GIMI/LabWiki Tutorial

Mike Zink, Divyashri Bhat,
Cong Wang, Thierry Rakotoarivelo

GEC20

March 22nd 2014, UC Davis, CA

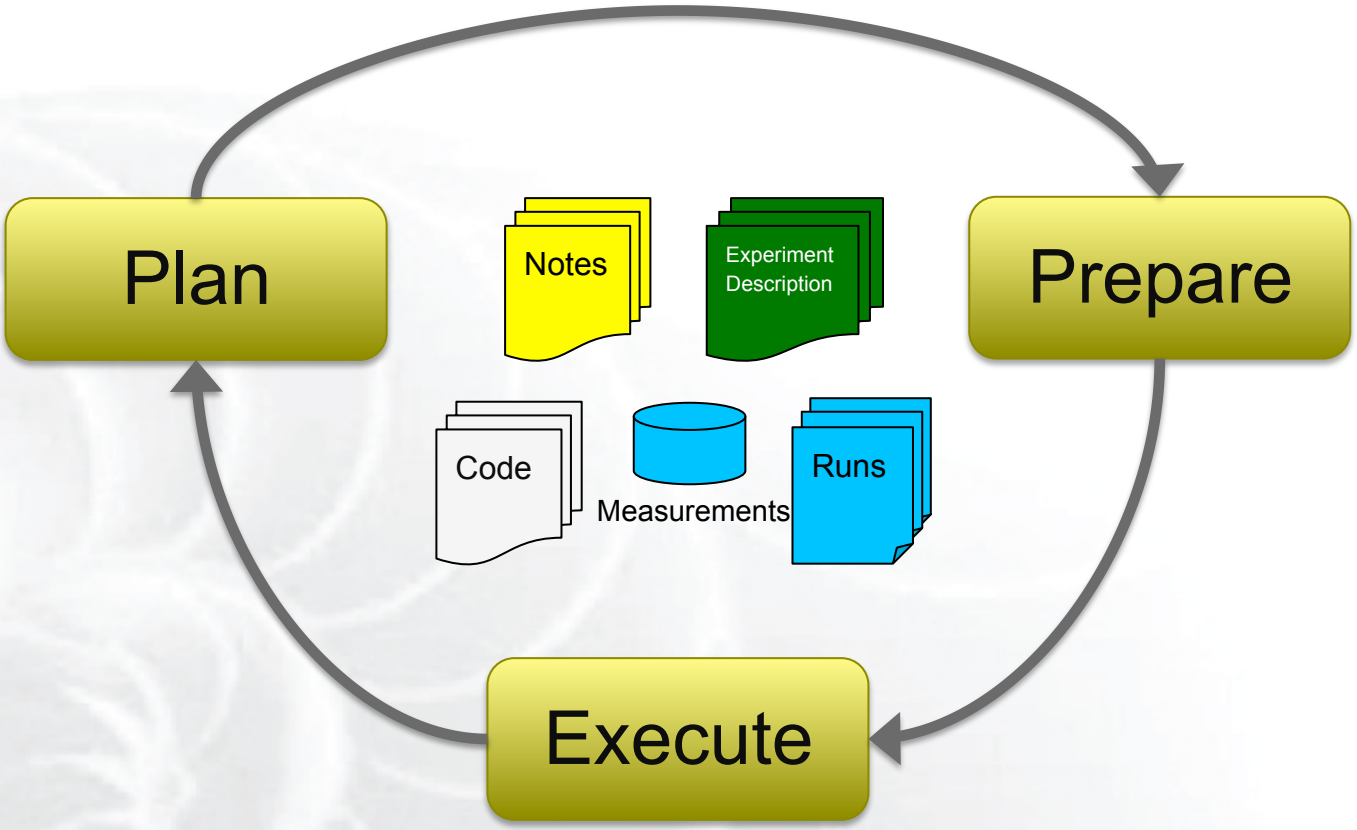
Sponsored by the National Science Foundation

Experiment Design

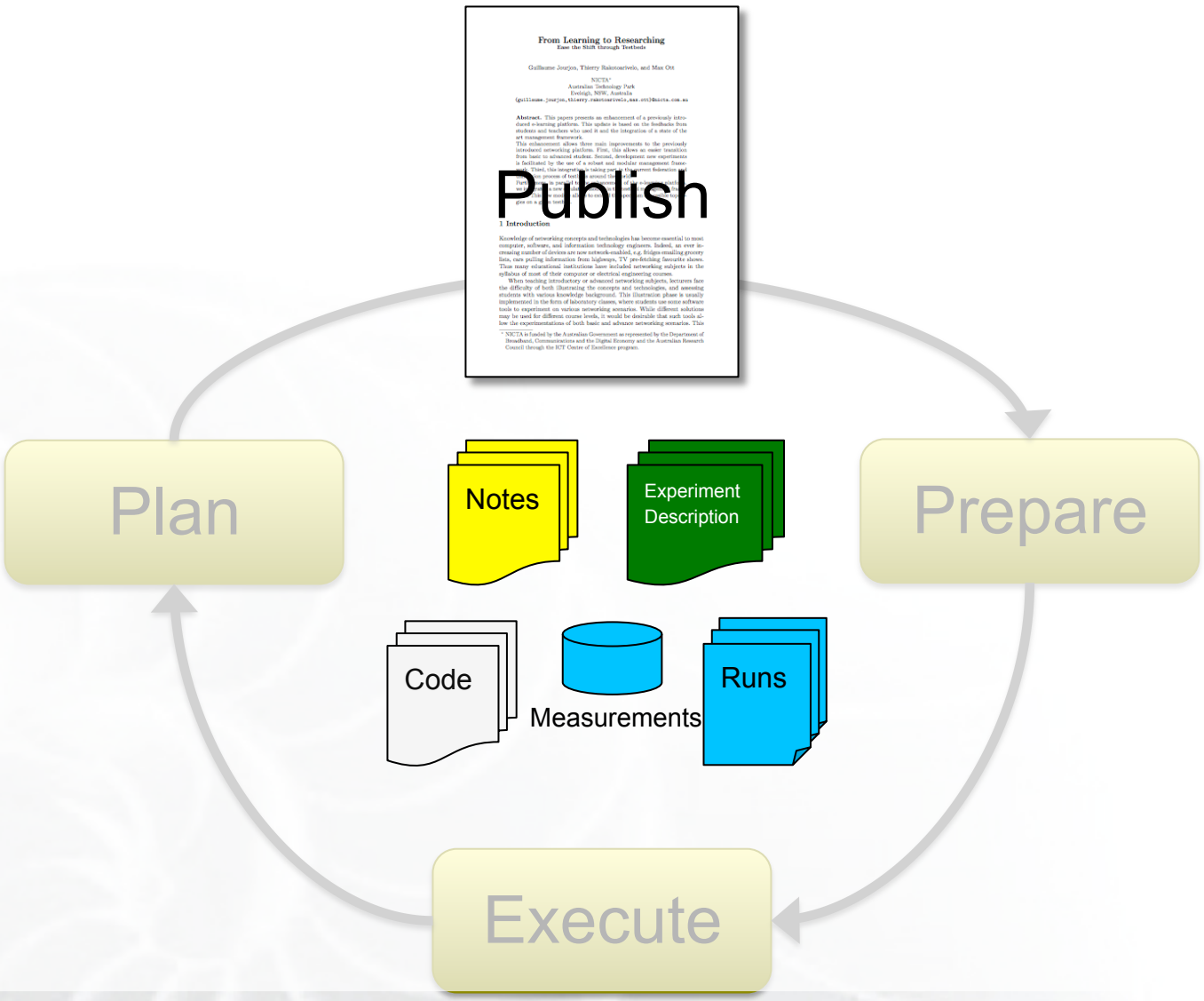
"Perform basic measurements on GENI testbeds"

- ✓ GENI slice (network & compute resources)
- ✓ Repeatable experiment
- ✓ Ability to monitor and document experiment

