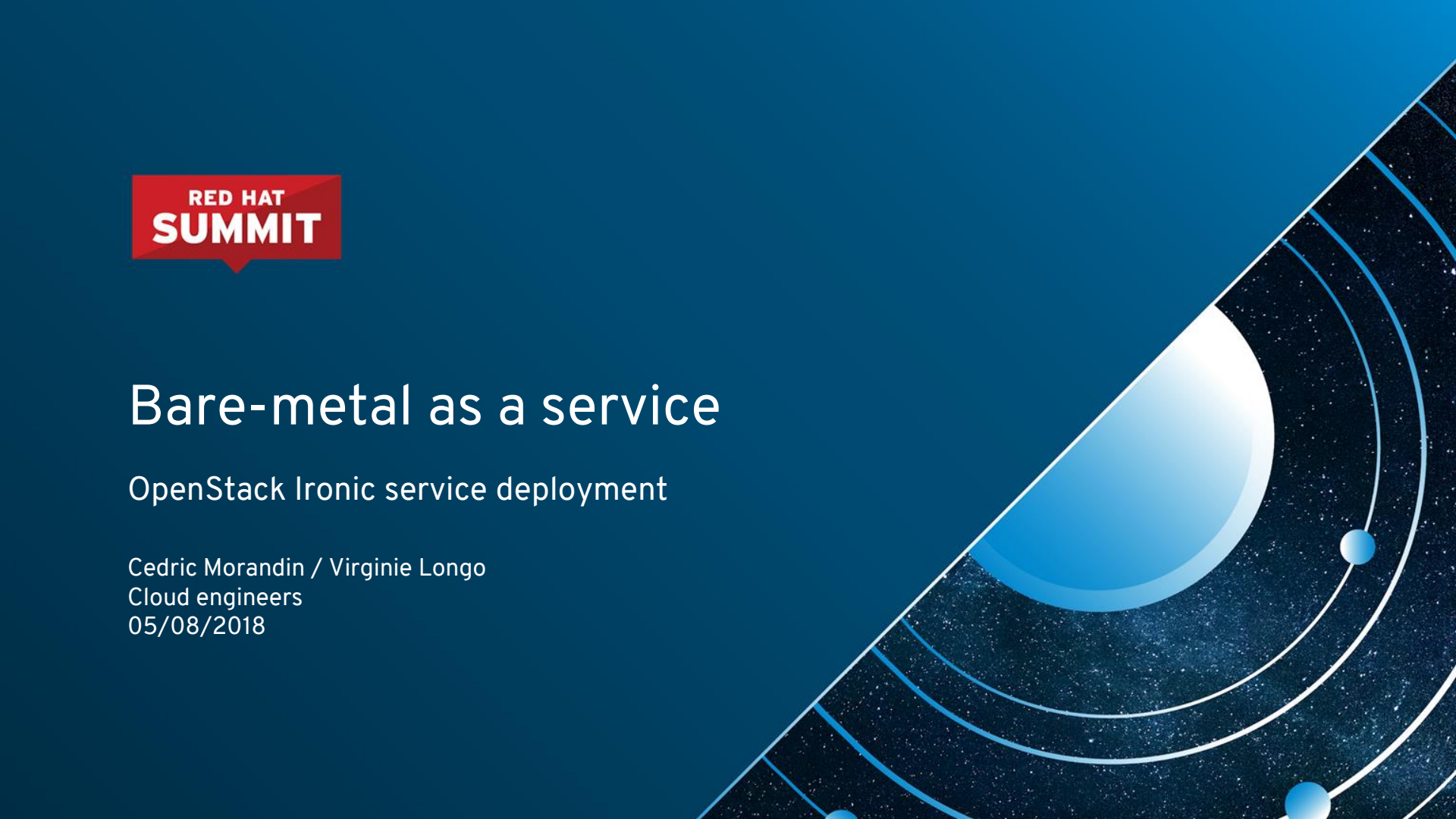


Bare-metal as a service

OpenStack Ironic service deployment

Cedric Morandin / Virginie Longo
Cloud engineers
05/08/2018

Our environment

Amadeus in few words

630+ million total bookings processed in 2017
using the Amadeus distribution platform

