redhat.

Analyzing Performance of OpenStack with Grafana Dashboards

GrafanaCon EU 2018

Alex Krzos
Senior Software Engineer
March 2nd 2018

Agenda

What is OpenStack

What is the problem?

What is the Monitoring Stack

Dashboards Dashboards Dashboards

A dashboard is worth a thousand words

Example Perf and Scale Analysis

What can be improved?

Questions

whoami

Alex Krzos

Senior Software Engineer @ Red Hat Inc

On Red Hat OpenStack Scale and Performance Team

2 years on OpenStack

2 years ManageIQ & RH Satellite Performance

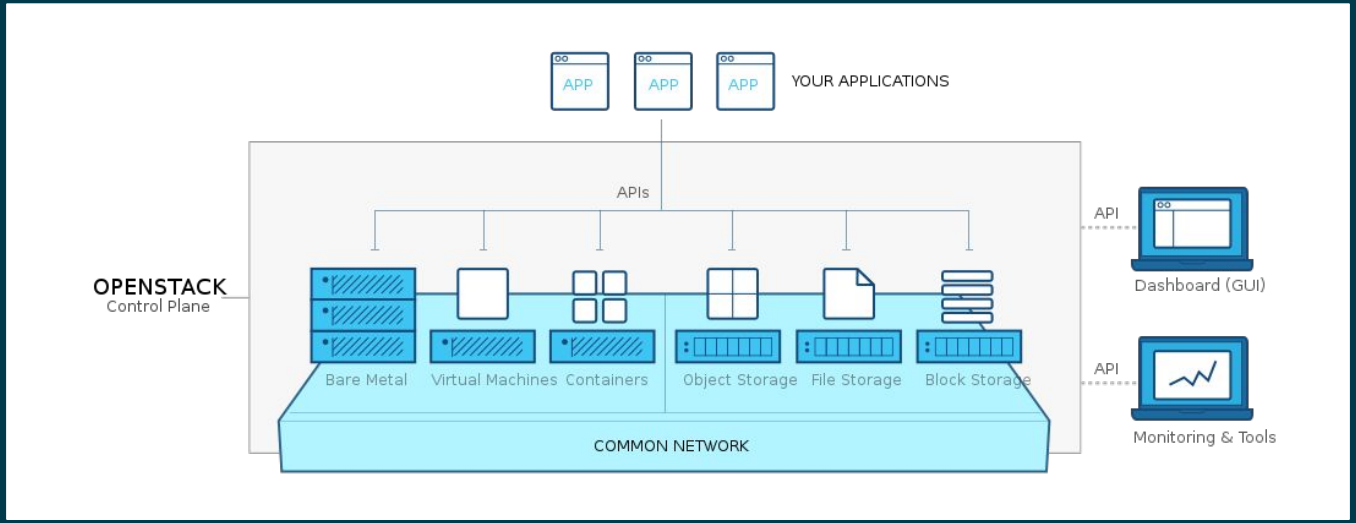
4 years at Cisco System/Network testing

Recently completed Army National Guard Obligation

What is OpenStack

Open Source Cloud Software - Cloud Operating System

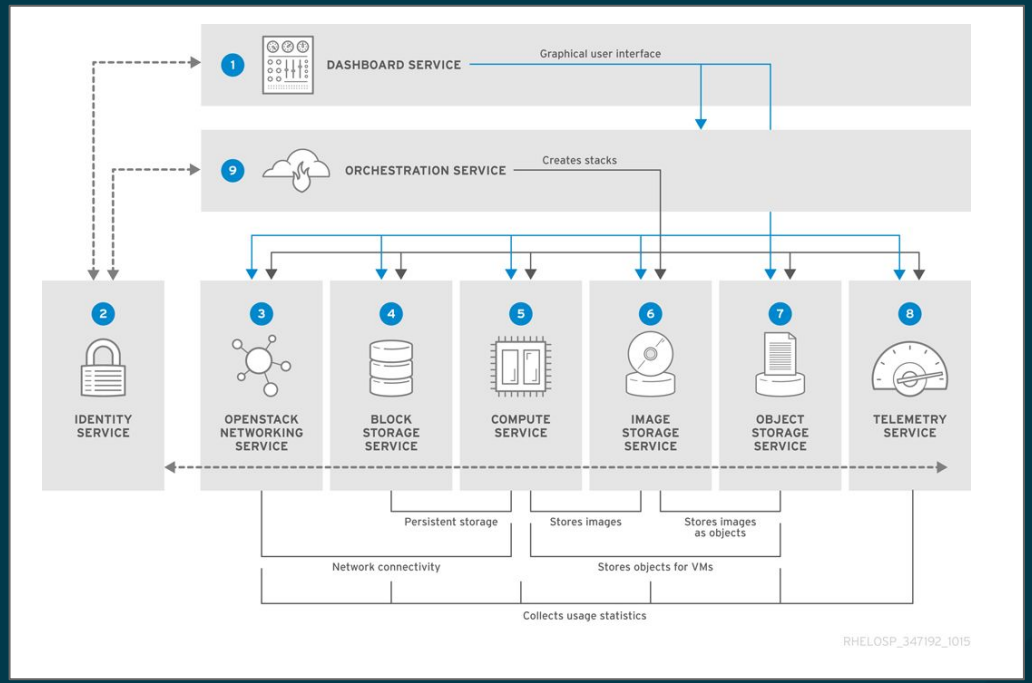
64 official projects, 1582 repos in github.com/openstack