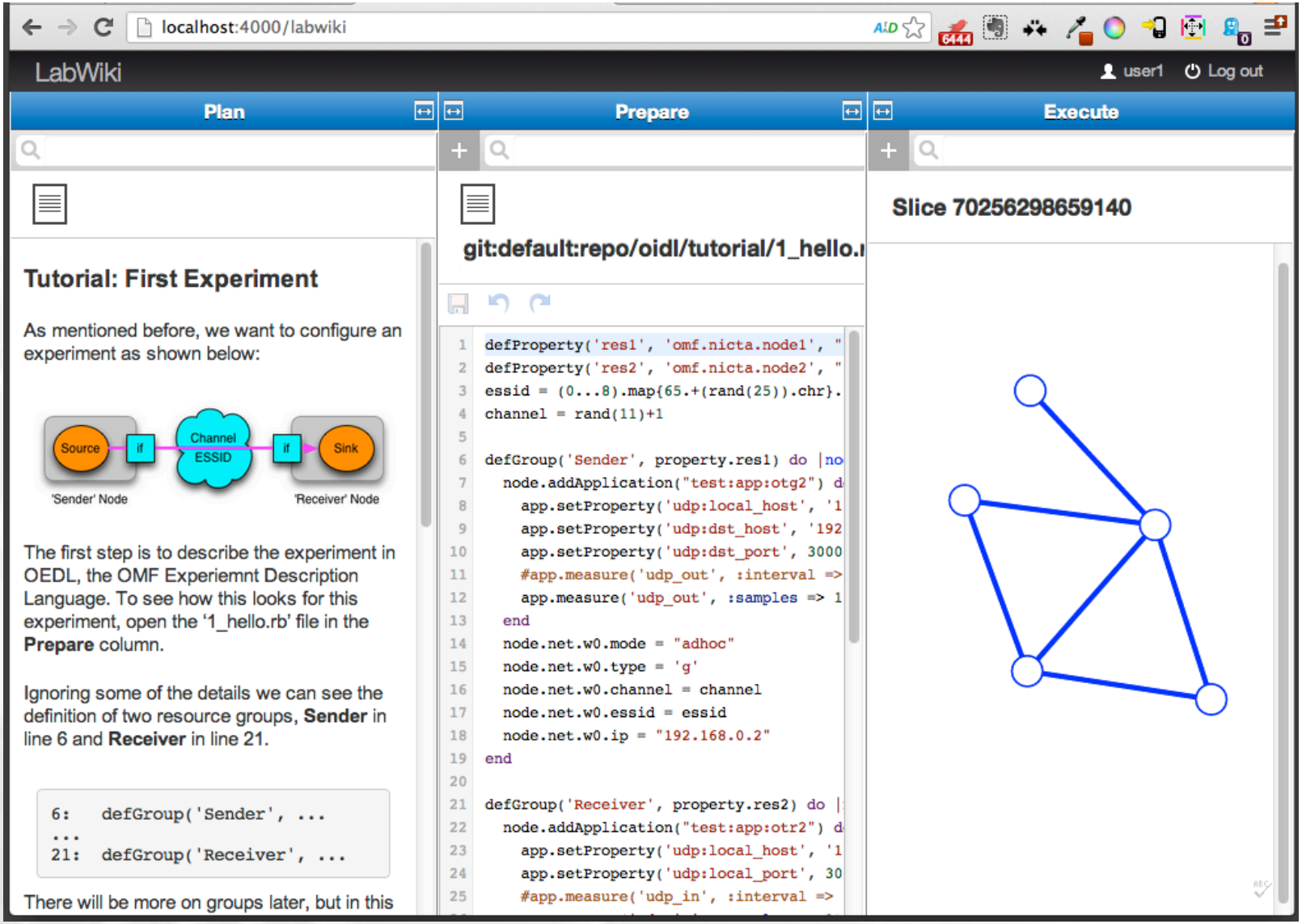
The “Experiment Cycle”

