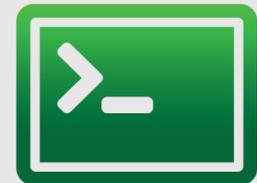


Getting Started with GENI - Part 2

ICDCS 2013
8 July 2013



Design/Setup



Execute



Finish

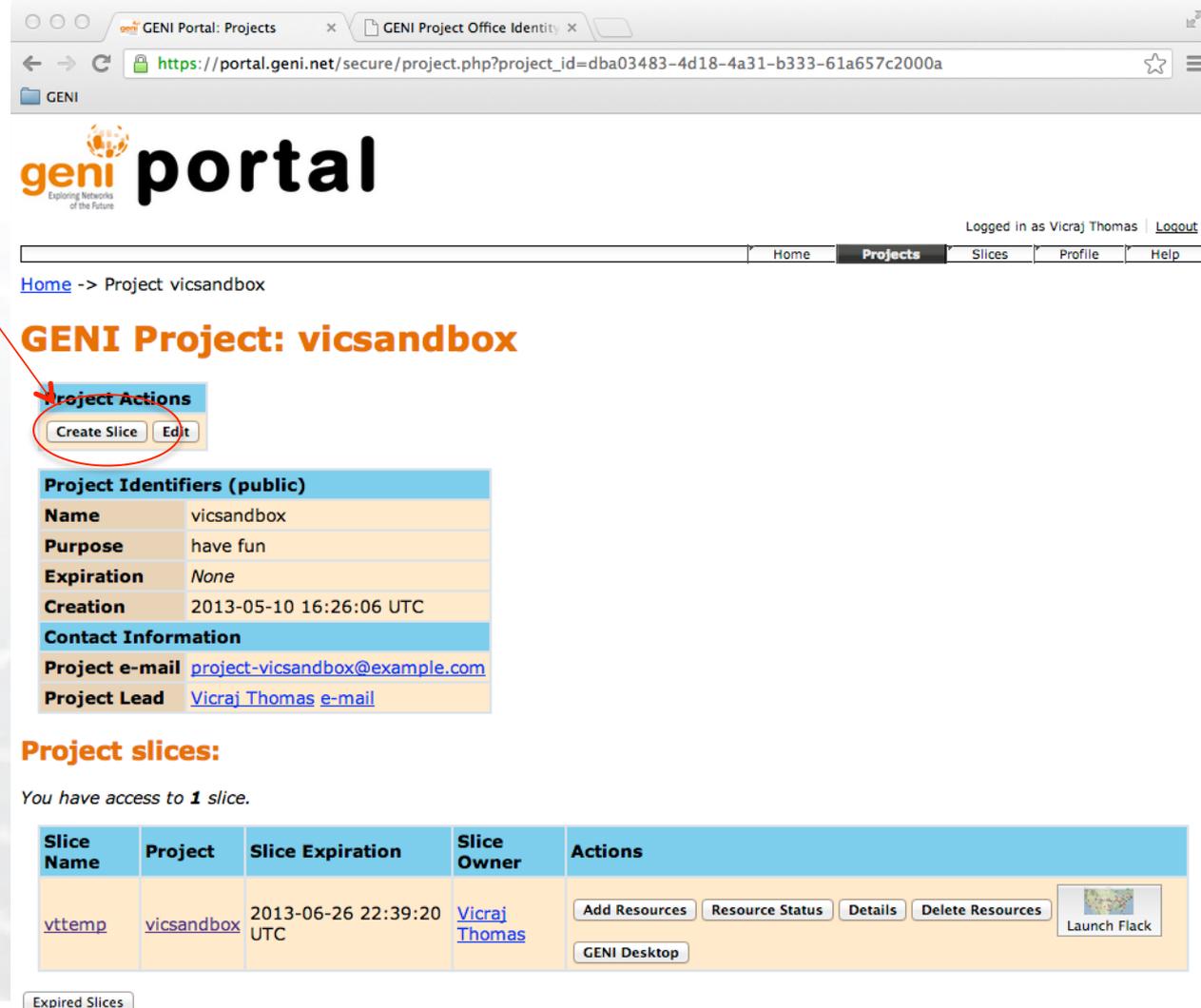
- Log into the **GENI** Portal



The screenshot shows a web browser window with the URL `https://portal.geni.net/eds/?entityID=https%3A%2F%2Fpanther.gpolab.bbn.com%2Fshibboleth&return=https%3A%2F%2Fporta...`. The page title is "IDP select test bed". The main content area features the GENI logo and the text "Please sign in using your account at one of our partners:". Below this, a red-bordered box contains a "Use a suggested selection:" section with the GENI logo and "GENI Project Office" link. Underneath is a text input field with the placeholder "Or enter your college, university, or organization's name", a "Continue" button, and a link "Allow me to pick from a list" with a "Get Help" button. To the right of the box is a larger GENI logo. At the bottom, there are links for "Can't login via any of the above organizations? Request a login from the GPO" and "Need help? Contact GENI Help". A footer bar contains "Web Design by Free Templates Online" and "GENI is sponsored by the National Science Foundation" with the NSF logo.

In Part 1 You Learned to...

- Log into the GENI Portal
- Create a **slice**



GENI Portal: Projects | GENI Project Office Identity

https://portal.geni.net/secure/project.php?project_id=dba03483-4d18-4a31-b333-61a657c2000a

GENI

geni portal

Logged in as Vicraj Thomas | Logout

Home | **Projects** | Slices | Profile | Help

Home -> Project vicsandbox

GENI Project: vicsandbox

Project Actions

Create Slice | Edit

Project Identifiers (public)

Name	vicsandbox
Purpose	have fun
Expiration	None
Creation	2013-05-10 16:26:06 UTC

Contact Information

Project e-mail project-vicsandbox@example.com

Project Lead [Vicraj Thomas e-mail](#)

Project slices:

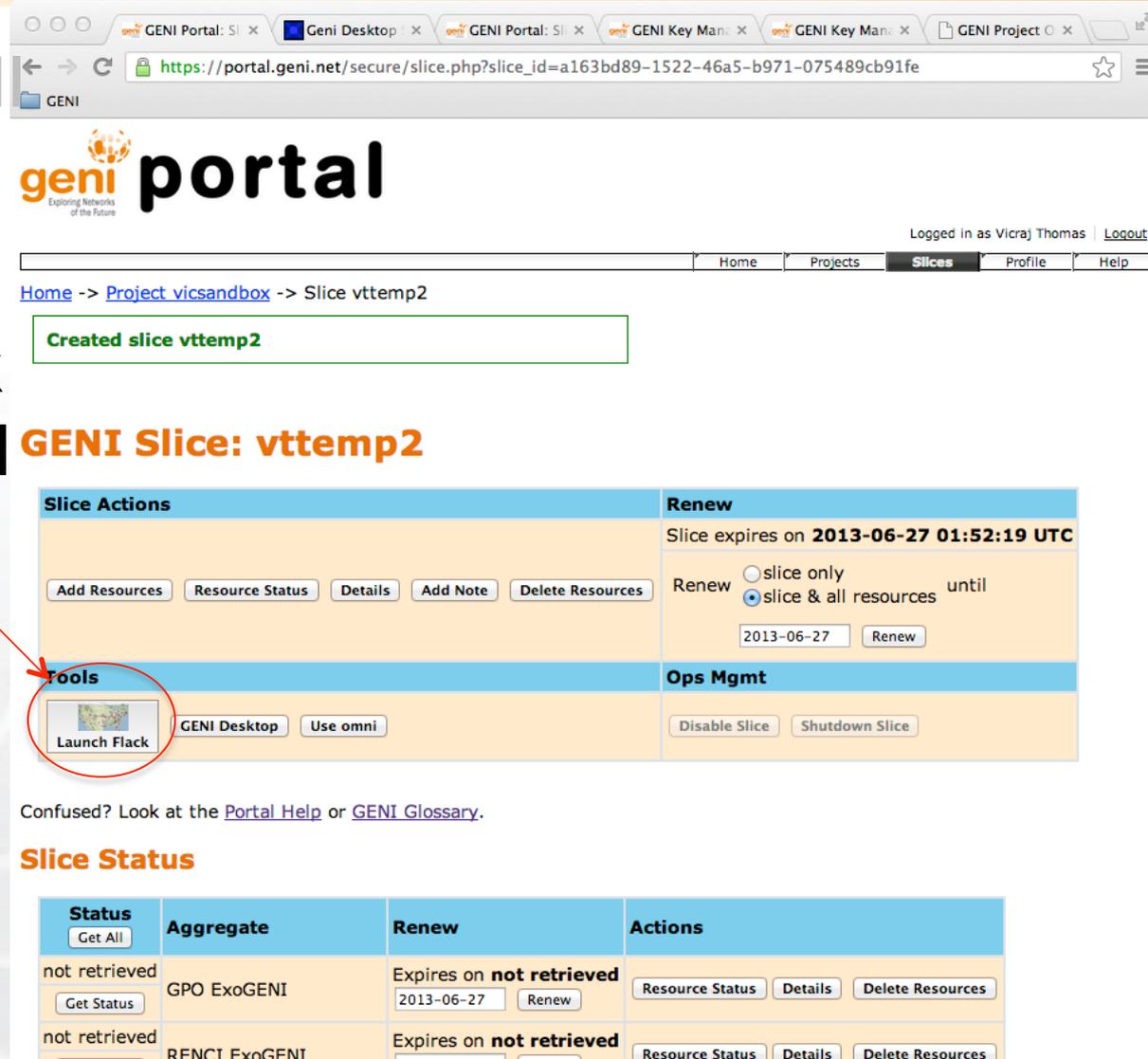
You have access to **1** slice.

Slice Name	Project	Slice Expiration	Slice Owner	Actions
vttemp	vicsandbox	2013-06-26 22:39:20 UTC	Vicraj Thomas	Add Resources Resource Status Details Delete Resources Launch Flask GENI Desktop

Expired Slices

In Part 1 You Learned to...

- Log into the GENI Portal
- Create a slice
- Launch the Flack experimenter tool



GENI Portal: SLI x Geni Desktop x GENI Portal: SLI x GENI Key Man: x GENI Key Man: x GENI Project O x

https://portal.geni.net/secure/slice.php?slice_id=a163bd89-1522-46a5-b971-075489cb91fe

geni portal

Logged in as Vicraj Thomas | Logout

Home Projects Slices Profile Help

Home -> Project vicsandbox -> Slice vttemp2

Created slice vttemp2

GENI Slice: vttemp2

Slice Actions	Renew
Add Resources Resource Status Details Add Note Delete Resources	Slice expires on 2013-06-27 01:52:19 UTC Renew <input type="radio"/> slice only <input checked="" type="radio"/> slice & all resources until 2013-06-27 Renew
Tools	Ops Mgmt
<div style="border: 2px solid red; border-radius: 50%; padding: 5px; display: inline-block;">  Launch Flack </div> GENI Desktop Use omni	Disable Slice Shutdown Slice

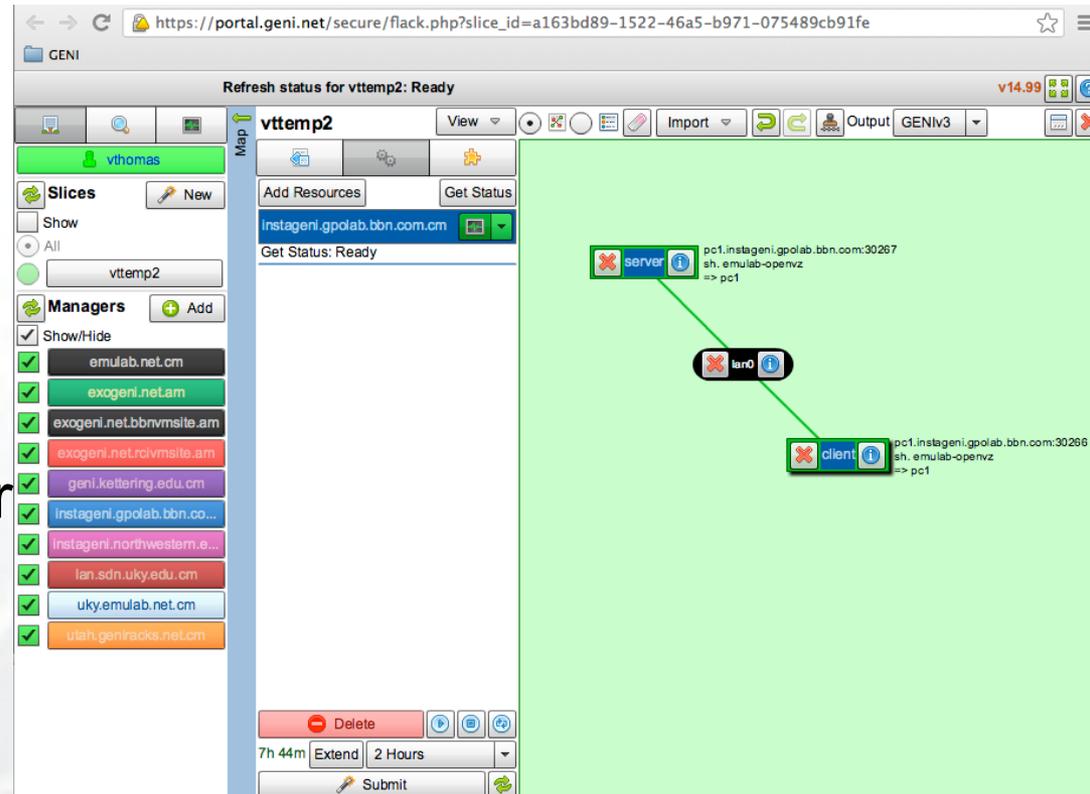
Confused? Look at the [Portal Help](#) or [GENI Glossary](#).