1.6 billion passengers boarded in 2017
with Amadeus and Navitaire solutions

1 of the world's top 15 software companies
Forbes 2017 global rankings

Why OpenStack

For the R&D environment

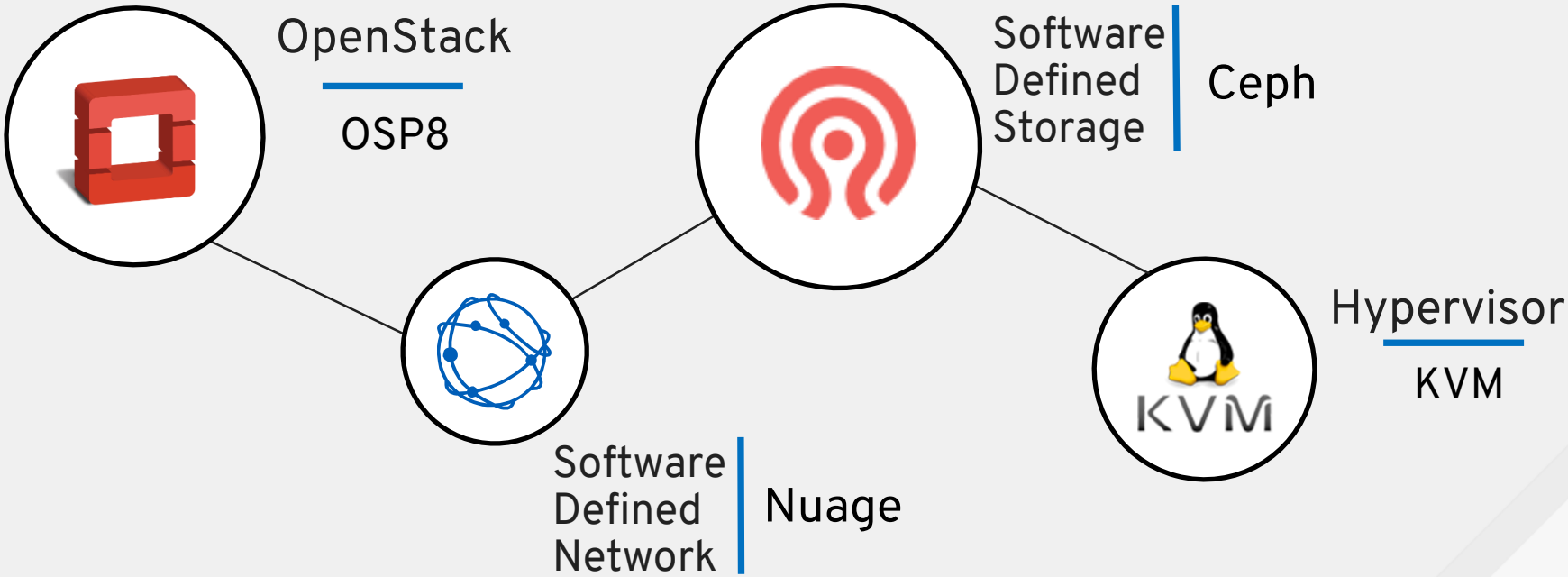
Developers want a platform allowing them to quickly prototype