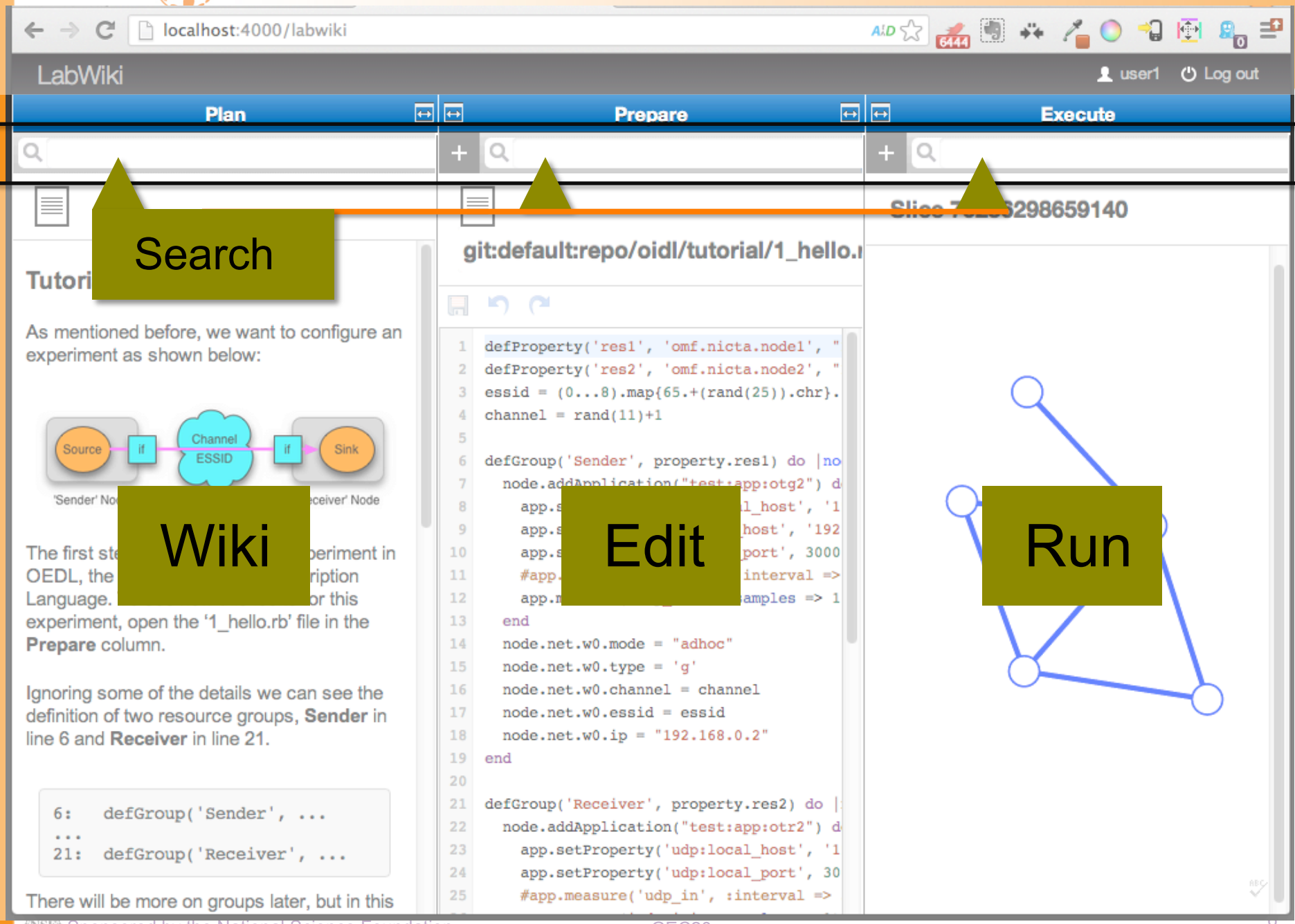


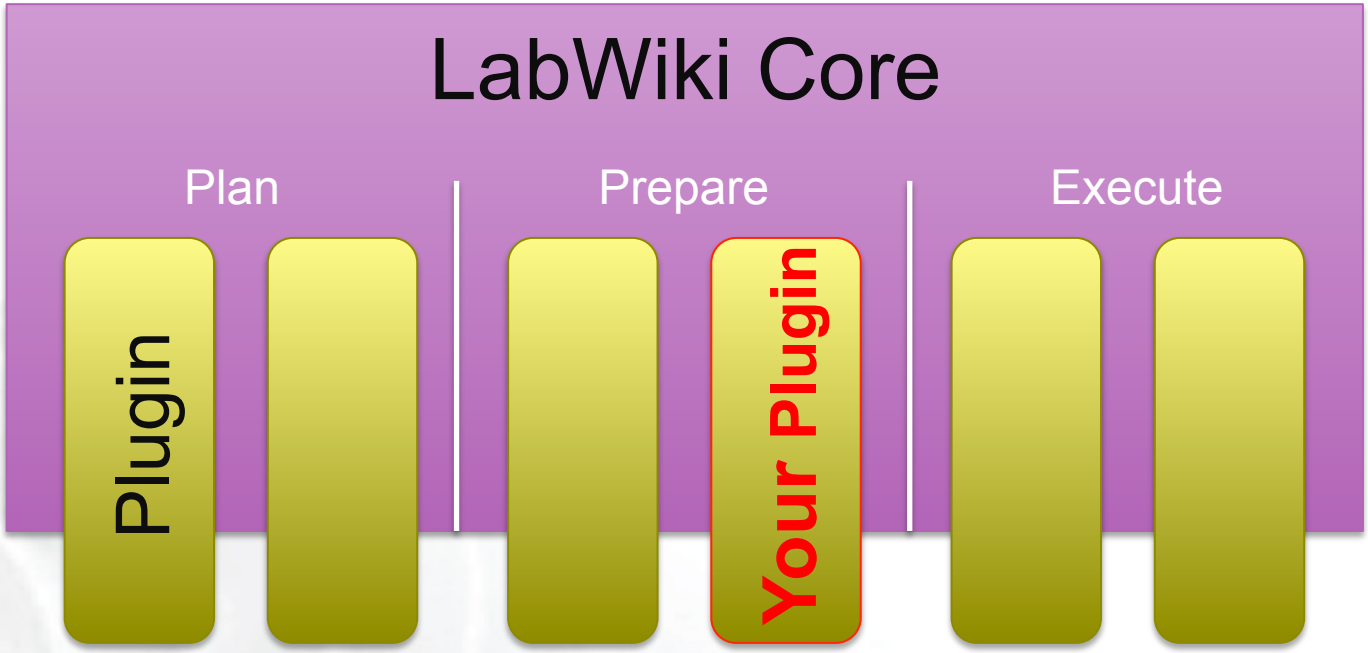
The "Successful Experiment Cycle"



The "Experiment Cycle" in a Tool: LabWiki



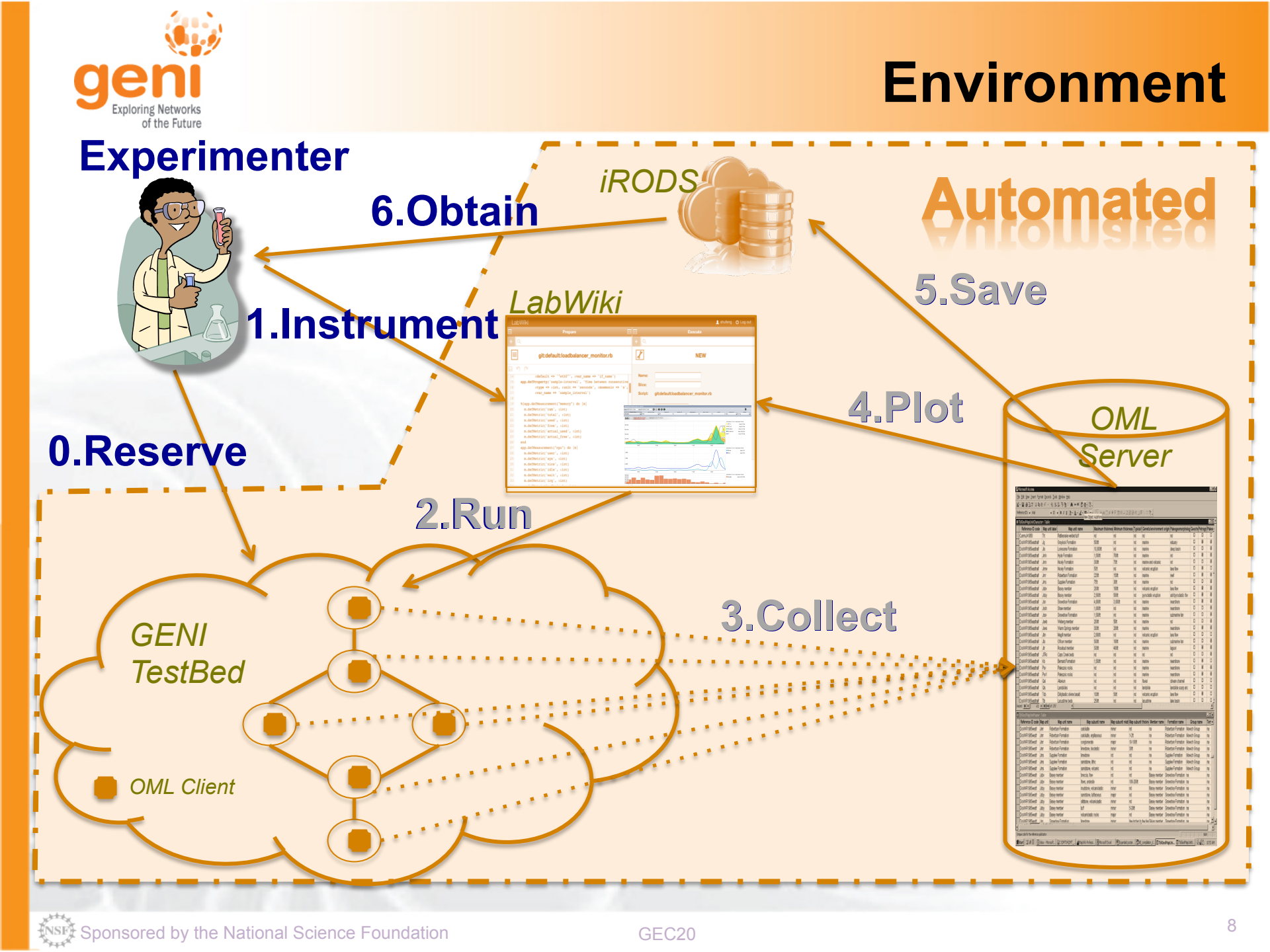

localhost:4000/labwiki
LabWiki
user1
Log out
Plan
Prepare
Execute
Search
Wiki
Edit
Run
Tutori
As mentioned before, we want to configure an experiment as shown below:
Source
if
Channel
ESSID
if
Sink
The first ste
periment in
OEDL, the
ription
Language.
or this
experiment, open the '1_hello.rb' file in the
Prepare column.
Ignoring some of the details we can see the definition of two resource groups, Sender in line 6 and Receiver in line 21.
6: defGroup('Sender', ...
...
21: defGroup('Receiver', ...
There will be more on groups later, but in this
git:default:repo/oidl/tutorial/1_hello.r
1 defProperty('res1', 'omf.nicta.node1', "
2 defProperty('res2', 'omf.nicta.node2', "
3 essid = (0...8).map{65.+(rand(25)).chr}.
4 channel = rand(11)+1
6 defGroup('Sender', property.res1) do |no
7 node.addApplication("test:app:otg2") d
8 app.s l_host', '1
9 app.s host', '192
10 app.s port', 3000
11 #app. interval =>
12 app.m amples => 1
13 end
14 node.net.w0.mode = "adhoc"
15 node.net.w0.type = 'g'
16 node.net.w0.channel = channel
17 node.net.w0.essid = essid
18 node.net.w0.ip = "192.168.0.2"
19 end
21 defGroup('Receiver', property.res2) do |
22 node.addApplication("test:app:otr2") d
23 app.setProperty('udp:local_host', '1
24 app.setProperty('udp:local_port', 30
25 #app.measure('udp_in', :interval =>
Slice 3298659140



