

# Tutorial: Introduction to Omni

Niky Riga, Sarah Edwards  
GENI Project Office  
23 October 2012

```
$ omni.py createsliver aliceslice myRSpec.xml
INFO:omni:Loading config file omni_config
INFO:omni:Using control framework pgeni
INFO:omni:Slice urn:publicid:IDN+pgeni.gpolab.
           expires within 1 day on 2011-07-07
INFO:omni:Creating sliver(s) from rspec file
INFO:omni:Writing result of createsliver for
INFO:omni:Writing to 'aliceslice-manifest-rspe
INFO:omni: -----
INFO:omni: Completed createsliver:

Options as run:
  aggregate: https://www.emulab.
  framework: pgeni
  native: True

Args: createsliver aliceslice myRSpec.xml

Result Summary: Slice urn:publicid:IDN+pgeni
Reserved resources on https://www.emulab.net/p
Saved createsliver results to aliceslice-man
INFO:omni: -----
```

Prework

Configure Omni

Run your experiment

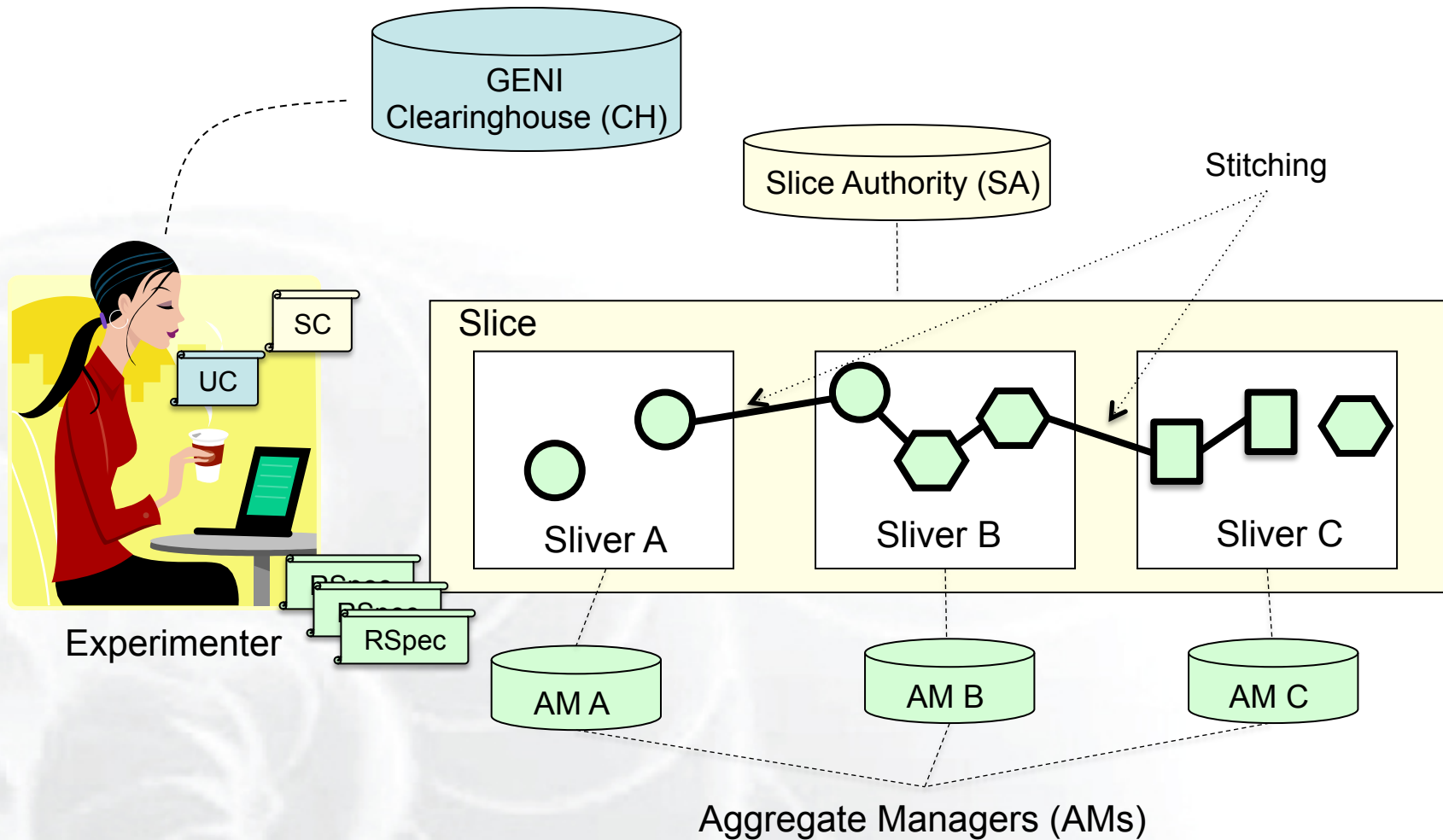
Cleanup resources

Wrap up

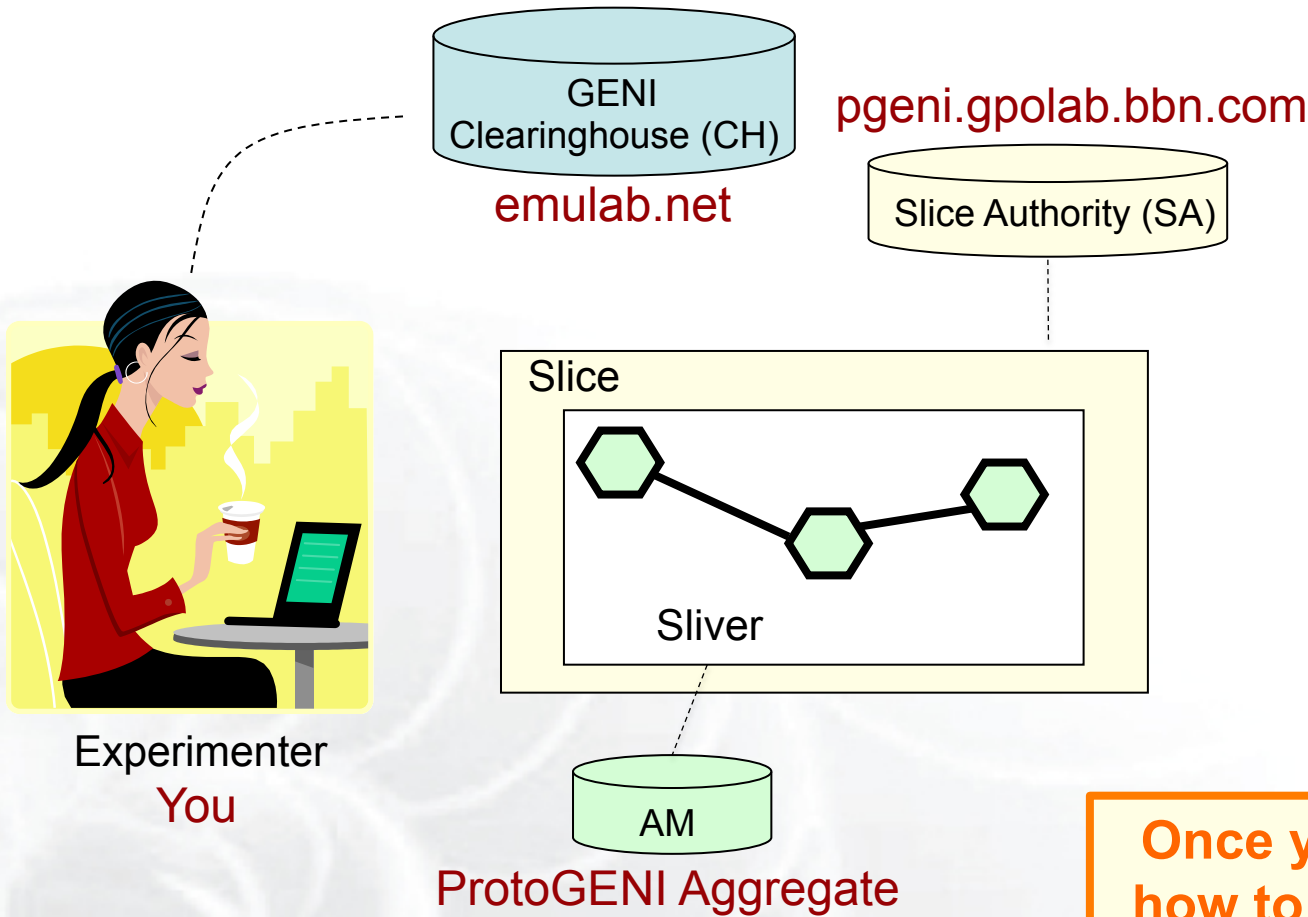
- Learn what is happening behind the scenes
  - How is an experiment setup ?
  - What are the key participants?
- Use Mesoscale for your experiment
  - Run Layer 2 (and up) experiments