Slice Status

Status	Aggregate	Renew	Actions
not retrieved Get All Get Status	GPO ExoGENI	Expires on not retrieved 2013-06-27 Renew	Resource Status Details Delete Resources
not retrieved	RENCI ExoGENI	Expires on not retrieved	Resource Status Details Delete Resources

In Part 1 You Learned to...

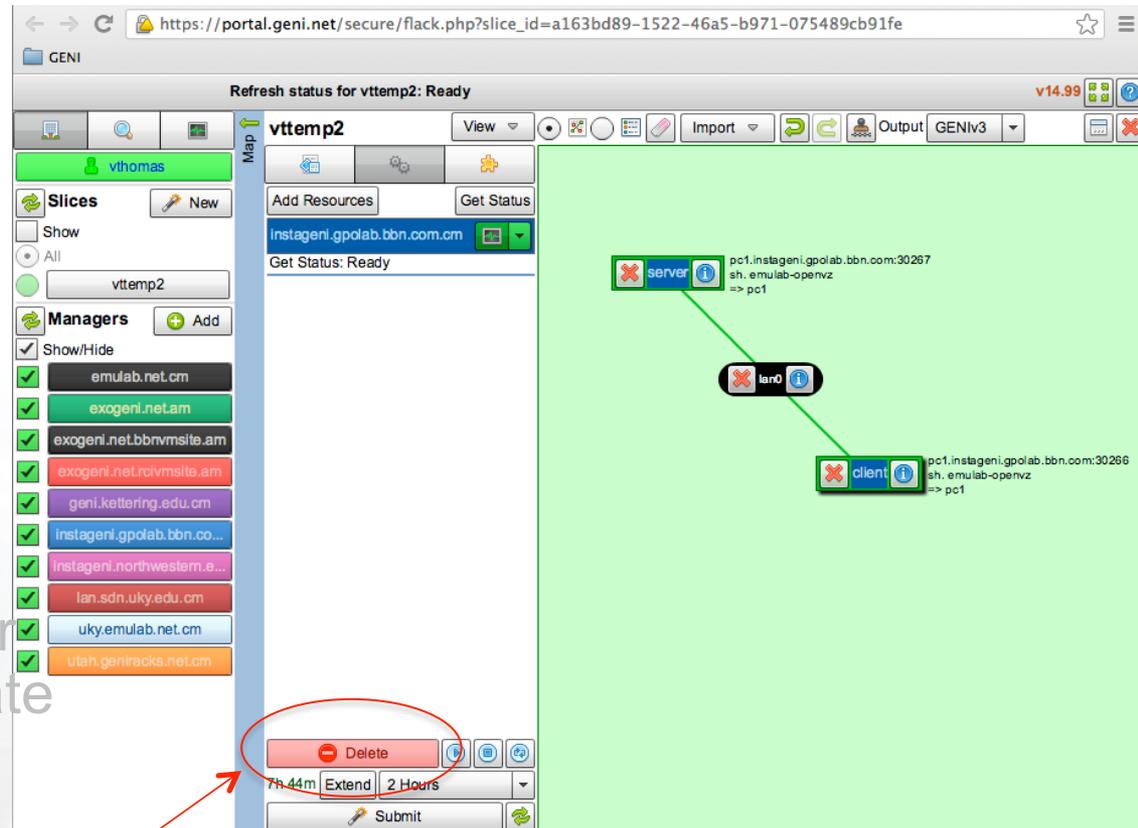
- Log into the GENI Portal
- Create a slice
- Launch the Flack experimenter tool
- Set up a simple experiment
 - Add **resources** to your slice from an **aggregate**
 - Use the resources in your slice



The screenshot shows the GENI portal interface for a slice named 'vttemp2'. The browser address bar shows the URL: https://portal.geni.net/secure/flack.php?slice_id=a163bd89-1522-46a5-b971-075489cb91fe. The interface includes a navigation bar with 'GENI' and 'Refresh status for vttemp2: Ready'. Below this, there are buttons for 'Add Resources' and 'Get Status'. A list of resources is shown, including 'instageni.gpolab.bbn.com.cm'. The main area displays a network diagram with nodes labeled 'server', 'lan0', and 'client'. The 'server' node is connected to 'lan0', which is connected to 'client'. The 'server' node has the IP address 'pc1.instageni.gpolab.bbn.com:30267' and the command 'sh. emulab-openvz => pc1'. The 'client' node has the IP address 'pc1.instageni.gpolab.bbn.com:30266' and the command 'sh. emulab-openvz => pc1'. The interface also shows a 'Delete' button, a timer for '7h 44m', and an 'Extend' button set to '2 Hours'. A 'Submit' button is at the bottom.