GENI CH/AM | OMF | GIMI Services | iRODS | **Your** Service

https://github.com/mytestbed/labwiki

Environment



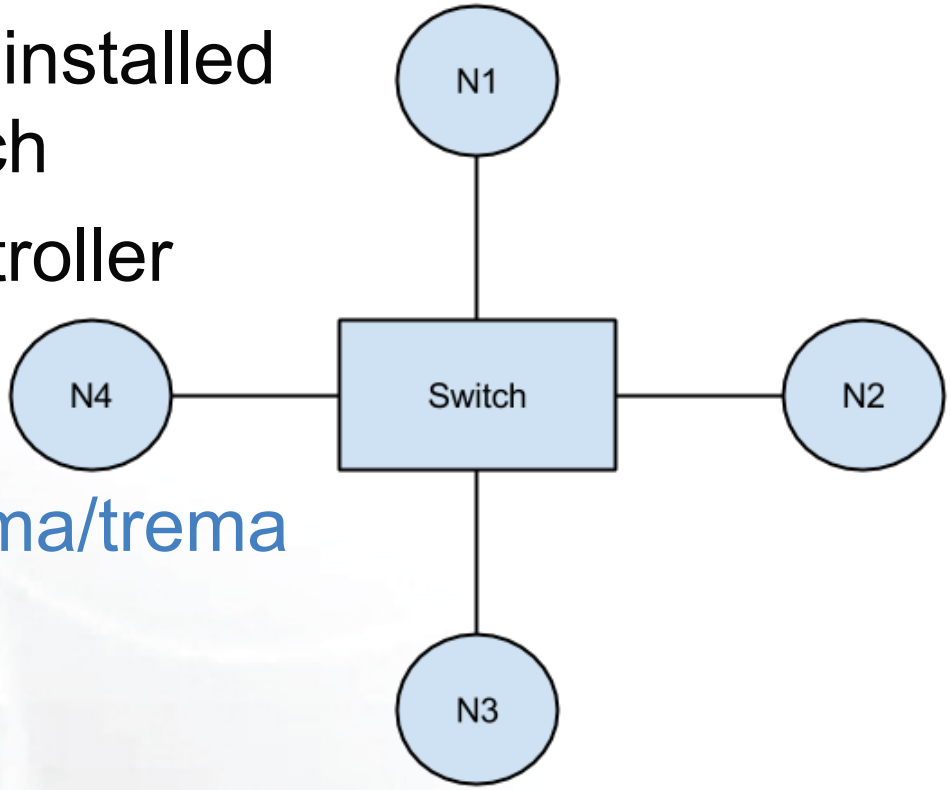
Preparation

- In earlier tutorials:
  - GENI account and credentials
  - iRODS account (set up via portal)
  - Know what resources are available
  - How to create a slice request (Rspec)
  - How to reserve a slice
  - How to log into VMs of your slice
- Familiar with GENI Portal!

Example Experiment Topology

- N1 – N4: regular end systems
- Switch: VM with OVS installed
  ➔ sw-based OF switch
- Trema-based OF controller running on Switch

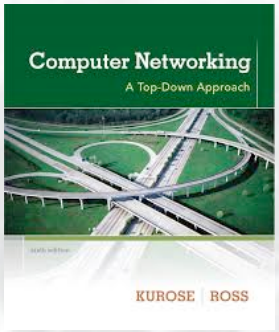
https://github.com/trema/trema

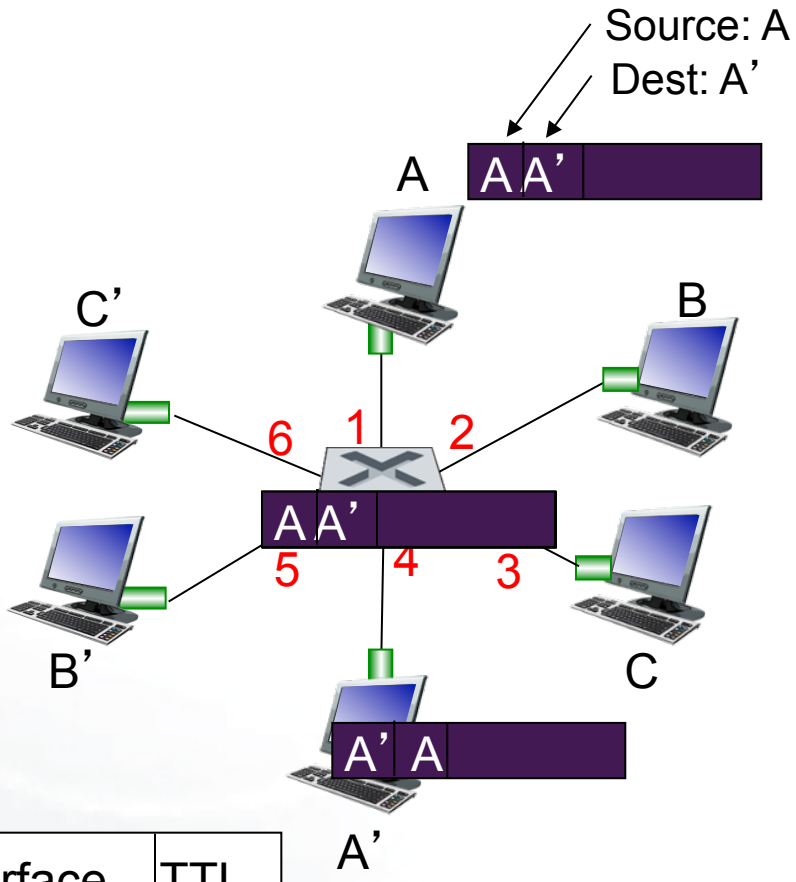


Goal: Implement learning switch and verify its correct behavior

Learning Switch

- frame destination unknown: *flood*
- destination A location known: selective send
- More info in chapter 6 of "Computer Networks", Kurose & Ross





MAC addr	interface	TTL
A	*1*	*60*
A'	*4*	*60*

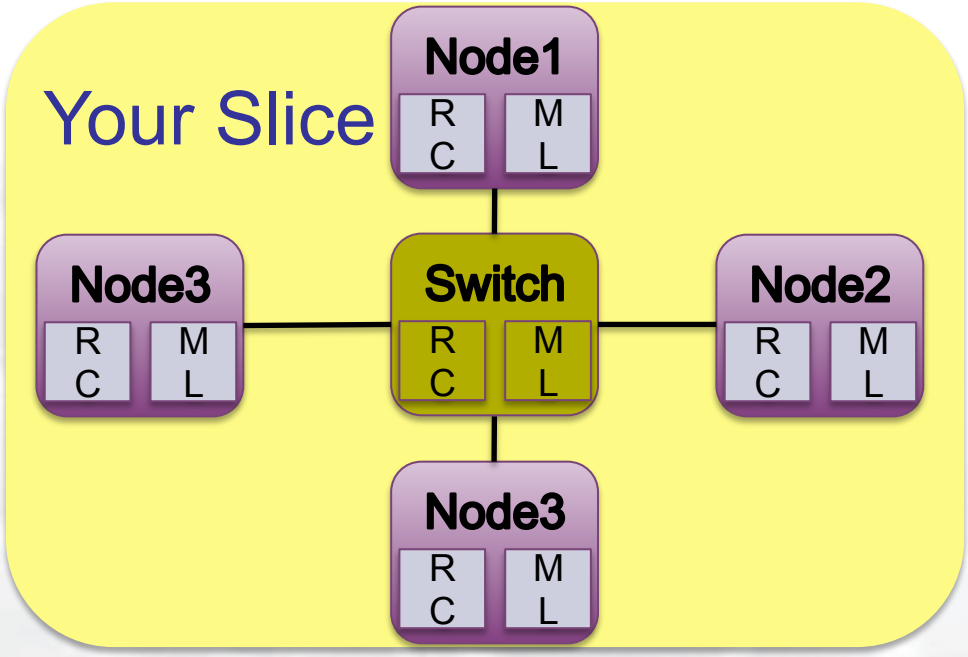
Switch table (initially empty)