# 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://groups.geni.net/geni/wiki/GeniExperiments>
- Email us with questions (help@geni.net)

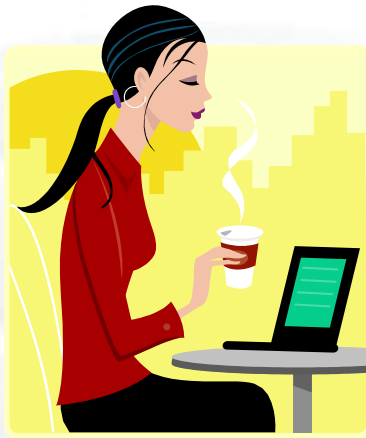


# For the Tutorial: Keep it simple



**Once you understand how to run this simple experiment, scaling up is easy!**

# GENI Accounts, GENI Credentials, GENI Certificates, GENI Keys



Experimenter

**GENI Account** : Usually this refers to your web UI account (e.g. pgeni.gpolab.bbn.com, emulab.net, etc)

**GENI Certificate** : identifies user; created by CH and you can download it through the webUI, it is an ssl certificate

**GENI Credential** : identifies user's permissions; you almost never have to handle; tools query it from CH, SA

**GENI Keys** : Access to compute resources are through ssh-keys, you can use your own or use the one GENI created for you

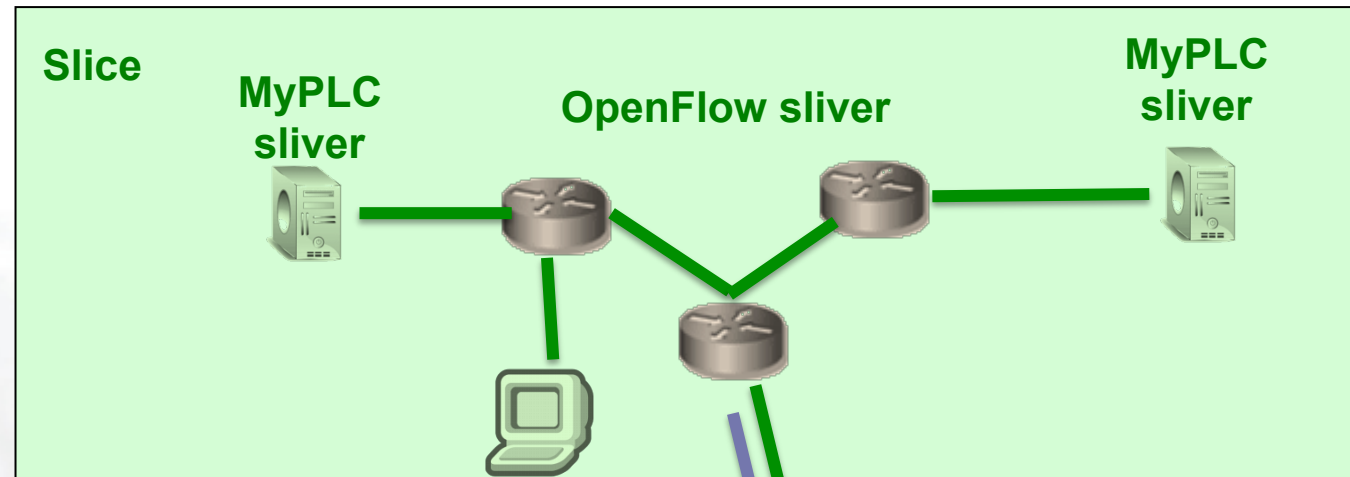
**Password** : GENI account password; use to login to ProtoGENI web UI