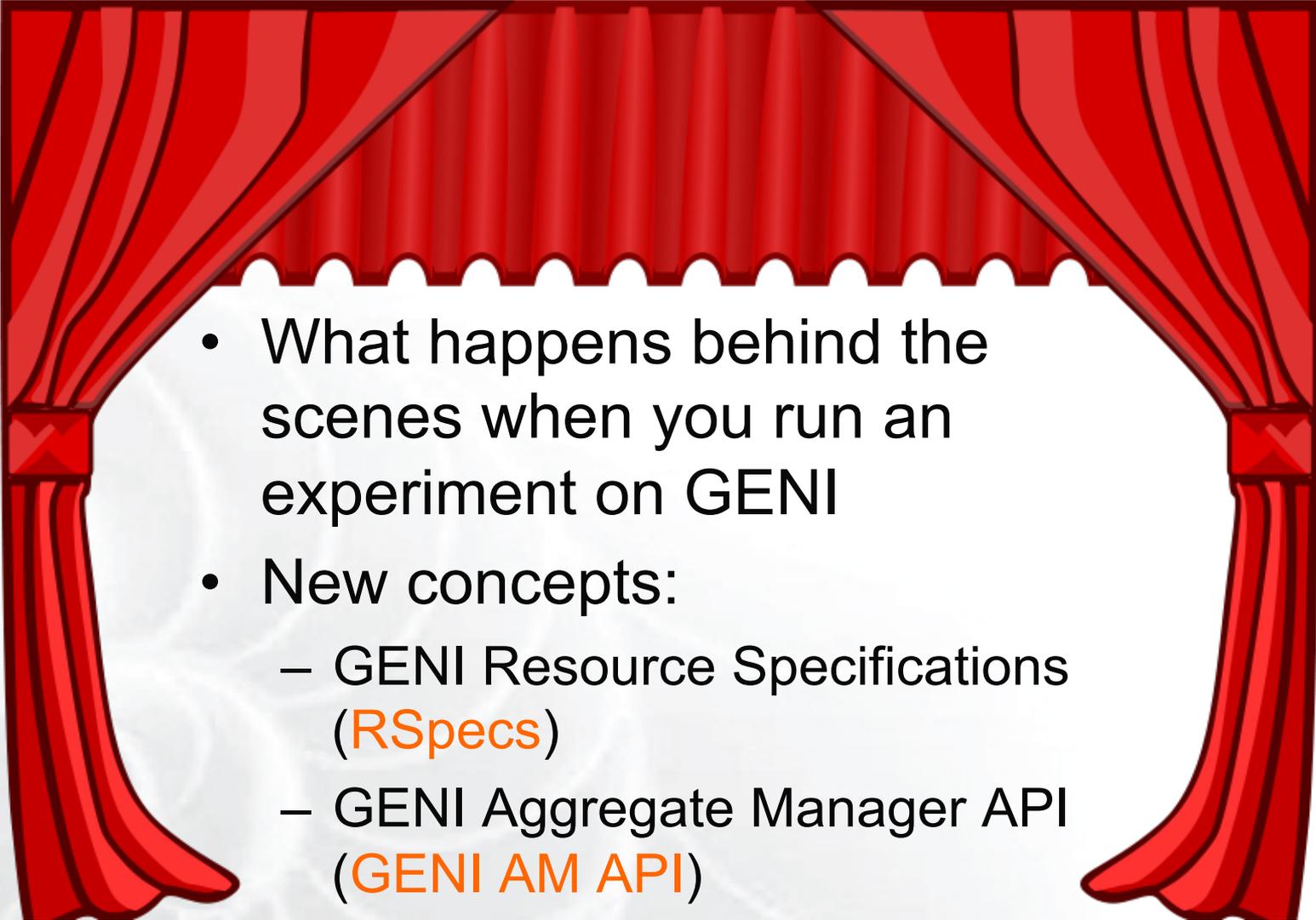
In Part 1 You Learned to...

- Log into the GENI Portal
- Create a slice
- Launch the Flack experimenter tool
- Set up a simple experiment
 - Add resources to your slice from an aggregate
 - Use the resources in your slice
- **Delete resources in your slice**



The screenshot shows the GENI portal interface for a slice named 'vttemp2'. The interface includes a navigation menu on the left with 'Slices' and 'Managers' sections. The main area displays the slice details, including a network diagram with nodes 'server', 'lan0', and 'client'. A red circle highlights the 'Delete' button in the bottom right corner of the slice management area, with a red arrow pointing from the text 'Delete resources in your slice' in the list above.

In Part 2 You will Learn...

- 
- What happens behind the scenes when you run an experiment on GENI
 - New concepts:
 - GENI Resource Specifications (**RSpecs**)
 - GENI Aggregate Manager API (**GENI AM API**)

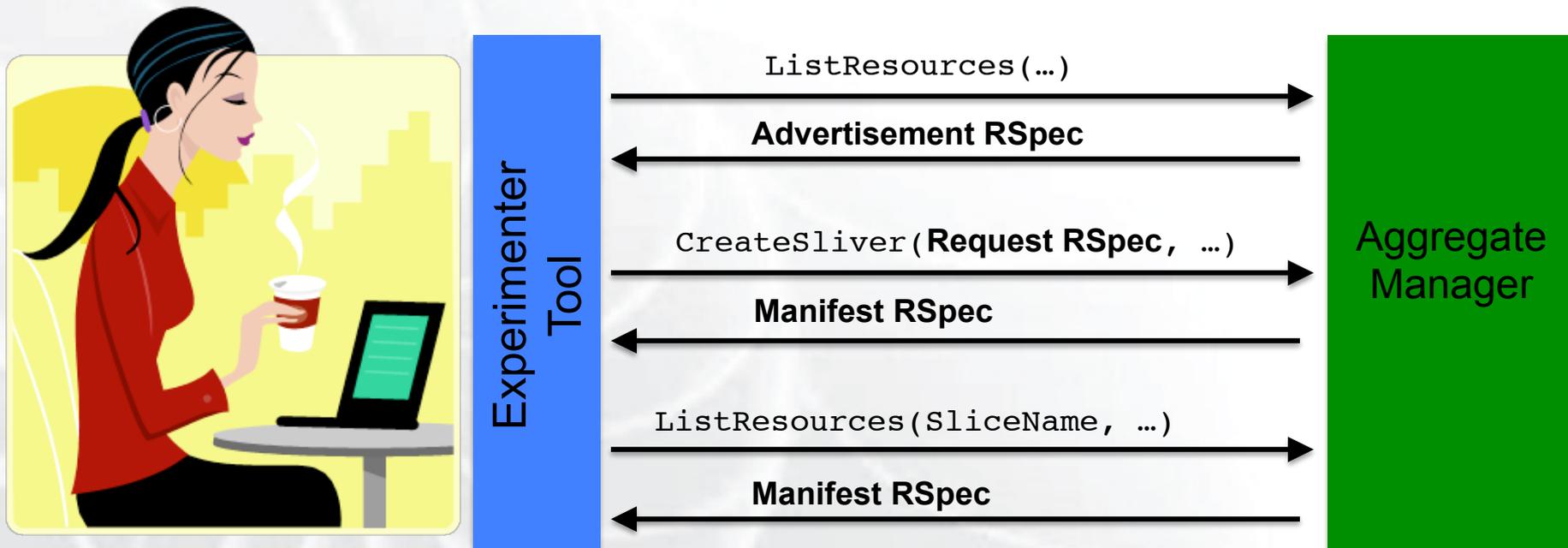
- RSpecs are XML documents that describe resources
 - VMs, links, etc.

