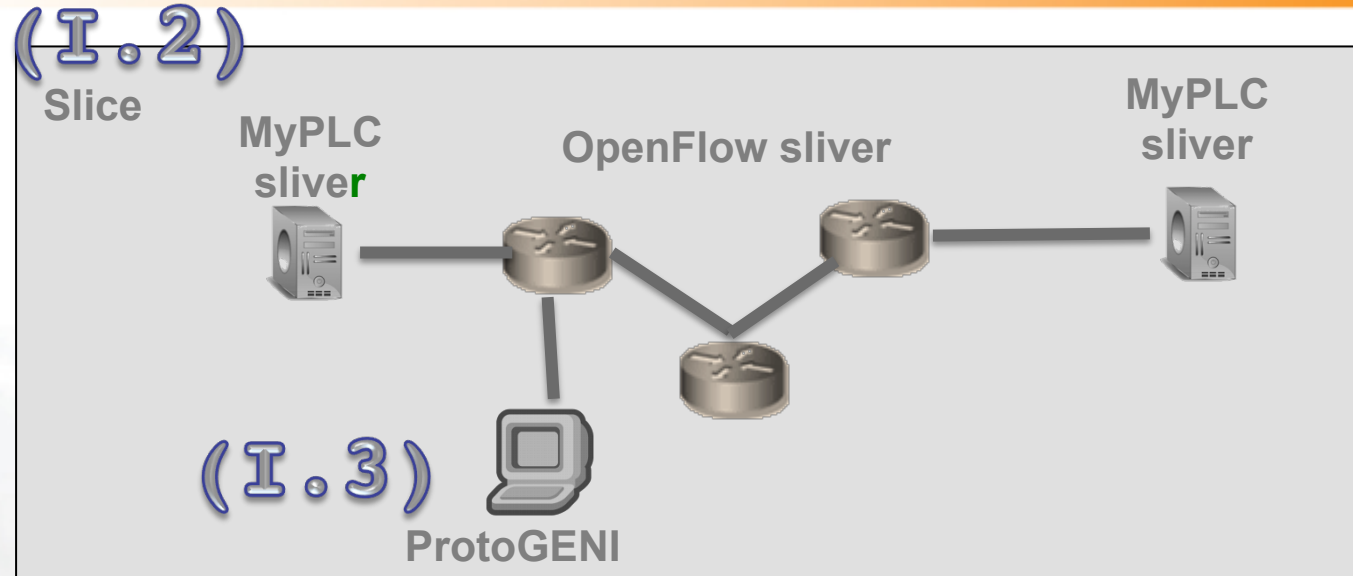
- Split into Groups
 - Using Openflow to affect your topology
 - Using software routers

- Wrap Up

... hands on ...



Omni



I.1 Get to know Omni

```
omni.py getversion
```

I.2 Make a slice

```
omni.py createslice slicename
```

```
omni.py renewslice slicename date
```

```
omni.py listmyslices username
```

I.3 Make a ProtoGENI sliver

```
omni.py createsliver slicename reqRSpec
```

```
omni.py sliverstatus slicename
```

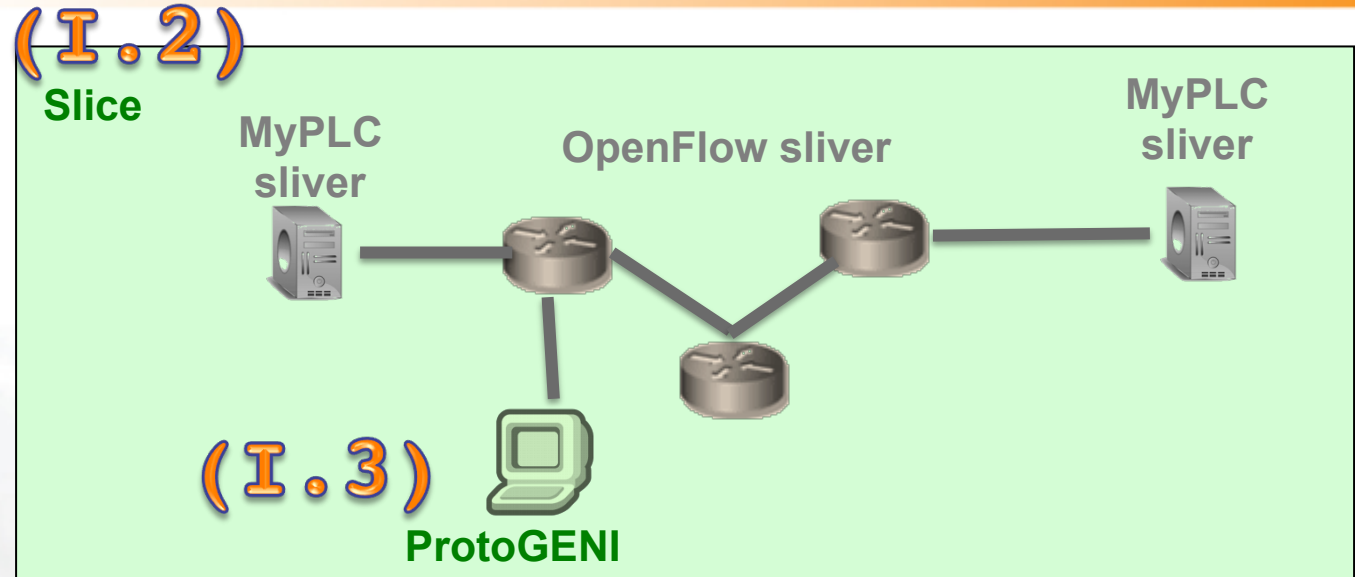
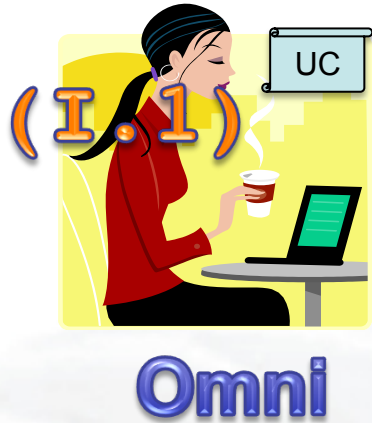
Note:

