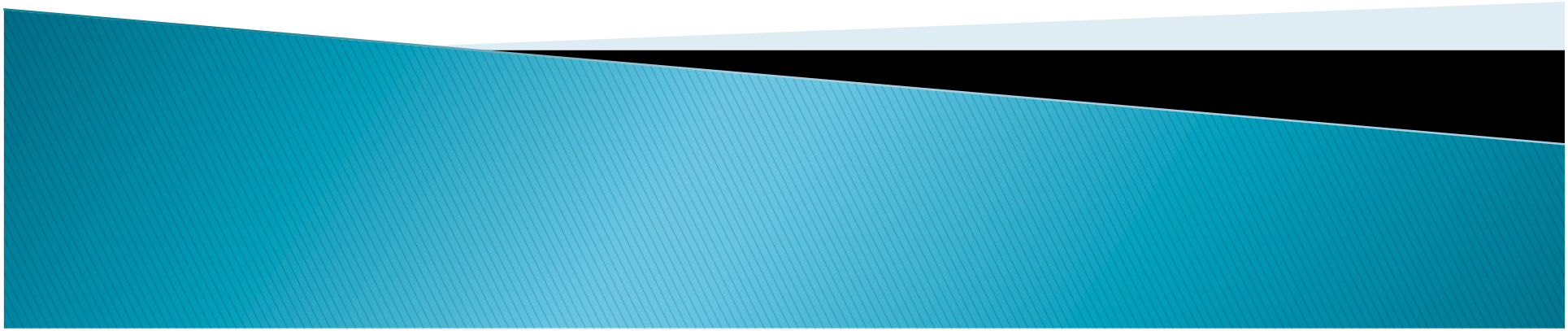
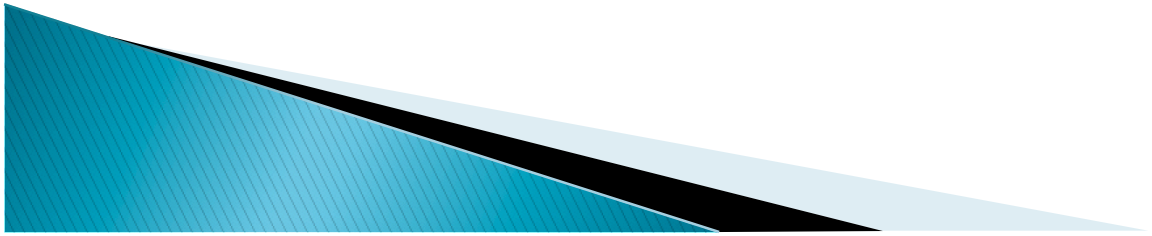
ExoGENI Tutorial

Ilia Baldine ibaldin@renci.org



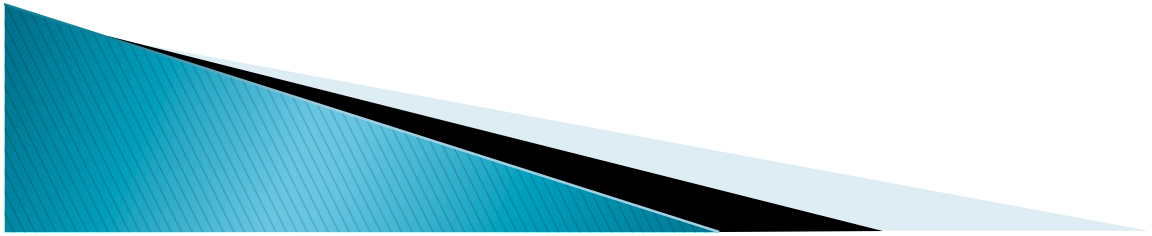
Tutorial sections

- Configure environment
- ExoGENI Ecosystem
- Flukes Overview
- Creating slices with Flukes
- ORCA and VM images
- Tutorial page:
  - http://groups.geni.net/geni/wiki/GEC15Agenda/ExoGENITutorial
  - Please open in your browser
  - Please open the presentation attached to the page



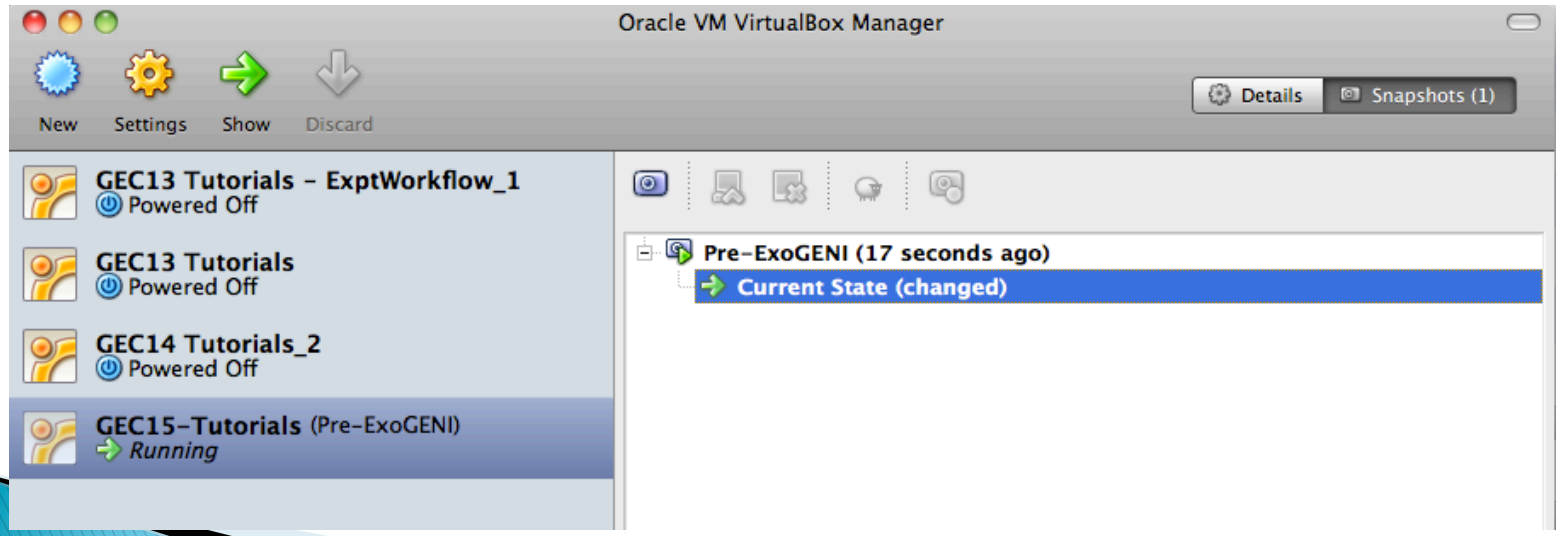
Goals of the tutorial

- Gain basic understanding of ExoGENI
- Gain basic understanding of the tools
- Learn how to build a variety of slices
  - Intra-rack
  - Inter-rack
  - Linked to NLR Mesoscale OpenFlow deployment
- Caveat: this is a complex distributed system and there are no guarantees. Things can and do fail. Be patient, ask for help.



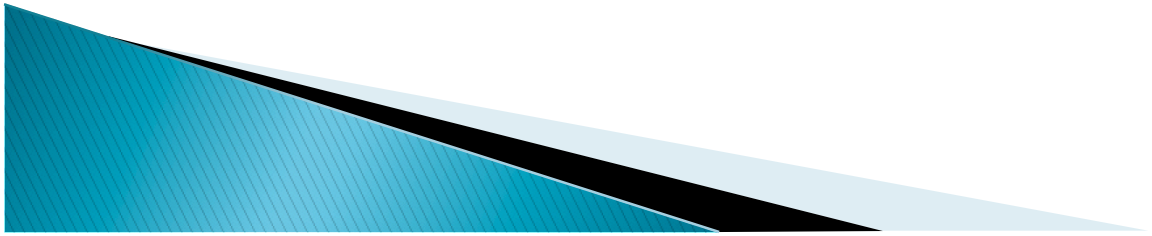
Before you begin

- If you are planning to attend the GIMI tutorial tomorrow and want to use the same VM
  - Please take a snapshot in VirtualBox before you fire off the VM
  - At GIMI tutorial start from the snapshot version



Configuring your Environment

- VM login password: gec14user
- Your user id is gimiXX where XX is 01-30
- All Flukes user properties are under $HOME/.flukes.properties – it is a text file
  - Edit $HOME/.flukes.properties
    - Open in an editor
    - Follow the wiki page to make the modifications
    - http://groups.geni.net/geni/wiki/GEC15Agenda/ExoGENITutorial
- Double-click on flukes.jnlp on your desktop to launch Flukes



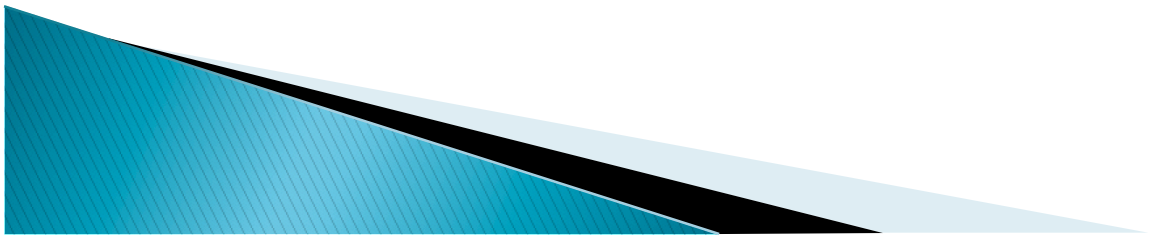
Important!