RSpec for a virtual machine with one interface:

```
<?xml version="1.0" encoding="UTF-8"?>
<rspec type="request" xsi:schemaLocation="http://www.geni.net/
resources/rspec/3 ... xmlns="http://www.geni.net/resources/rspec/3">
  <node client_id="server" component_manager_id="urn:publicid:IDN
+instageni.gpolab.bbn.com+authority+cm">
    <sliver_type name="emulab-opensvz"/>
    <interface client_id="server:if0"> </interface>
  </node>
</rspec>
```

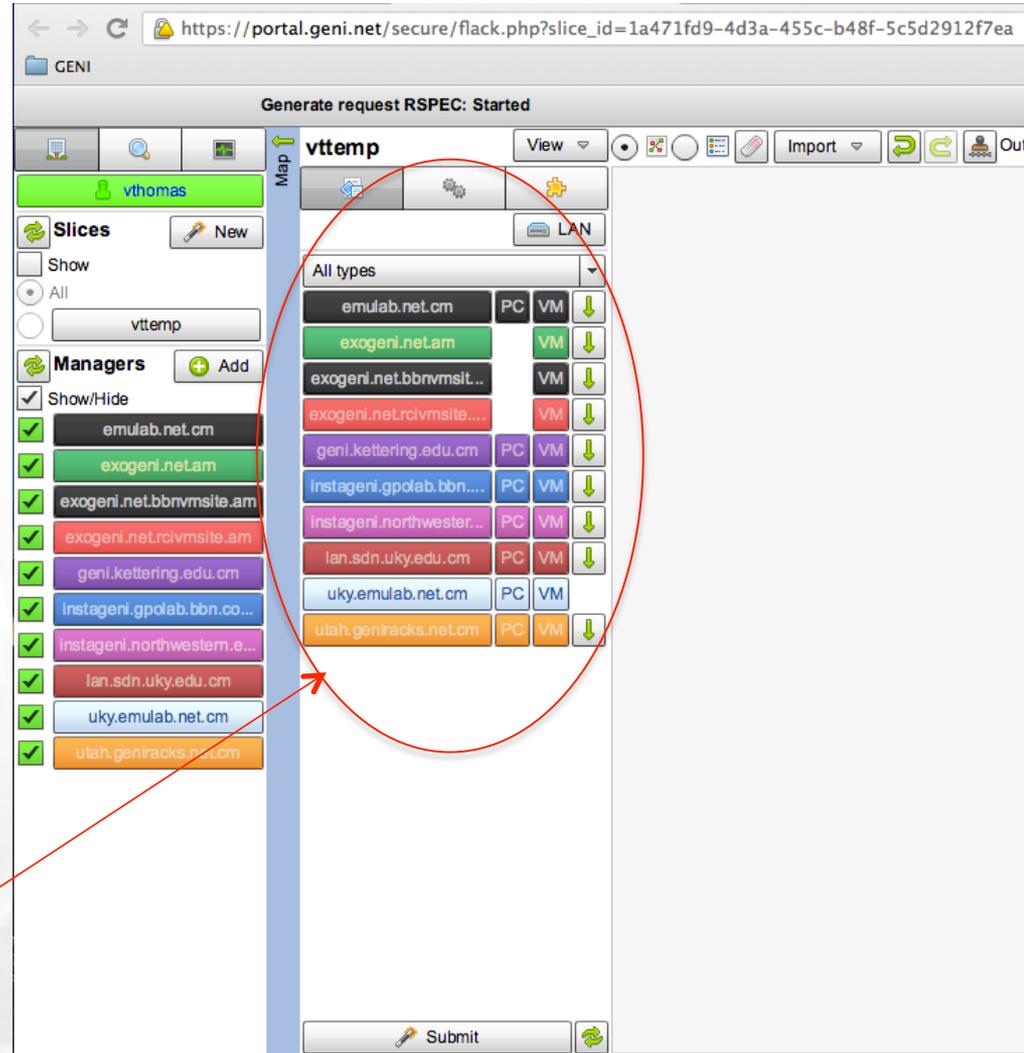
- RSpec documents are exchanged by experimenter tools (e.g. Flack) and aggregates
 - Aggregates use RSpecs to describe what they have – **Advertisement RSpecs**
 - Experimenters use RSpecs to describe the resources they want – **Request RSpecs**
 - Aggregates use RSpecs to describe the resources allocated to an experimenter – **Manifest RSpecs**

- Experimenter tools and aggregates talk to each other using the GENI Aggregate Manager API (**GENI AM API**)



Putting it all Together...

- Flack calls **ListResources** on all aggregates it knows about
- Aggregates send back **advertisement RSpecs**
- Flack uses information in the advertisements to populate its palette of resources

