Examples of REST Modelling
(OpenStack - Compute)

Note: Slides prepared by Yassine Jebbar, Teaching Assistant

Roch Glitho, PhD
Associate Professor and Canada Research Chair
My URL - http://users.encs.concordia.ca/~glitho/
OpenStack Compute API

- REST Modelling procedure
- OpenStack Compute key concepts
- Applying the procedure
The procedure – First Part

- Figure out the data set
- Split the data set into resources
The procedure – Second Part

For each resource:

- Name the resources with URIs
- Identify the subset of the uniform interface that is exposed by the resource
- Design the representation(s) as received (in a request) from and sent (in a reply) to the client
- Consider the typical course of events by exploring and defining how the new service behaves and what happens during a successful execution
OpenStack Compute (REST-based) Key Concepts

- OpenStack Compute is a compute service that provides server capacity in the cloud.

- Compute Servers come in different flavors (virtual hardware configuration) of memory, cores, disk space, and CPU, and can be provisioned in minutes.

- Interactions with Compute Servers can happen programmatically with the OpenStack Compute API.
OpenStack Compute Key Concepts

- **Server**: A virtual machine (VM) instance, physical machine or a container in the compute system.

- **Flavor**: Virtual hardware configuration for the requested server. Each flavor has a unique combination of disk space, memory capacity and priority for CPU time.

- **Image**: A collection of files used to create or rebuild a server. Operators provide a number of pre-built OS images by default.
OpenStack Key Concepts

- **Server Management:** Enable all users to perform an action on a server.
  
  Example: ➢ Create/Delete/Resize/Reboot Server
  ➢ Show Server(s) Details

- **Flavor Management:** Show and manage server flavors.
  
  Example: ➢ Create/Delete/Update Flavor
  ➢ Show Flavor(s) Details

- **Image Management:** Show details and manage images.
  
  Example: ➢ List Images
  ➢ Show Image Details
  ➢ Delete Image
Applying the procedure – Data Set

• Servers

• Flavors

• Images
Applying the procedure – Split Data Set into Resources

• Each server is a resource
• Each flavor is a resource
• Each image is a resource
• One special resource that lists servers
• One special resource that lists flavors
• One special resource that lists images
### Applying the procedure – Name Resources with URIs

<table>
<thead>
<tr>
<th><strong>Server URI</strong></th>
<th><strong>Flavor URI</strong></th>
<th><strong>Image URI</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>/servers</td>
<td>GET</td>
<td>/images/detail</td>
</tr>
<tr>
<td>/servers/list</td>
<td>GET</td>
<td>List Images With Details</td>
</tr>
<tr>
<td>/servers/</td>
<td>POST</td>
<td>/images/{image_id}</td>
</tr>
<tr>
<td>/servers/</td>
<td>GET</td>
<td>Show Image Details</td>
</tr>
<tr>
<td>/servers/</td>
<td>PUT</td>
<td>/images/{image_id}</td>
</tr>
<tr>
<td>/servers/</td>
<td>DELETE</td>
<td>Delete Image</td>
</tr>
<tr>
<td>/servers/</td>
<td>POST</td>
<td>/images/{image_id}</td>
</tr>
<tr>
<td>/servers/</td>
<td>POST</td>
<td>Update Flavor Description</td>
</tr>
<tr>
<td>/servers/</td>
<td>DELETE</td>
<td>Delete Flavor</td>
</tr>
</tbody>
</table>

#### Examples:

- **GET**: List Servers
- **POST**: Create Server
- **GET**: Show Server Details
- **PUT**: Update Server
- **DELETE**: Delete Server
- **POST**: Reboot Server (reboot Action)
- **DELETE**: Delete Flavor
Example: Listing and Creating Server

1: GET:/servers

2: 200 OK

{ "servers": [ { "id": "1", "links": [ { "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/1", "rel": "self" }, ....].

3: POST:/servers

2: 200 OK

{ "server": { "name": "auto-allocate-network", "imageRef": "70a599e0-31e7-49b7-b260-868f441e862b", "flavorRef": "http://openstack.example.com/flavors/1", "networks": "auto" } }
Example: Resizing Server

Alice
Openstack User

Openstack Compute
REST Server

3: POST:/servers/1/action
{"resize": { "flavorRef": "2", "OS-DCF:diskConfig": "AUTO" }}

2: 200 OK

Resize Server

If successful, this method does not return content in the response body.
References

https://docs.openstack.org/api-guide/compute/general_info.html

https://docs.openstack.org/api-ref/compute/?expanded=