-a *aggregate*

to specify an aggregate manage

-o

to save the output to a file



I.1 Get to know Omni

```
omni.py getversion
```

I.2 Make a slice

```
omni.py createslice slicename
```

```
omni.py renewslice slicename date
```

```
omni.py listmyslices username
```

I.3 Make a ProtoGENI sliver

```
omni.py createsliver slicename reqRSpec
```

```
omni.py sliverstatus slicename
```

Note:

-a *aggregate*

to specify an aggregate manage

-o

to save the output to a file

- Overview (tutorial, resources, Omni)

... prework ...

- Mesoscale GENI Infrastructure

... hands on ...

- Split into Groups
 - Using Openflow to affect your topology
 - Using software routers

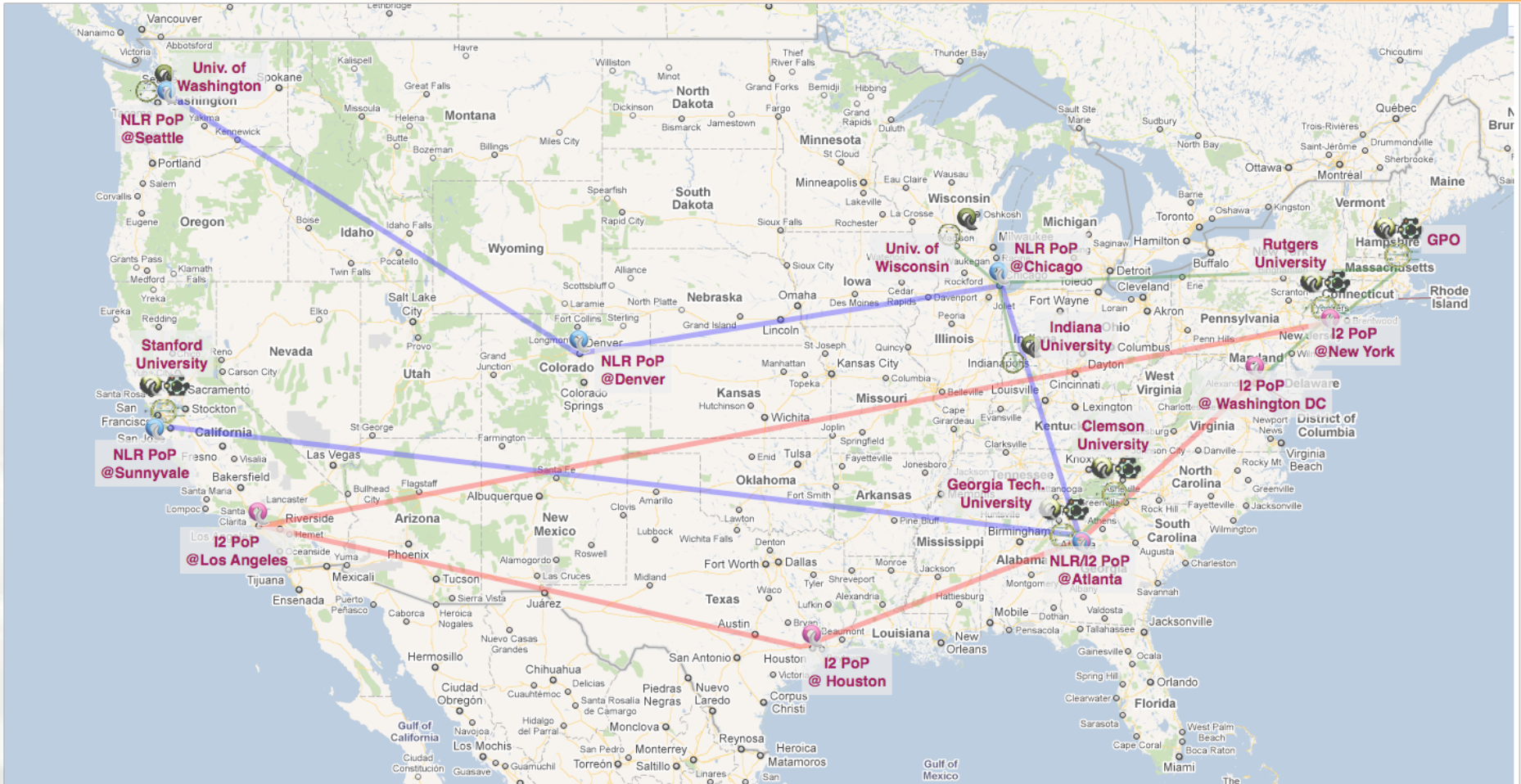
- Wrap Up

... hands on ...

OpenFlow Mesoscale deployment :

- is a prototype GENI infrastructure
- spans multiple sites connected over Layer 2
 - 2 backbone, 7 regionals, 8 campuses
- is open to experimenters that want to gain early access to a Layer 2 infrastructure that combines multiple aggregates.
- includes :
 - OpenFlow aggregates
 - Private PlanetLab aggregates (MyPLC)
 - ProtoGeni aggregates

Where are the tutorial resources?