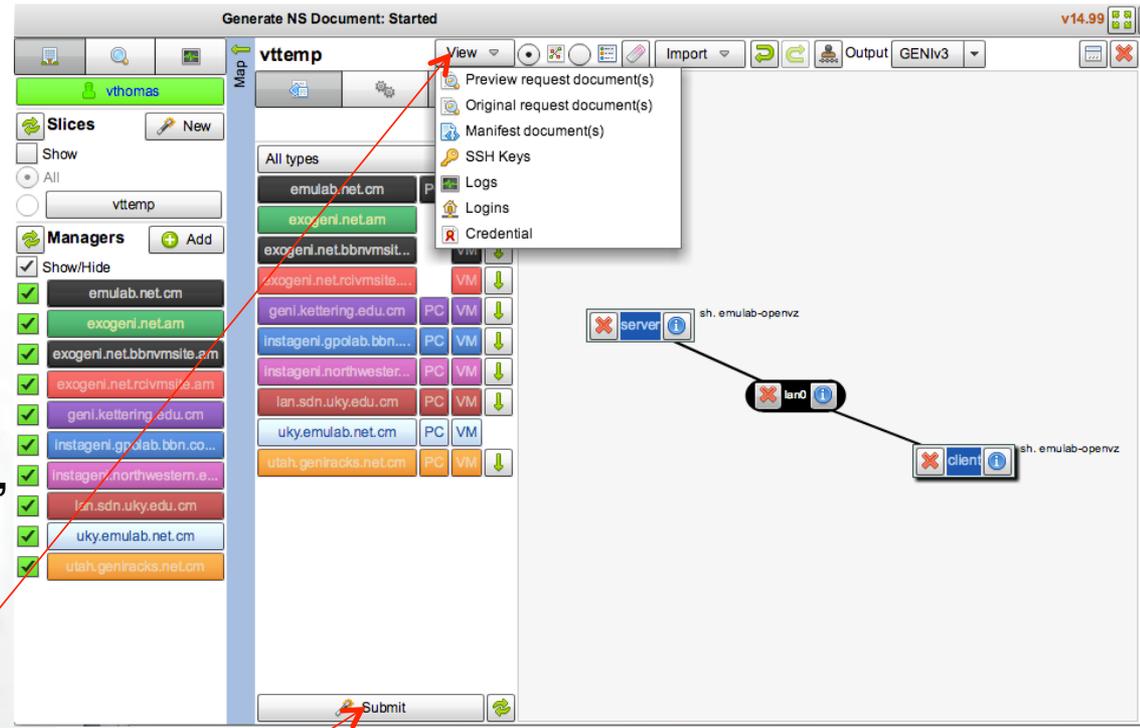


The screenshot shows the GENI portal interface for a slice named 'vtemp'. The interface includes a navigation bar, a search bar, and a 'Map' button. The main content area displays a list of resources available for the slice, categorized by type (PC and VM). A red circle highlights the resource palette, and a red arrow points from the text in the adjacent slide to this area.

Resource Name	Type	VM	Down Arrow
emulab.net.cm	PC	VM	↓
exogeni.net.am	VM	VM	↓
exogeni.net.bbrvmsit...	VM	VM	↓
exogeni.net.rcivmsite...	VM	VM	↓
geni.kettering.edu.cm	PC	VM	↓
instageni.gpolab.bbn...	PC	VM	↓
instageni.northwester...	PC	VM	↓
lan.sdn.uky.edu.cm	PC	VM	↓
uky.emulab.net.cm	PC	VM	↓
utah.geniracks.net.cm	PC	VM	↓

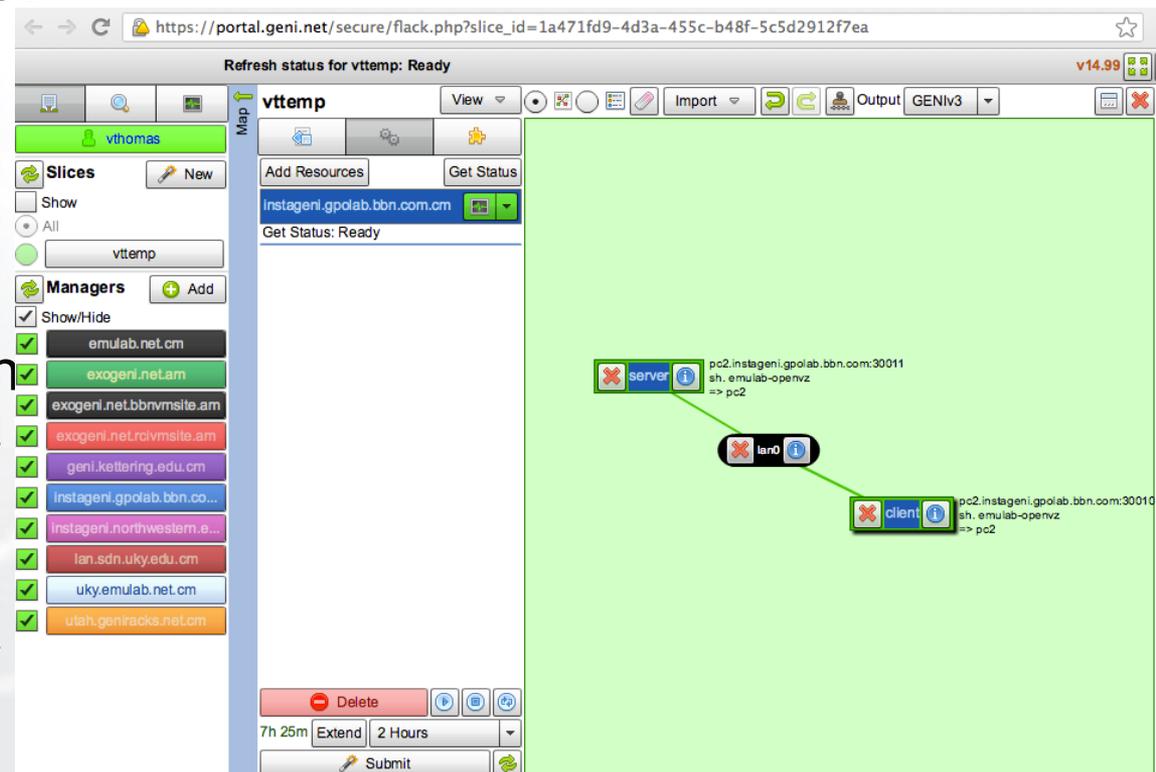
Putting it all Together...

- When you drag and drop resources on to the Flack canvas, it creates **request RSpecs** for these resources
 - To view the request Rspec click on “View” and select “Preview request documents”
- When you click “Submit”, Flack makes **createSliver** calls on the aggregates



Putting it all Together...