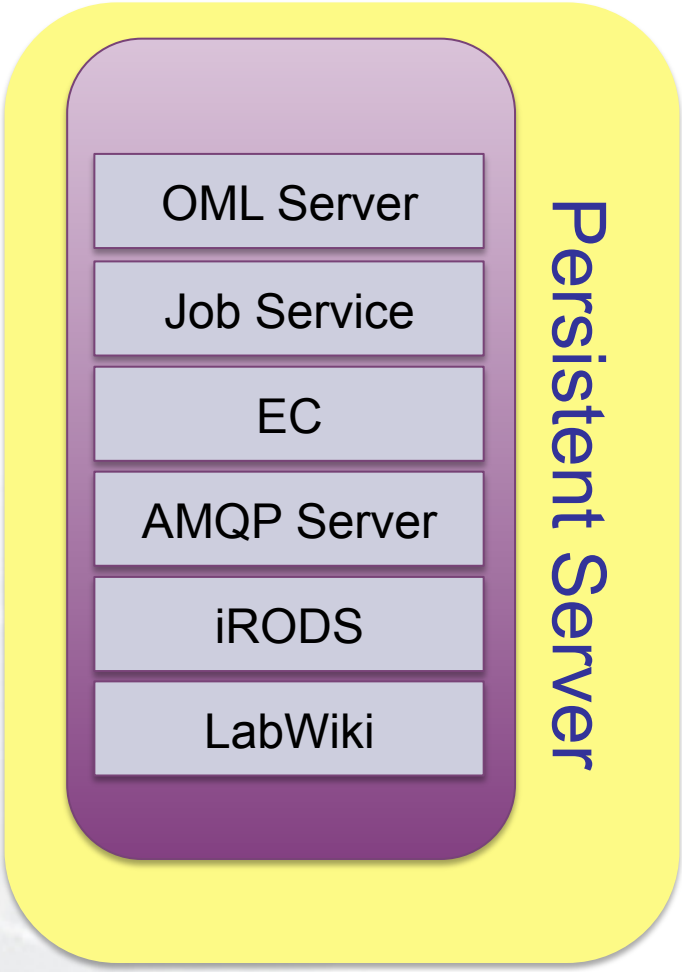
Module A

http://groups.geni.net/geni/wiki/GEC20Agenda/LabWiki/ModuleA

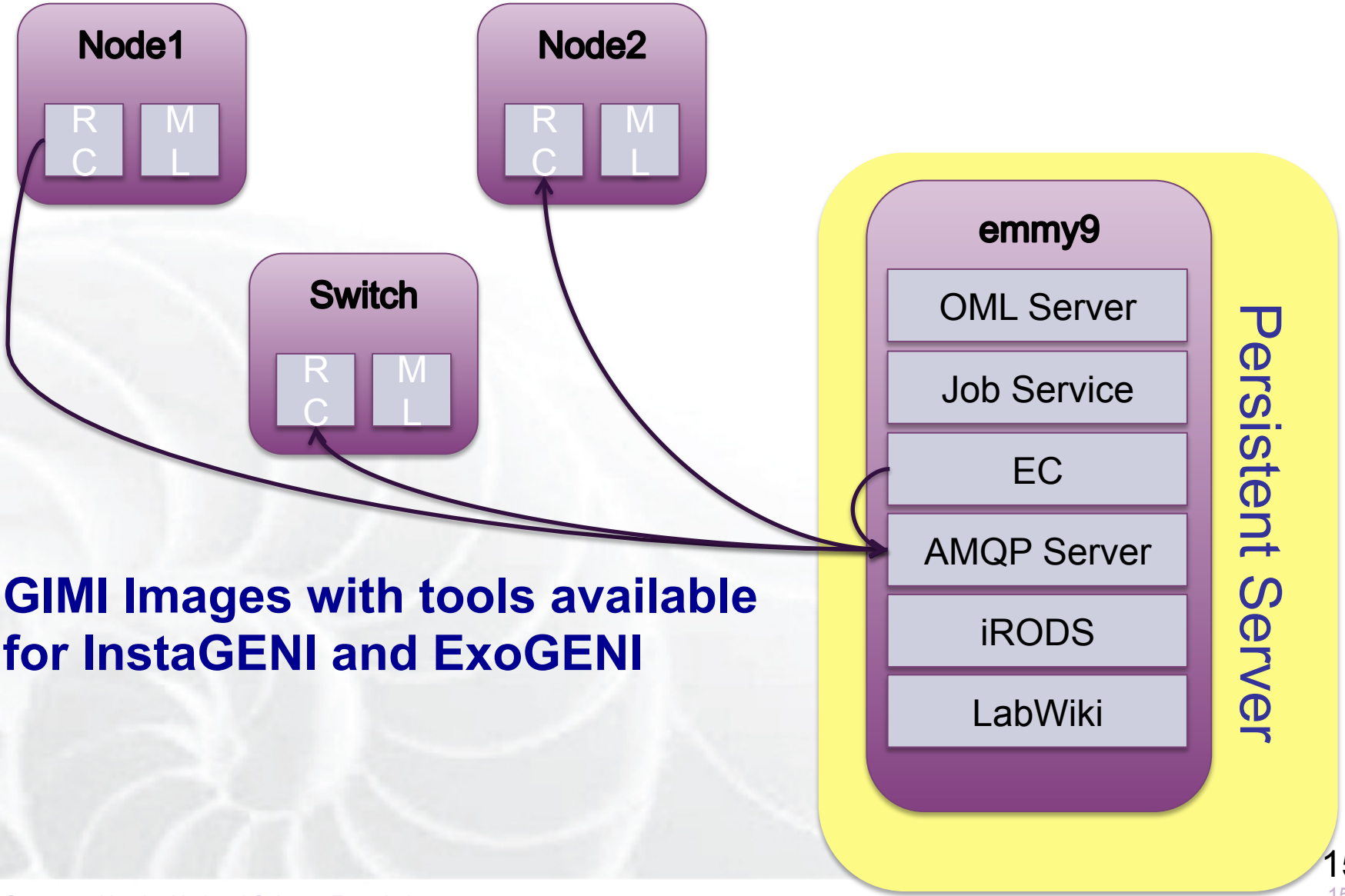
Module B

Architecture



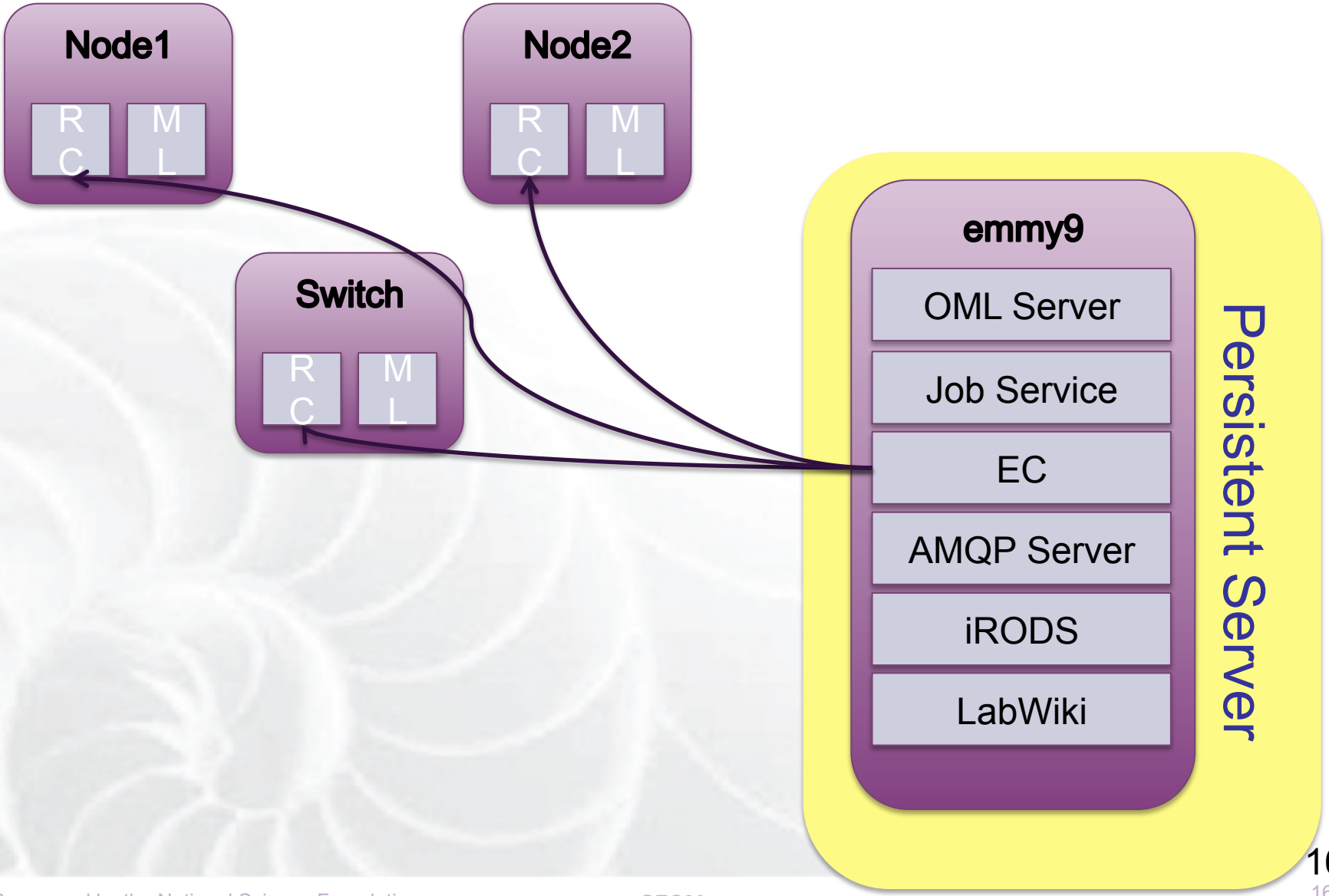


AMQP Messaging

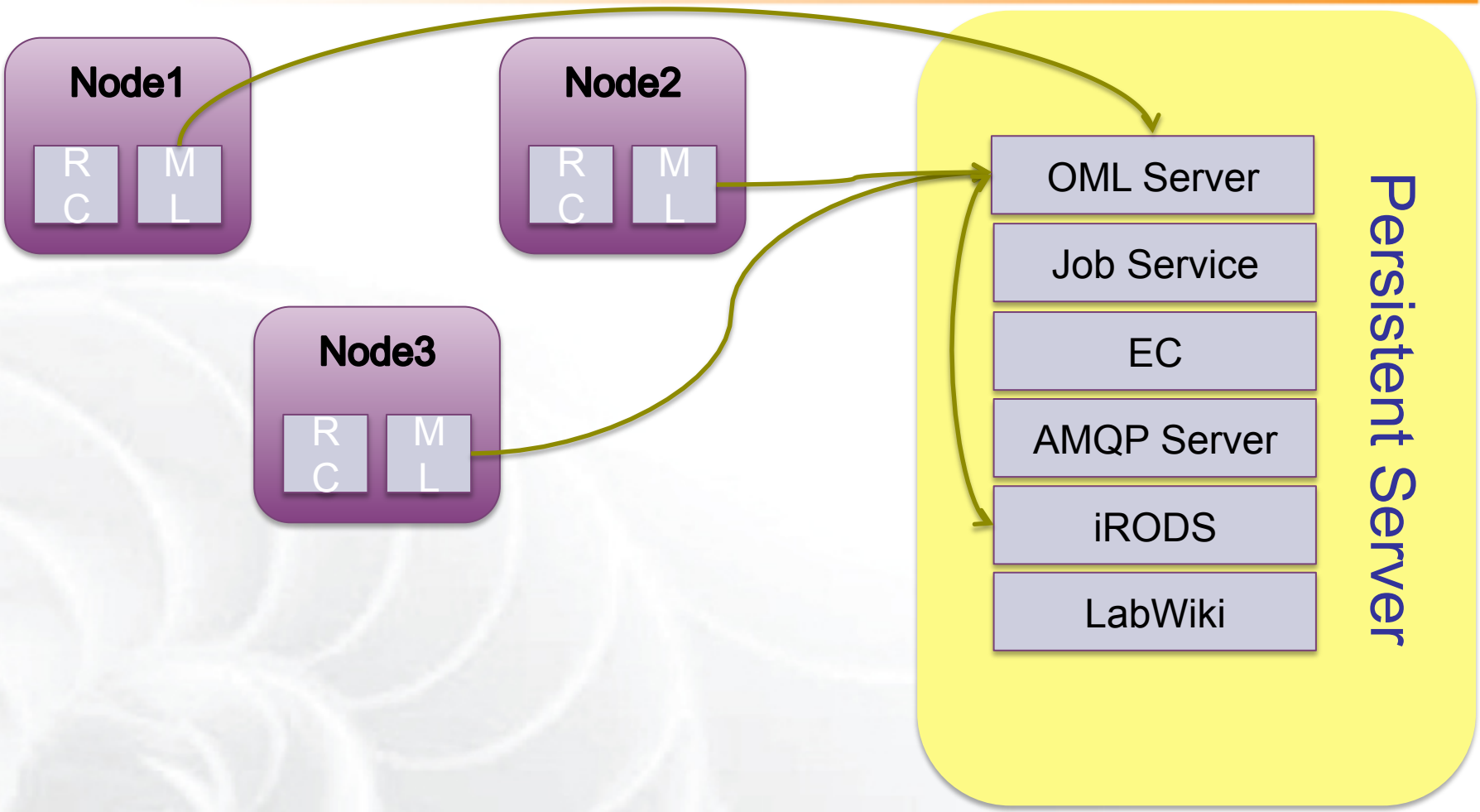