And a safe place where to test new technologies

They want to easily spawn development and test platforms

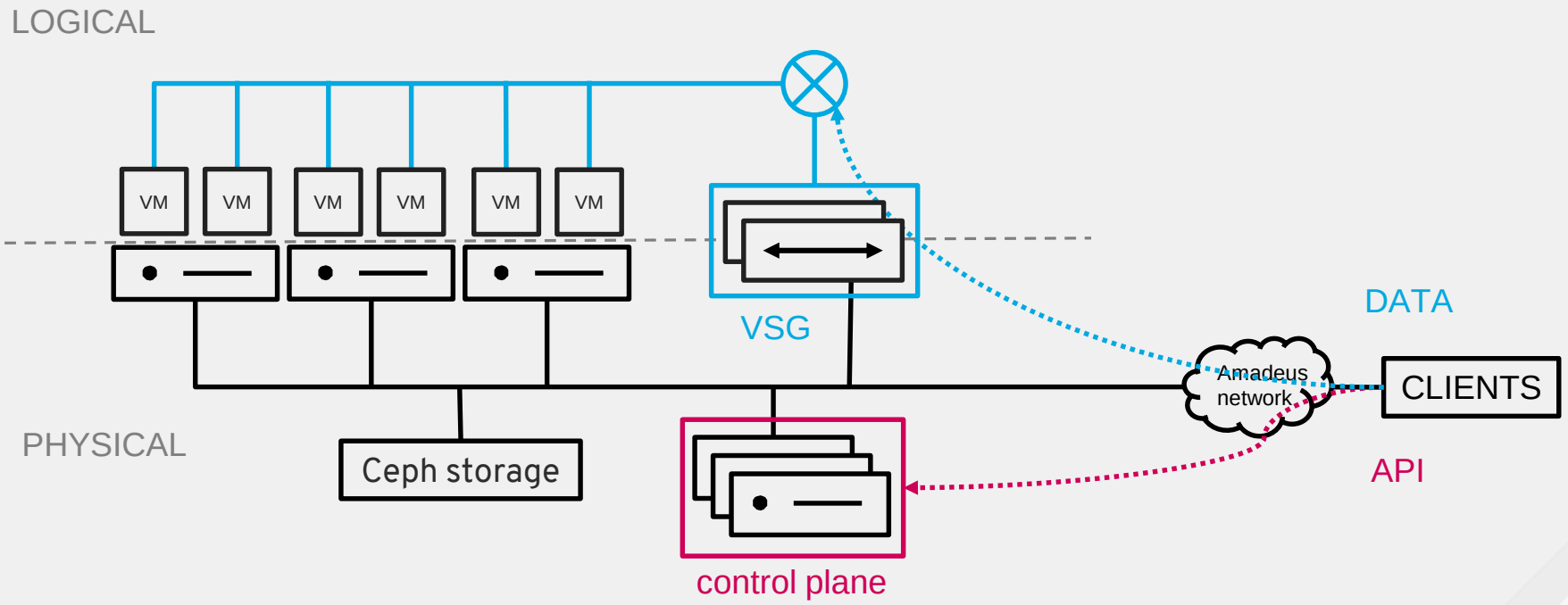
And automate infrastructure deployment

R&D cloud platform anatomy

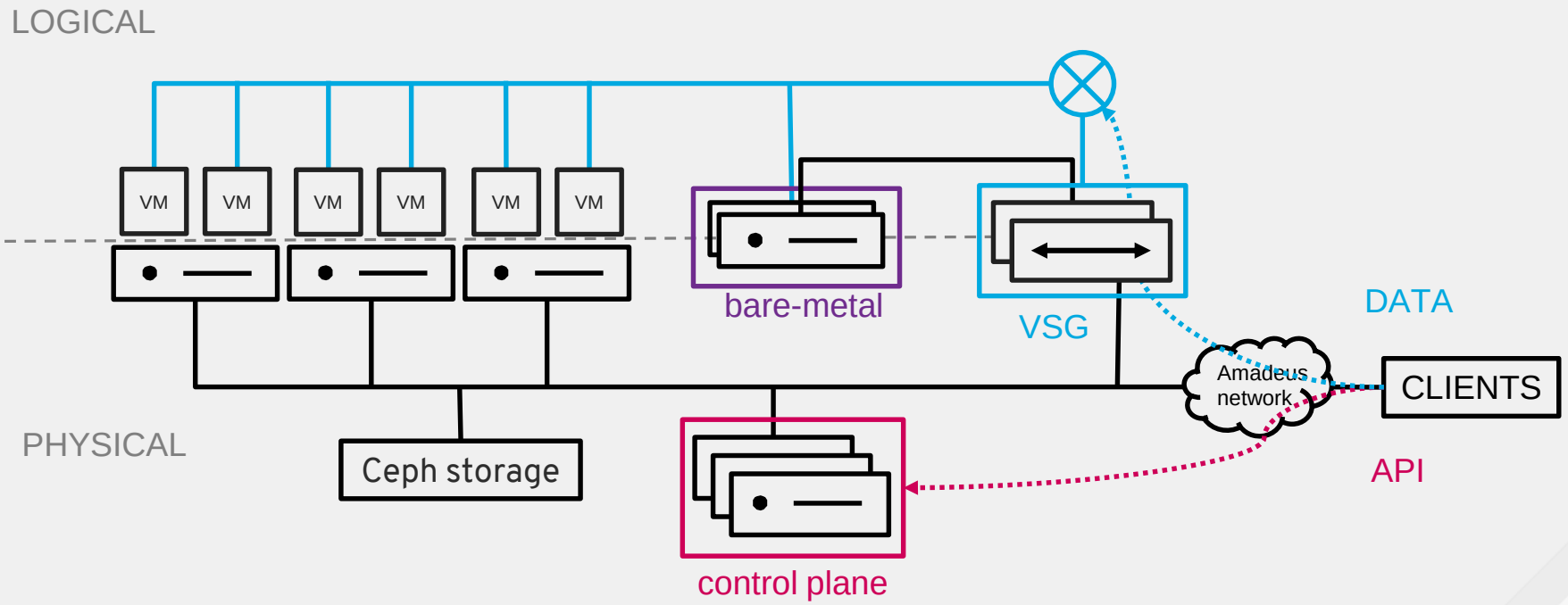


Under the hood

The platform



What do we want?