**Passphrase** : like a password but for you certificate and you ssh keys

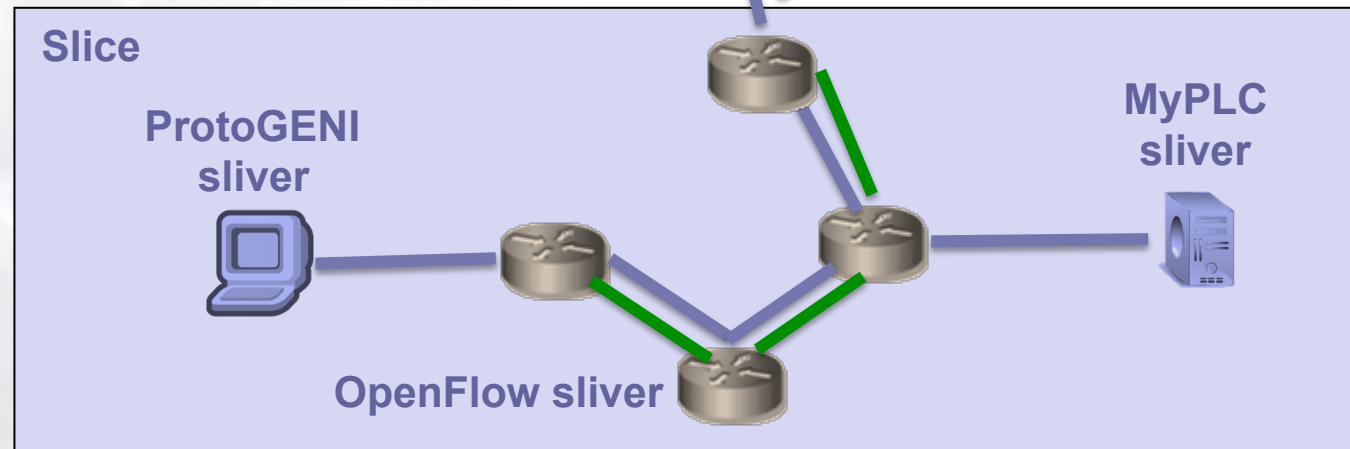
# Today's GENI Experiments



Omni



Omni





# Omni: Resource Reservation tool

- A command line experimenter tool
- Create slices and slivers using the GENI AM API
- Written in and scriptable from Python
- Use existing accounts
  - ProtoGENI
  - PlanetLab
  - GENI Portal
- **Works with aggregates that implement the GENI AM API**
  - ProtoGENI, PlanetLab, OpenFlow, InstaGENI, ExoGENI

```
$ omni.py createsliver aliceslice myRSpec.xml
INFO:omni:Loading config file omni_config
INFO:omni:Using control framework pgeni
INFO:omni:Slice urn:publicid:IDN+pgeni.gpolab.
