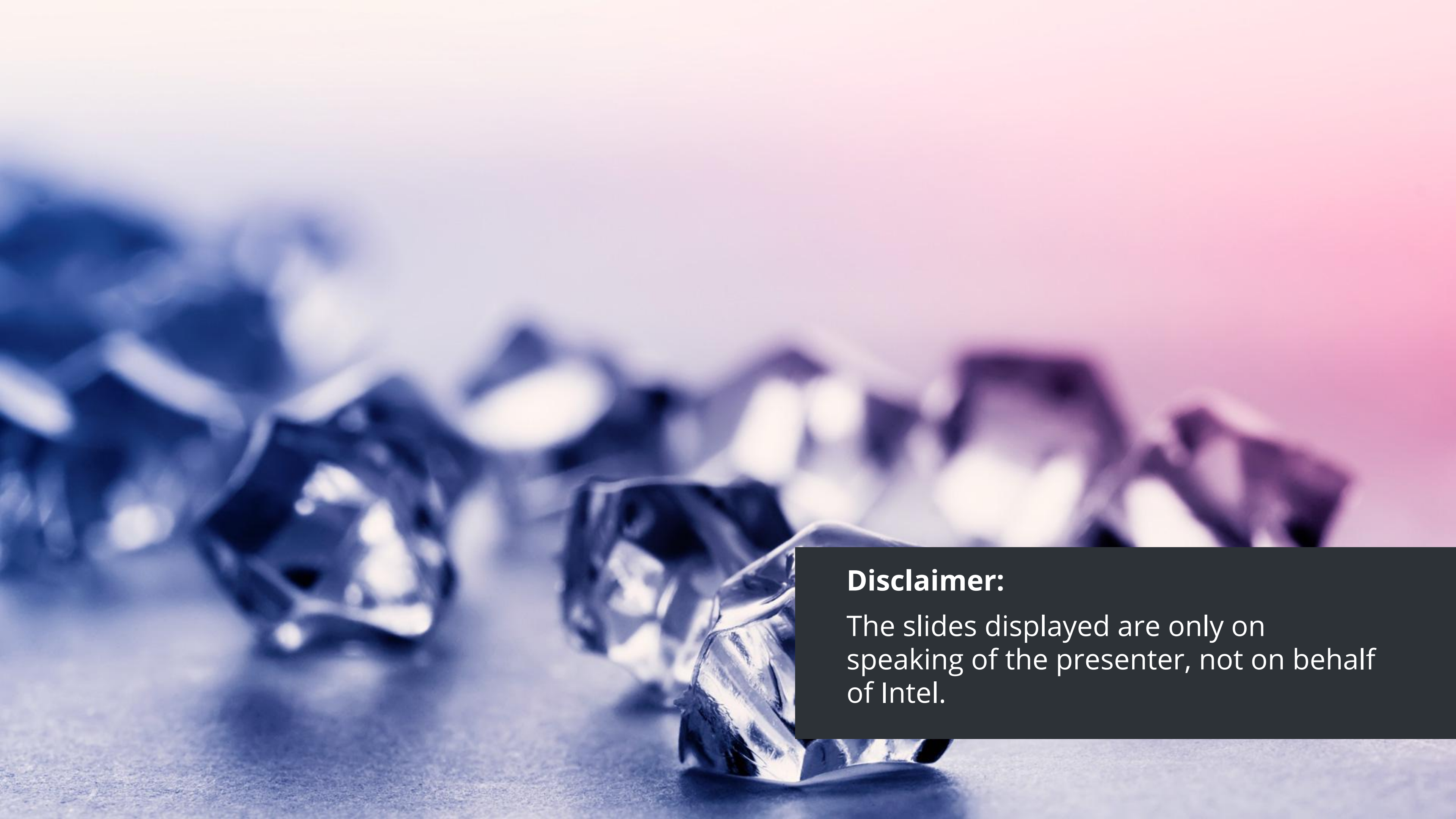
Stateless OS:
From an Openstack perspective

Marcos Simental

Clear Linux* Project
for Intel® Architecture



Disclaimer:

The slides displayed are only on speaking of the presenter, not on behalf of Intel.