10 OpenFlow AM (2 backbones (NLR, 12) + 8 campuses)

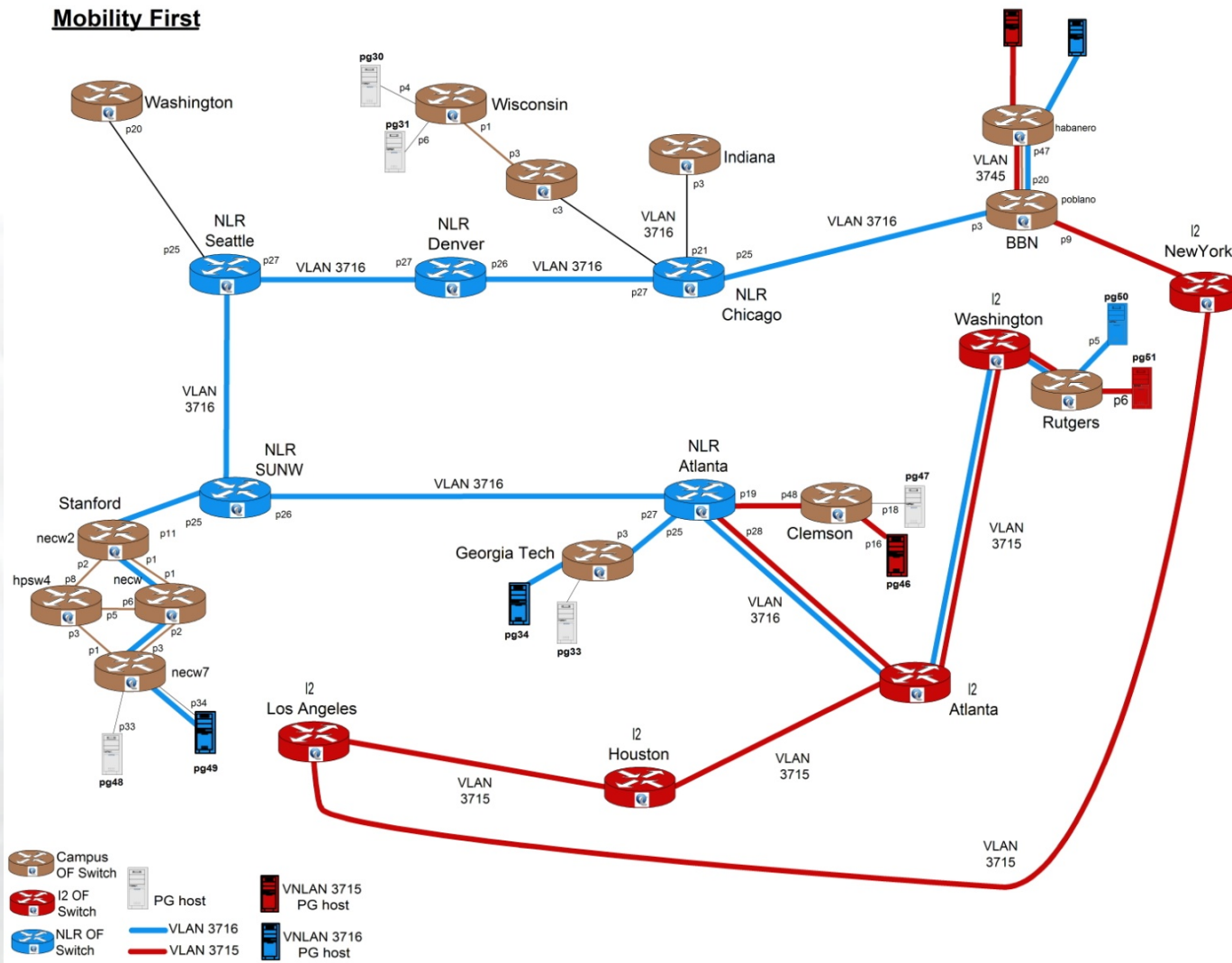
8 MyPLC AM

- Clemson, GaTech, GPO, Indiana, Rutgers, Stanford, Wisconsin, Washington

2 ProtoGENI AM (GPO, Utah)

- Separate control and data plane
 - Control plane over commodity Internet
 - Data plane is Layer 2 over GENI backbone
- All hosts have one interface directly connected to an OpenFlow switch
- Pre-provisioned ~60 Dataplane IP subnets
- Out of band reservation of IP subnet
- Out of band reservation of eth_types for L2 experiments.
- **2 VLANs over the same resources, providing different topologies**

MobilityFirst: Backbone Physical Topology

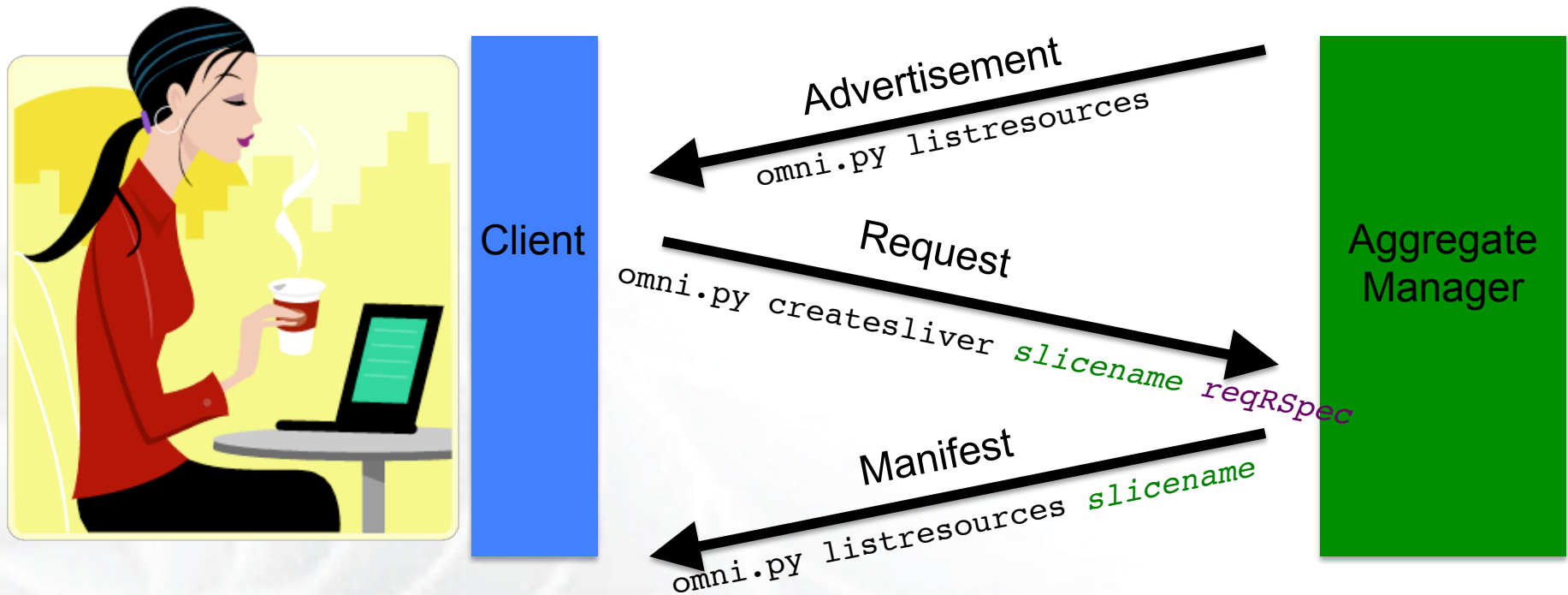


Resource Specification Document (RSpec)

- XML document that describes resources
 - hosts, links, switches, etc
- today only 1 RSpec version is used
 - **GENI v3**
 - AM extensions
 - Client extensions
 - Tool extensions
- Other versions are supported
 - SFA
 - ProtoGENI v2.0

```
<?xml version="1.0" encoding="UTF-8"?>
<rspec xmlns="http://www.protogeni.net/
resources/rspec/2"
xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance"
xsi:schemaLocation="http://
www.protogeni.net/resources/rspec/2
http://
www.protogeni.net/resources/rspec/2/
request.xsd" type="request" >
  <node client_id="my-node"
exclusive="false">
    <sliver_type name="emulab-openvz" />
  </node>
</rspec>
```


The Three Types of RSpecs



Advertisement RSpec : What does the AM have?