- Note: several important steps have been skipped:
  - Acquiring GENI credentials from the GPO
    - Either as a PI or a participant in an existing project
  - Converting credentials into an Flukes-compatible form
    - See ExoGENI wiki
  - Signing up on the geni-orca-users@googlegroups.com to be whitelisted on ExxoGENI
    - See ExoGENI wiki

A word about keys/credentials

- There are two types of credentials typically used in GENI
  - GPO-issued credential – an X.509 signed certificate confirming your identity
    - It can take several forms – a .pem file, a .jks file, a .p12 file
    - Your private key is locked with a password
  - SSH keys are used to login to the provisioned nodes
    - The tools grab your public key from your home directory and install it into the nodes
    - Your private SSH key may or may not be locked with a password (it is in this tutorial)
- In this tutorial the passwords locking your private X.509 key and ssh key are the same
  - They don't have to be
  - The two are not related

Section: ExoGENI Ecosystem



ExoGENI Testbed

- 14 GPO-funded racks
  - Partnership between RENCI, Duke and IBM
  - IBM x3650 M3/M4 servers
    - 48G RAM
    - Dual-socket 8-core Intel X5650 2.66Ghz CPU
    - 10G dual-port Chelseo adapter
  - BNT 8264 10G/40G OpenFlow switch
  - DS3512 6TB sliverable storage
- Each rack is a small networked cloud
  - OpenStack- and xCAT based
  - EC2 nomenclature for VM node sizes (m1.small, m1.large etc)
  - Baremetal node provisioning
  - Interconnected by combination of dynamic and static L2 circuits through regionals and national backbones
- http://wiki.exogeni.net

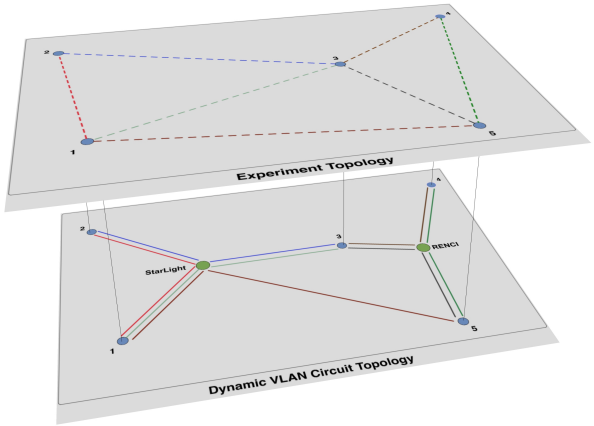
ExoGENI Status

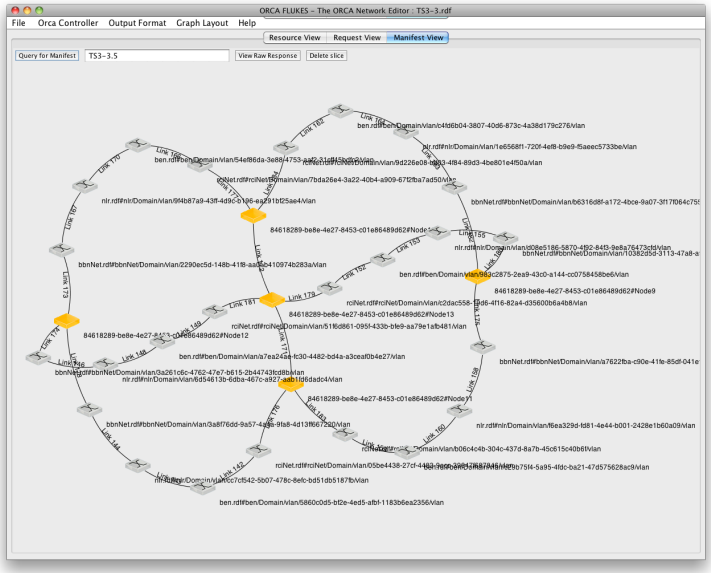
- 3 new racks deployed
  - RENCI, GPO and NICTA
- 2 more racks delivered, being configured
  - FIU and UH

- Connected via BEN (http://ben.renci.org), LEARN and NLR FrameNet, (eventually I2)

ExoGENI slice isolation

- Strong performance isolation is the goal
- Compute instances are KVM based and get a dedicated number of cores (ExoGENI does not over-provision cores)
  - Currently all instances get 1 core (different RAM and disk).
- VLANs are the basis of connectivity
  - VLANs can be best effort or bandwidth-provisioned (within and between racks)
  - Current hardware in the racks allows best-effort VLANs only – will be remedied within a month





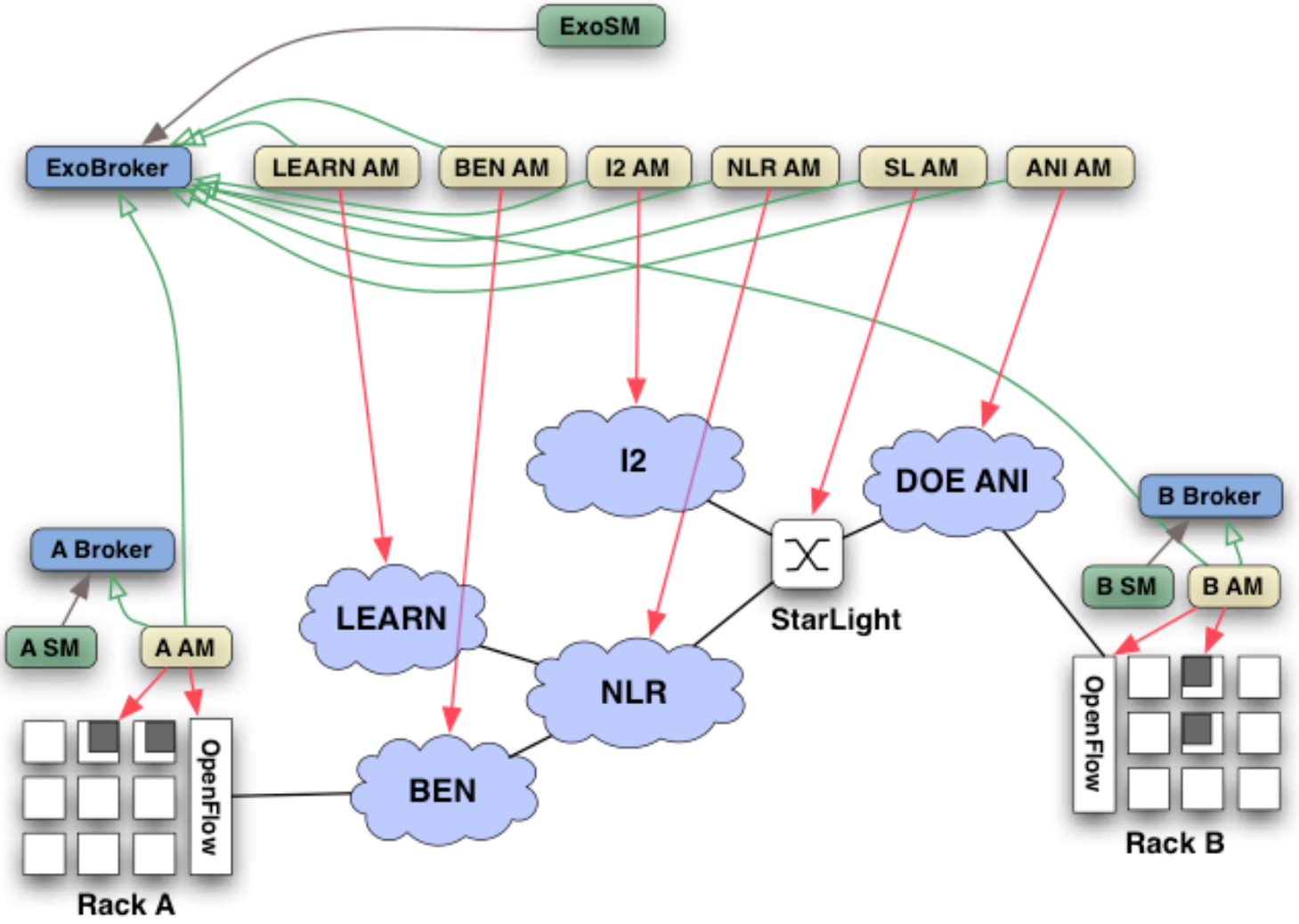
ORCA Overview

- ORCA is the control framework for ExoGENI
- Originally developed by Jeff Chase and his students at Duke
- Funded as Control Framework Candidate for GENI
  - Jointly developed by RENCI and Duke for GENI since 2008.
- A federation of networked clouds with a variety of interfaces
  - Native ORCA
  - GENI AM API
- Unique feature of ExoGENI: experimenter can
  - Operate on individual racks as independent aggregates
  - Operate on entire testbed and link racks together using ExoSM