What is OpenStack

Major Projects:

- Keystone - Identity
- Nova - Compute
- Neutron - Networking
- Swift - Object Storage
- Cinder - Block Storage
- Glance - Image Storage
- Telemetry - Time-Series Data
- Horizon - Dashboard GUI



What are the problems?

Many services, daemons, databases and messaging bus to monitor

Complexity of service interactions

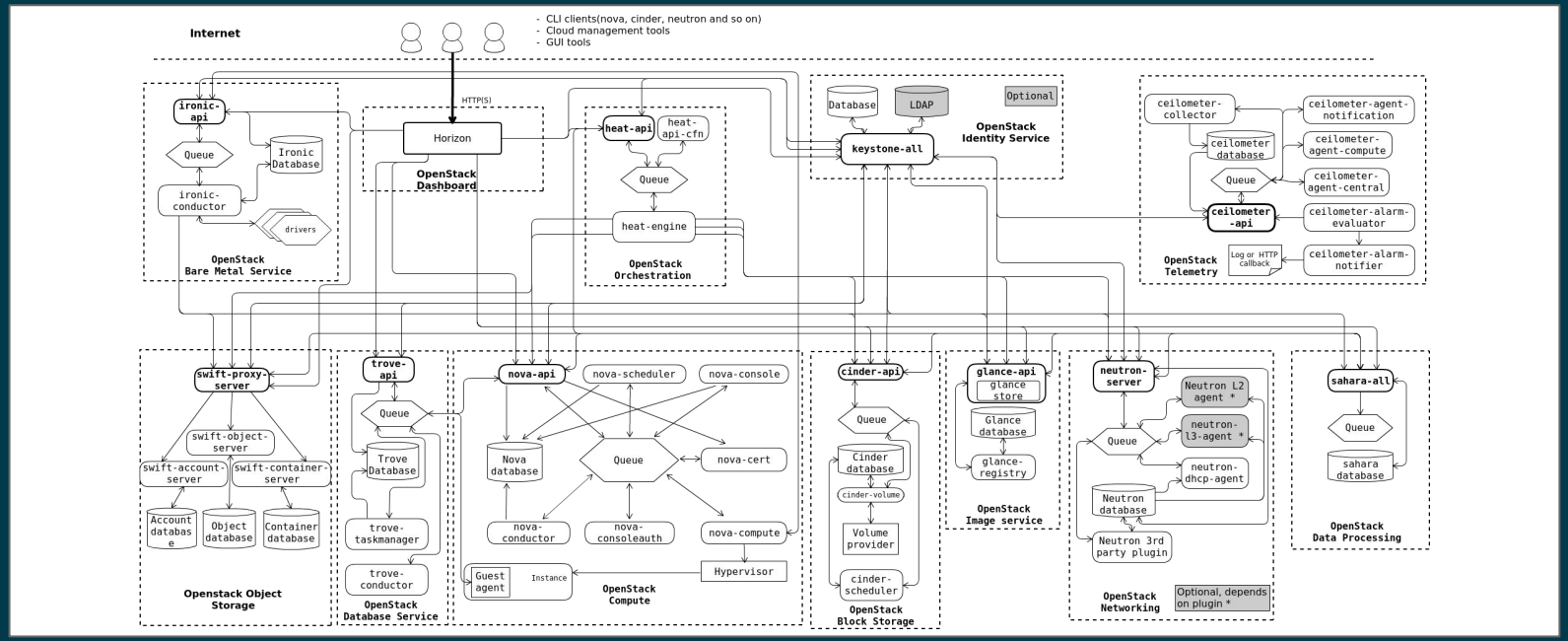
Varying node counts and node types

Large configuration files

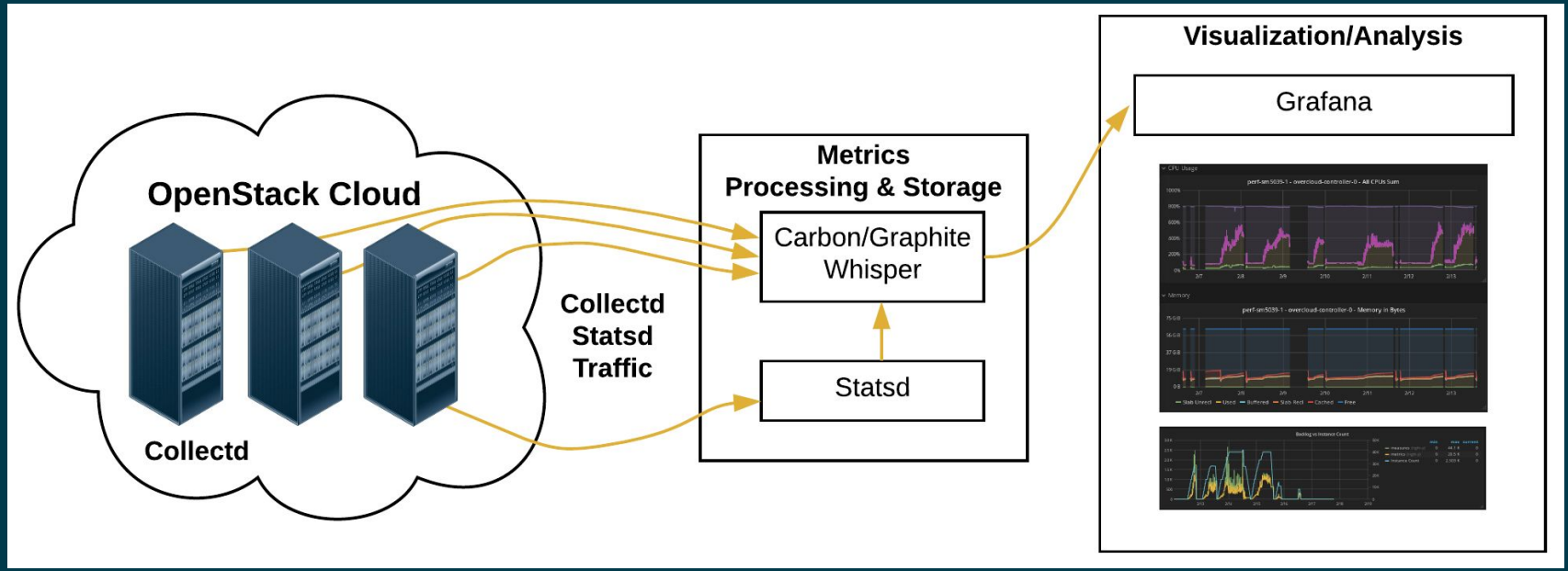