Request RSpec : What does the Experimenter want?

Manifest RSpec: What does the Experimenter have?

Too many RSpecs

- There is an art in writing well formed RSpecs
- Do not try to write one from scratch
 - Find example RSpecs and use them as your base
 - Use tools, like Flack, to generate sample RSpecs for you
 - When appropriate modify advertisement RSpecs



- Many people will be accessing the resources, so some calls might fail. Wait a bit and try again!
- Omni is a command line tool, copy-paste is your friend
- You can copy-paste between your computer and the VM.

- Overview (tutorial, resources, Omni)

... prework ...

- Mesoscale GENI Infrastructure

... hands on ...

- Split into Groups
 - Using Openflow to affect your topology
 - Using software routers

- Wrap Up

... hands on ...

PlanetLab: Modifying an ad RSpec

```
<RSpec type="SFA">
  <network name="plc">
    <site id="s1">
      <name>MyPLC</name>
      <node id="n1">
        <hostname> host1.geni.net </hostname>
        <sliver/>
      </node>
      <node id="n2">
        <hostname> host2.geni.net </hostname>
      </node>
    </site>
  </network>
</RSpec>
```

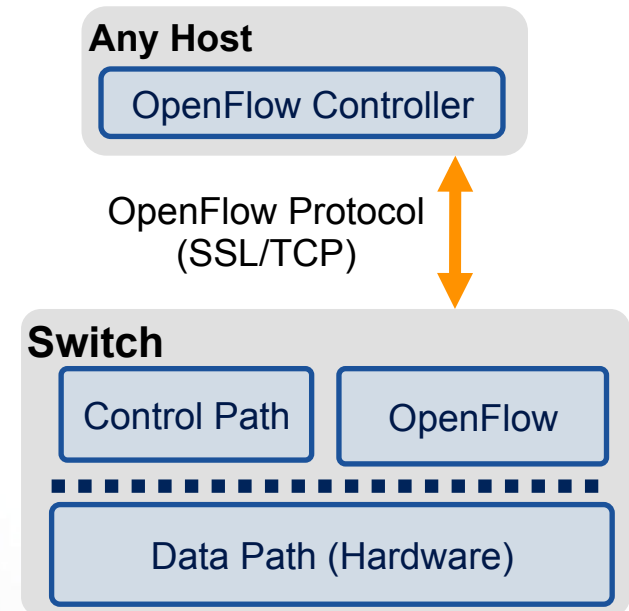
Insert a **<sliver/>** tag in the node tag, for the node you want to reserve

OpenFlow is an API

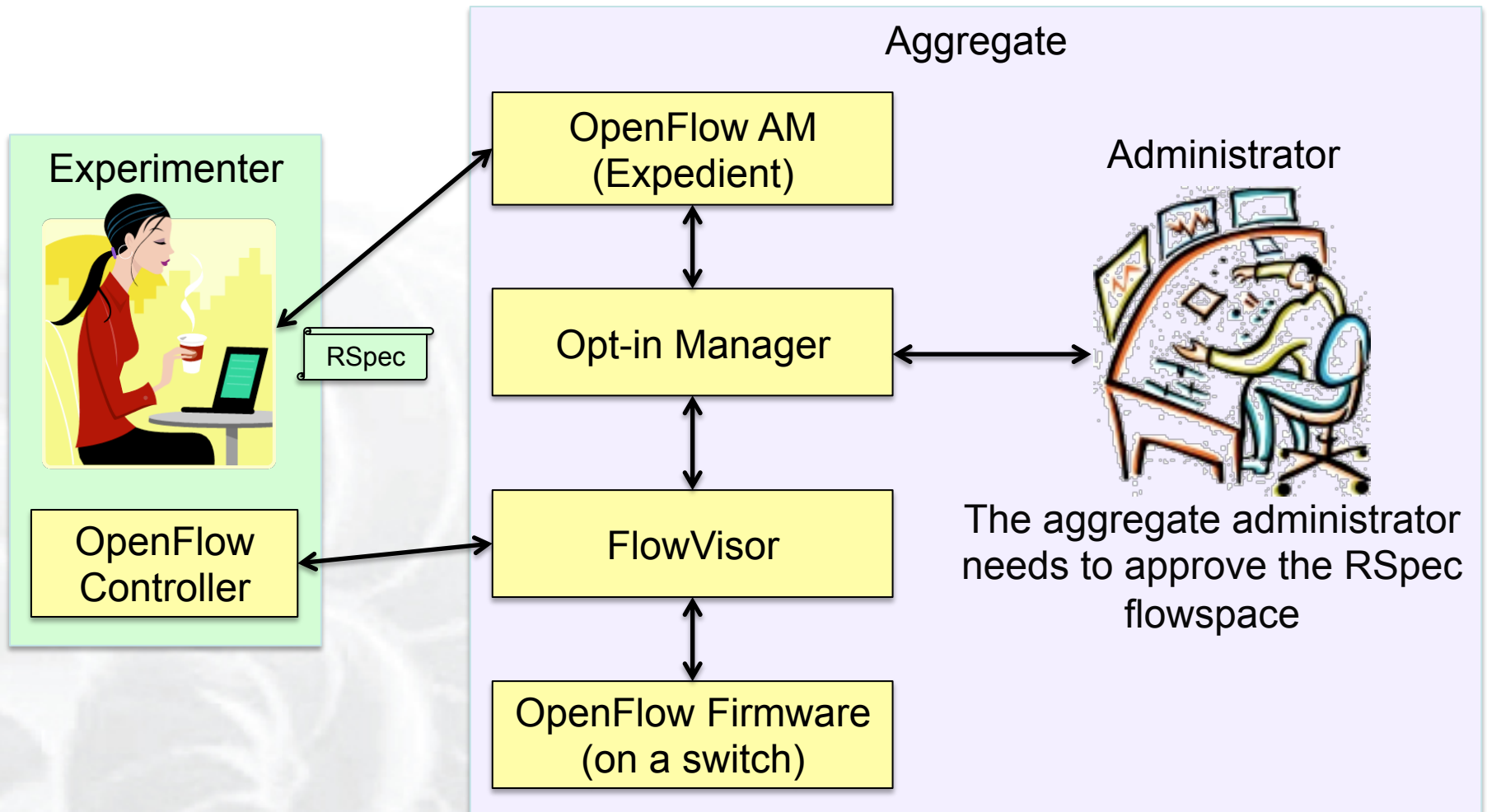
- Controls how packets are forwarded
- Implemented on COTS hardware
- Make deployed networks programmable