Introducing Ironic

*“**Ironic** aims to provision bare metal machines instead of virtual machines, forked from the Nova bare-metal driver. By default, it will use PXE and IPMI in concert to provision and turn on/off machines”*

Why using Ironic

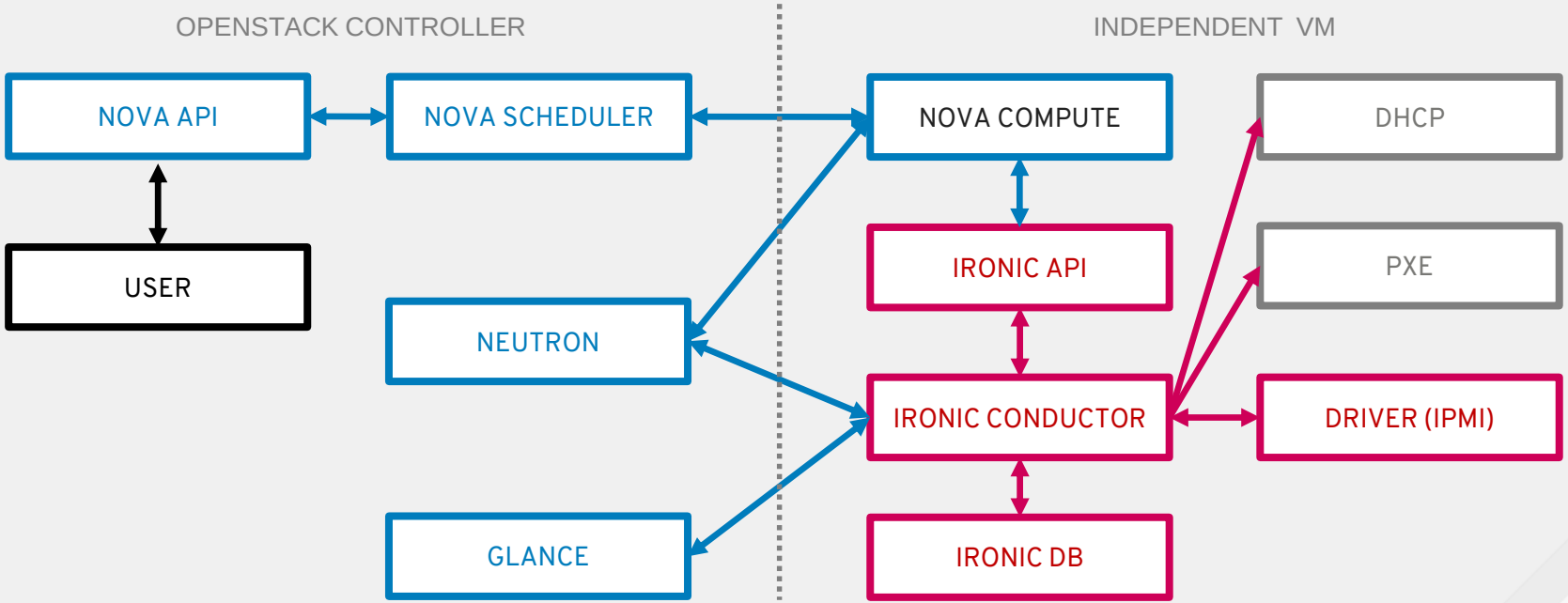
and what are the requirements?

Performance testing of applications
Hosting applications with **built-in redundancy**

Multi-tenancy for bare-metal
Standard **Nova API** for manipulating bare-metal servers
Images stored in **Glance**

How does it work?

https://docs.openstack.org/ironic/pike/user/ - section 1.2



Implementation and usage

Let’s deploy Ironic!

Liberty version

#1 Configure Nuage SDN

Let’s deploy Ironic!

Liberty version

#1 Configure Nuage SDN

#2 Deploy a VM with nova compute and all Ironic services

Let’s deploy Ironic!