Agenda

- What is a Stateless OS
- Stateless + Bundles = Consistency
- Stateless Openstack: How we made it
- Demo

What is a Stateless OS

A brief look at history

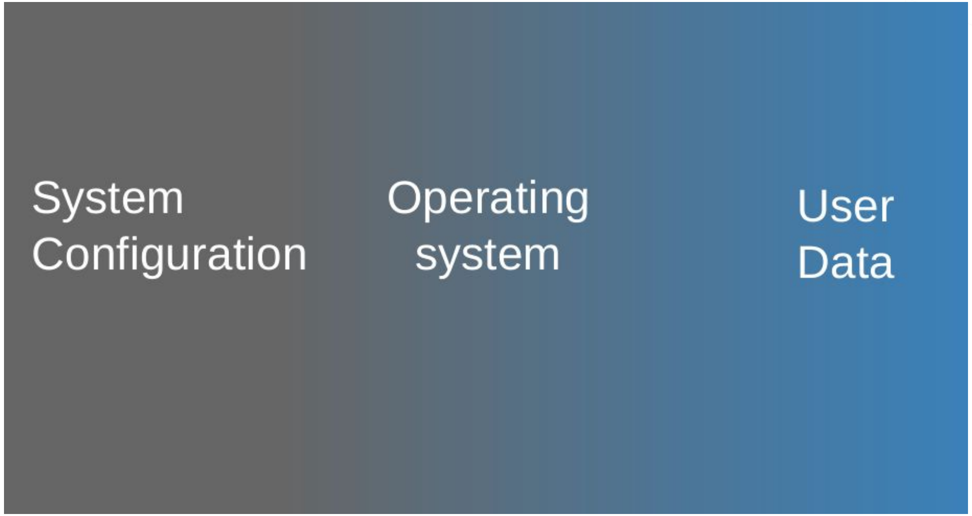
Stateless Operating System concepts have been explored for over a decade now.

- First prototypes in ~2004 by Red Hat*
- Moving from stateful to stateless approach
- Stateless was not a replacement for stateful systems

**Other names and brands may be claimed as the property of others*

Stateful OS

Operating system, its configuration and user data are **not strictly separated**.