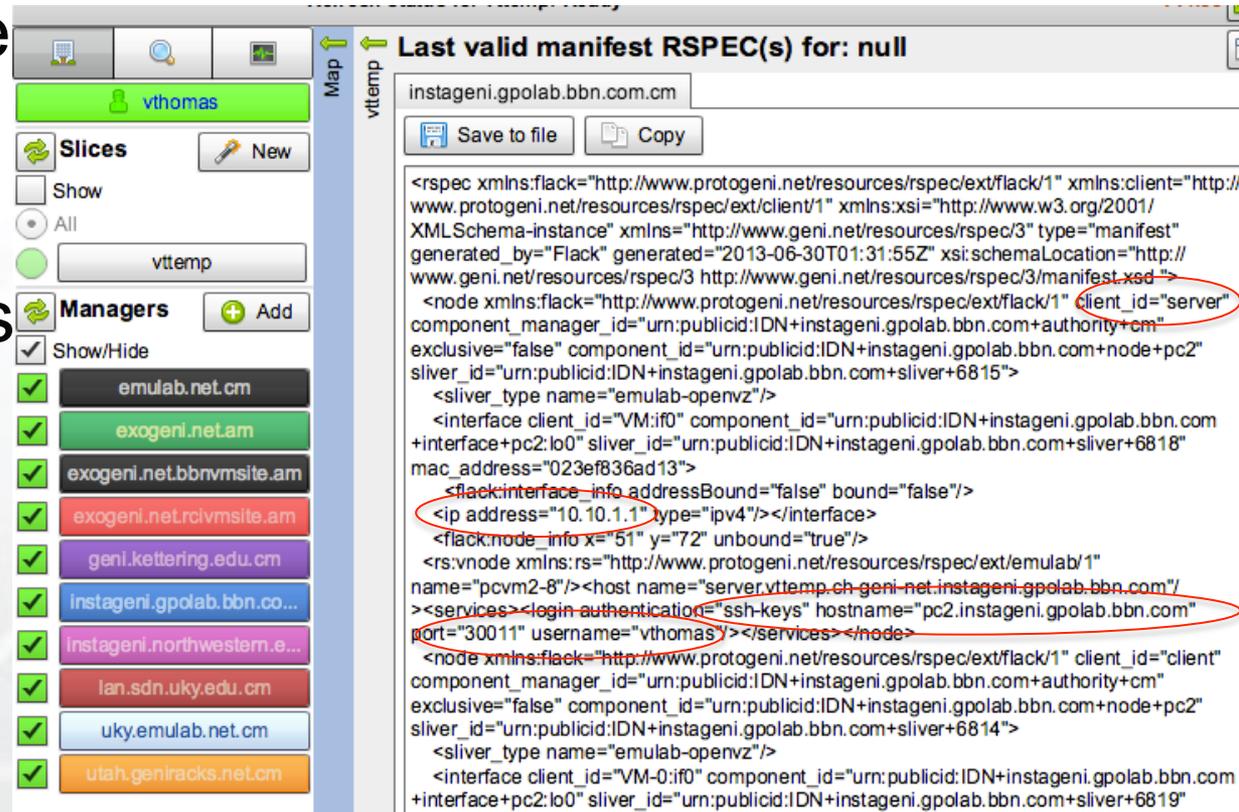
- Flack periodically calls `sliverStatus` on the aggregates to check on the status of your request
- When `sliverStatus` shows:
 - Resources have been allocated, Flack turns its canvas yellow
 - Resources are ready to use, Flack turns its canvas green



The screenshot shows the GENI portal interface for a virtual network slice named 'vtemp'. The browser address bar shows the URL: `https://portal.geni.net/secure/flack.php?slice_id=1a471fd9-4d3a-455c-b48f-5c5d2912f7ea`. The page title is 'Refresh status for vtemp: Ready'. The interface includes a sidebar with 'Slices' and 'Managers' sections. The 'Managers' list includes various network providers like emulab.net, exogeni.net, and others. The main area shows the slice 'vtemp' with a status of 'Ready'. A network diagram is displayed on the right, showing a 'server' node connected to a 'client' node via a 'lan0' interface. The server node is labeled 'pc2.instageni.gpolab.bbn.com:30011 sh. emulab-openvz => pc2'. The client node is labeled 'pc2.instageni.gpolab.bbn.com:30012 sh. emulab-openvz => pc2'. The interface also includes a 'Delete' button, a timer set to '7h 25m', and a 'Submit' button.

Putting it all Together...

- listResources with a slice name returns a **manifest RSpec**
- Manifest includes names and ports used to ssh into VMs
 - Flack uses this information to help you log into your resources