        expires within 1 day on 2011-07-07
INFO:omni:Creating sliver(s) from rspec file
INFO:omni:Writing result of createsliver for
INFO:omni:Writing to 'aliceslice-manifest-rspe
INFO:omni: -----
INFO:omni: Completed createsliver:

Options as run:
                aggregate: https://www.emulab.
                framework: pgeni
                native: True

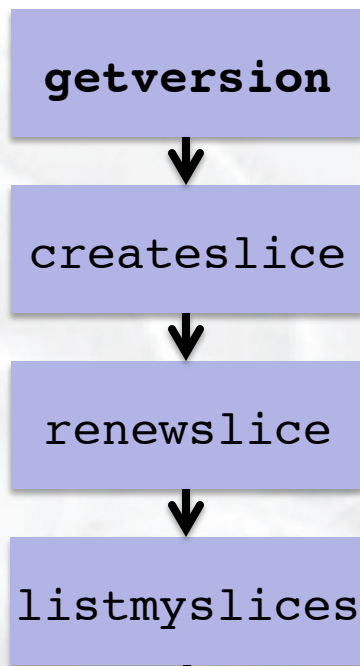
Args: createsliver aliceslice myRSpec.xml

Result Summary: Slice urn:publicid:IDN+pgeni
Reserved resources on https://www.emulab.net/p
Saved createsliver results to aliceslice-man
INFO:omni: =====
```

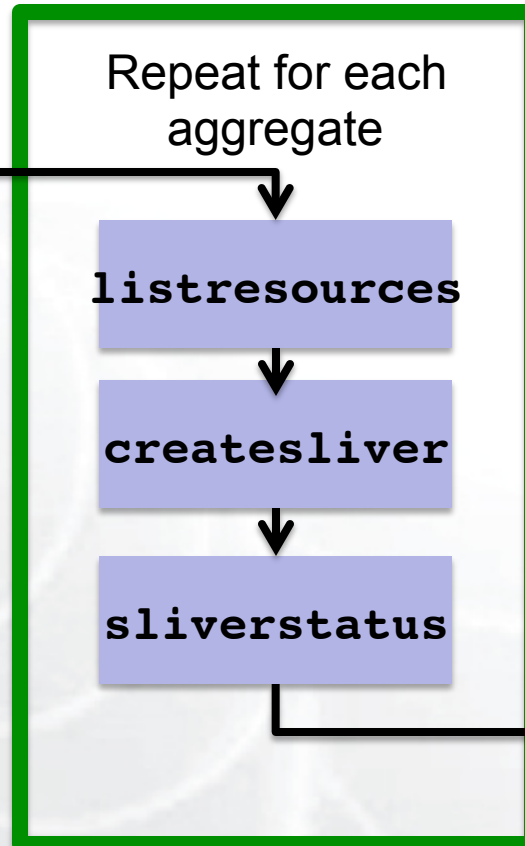
<http://trac.gpolab.bbn.com/gcf/wiki/Omni>