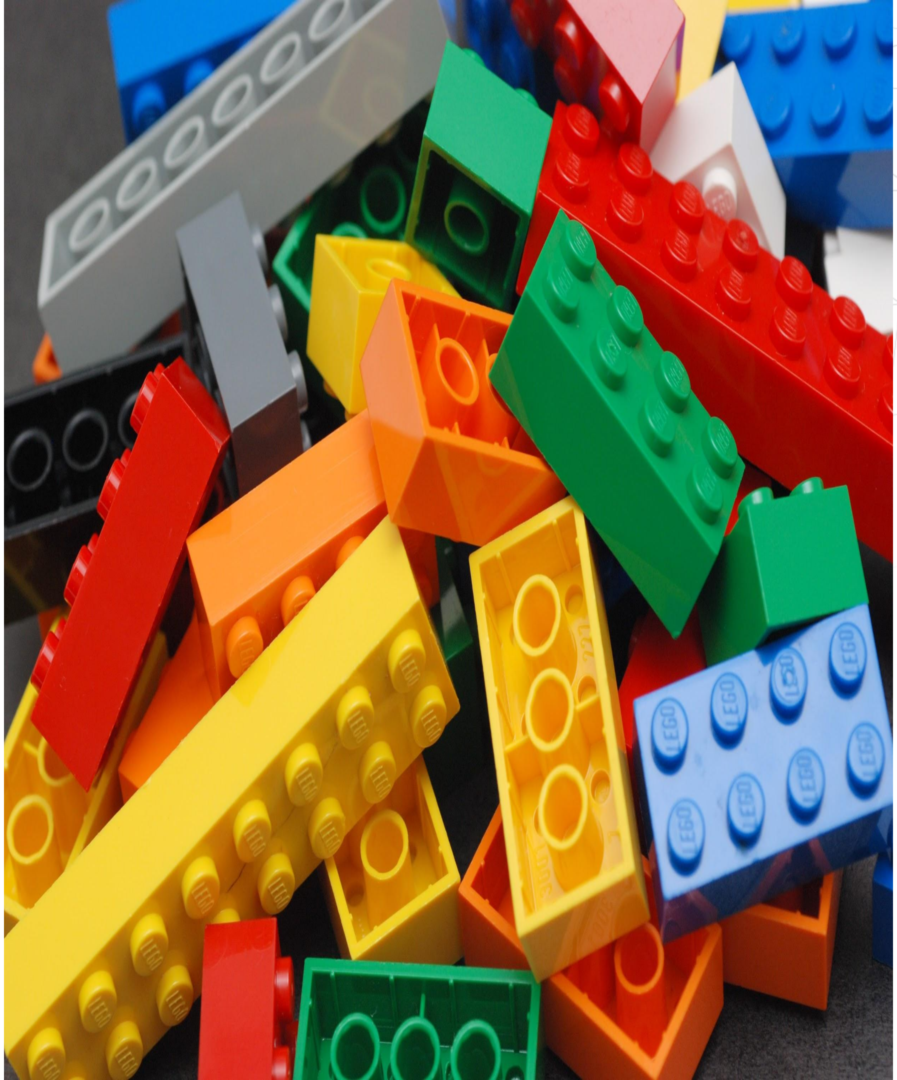
What is a Stateless OS

Stateful OS

A single OS version could be **permutated** into **thousands of combinations**.

That’s really hard to manage without a configuration manager (Puppet*, Ansible*, Chef*, etc)

*Other names and brands may be claimed as the property of others

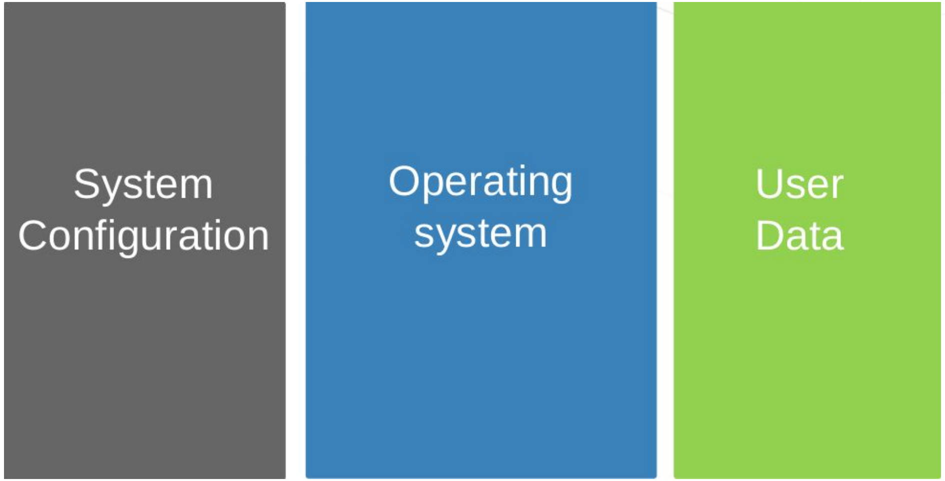


What is a Stateless OS

Stateless OS

Operating system, configuration and user data are **strictly separated**.

- Read-only root FS
- Default configuration provided
- Allow sysadmin to override default configurations
- Factory reset allowed!



Stateless + Bundles = Consistency

I just want a web server...

Sure! you just have to:

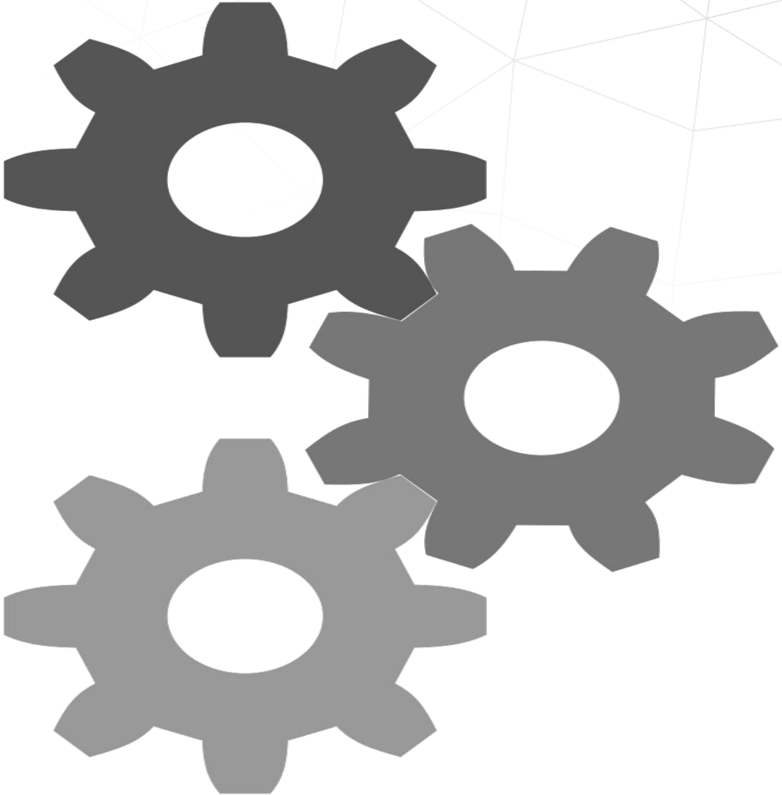
- Install X, Y , Z packages
- Look over a whole messy config files in /etc to make it work
- start your web server!

Stateless + Bundles = Consistency

I just want a nova-compute node...

Sure! you just have to:

- Install X, Y , Z packages
- Look over a whole messy config files in /etc to make it work
- start your nova-compute node!



Stateless + Bundles = Consistency

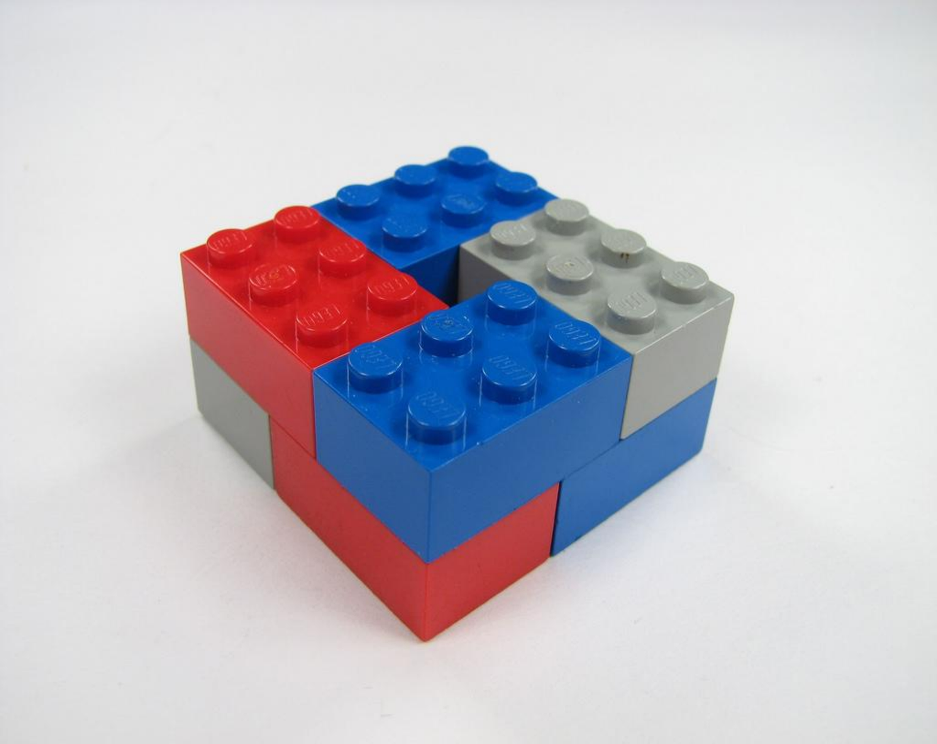
I just want X functionality, or …

Stateless + Bundles = Consistency

So what if... bundles!

Functionality over granularity (packages)

- want a nova-compute node? just install the ***openstack-compute*** bundle
- want X functionality? just install the X bundle!



Stateless + Bundles = Consistency

What’s the gain with bundles?