ORCA deployment in ExoGENI

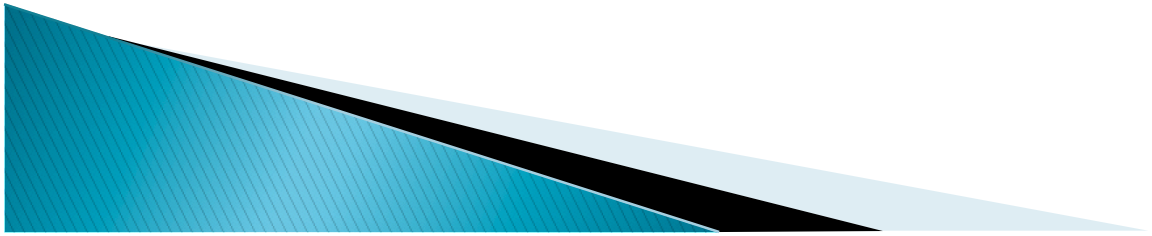
- Each rack runs its own Orca actor called the 'SM' that exposes
  - ORCA native APIs
  - GENI AM API
- Rack-local SM can only create slices with resources **within that rack** (virtual machines and VLANs)
- Special 'ExoSM' has global visibility
  - Has access to a fraction of resources **resources in all racks**
  - Has access to **network backbone resources for stitching topologies between racks**
- ExoSM
  - https://geni.renci.org:11443/orca/xmlrpc
- Rack SMs listed on wiki:
  - https://wiki.exogeni.net/doku.php?id=public:experimenters:orca_sm

Orca Deployment in ExoGENI



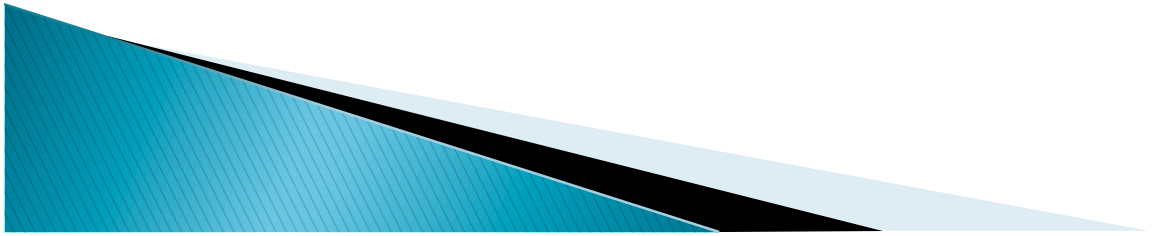
How are resources split?

- Resources in each rack are split between rack SM and ExoSM.
- Currently the split is 50/50 for VMs and internal VLANs between rack SM and ExoSM
- Baremetal nodes are usable only by ExoSM
- Stitching links from regional and national providers are delegated to ExoSM only.



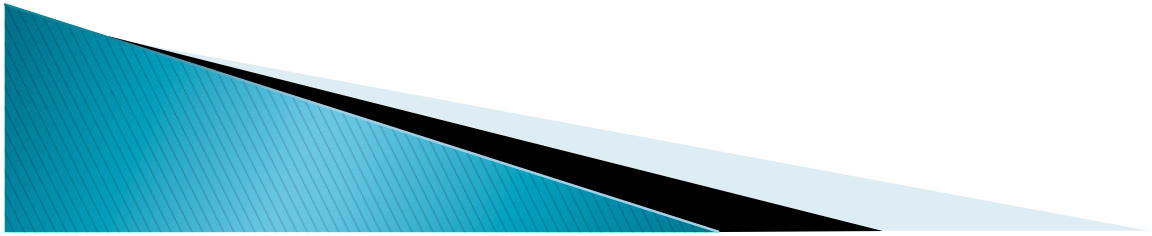
Reasons to use ExoSM

- Simplicity
- Inter-rack slices
- Bare-metal nodes



Reasons to use rack SM

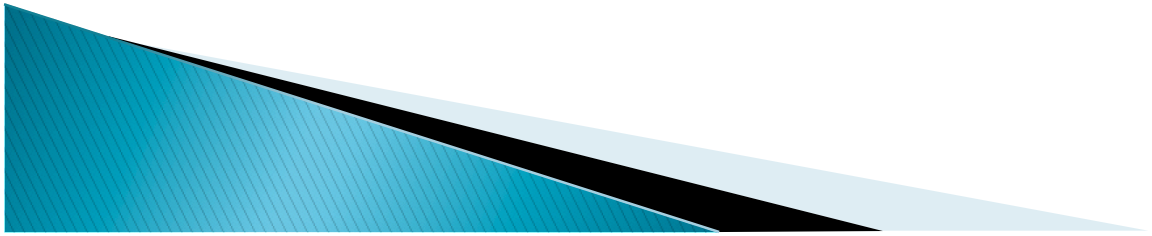
- Resource availability
- Single-rack experiments with VMs only



How do I use it?

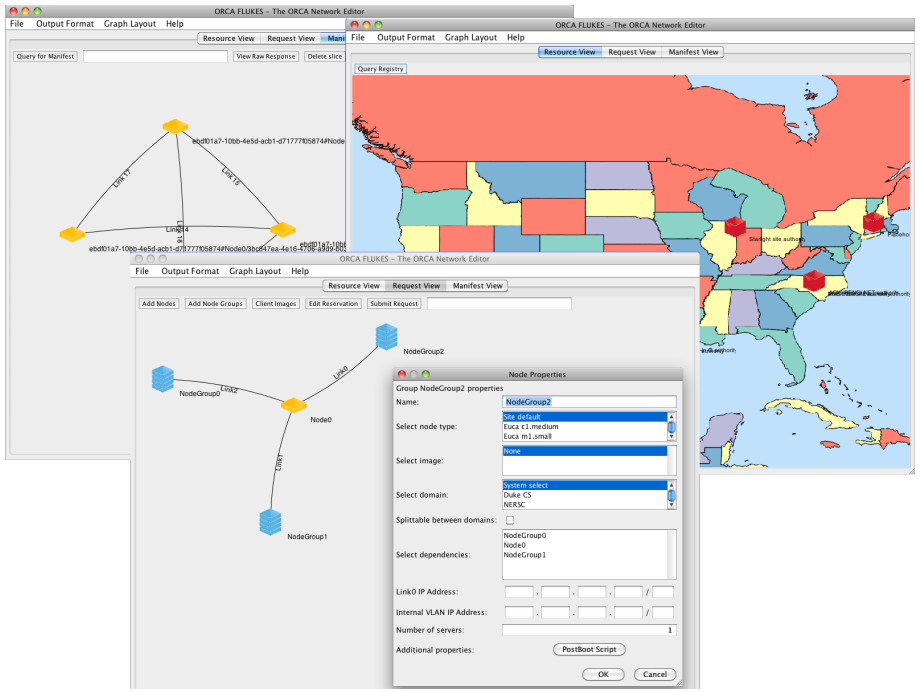
- Request credentials through the GPO
- Build your own VM Image [optional]
- https://wiki.exogeni.net/doku.php?id=public:experimenters:images

- Define slice topology and submit request
  - Use either Omni or Flukes
- Maximum slice lifetime is 2 weeks



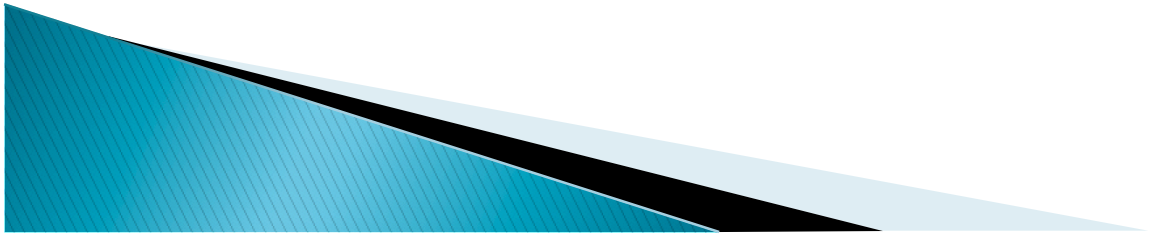
Section: Flukes Overview

- Graphical tool for creating and managing slice topologies in ORCA
  - JAVA (JNLP)