# Omni Command Workflow

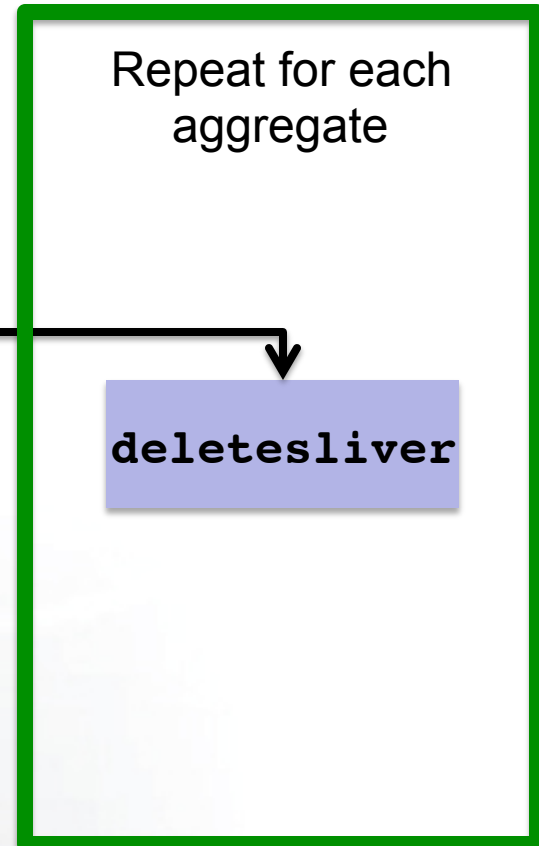
## Create Slice



## Create Sliver



## Cleanup



Legend: **AM API command**

Run this script for standard configuration:

1. Download GENI certificate
2. Run the script

```
[omni]
default_cf = pgeni
users = gpousr21
# ----- Users -----
[gpousr21]
urn = urn:publicid:IDN
+pgeni.gpolab.bbn.com+user
+gpousr21
keys = ~/.ssh/geni_key
...
```

- ✓ You only need your GENI certificate passphrase
- ✓ It will create the ssh key for uploading to compute resources
- ✓ Standard AM nicknames

# clear-passphrases.py

Use this script with **caution**.  
Don't use it on public computers.

Removes the passphrase from

- GENI certificate
- GENI ssh key

```
[omni]
default_cf = pgeni      # Tutorial accounts are on GPO's PG
users = gpousr21       # gpousr21's keys loaded on the VM to
allow login

# ----- Users -----
[gpousr21]
urn = urn:publicid:IDN+pgeni.gpolab.bbn.com+user+gpousr21
# Really important to get the keys correct!!!
#key to load on VM
keys = ~/Tutorials/Omni/gpousr20/ssh/gpousr20_key.pub

# default aggregates to run omni commands on
aggregates = http://emulab.net/protogeni/xmlrpc/am,
https://pgeni.gpolab.bbn.com/protogeni/xmlrpc/am, ...
```

```
# ----- Frameworks -----  
[pgeni]  
type = pg  
ch = https://www.emulab.net:443/protogeni/xmlrpc/ch  
sa = https://www.pgeni.gpolab.bbn.com:443/protogeni/xmlrpc/  
sa  
  
# Tutorial certificate and key  
cert = ~/omni_tutorial/ssh/gpousr21_cert_ct.pem  
key = ~/omni_tutorial/ssh/gpousr21_cert_ct.pem  
  
#----- AM nicknames -----  
[aggregate_nicknames]  
pg-gpo=urn:publicid:IDN+pgeni.gpolab.bbn.com+authority  
+am,https://pgeni.gpolab.bbn.com/protogeni/xmlrpc/am  
plc=,https://www.planet-lab.org:12346
```

```
geni@geni-vm:~/omni_tutorial$ omni.py getversion -a http://www.planet-lab.org:12346
INFO:omni:Loading config file omni_config
INFO:omni:Using control framework pgeni
INFO:omni:AM URN: unspecified_AM_URN (url: http://www.planet-lab.org:12346) has version:
INFO:omni:{  'ad_rspec_versions': [  {  'extensions': [  'http://www.protojeni.net/resource
                                                    'http://www.protojeni.net/resources/rspec/ex
                                                    'namespace': 'http://www.protojeni.net/resources/rspec/2',
                                                    'schema': 'http://www.protojeni.net/resources/rspec/2/ad.xsd',
                                                    'type': 'ProtoGENI',
                                                    'version': '2'},
          {  'extensions': [],
            'namespace': None,
            'schema': None,
            'type': 'SFA',
            'version': '1'}],
  'code_tag': '1.0-27',
  'code_url': 'git://git.onelab.eu/sfa.git@sfa-1.0-27',
  'default_ad_rspec': {  'extensions': [],
                        'namespace': None,
                        'schema': None,
                        'type': 'SFA',
                        'version': '1'},
```

```
INFO:omni: -----
INFO:omni: Completed getversion:

Options as run:
    aggregate: http://www.planet-lab.org:12346
    framework: pgeni
    native: True

Args: getversion

Result Summary:
Got version for 1 out of 1 aggregates

INFO:omni: =====
```

- `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>



- 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.