FlowSpace describes packet flows :

- **Layer 1**: Incoming port on switch
- **Layer 2**: Ethernet src/dst addr, type, vlanid
- **Layer 3**: IP src/dst addr, protocol, ToS
- **Layer 4**: TCP/UDP src/dst port



An **experimenter** can control multiple FlowSpaces



```
<resv_rspec type="openflow" version="2">  
  <user ... />  
  <project ... />  
  <slice controller_url="tcp:host:port" expiry="1326139200" />  
  <flowspace>  
    <switch urn="urn: ... " />  
    <port urn="urn: ... " />  
    <pkt_field from="..." to="..." />  
  </flowspace>>
```

expiry is in Unix timestamp

pkt_field can be :

- dl_src, dl_dst, dl_type, dl_vlan
- nw_src, nw_dst, nw_proto, nw_tos
- tp_src, tp_dst

- Use OpenFlow to route your packets in the network
- Use OpenFlow to rewrite header fields :
 - OFPAT_SET_VLAN_VID, /* Set the 802.1q VLAN id. */
 - OFPAT_SET_VLAN_PCP, /* Set the 802.1q priority. */
 - OFPAT_STRIP_VLAN, /* Strip the 802.1q header. */
 - OFPAT_SET_DL_SRC, /* Ethernet source address. */
 - OFPAT_SET_DL_DST, /* Ethernet destination address. */
 - OFPAT_SET_NW_SRC, /* IP source address. */
 - OFPAT_SET_NW_DST, /* IP destination address. */
 - OFPAT_SET_NW_TOS, /* IP ToS (DSCP field, 6 bits). */
 - OFPAT_SET_TP_SRC, /* TCP/UDP source port. */
 - OFPAT_SET_TP_DST, /* TCP/UDP destination port. */
- Caveat, not all actions are done in hardware (OVS)

- Not wire-speed
- Easy to deploy
- Provide ability to implement new routing protocols over a L2 network fast
 - XIA , MobilityFirst demos
- An easy way to bandwidth limit your slice

- Specify in the rspec
 - Where to download software from
 - Usually comes with an install-script.sh
 - What commands to run at boot time
- It runs in the context of a the “geni” user
 - It is not your user account
 - It does have sudo priviledges