Section Overview

- Configuring Flukes prior to launch (DONE!)
- Launching Flukes
  - GUI Overview
  - Nodes, NodeGroups and Link parameters
  - Node-level vs. reservation level options
- Building slice request topologies
- Launching slice requests
- Inspecting slice manifests
- Logging into nodes in the slice
- NEuca-py tools

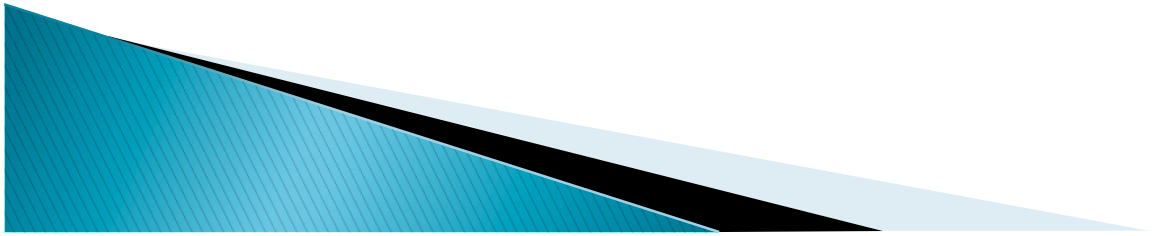


Flukes

- Permanent stable version link
  - http://geni-images.renci.org/webstart/flukes.jnlp
  - You can simply retrieve this file and either double-click on it (if you OS recognizes the file type) or launch it using 'javaws flukes.jnlp'
  - Oracle Java 6 recommended
- Can I use Flukes outside of ExoGENI?
  - No. Flukes uses semantic web mechanisms (RDF and OWL) to describe resources that is only compatible with ORCA and ExoGENI.
- Can I use GENI tools with ExoGENI?
  - Yes. You can use e.g. omni
  - See https://geni-orca.renci.org/trac/wiki/orca-and-rspec for conventions in using GENI RSpec in ExoGENI

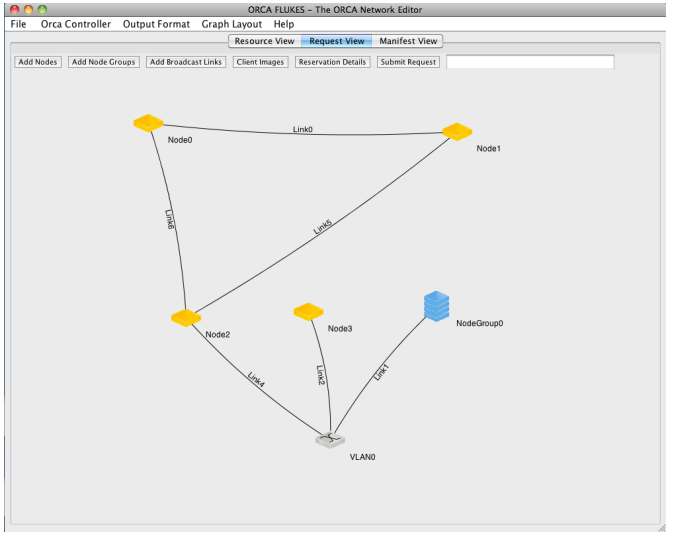
GUI Overview

- Tabs
  - Resources, Request, Manifest
- Menus
  - Current properties
  - Overwriting properties ($HOME/.flukes.properties)
- Adding nodes, nodegroups and links



Node and Link parameters

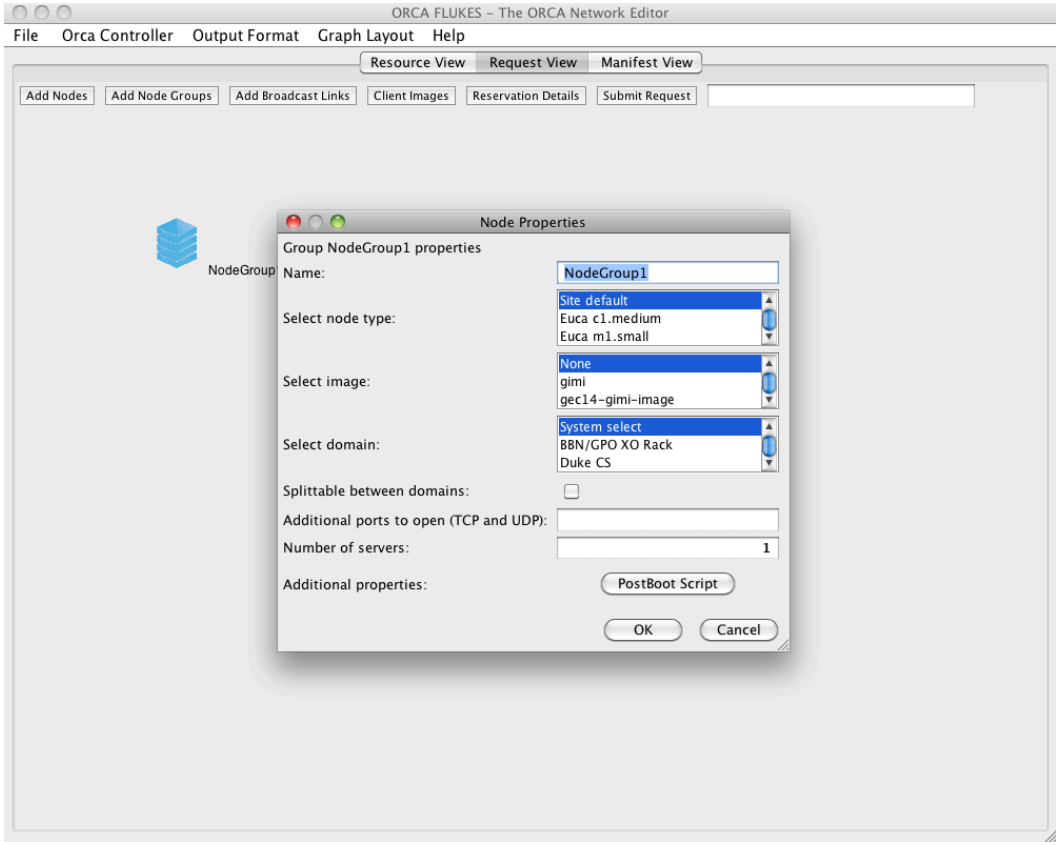
- Create a single node
- Right-click on the Node
  - Look at properties
  - Edit properties
    - Node type (size)
    - VM image
    - Domain (binding)
    - PostBoot script



- Create another node, link the two together
- Right-click and open properties again
  - Specify IP address on the link
  - Node functional dependencies
- Right-click on links
  - Inspect and edit link properties
  - Note only bandwidth is currently respected (and not everywhere due to hardware limitations)
- Broadcast links
- Specifying vlan tags
  - Only ‘special’ shared tags can be specified

NodeGroup parameters

- Create a single unattached node group
- Right-click to inspect and edit properties
  - Group sizes
  - Splittable groups



Nodes and NodeGroups

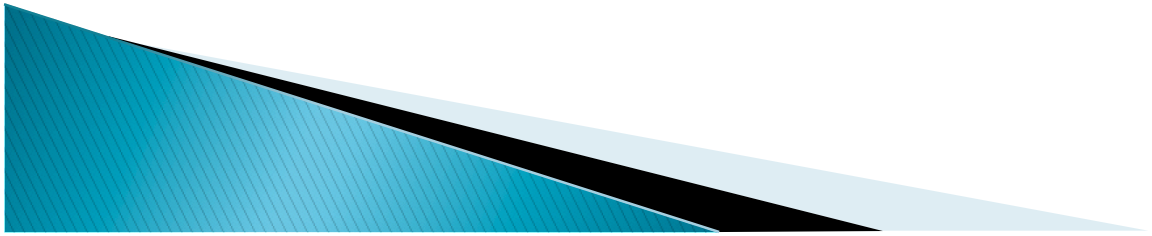
- A Node is an individual compute element
  - Typically a VM or a hardware node
  - IP address(es) on links, size, image, site binding, post boot script
  - Can I control management IP address assignment? NO!
- A NodeGroup is a group of identically configured nodes
  - A lot like a node except
    - PostBoot script is templated using Velocity template engine
      - https://geni-orca.renci.org/trac/wiki/flukes
    - IP address assignment is semi-automatic (starting with a user-specified address)
    - Node groups can be splittable between sites
      - A node group that is too big for any site and that is not explicitly bound may be split across sites

What do I get when I …?