Liberty version

#1 Configure Nuage SDN

#2 Deploy a VM with nova compute and all Ironic services

#3 Reconfigure OpenStack controllers

Let’s deploy Ironic!

Liberty version

#1 Configure Nuage SDN

#2 Deploy a VM with nova compute and all Ironic services

#3 Reconfigure OpenStack controllers

#4 Create Keystone service catalog entries for OpenStack service

Let’s configure Ironic!

#5 Register and configure the bare-metal nodes

Let’s configure Ironic!

#5 Register and configure the bare-metal nodes

```
$ openstack baremetal create nceospp01bms663.yaml
$ openstack baremetal node set --property capabilities=boot_option:local\
--driver-info deploy_kernel=de99de5d-341d-4c8a-bc72-2cd04416f77f\
--driver-info deploy_ramdisk=86a2aa1f-3b04-4f3d-9330-dcac08a283e8 nceospp01bms663

$ openstack baremetal port set dd7428df-3f8b-4867-a633-d590ebade831 --extra
gateway_name=10.255.110.248 --extra gateway_port=1/1/5 --extra gateway_vlan=0
```

Let’s configure Ironic!

#5 Register and configure the bare-metal nodes

```
$ openstack baremetal create nceospp01bms663.yaml
$ openstack baremetal node set --property capabilities=boot_option:local\
--driver-info deploy_kernel=de99de5d-341d-4c8a-bc72-2cd04416f77f\
--driver-info deploy_ramdisk=86a2aa1f-3b04-4f3d-9330-dcac08a283e8 nceospp01bms663

$ openstack baremetal port set dd7428df-3f8b-4867-a633-d590ebade831 --extra
gateway_name=10.255.110.248 --extra gateway_port=1/1/5 --extra gateway_vlan=0

$ ironic node-list --fields name power_state provision_state instance_uuid
+-----------------+-------------+--------------------+--------------------------------------+
| Name            | Power State | Provisioning State | Instance UUID                        |
+-----------------+-------------+--------------------+--------------------------------------+
| nceospp01bms651 | power on    | active             | 79148cc7-f375-4b98-a5f0-fe96c587b406 |
| nceospp01bms663 | power off   | available          | None                                 |
| nceospp01bms664 | power on    | deploying          | a4dc68ab-9bbe-4ff7-b5a5-d18df81f7400 |
+-----------------+-------------+--------------------+--------------------------------------+
```

Let’s configure Ironic!

#6 Create the appropriate flavor

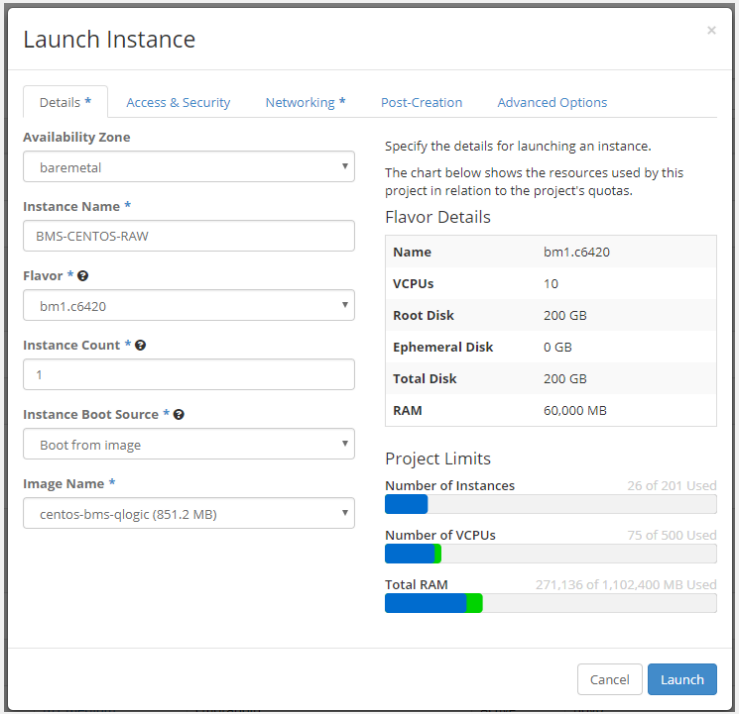
Let’s configure Ironic!