GIMI Images with tools available for InstaGENI and ExoGENI

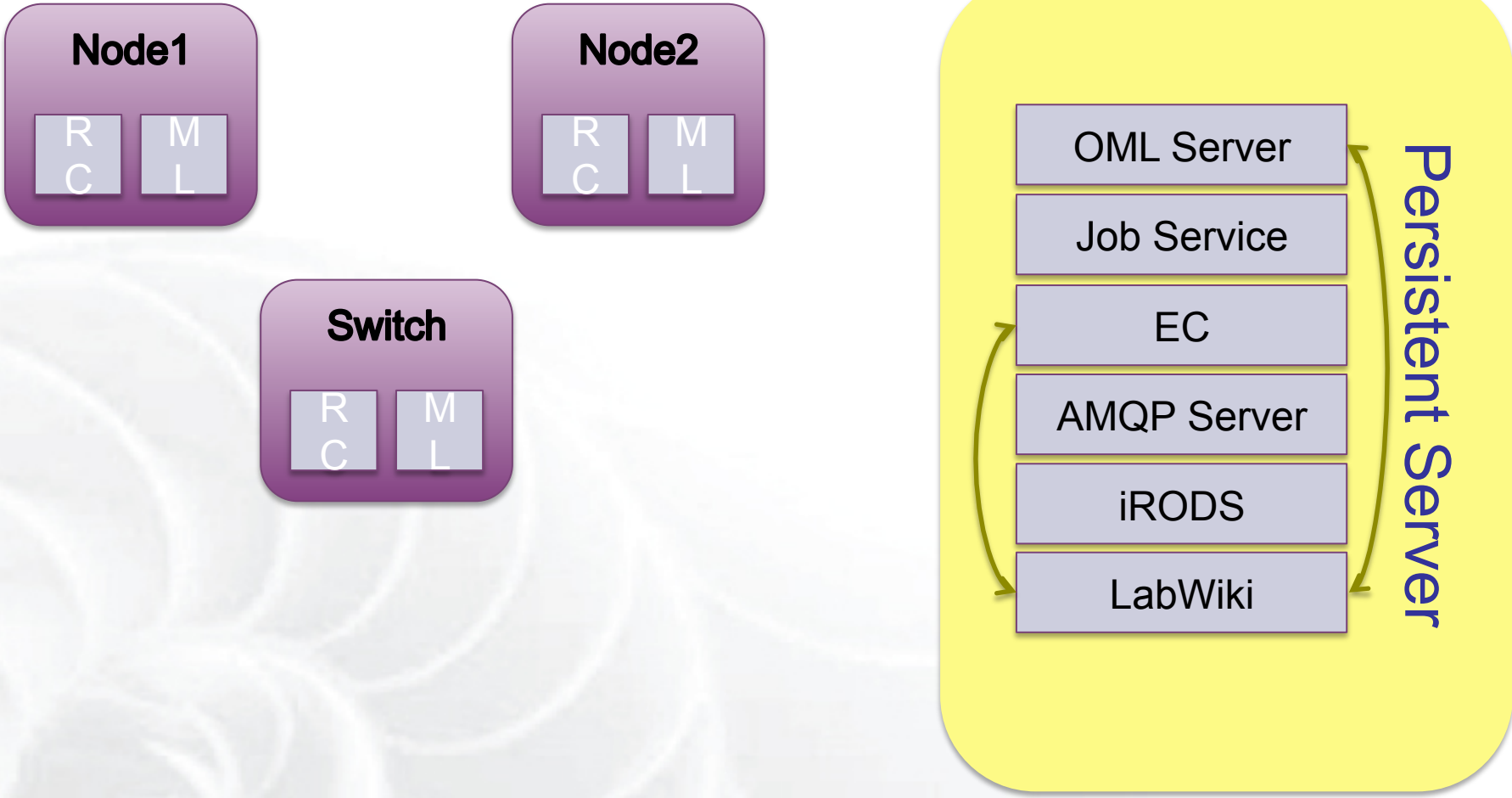
EC/RC Messaging



Data Collection



LabWiki


Node1
R C
M L
Node2
R C
M L
Switch
R C
M L
OML Server
Job Service
EC
AMQP Server
iRODS
LabWiki
Persistent Server