Prework

Configure Omni

Run your experiment

Cleanup resources

Wrap up

Prework

Configure Omni

Run your experiment

Cleanup resources

Wrap up

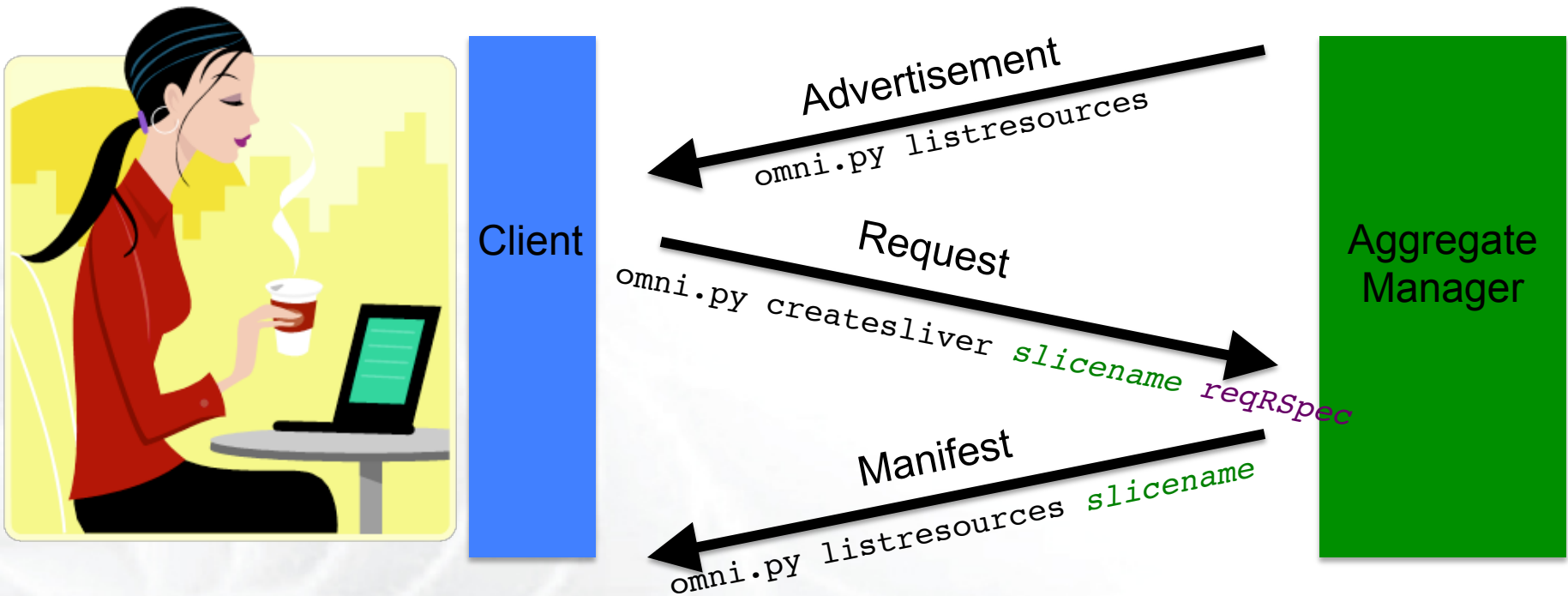
## Resource Specification Document (RSpec)

- 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>
```

```
<rspec type="request" xsi:schemaLocation="... " xmlns=""http://  
www.geni.net/resources/rspec/3"">  
  <node client_id="..." component_manager_id="urn:..."  
component_id="urn:..." component_name="..." exclusive="true">  
  <sliver_type name="raw-pc">  
    <disk_image name="urn:...">  
  </sliver_type>  
  <services>  
    <execute command="..." shell="..." />  
    <install install_path="..."  
      <url="..."  
      <file_type="..." />  
  </services>  
</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