#6 Create the appropriate flavor

```
(openstack) flavor show bm1.c6420
+----------------------------+-------------------------------------------------------+
| Field                      | Value                                                 |
+----------------------------+-------------------------------------------------------+
| disk                       | 200                                                   |
| id                         | 7e961875-3cae-4e4d-ac3a-ca8e369f010b                  |
| name                       | bm1.c6420                                             |
| os-flavor-access:is_public | True                                                  |
| properties                 | aggregate_instance_extra_specs:baremetal='true',      |
|                            | capabilities:boot_option='local'                      |
| ram                        | 60000                                                 |
| rxtx_factor                | 1.0                                                   |
| vcpus                      | 10                                                    |
+----------------------------+-------------------------------------------------------+
```

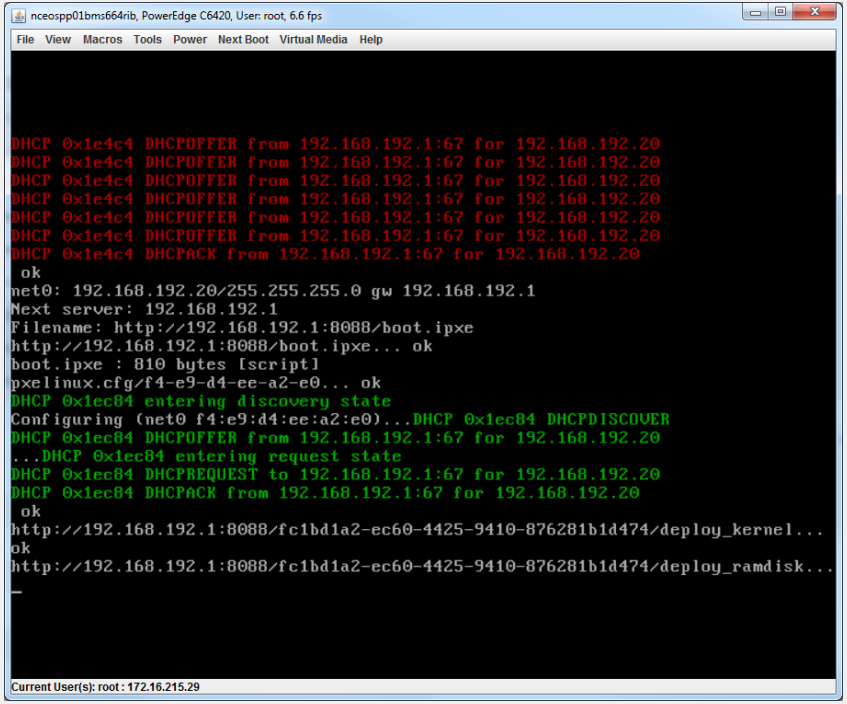
Pixie boots rockin’

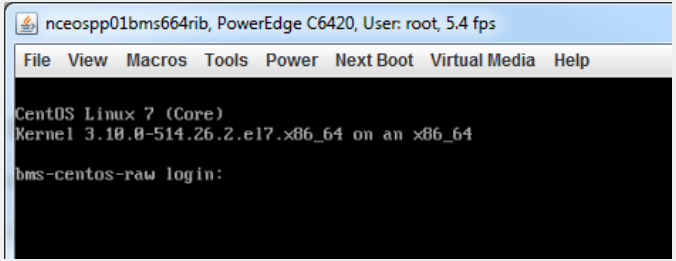
The first bare-metal server




```
$ nova boot bms-centos-raw --image centos-bms-qlogic\
--flavor bm1.c6420 --config-drive true\
--nic net-id=b08e83b8-f9b0-4922-9b63-149dc7fdc79c\
--key-name mykey
```

The first bare-metal server






```
[vlongo@bastion01 ~]$ ssh centos@172.16.156.12
[centos@jump ~]$ ssh centos@192.168.1.29 -i mykey.pem
[centos@bms-centos-raw ~]$
```

It works!

Multi-tenancy achieved thanks to VSGs

Easily usable by people as because **integrated with Nova**

Images are **stored in Glance** and managed as **normal images**

But it could be even better!

Connectivity	No NAT operation possible at this time
Quotas	No quota to limit the number of bare-metal servers per tenant
Metadata	Available only through config drive
Images	Because of the config drive being used for metadata specific images have to be built for Ironic

The journey continues...

What's next?

Deployment of OSP12 (**Pike**) **solving** most of the problems

Onboarding of new customers and **adding more nodes**

Design a solution **allowing NAT connectivity**

THANK YOU


plus.google.com/+RedHat


facebook.com/redhatinc


linkedin.com/company/red-hat


twitter.com/RedHat


youtube.com/user/RedHatVideos