- Create a standalone node?
  - You get a single compute element at one of the sites with a single network interface to the management network through which you can SSH into the node
  - Management interface is always eth0
- Connect two nodes together?
  - You get two compute elements each with two network interfaces – one for management access and one for the link between two nodes.
  - User-controlled interfaces start with eth1
  - You can control IP address assignment on the interfaces linking the two nodes (use RFC1918; suggested range: 172.16.0.0/16)

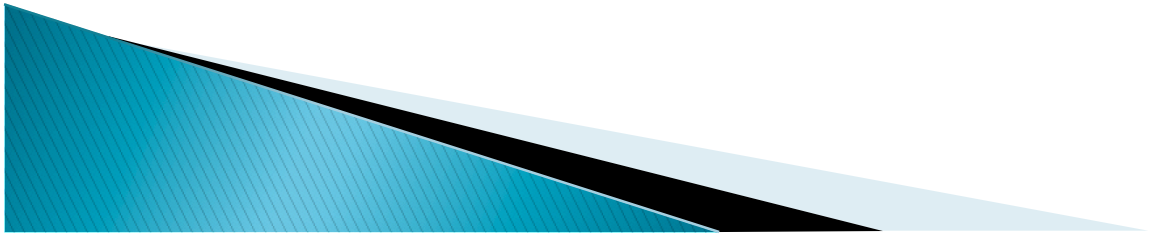
What do I get when I …?

- Create a standalone NodeGroup?
  - You get some number of nodes (specified in the group size) each with a single interface to the management network (eth0)
  - Nodes typically will be within the same rack
  - If node group is marked splittable nodes may be split across sites
- Connect a node group to a node or another node group?
  - All nodes within the group and the adjacent node (or all nodes in both groups) have interfaces on a common VLAN. They also have management interfaces (eth0)
  - IP address is specified similarly to private VLAN
  - Beware of address clashes! (i.e. here is a piece of rope, feel free to shoot yourself in the foot)
- Connect a node group to a broadcast link?
  - You get a cluster with a dedicated Layer2 backplane.
  - You can assign IP addresses to interfaces in this backplane.



Can I tell which interface in the node will be eth1, eth2 etc?

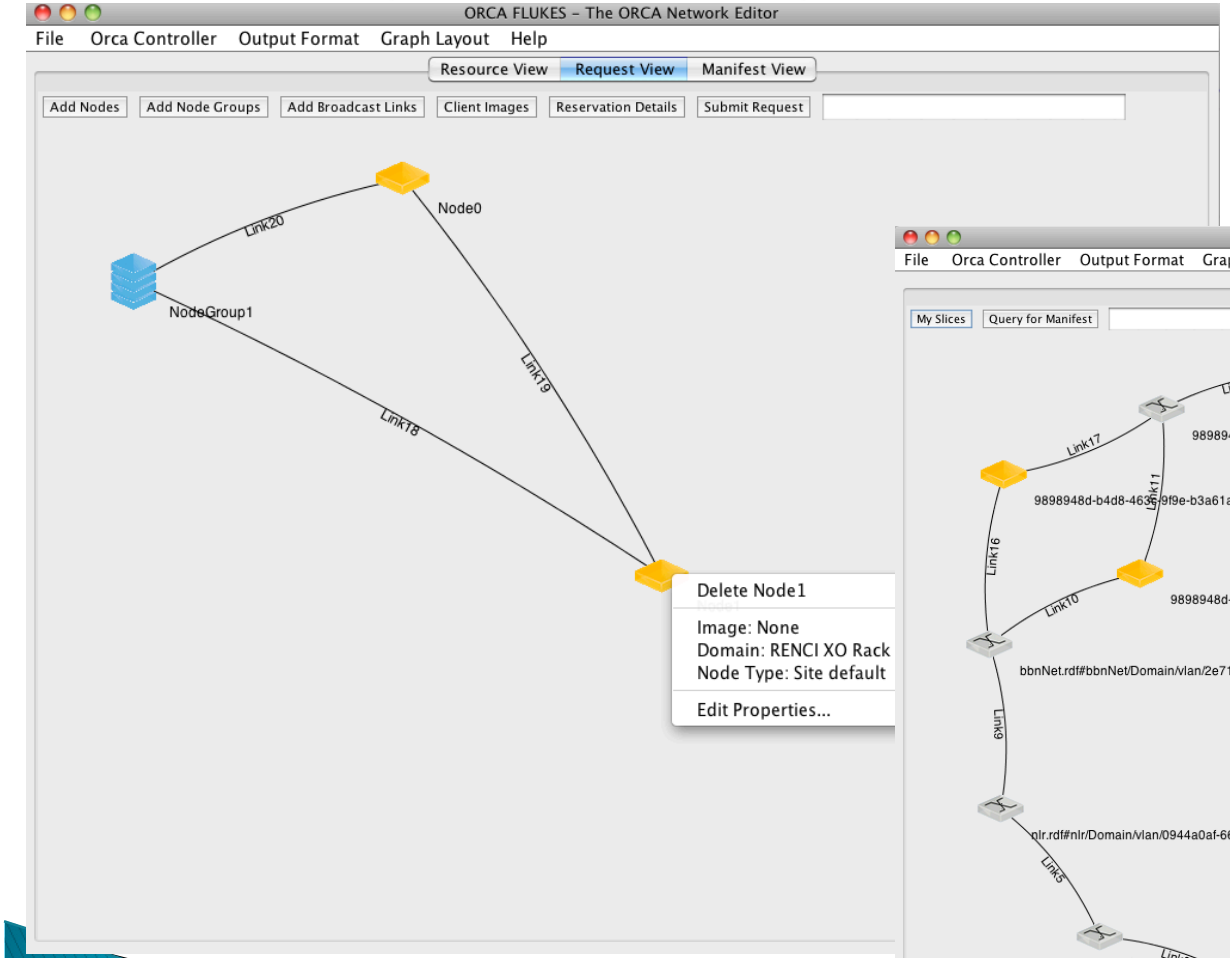
- No, nor should you need to. Interfaces are identified by links they belong to.
- Note that different OSs name interfaces differently



Domain binding

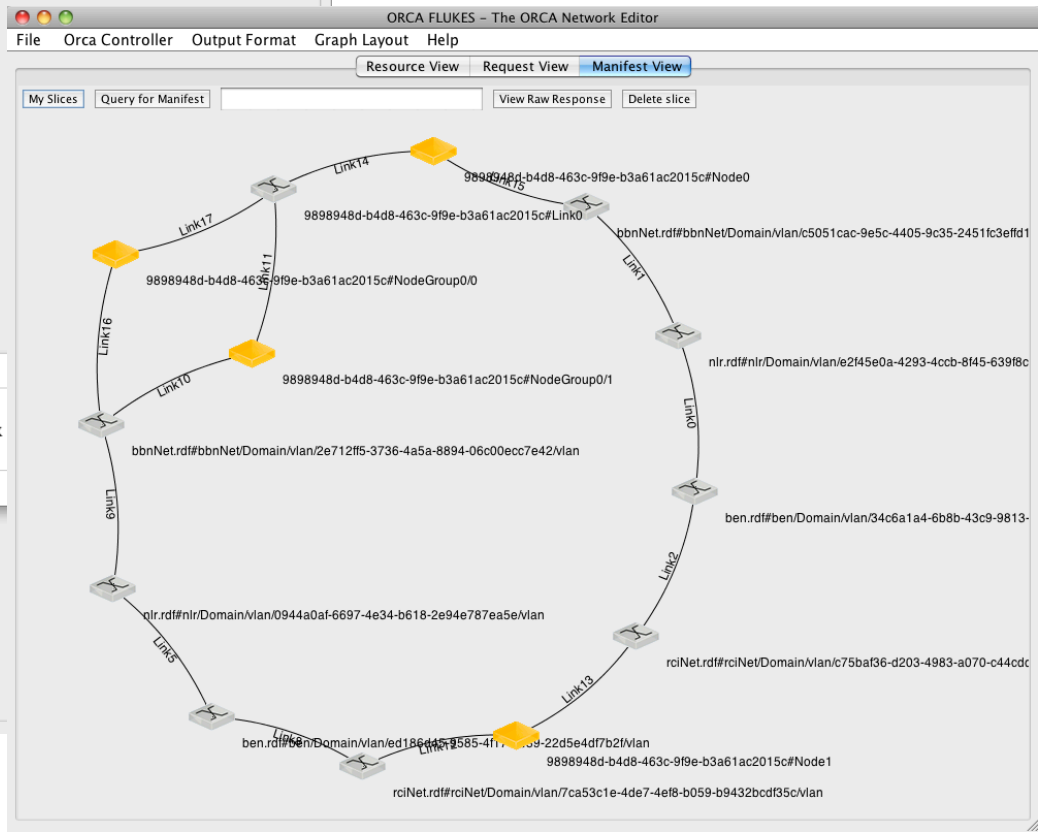
- Binding refers to the selection of specific domains/sites/racks for parts of your slice
- Binding can be done on individual slice elements (nodes) or the entire reservation
- Unbound requests are automatically bound to domains with available resources.
  - This depends on the visibility of the SM.
  - ExoSM can bind to any rack
  - Rack SM will always bind to its rack
- Bound requests are honored if resources are available
  - To create an inter-rack request, bind some of the nodes in it to one rack, and others to another
  - Can only be done via ExoSM!