Prework

Configure Omni

Run your experiment

Cleanup resources

Wrap up



- You can write custom Python scripts
  - Call existing Omni functions
  - Parse the Output
- Example: readyToLogin.py
  - Calls getversion, sliverstatus, listresources
  - Parses output
  - Determines ssh command to log into node
  - Print ssh configuration blocks to place in your ssh config file
- More examples distributed with Omni

- 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



Prework

Configure Omni

Run your experiment

Cleanup resources

Wrap up

Prework

Configure Omni

Run your experiment

Cleanup resources

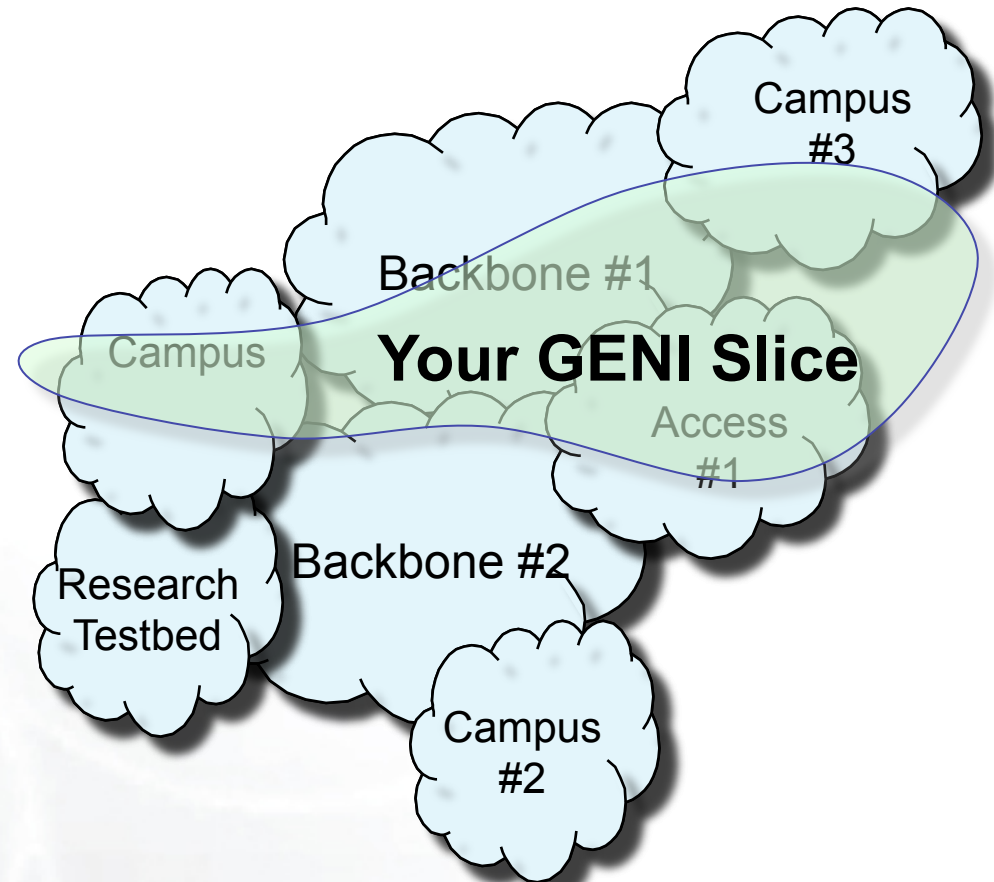
Wrap up

# Running Experiments on GENI

- Get an account to run experiments on GENI

- Contact us at [help@geni.net](mailto: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**

Prework

Configure Omni

Run your experiment

Cleanup resources

Wrap up



**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](mailto:gcf-dev@geni.net)
  - For general GENI help: [help@geni.net](mailto: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