Documentation

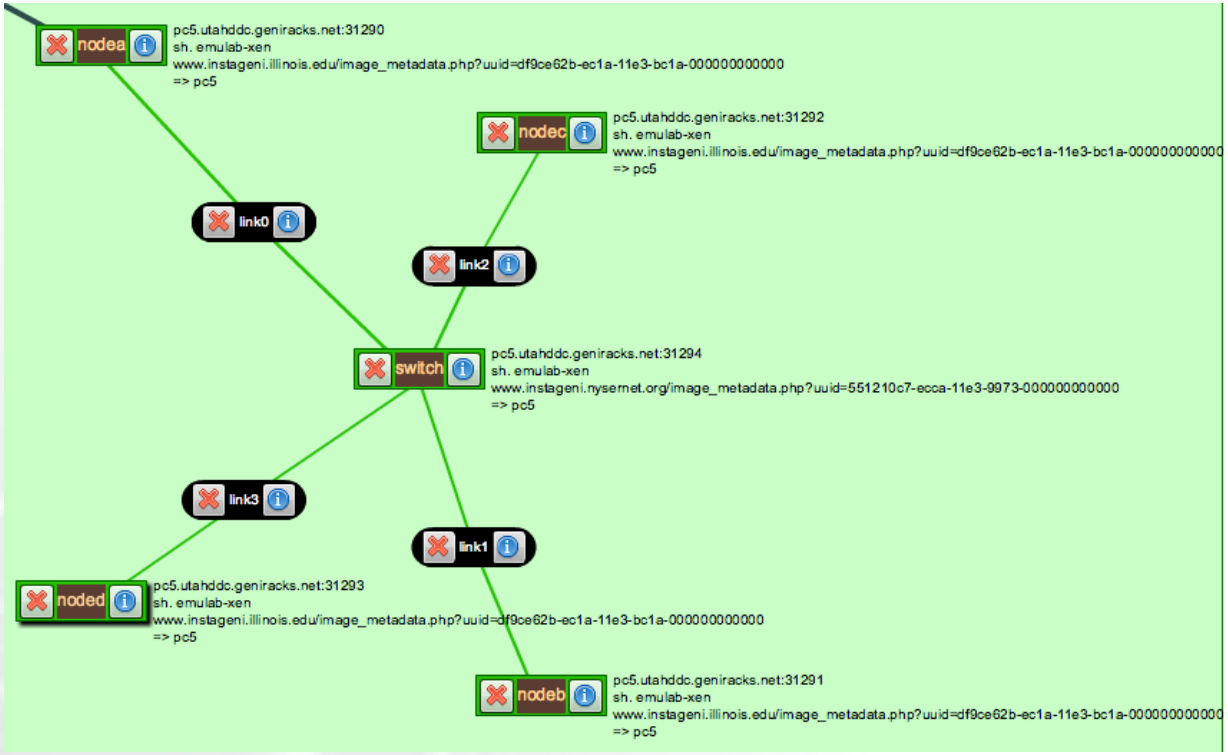
- OML: http://oml.mytestbed.net/projects/oml
- Job Service: https://github.com/mytestbed/omf_job_service
- OMF: http://oml.mytestbed.net/projects/omf
- AMQP: http://www.rabbitmq.com/
- iRODS: http://irods.org/
- LabWiki: https://github.com/mytestbed/labwiki

Steps

1. Reserve slice with VMs that boot GIMI image
2. Verify that slice is up
3. Define experiment to be executed in LabWiki
4. Run and observe experiment
5. Archive results
6. Document results
7. Delete slice (be a good GENI citizen by sharing!!)

Nodes 1-4

- Image
- Postboot script
- ssh into node



Nodes 1-4: Image

- ExoGENI
  - Debian 6
  - OMF
  - OML client & apps
  - URL to image: http://emmy9.casa.umass.edu/Disk_Images/ExoGENI/deb6ovsgimiv4.xml
- InstaGENI
  - Ubuntu 12.04
  - OMF
  - OML client & apps
  - URL to image: https://www.instageni.illinois.edu/image_metadata.php?uuid=df9ce62b-ec1a-11e3-bc1a-000000000000

Nodes 1-4: Postboot Script IG

- Postboot script (http://emmy9.casa.umass.edu/GEC-20/gimidev.tar.gz)

```
#!/bin/bash
read -r slice1 </var/emulab/boot/nickname
slicename=$(echo $slice1 | cut -f2 -d.)
host1=$(hostname)
host=$(echo $host1 | cut -f1 -d.)
slice=`ruby -e "print '$slicename'[/[^+]*$/]"`
echo $host > /etc/hostname
/bin/hostname -F /etc/hostname
echo "---
:uid: $host-$slice
:uri: amqp://gimi3.casa.umass.edu
:environment: production
:debug: false" > /etc/omf_rc/config.yml
restart omf_rc
wget http://emmy9.casa.umass.edu/GEC-20/fdb.rb -P /root/
wget http://emmy9.casa.umass.edu/GEC-20/Insta_post/learning-switch_copy.rb -P /root
chmod +x /root/learning-switch_copy.rb
wget http://emmy9.casa.umass.edu/GEC-20/Insta_post/learn_ofcollect.rb -P /usr/local/bin
chmod +x /usr/local/bin/learn_ofcollect.rb
wget http://emmy9.casa.umass.edu/GEC-20/Insta_post/ovs-setup2.sh -P /root
chmod +x /root/ovs-setup2.sh
```

Initiates connect to AMQP server

Identical for both PB scripts

Nodes 1-4: Postboot Script EG

- Postboot script (http://emmy9.casa.umass.edu/GEC-20/gimidev.tar.gz)

```
#!/bin/bash
export PATH=/usr/local/bin/rubydir/bin:$PATH
host=$1
slice=`ruby -e "print '$2'[/[^+]*$/]"`
echo $host > /etc/hostname
/bin/hostname -F /etc/hostname
apt-get install psmisc
echo "---
:uid: $host-$slice
:uri: amqp://gimi3.casa.umass.edu
:environment: production
:debug: false" > /etc/omf_rc/config.yml
echo $PATH > /tmp/curpath.out
if grep 'Ubuntu' /etc/issue; then
restart omf_rc
else
/etc/init.d/omf_rc restart
fi
```

Initiates connect to AMQP server

Identical for both PB scripts

Switch

- Image
- Postboot script
- ssh into node

Switch: Image

- ExoGENI
  - Debian 6
  - OMF
  - OML
  - OVS
  - URL to image: http://emmy9.casa.umass.edu/Disk_Images/ExoGENI/deb6ovsgimiv4.xml
- InstaGENI
  - Ubuntu 12.04
  - OMF
  - OML
  - OVS
  - URL to image: https://www.instageni.nysernet.org/image_metadata.php?uuid=551210c7-ecca-11e3-9973-000000000000

Switch: Postboot Script

- Postboot script:
- Rspec:

```
<execute command="wget http://emmy9.casa.umass.edu/GEC-20/fdb.rb -P /
root/; wget http://emmy9.casa.umass.edu/GEC-20/learning-switch.rb -P /
root/; sh /gimidev/gimibot.sh $self.Name() $sliceName; wget http://
emmy9.casa.umass.edu/GEC-20/ovs-setup.sh -P /root/; sh /root/ovs-setup.sh"
shell="sh"/>
```

- ovs-setup.sh:

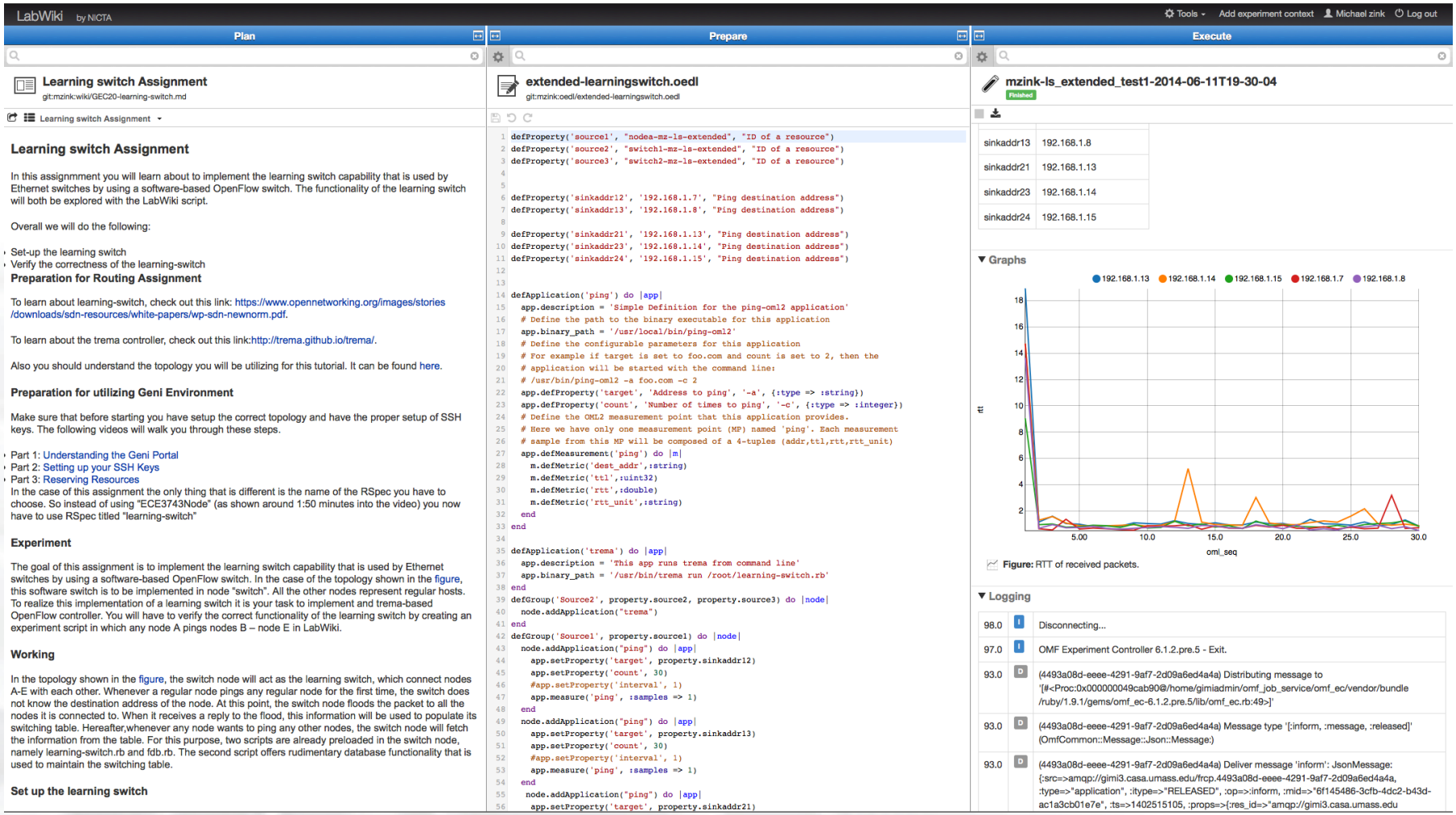
```
#!/bin/bash
ethnum=$(ifconfig | grep 'Link encap:' | awk '{print $1}')
while [[ "$ethnum" != *eth1* || "$ethnum" != *eth2* || "$ethnum" != *eth3* || "$ethnum" != *eth4*]]; do
        /usr/local/bin/neuca-netconf
        sleep 10
        ethnum=$(ifconfig | grep 'Link encap:' | awk '{print $1}')
done
ovs-vsctl add-br test
ovs-vsctl add-port test eth1
ovs-vsctl add-port test eth2
ovs-vsctl add-port test eth3
ovs-vsctl add-port test eth4
#ovs-vsctl set-controller test tcp:127.0.0.1:6653 ptcp:6633:127.0.0.1
ovs-vsctl set-controller test tcp:127.0.0.1:6653
```

Debugging the RC

- ssh into node
  `ssh –i ~/.ssh/`*`yourkey management_ip_of_node`*
- On the node:
  /var/log/omf_rc.log

```
[2014-06-20 17:48:03 +0000] INFO  OmfRc::Runner: Starting OMF Resource Controller version
'6.1.2.pre.5'
[2014-06-20 17:48:04 +0000] INFO  OmfRc::Runner: Connected using
{:proto=>:amqp, :user=>"guest", :domain=>"127.0.0.1"}
[2014-06-20 17:49:05 +0000] WARN  OmfCommon::Comm::AMQP::Communicator: Lost connectivity.
Trying to reconnect...
[2014-06-20 17:49:05 +0000] INFO  OmfRc::Runner: Starting OMF Resource Controller version
'6.1.2.pre.5'
[2014-06-20 17:49:05 +0000] INFO  OmfRc::Runner: Connected using
{:proto=>:amqp, :user=>"guest", :domain=>"127.0.0.1"}
[2014-06-20 17:56:06 +0000] INFO  OmfRc::ResourceProxy::AbstractResource: App Event from
'ping_cxt_0' (#0) - STARTED: 'env -i /usr/local/bin/ping-oml2 -a 192.168.1.7 -c 30 --oml-
config /tmp/854849b2-19a2-475d-9588-278271d894aa-1403286966.xml'
[2014-06-20 17:56:06 +0000] INFO  OmfRc::ResourceProxy::AbstractResource: App Event from
'ping_cxt_1' (#0) - STARTED: 'env -i /usr/local/bin/ping-oml2 -a 192.168.1.8 -c 30 --oml-
config /tmp/96c56027-f62e-41b1-873d-ba39fe429ab9-1403286966.xml'
[2014-06-20 17:56:06 +0000] INFO  OmfRc::ResourceProxy::AbstractResource: App Event from
'ping_cxt_2' (#0) - STARTED: 'env -i /usr/local/bin/ping-oml2 -a 192.168.1.9 -c 30 --oml-
config /tmp/53a86677-61d6-4b31-8f65-4e272d7b2343-1403286966.xml'
```

LabWiki Tour



OEDL Script

- We will look at this in LabWiki
- More info on OEDL scripts: http://omf.mytestbed.net/projects/omf6/wiki/OEDLOMF6

MD Script

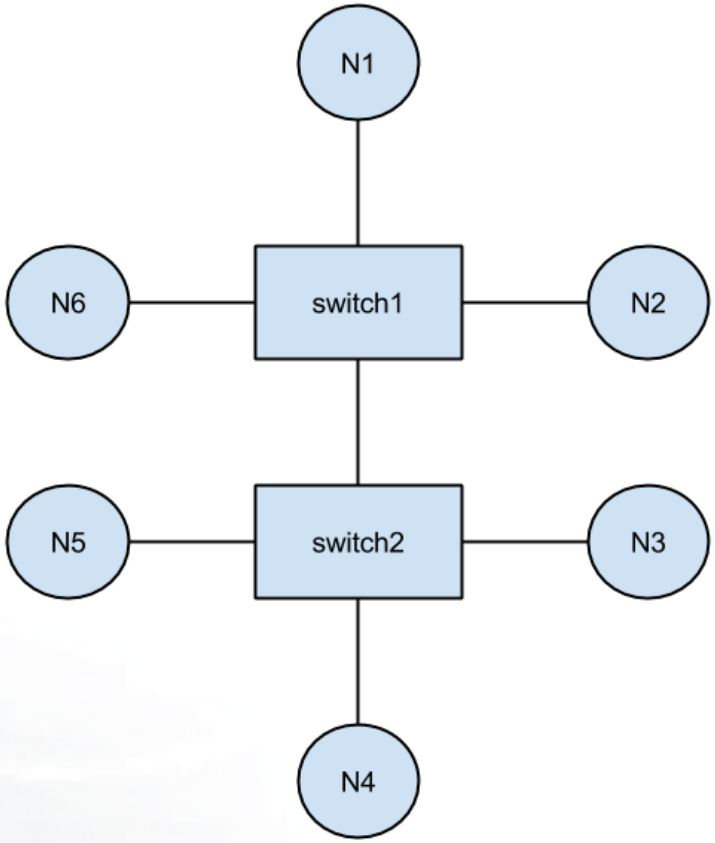
- We will look at this in LabWiki

Module C

http://groups.geni.net/geni/wiki/GEC20Agenda/LabWiki/ModuleC

Extended Experiment Topology

- N1 – N6: regular end systems
- Switch 1-2: VM with OVS installed ➔ sw-based OF switch
- Trema-based OF controller running on Switch



Goal: Implement learning switch and verify its correct behavior

Measure FlowStats

- Observe performance of flows
- Done via flow_stats in OF controller
- OML app to *capture* information and transmit to OMLserver
- Use LabWiki to observe FlowStats live

Want to Know More?

- Sample assignments: http://groups.geni.net/geni/wiki/GENIEducation/SampleAssignments
- Demo Night, immediately after this session