An Example – an inter-domain slice (RCI–BBN)



Slice Request

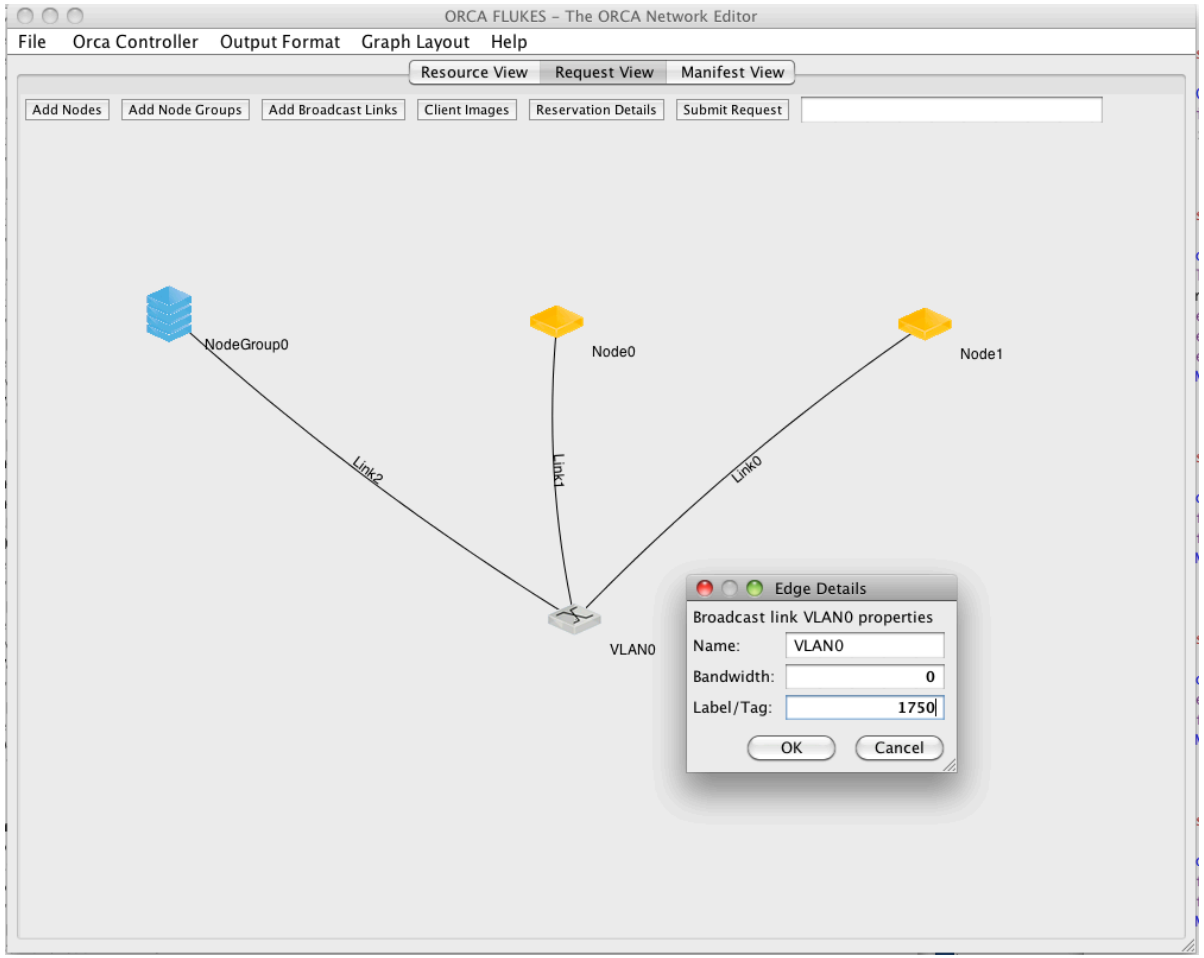
Slice Manifest



Attaching nodes to Mesoscale

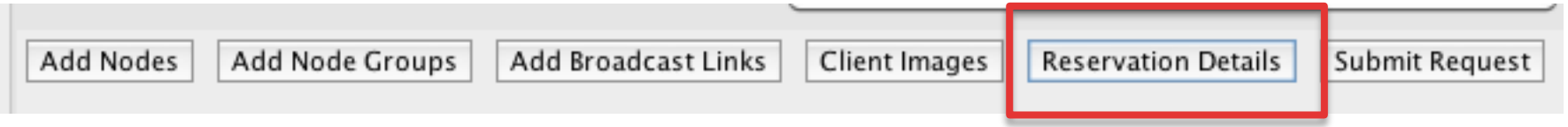
- ExoGENI racks have layer 2 connections to NLR's OpenFlow MesoScale deployment
- Each rack attaches to it via a unique VLAN id
  - https://wiki.exogeni.net/doku.php?id=public:experimenters:resource_types:start
- To attach a VM or a baremetal node to a mesoscale VLAN, you need to
  - Decide which rack you will be using (your request must be bound)
  - Look up the VLAN id of mesoscale at this site (see URL above)
  - Create a topology with some number of nodes
  - Add a 'Broadcast Link'
  - Right-click on the broadcast link, select 'Edit Properties' and enter the tag in that 'Label/Tag' field
  - No QoS support on mesoscale vlans.

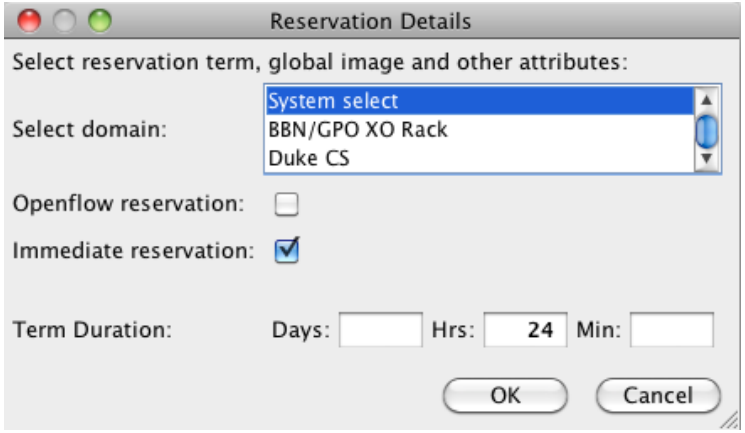
Example of attaching nodes to mesoscale



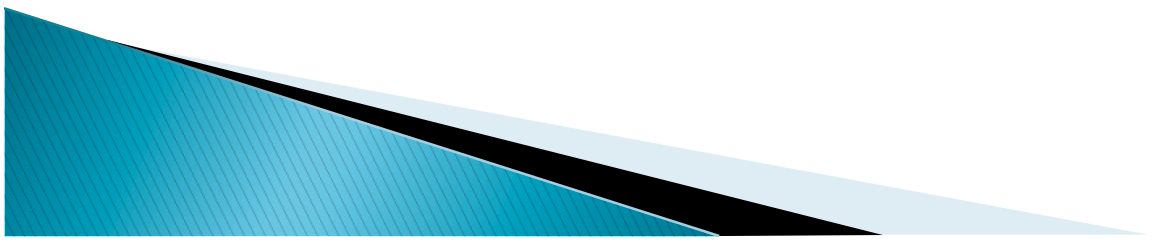
Reservation-level options

- Domain binding for the entire slice
- OpenFlow slice parameters can currently only be specified at reservation level
- Slice duration