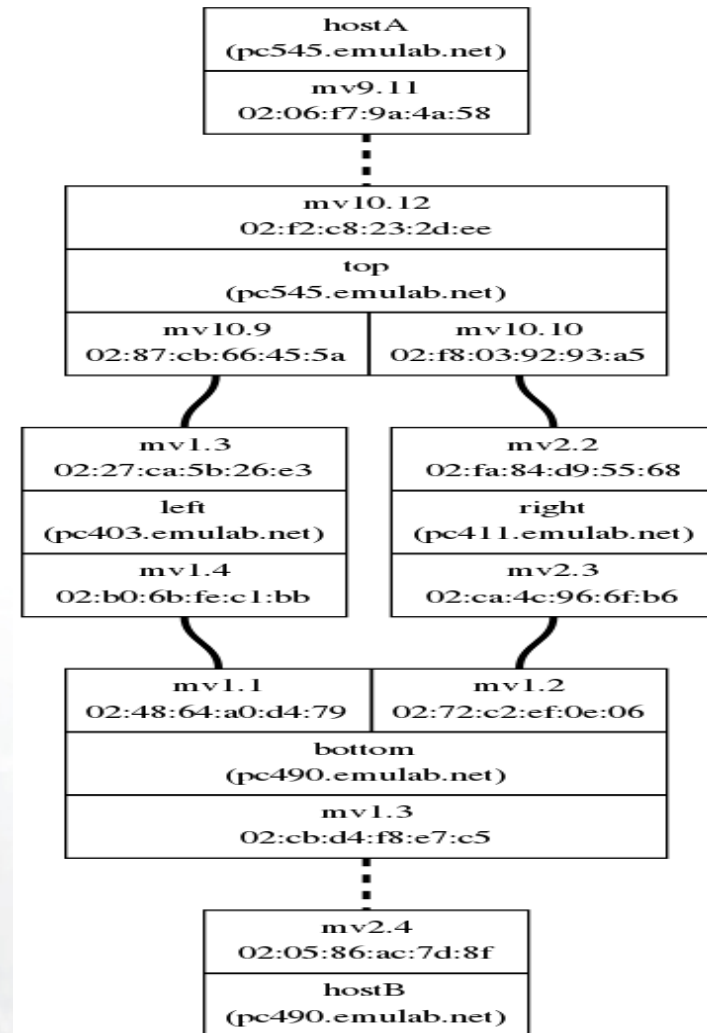

```
# Change to dir. with installed software  
cd /local
```

```
##### Check if file is there #####  
if [ ! -f "./installed.txt" ]  
then  
    ##### Create the file #####  
    sudo touch "./installed.txt"  
    ##### Run one-time commands #####  
    #Install necessary packages  
    # Install custom software  
    ## Reboot if needed  
  
fi  
##### Run Boot-time commands  
# Start common services  
hn=`echo $HOSTNAME | cut -d'.' -f 1`  
if [ $hn == "server" ]  
then  
    # Host specific commands  
fi
```


- Overview (tutorial, resources, Omni)
... prework ...
- Mesoscale GENI Infrastructure
... hands on ...
- Split into Groups
 - Using Openflow to affect your topology
 - Using software routers
- Wrap Up
... hands on ...

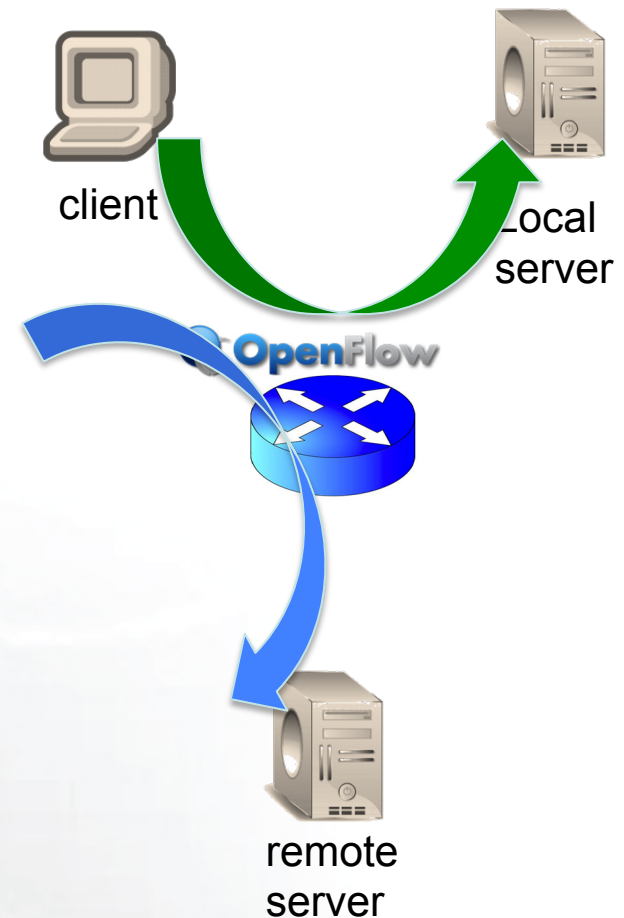
Click Example Experiment

- You will run your own non-IP multipath router
 - Odd checksums left
 - Even checksums right
- 6 VMs in Utah
 - 4 routers
 - 2 end hosts



OpenFlow Experiment

- You will run your own OpenFlow experiment
- Use Mesoscale resources
- In-network diverged traffic
 - Send requests to a local host



- Overview (tutorial, resources, Omni)
... prework ...
- Mesoscale GENI Infrastructure
... hands on ...
- Split into Groups
 - Using Openflow to affect your topology
 - Using software routers
- Wrap Up
... hands on ...