Last valid manifest RSPEC(s) for: null

instageni.gpolab.bbn.com.cm

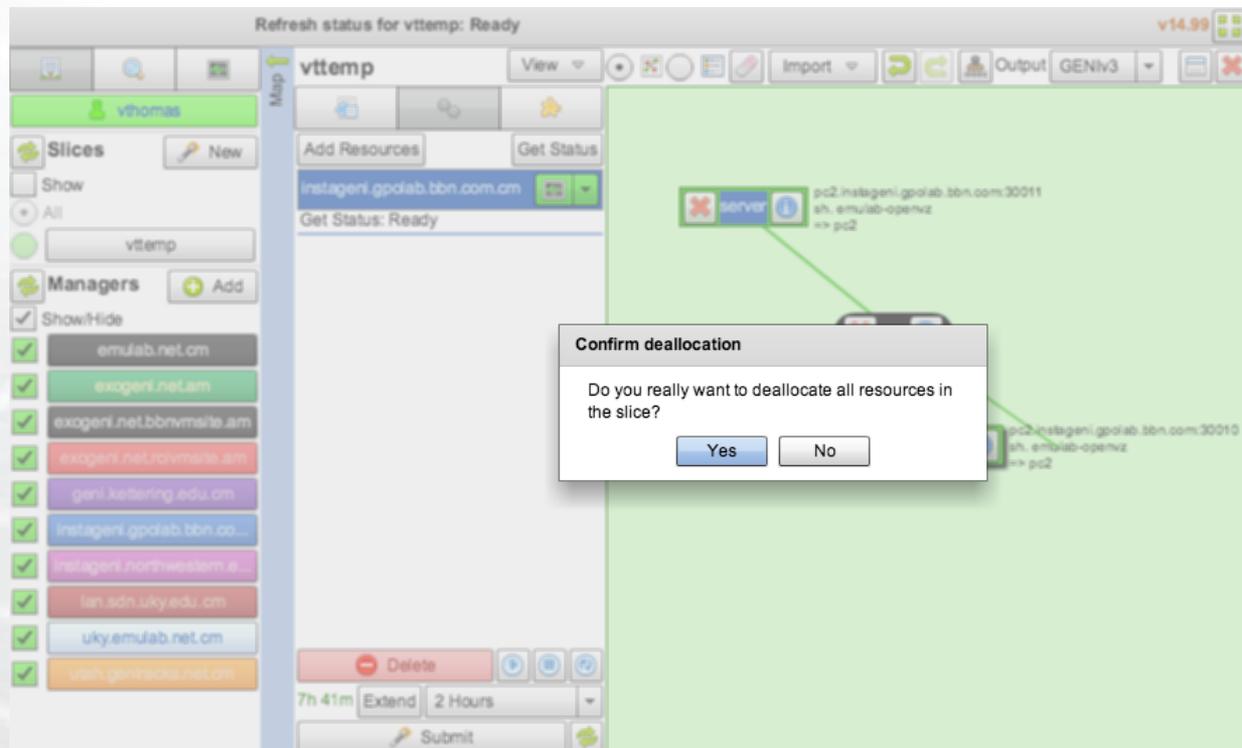
Save to file Copy

```

<rspec xmlns:flack="http://www.protogeni.net/resources/rspec/ext/flack/1" xmlns:client="http://www.protogeni.net/resources/rspec/ext/client/1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.geni.net/resources/rspec/3" type="manifest" generated_by="Flack" generated="2013-06-30T01:31:55Z" xsi:schemaLocation="http://www.geni.net/resources/rspec/3 http://www.geni.net/resources/rspec/3/manifest.xsd">
  <node xmlns:flack="http://www.protogeni.net/resources/rspec/ext/flack/1" client_id="server" component_manager_id="urn:publicid:IDN+instageni.gpolab.bbn.com+authority+cm" exclusive="false" component_id="urn:publicid:IDN+instageni.gpolab.bbn.com+node+pc2" sliver_id="urn:publicid:IDN+instageni.gpolab.bbn.com+sliver+6815">
    <sliver_type name="emulab-openvz"/>
    <interface client_id="VM:i0" component_id="urn:publicid:IDN+instageni.gpolab.bbn.com+interface+pc2:lo0" sliver_id="urn:publicid:IDN+instageni.gpolab.bbn.com+sliver+6818" mac_address="023ef836ad13">
      <flack:interface_info addressBound="false" bound="false"/>
      <ip address="10.10.1.1" type="ipv4"/></interface>
      <flack:node_info x="51" y="72" unbound="true"/>
    <rs:vnode xmlns:rs="http://www.protogeni.net/resources/rspec/ext/emulab/1" name="pcvm2-8"/><host name="server.vttemp.ch.geni.net.instageni.gpolab.bbn.com"/><services><login_authentication ssh-keys hostname="pc2.instageni.gpolab.bbn.com" port="30011" username="vthomas"/></services></node>
    <node xmlns:flack="http://www.protogeni.net/resources/rspec/ext/flack/1" client_id="client" component_manager_id="urn:publicid:IDN+instageni.gpolab.bbn.com+authority+cm" exclusive="false" component_id="urn:publicid:IDN+instageni.gpolab.bbn.com+node+pc2" sliver_id="urn:publicid:IDN+instageni.gpolab.bbn.com+sliver+6814">
      <sliver_type name="emulab-openvz"/>
      <interface client_id="VM-0:i0" component_id="urn:publicid:IDN+instageni.gpolab.bbn.com+interface+pc2:lo0" sliver_id="urn:publicid:IDN+instageni.gpolab.bbn.com+sliver+6819"
  
```

Putting it all Together...

- When you deleted your resources, Flack called **deleteSliver** on the aggregates



- Reinforce understanding of the new concepts by:
 - Viewing and editing RSpec documents
 - Making the AM API calls ourselves using the **Omni experimenter tool**

- Repeat the experiment from Part 1 by loading an RSpec into Flack
 - Instead of drawing the topology ourselves (saves time)
- Edit the RSpec using Flack but don't "submit"
- Save the request RSpec generated by Flack into a file
- Use Omni to make GENI AM API calls to send the request RSpec, check status of resources, etc.

- `listresources`: Get an advertisement rspec listing the resources at an aggregate
- `createsliver`: Request resources from an aggregate
- `sliverstatus`: Get status of resources allocated to a slice at an aggregate
- `deletesliver`: Delete resources allocated to a slice by an aggregate

For a complete list of GENI AM API calls see:
http://groups.geni.net/geni/wiki/GAPI_AM_API

- A command line experimenter tool
- Useful for making AM API calls on aggregates
- Written in and scriptable from Python
- **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.py -a aggregatename listresources`
- `omni.py -a aggregatename createsliver slicename requestRSpec`
- `omni.py -a aggregatename sliverstatus slicename`
- `omni.py -a aggregatename listresources slicename`
- `omni.py -a aggregatename deletesliver slicename`

- A useful utility (distributed with Omni):
`readyToLogin.py`
 - Gives you the ssh commands you need to log into your nodes
 - `readyToLogin.py` parses the output of `sliverStatus` to determine the hostname, portname and username for the ssh commands

- Omni reads a configuration file `omni_config` to:
 - Get usernames for accounts to be created on compute resources
 - Find locations of ssl certs and ssh key files
 - ssl certs are used to secure communication between Omni and the aggregates
 - ssh key pairs are used log into compute resources
 - Find standard nicknames for aggregates
 - E.g. you can refer to the InstaGENI rack at BBN as `ig-bbn` instead of `https://boss.instageni.gpolab.bbn.com:12369/protogeni/xmlrpc/am/2.0`

Creating an Omni Config File

1. Download the GENI bundle from the GENI Portal
2. Run the script `omni-configure.py`
 - Distributed with Omni
 - Already installed on your virtual machine

Download omni bundle

Instructions:

1. Choose a project below as your default omni project.
2. Click "Download omni bundle"
3. Run "`omni-configure.py -f portal <location of bundle>`"

Choose project as omni default:

```
geni@NSDI13-Tutorials:~$>
```

```
omni-configure.py -f portal
```

Watch Instructor do the Exercise

- By the end of this tutorial you should:
 - Feel comfortable running simple experiments on GENI
 - Have a basic understanding of how GENI works
- Later tutorials may skip some of these basic steps to focus on new material
 - You may be given an RSpec to use rather than have you create one
 - You may use slices that have already been created and resources added to them

For a description of the GENI concepts you have learned, see:
<http://groups.geni.net/geni/wiki/GENIConcepts>

- You should have a printout with detailed instructions
- Online instructions:

```
http://groups.geni.net/geni/wiki/  
GENIExperimenter/Tutorials/PortalOmniExample
```