Section: Creating slices with Flukes

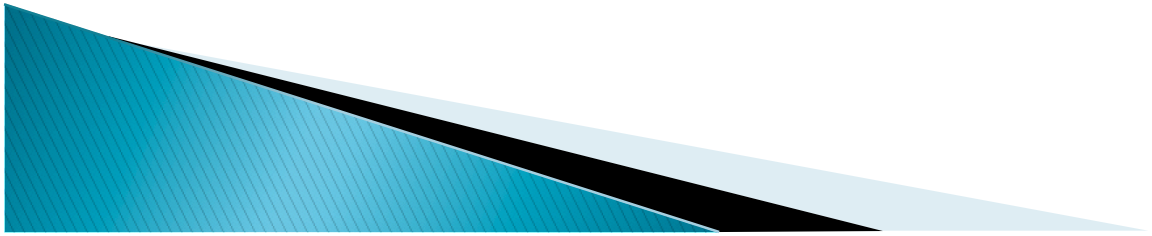


Launch a slice

- Click ‘File | New Request’
- Draw slice topology (nodes or groups)
  - Select m1.large size
  - Select image deb6-2g-zfilesystem
- Specify slice duration (click ‘Edit Reservation’) or binding
  - You can try binding to BBN XO Rack, RCI XO Rack or Duke CS
  - You can also let ORCA pick an available rack for you and leave the request unbound
- If you want to create a slice ‘In the land Down Under’
  - Select ‘https://nicta-hn.exogeni.net’ under ‘Orca Controllers’
  - Leave the slice elements unbound

Launch a slice

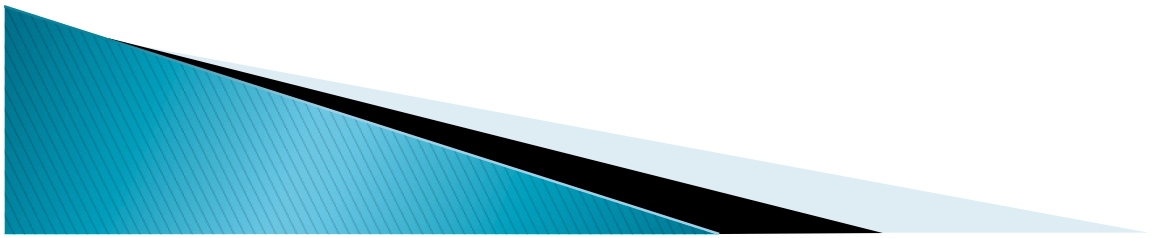
- Fill in slice name (must be unique, e.g. gimiXX)
- Click ‘Submit Request’
  - Type in the alias of the key in the keystore (‘gimiXX’)
  - Type in the password [given to you]
  - ‘OK’
- Inspect the output window
  - Mainly a debugging tool. Will go away in the future.



Inspect slice manifest

- Switch to Manifest tab
- Click on ‘My Slices’ and select your slice
  - Click OK to display manifest and reservation states
- Periodically poll ‘Query for Manifest’
  - Inspect the states of reservations. ‘Ticketed’ means it is proceeding. ‘Active’ means it is ready
  - If you see ‘Failed’ you have a problem
  - Click on the reservation to see a detailed error message
  - Visit https://geni-orca.renci.org/trac/wiki/orca-errors
- Play around with layouts (‘Graph Layout’) to get something pleasing

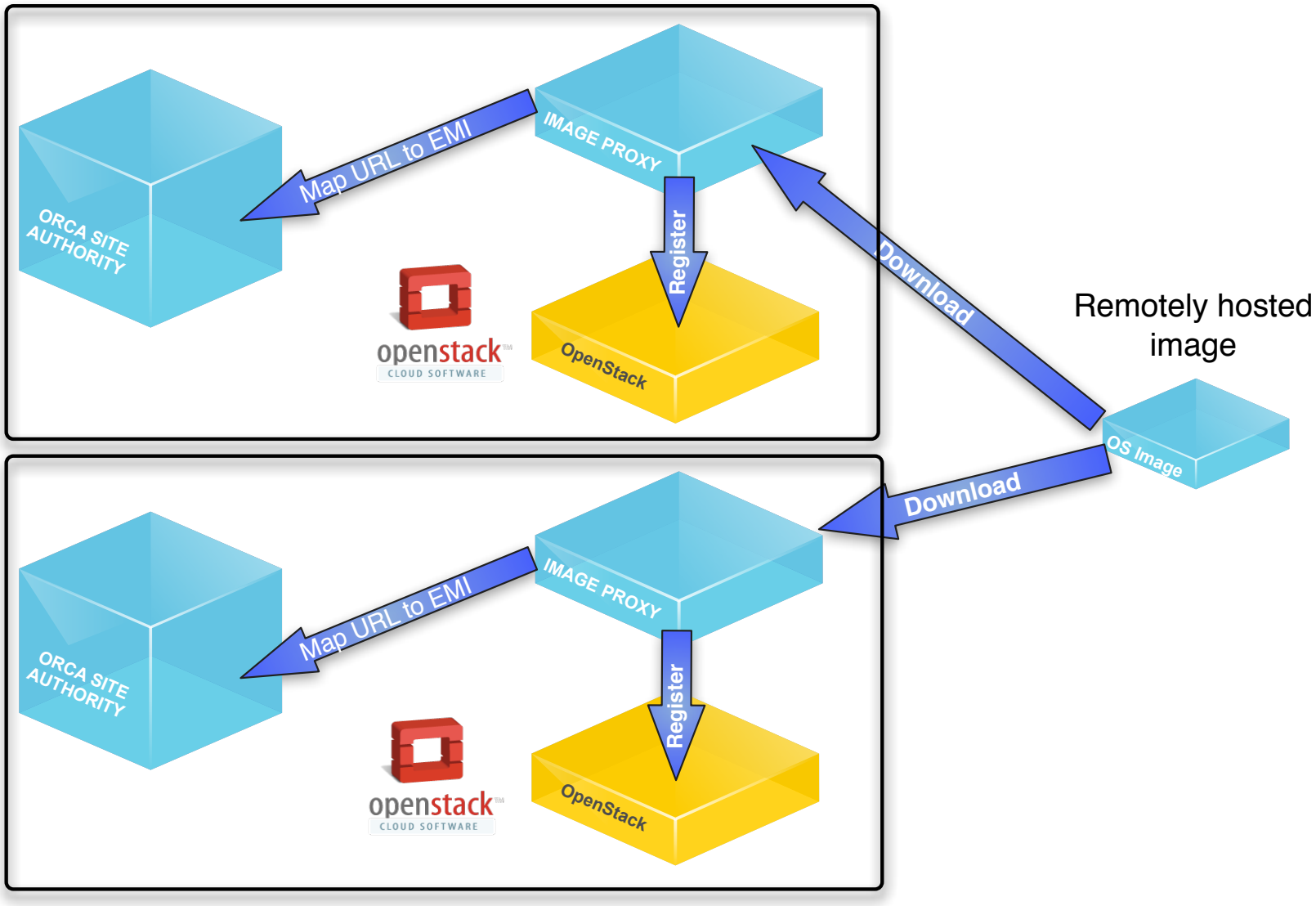
ORCA and VM Images



VM images

- Creating your own image
- Specifying your own image for ORCA
- Regarding delays:
  - Images are downloaded and registered with the site at the time of slice creation
    - If you repeatedly use the same image and the site already has it, this step is skipped
    - Images may be cached-out causing longer delays (to download and re-register)
- Are there examples of known good images?
  - https://wiki.exogeni.net/doku.php?id=public:experimenters:images
  - Alternatively: http://wiki.exogeni.net, click on 'experimenters', then 'images'.

VM Images

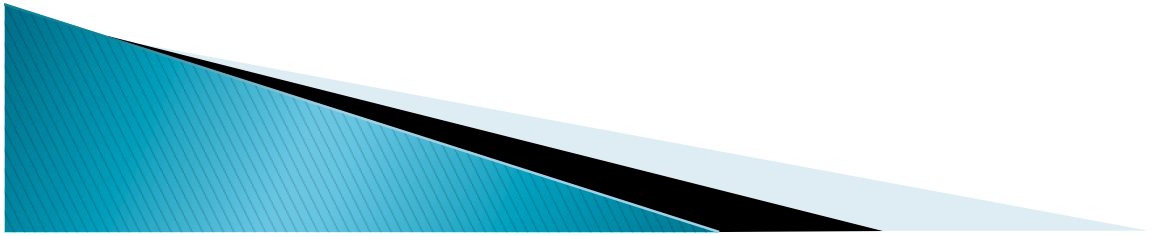

ORCA SITE AUTHORITY
Map URL to EMI
IMAGE PROXY
Register
openstack
CLOUD SOFTWARE
OpenStack
Download
Remotely hosted image
OS Image
Download
ORCA SITE AUTHORITY
Map URL to EMI
IMAGE PROXY
Register
openstack
CLOUD SOFTWARE
OpenStack