- When your experiment is done, you should always release your resources.
 - Archive your data
 - Delete all your slivers
 - OpenFlow slivers might outlive your slice, make sure you delete them before your slice expires
 - When appropriate delete your slice

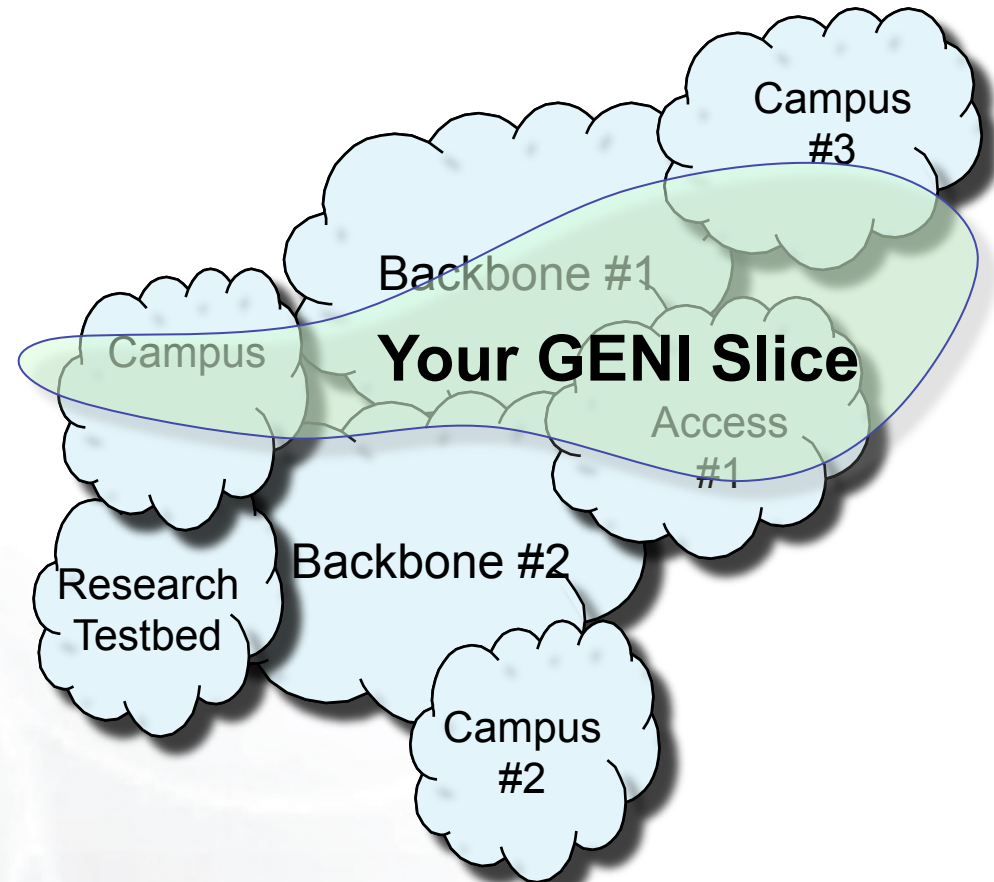


Running Experiments on GENI

- Get an account to run experiments on GENI

- Contact us at help@geni.net

- More information on Experimenter Portal:
 - <http://groups.geni.net/geni/wiki/ExperimenterPortal>



- Fill out the survey
- Attend more tutorials
- Come to Experimenter Drop-in session
Wednesday 3-5pm
- Come to coding sprint and start your experiment
Thursday 1-5pm



Omni



Omni

Happy experimenting!

Backup Slides

- Primary Information
 - `omni.py -h`
 - Omni Troubleshooting page:
<http://trac.gpolab.bbn.com/gcf/wiki/OmniTroubleShoot>
 - For Omni specific help: gcf-dev@geni.net
 - For general GENI help: help@geni.net
- Omni Wiki (install instructions, documentation, bug reporting):
<http://trac.gpolab.bbn.com/gcf/wiki/Omni>
- For an overview of GENI Experimentation using Omni:
 - <http://groups.geni.net/geni/wiki/GENIExperimenter>
- Example experiment walk-through:
 - <http://groups.geni.net/geni/wiki/GENIExperimenter/ExperimentExample>
- Example script walk-throughs:
 - <http://trac.gpolab.bbn.com/gcf/wiki/OmniScriptingWithOptions> and
<http://trac.gpolab.bbn.com/gcf/wiki/OmniScriptingExpiration>

- `omni.py getversion`
- `omni.py createslice slicename`
- `omni.py renewslice slicename date`
- `omni.py listmyslices username`
- `omni.py createsliver slicename requestRSpec`
- `omni.py sliverstatus slicename`
- `omni.py listresources [slicename]`
-t ProtoGENI 2 to request PGV2 Rspecs
- `omni.py deletesliver slicename`

Other Omni command line arguments

- c *omni_config* to use another *omni_config*
- f *plc* to use a different framework
- t ProtoGENI 2 to specify the version of the Rspec