The Clear Linux* project approach:

“Update from one consistent OS version to other consistent version”

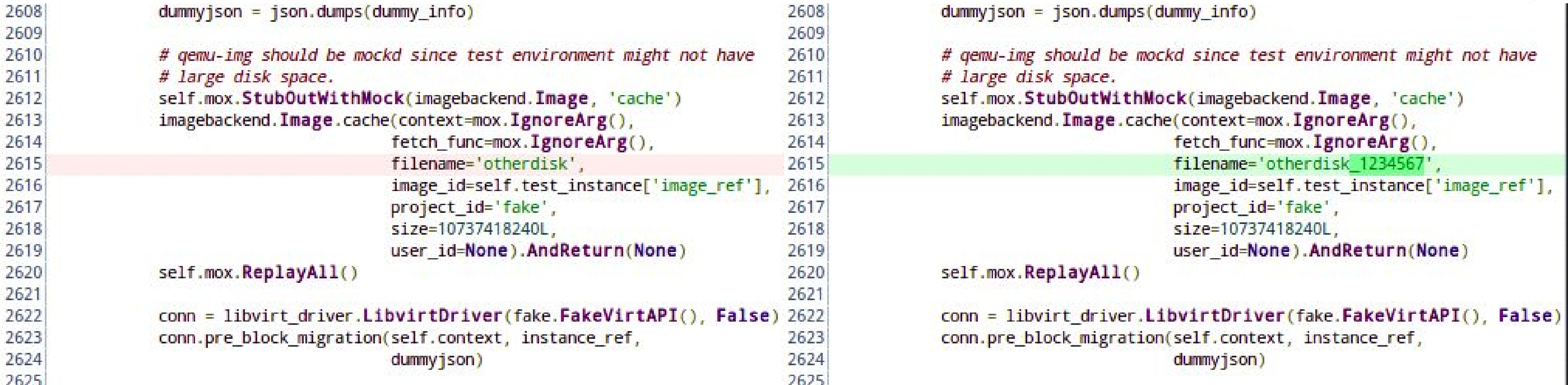
- When updating from K to K+1 version **Everything gets updated**
- The whole deployment is defined just by the OS version

*Other names and brands may be claimed as the property of others

Stateless + Bundles = Consistency

Whole distro update?! That’s... big?

The update technology **only updates files that actually changed**, using so-called binary-delta technology for efficiency.



```
        dummyjson = json.dumps(dummy_info)

        # qemu-img should be mockd since test environment might not have
        # large disk space.
        self.mox.StubOutWithMock(imagebackend.Image, 'cache')
        imagebackend.Image.cache(context=mox.IgnoreArg(),
                                 fetch_func=mox.IgnoreArg(),
                                 filename='otherdisk',
                                 image_id=self.test_instance['image_ref'],
                                 project_id='fake',
                                 size=10737418240L,
                                 user_id=None).AndReturn(None)
        self.mox.ReplayAll()

        conn = libvirt_driver.LibvirtDriver(fake.FakeVirtAPI(), False)
        conn.pre_block_migration(self.context, instance_ref,
                                 dummyjson)
```

```
        dummyjson = json.dumps(dummy_info)

        # qemu-img should be mockd since test environment might not have
        # large disk space.
        self.mox.StubOutWithMock(imagebackend.Image, 'cache')
        imagebackend.Image.cache(context=mox.IgnoreArg(),
                                 fetch_func=mox.IgnoreArg(),
                                 filename='otherdisk_1234567',
                                 image_id=self.test_instance['image_ref'],
                                 project_id='fake',
                                 size=10737418240L,
                                 user_id=None).AndReturn(None)
        self.mox.ReplayAll()

        conn = libvirt_driver.LibvirtDriver(fake.FakeVirtAPI(), False)
        conn.pre_block_migration(self.context, instance_ref,
                                 dummyjson)
```

Stateless + Bundles = Consistency

In summary

- Consistency over multiple OS installations
- Look at a higher level: I need X functionality
- Server messed up? just **`rm -rf /etc /var`** and reboot, just like new!
- Don’t mess up with tons of configuration lines, just set up what you know you need