Example image metafile

```
<images>
    <image>
        <type>ZFILESYSTEM</type>
        <signature>b54ed5a42cd99475c3d5d7c7a9839b69cf2076d5</signature>
        <url>http://geni-images.renci.org/images/workflows/pegasus/images/
    pegasus-4.0-v0.3.sparse.img.tgz</url>
    </image>
    <image>
        <type>KERNEL</type>
        <signature>f8a64d3bc429e8fb46c94ff3b11a932a27c142bc</signature>
        <url>http://geni-images.renci.org/images/workflows/debian-squeeze-
    kernel/vmlinuz-2.6.28-11-generic</url>
    </image>
    <image>
      <type>RAMDISK</type>
      <signature>6225968f43299aa40f6b1491360f3ce080bd16c4</signature>
      <url>http://geni-images.renci.org/images/workflows/debian-squeeze-kernel/
    initrd.img-2.6.28-11-generic</url>
    </image>
</images>
```

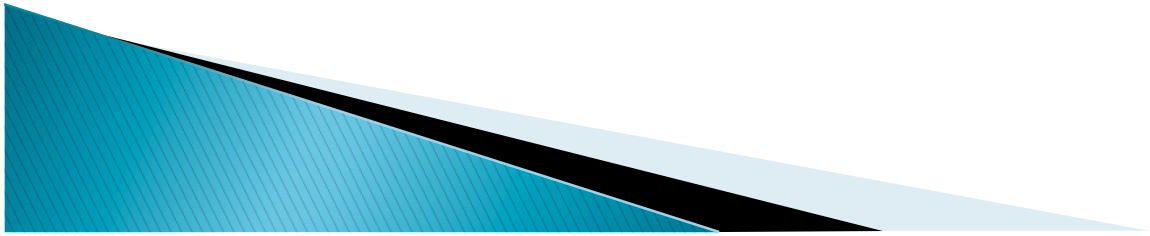
How does ORCA refer to an image

- URL of a metafile (can be same or different webserver as the image)
- SHA1 checksum of the metafile (to ensure it has not been modified)
- Workflow
  - Create filesystem, kernel, ramdisk
  - Place on some webserver
  - Take SHA1 signatures of each file
  - Generate metafile
  - Take SHA1 signature of metafile and its URL and add it to .flukes.properties or put it in Rspec



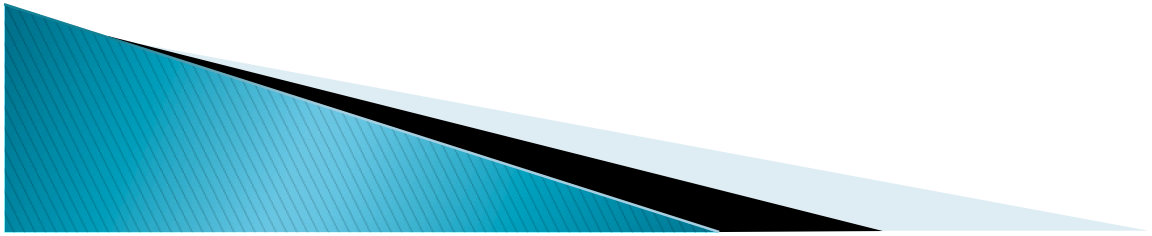
ORCA and VM images

- Can I put my image on your server?
  - Sorry, no.
- Will my image always remain cached at the racks?
  - No, depending on the use, your image may be cached out.
  - All this means is that sometimes you will experience a longer delay bringing up your slice



Building your own image

- Will it always remain this complicated?
  - Maybe. Building a valid OS image is still an advanced task.
  - In many cases you can use post-boot scripts to automate the customization of your instances *without* modifying the OS image.
  - We will be offering the use of an 'image playpen' facility – a separate installation of OpenStack with a convenient portal, where you can test and debug your images without trying them out directly on ExoGENI



Where to find information

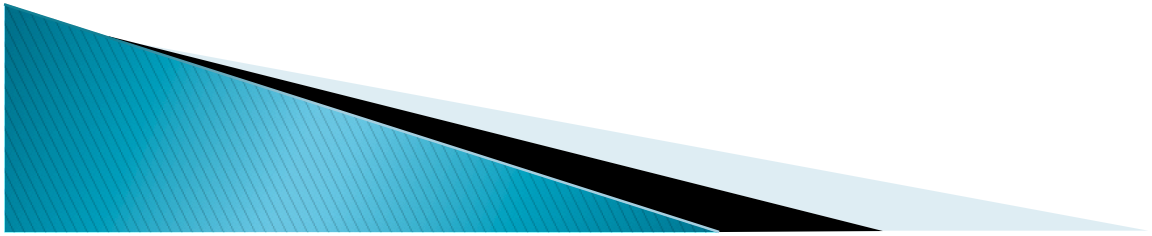
- This is all too confusing. How will I ever be able to find all this information on my own?
  - All the information is linked to by the main ExoGENI experimenter page
  - http://wiki.exogeni.net, click on ‘experimenters’

Back to manifests: Logging into nodes

- Right click on node in manifest
- Select 'Login to node'
  - In terminal window type in SSH key password, same as the key password you entered when you submitted the slice
- Inspect uptime

  $ uptime
- Inspect the output of your boot script
- Inspect interfaces

  $ ifconfig
- Try to ping node neighbors



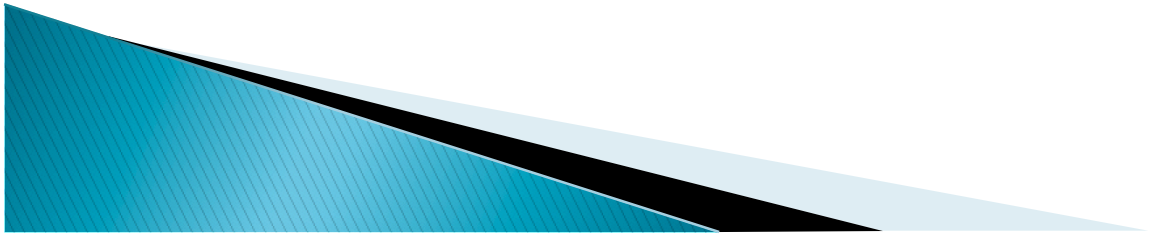
NEuca-py tools

- Neuca-py tools are loaded in the image
  - They configure network interfaces at boot time
  - They execute the post boot script
  - An image without NEuca tools will do neither of those things
    - You can still configure interfaces manually
- Allow you to inspect the VM configuration
- Run 'neuca' to get the list of neuca tools

  $ neuca
- Run 'neuca-user-data'
  - Note your boot script
- If you create your own VM image you are strongly encouraged to install NEuca tools on it
  - Visit https://geni-orca.renci.org/trac/wiki/NEuca-guest-configuration for instructions

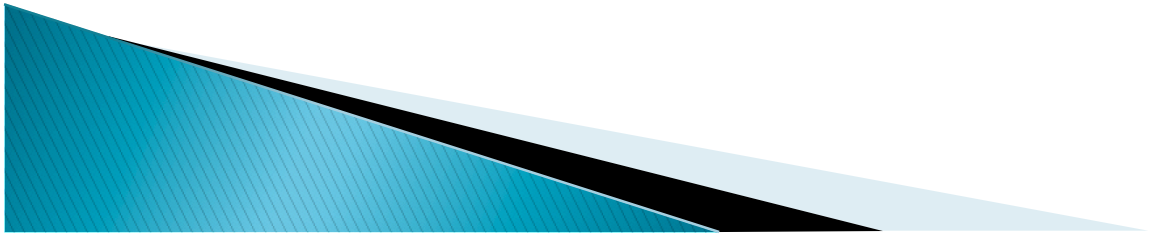
OpenFlow slices on the racks

- You can start an OF controller on some publically reachable host (it can be a single-node slice in ExoGENI)
- You can create a slice topology and point ORCA to your controller
  - Under ‘Reservation Details’ in Request pane, select ‘OpenFlow reservation’ and fill in the details of your controller
- When your slice is created, vlans in it will be under the control of your controller



OpenFlow slices and mesoscale

- You can use FOAM controller in each rack to create an OpenFlow slice
- You can get compute nodes attached to the mesoscale VLANs in each rack using ORCA (Flukes or Omni)
- Put the two together and you have a mesoscale OpenFlow slice



What is coming to ExoGENI in the next 2-4 months?

- Upgrade to OpenStack Essex
  - Support for QoS in slice links will be back
  - Better stability
  - Being tested now, will begin early deployment immediately following the GEC
- Image playpen
  - To help experimenters create custom images faster
- More racks
  - UH and FIU racks are in place, being configured
  - More racks will be coming in 2013.
- New firmware for rack backplane switches
  - Improved QoS support, better scalability

Thank you for attending the tutorial!

- More ExoGENI information
  - http://www.exogeni.net
- More Orca information
  - http://geni-orca.renci.org