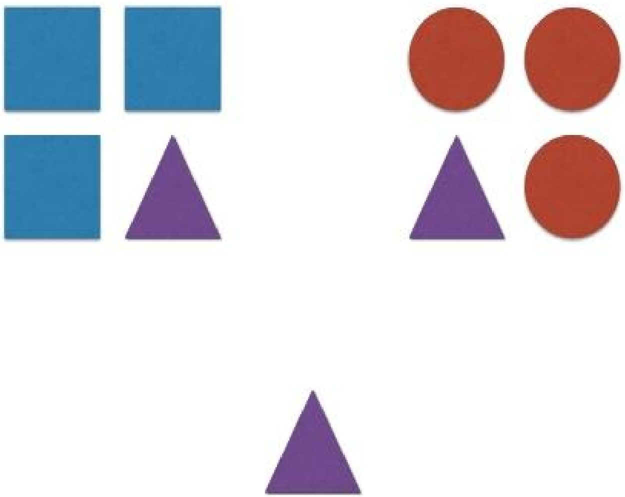
Stateless OpenStack

- We need to provide an easier way to deploy openstack
- If our node is b0rked, just `rm -rf /var /etc`, reboot and reconfigure with your previously configured values
- No need to worry for default values we didn’t touch; they are still there

Stateless OpenStack: How we made it

Most Valuable Player: oslo.config

- (Almost)[1] All OpenStack components use a single python module (**oslo.config**) to handle configurations
- stateless oslo.config = stateless openstack



[1] *We found out some corner cases where this is not true*

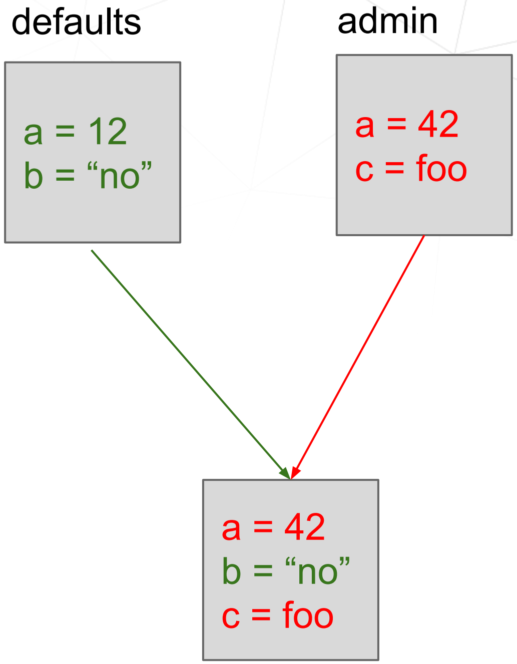
The logic

the basics of our stateless approach:

# Load default configs located in stateless dir (e.g: /usr/share/defaults/nova/nova.conf)

is there admin configs? (/etc/nova/nova.conf)

Read the admin configs and **override** options already **defined by defaults**



Demo

Please don’t disturb Murphy!

Q&A

For more information please go to: https://clearlinux.org/

Stateless OS:
From an Openstack perspective

Marcos Simental

Clear Linux* Project
for Intel® Architecture

References & Image sources

- https://clearlinux.org/
- https://web.archive.org/web/20130601131441/http://citethisbook.net/Red_Hat_Introduction_to_Stateless_Linux.html
- https://upload.wikimedia.org/wikipedia/commons/3/32/Lego_Color_Bricks.jpg
- https://upload.wikimedia.org/wikipedia/commons/thumb/f/fc/Server-web.svg/500px-Server-web.svg.png
- https://upload.wikimedia.org/wikipedia/commons/thumb/a/ac/Cog-icon-grey.svg/806px-Cog-icon-grey.svg.png
- https://upload.wikimedia.org/wikipedia/commons/f/f4/20060513_toolbox.jpg
- https://www.flickr.com/photos/oskay/2157692222
- https://upload.wikimedia.org/wikipedia/commons/thumb/d/de/Gnome-system-software-update.svg/768px-Gnome-system-software-update.svg.png
- http://docs.openstack.org/infra/publications/2014-lca-overview/images/gerrit-diff.png
- https://upload.wikimedia.org/wikipedia/commons/thumb/8/80/The_OpenStack_logo.svg/2000px-The_OpenStack_logo.svg.png
- http://image.slidesharecdn.com/oslo-long-view-atlanta-2014-140919095600-phpapp01/95/oslo-program-overview-openstack-atlanta-11-638.jpg?cb=1411120621