Data capture on testbeds

Errors / Misconfiguration

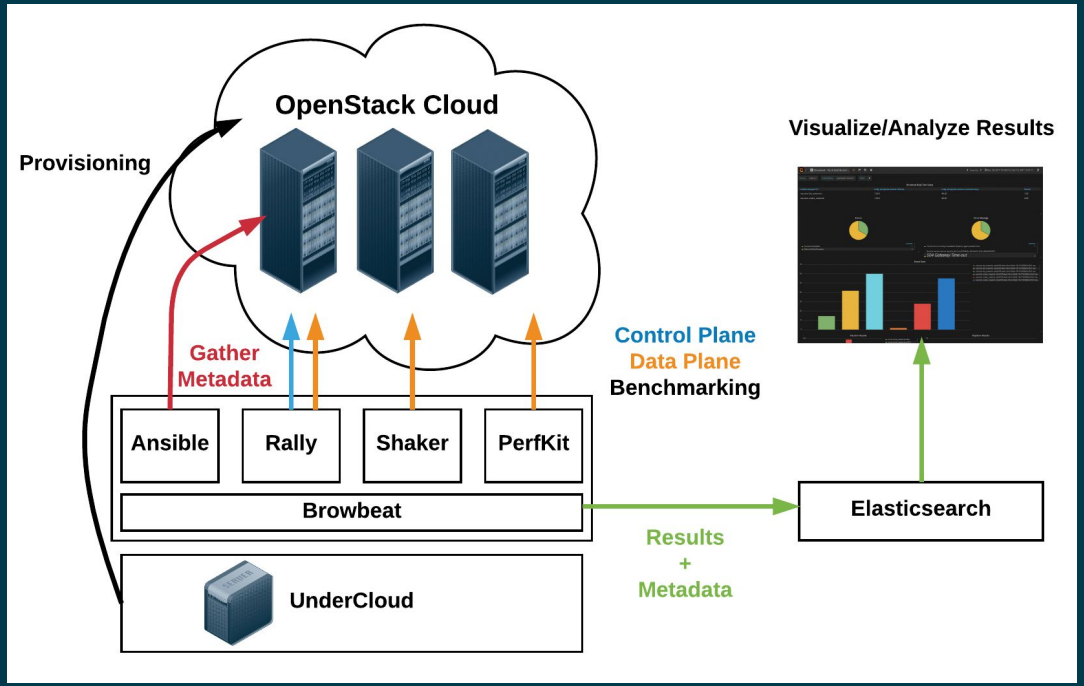
What are the problems? - Complexity



What is our Monitoring Stack



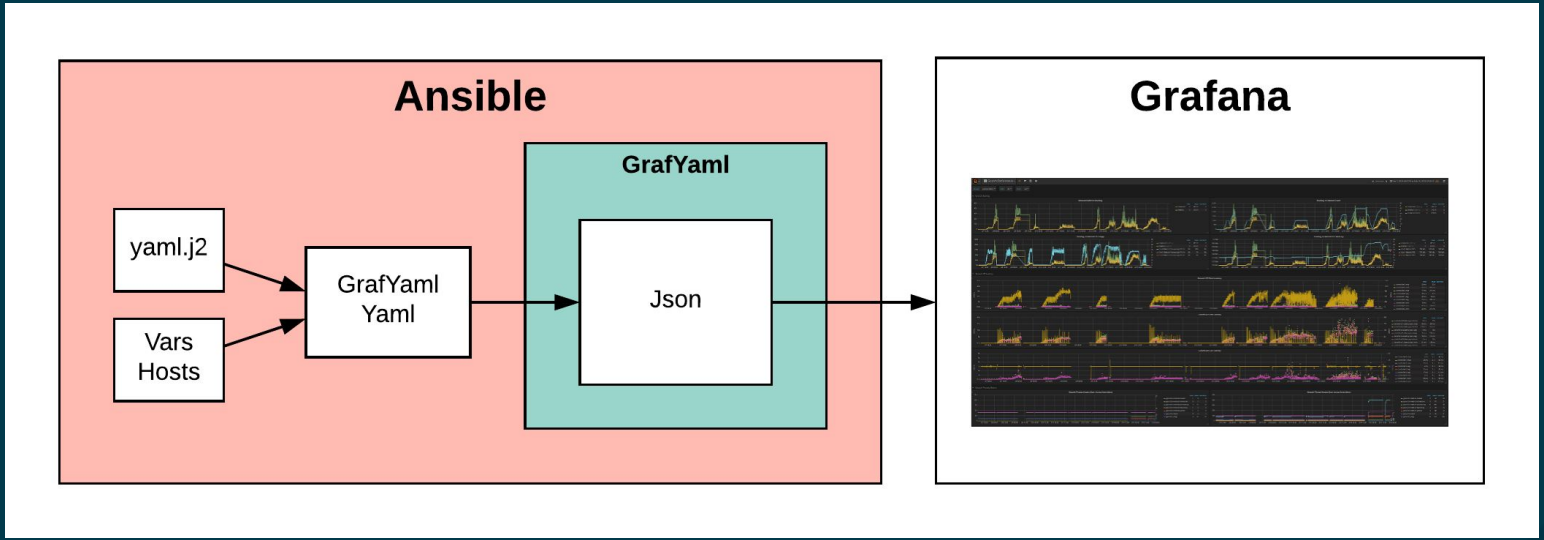
Benchmarking Tools



Deploying the Dashboards

Deploying the Dashboards via Ansible:

ansible-playbook -i hosts install/grafana-dashboards.yml



How Dashboards are Stored

Originally: .json and .json.j2

Converted to GrafYaml (.yaml, .yaml.j2)

Yaml advantages:

- Less Lines
- Less Curly braces and quotes
- Comments

GrafYaml Advantages:

- Manages ID / RefID
- Defaults reduce lines stored in Yaml
- Simple CLI

Jinja2 Template Advantages

- Reduce duplication
- Reuse by organizing into separate files

GrafYaml+Jinja2 reduced line count:

40,000 lines to 6,500 lines

Dashboards, Dashboards, Dashboards

4 Types of Dashboards

- Static Dashboards (.yaml)
- Templated “Static” Dashboards
- Templated General Dashboards
- Cloud Specific Dashboards (Per-Cloud)

Static Dashboards

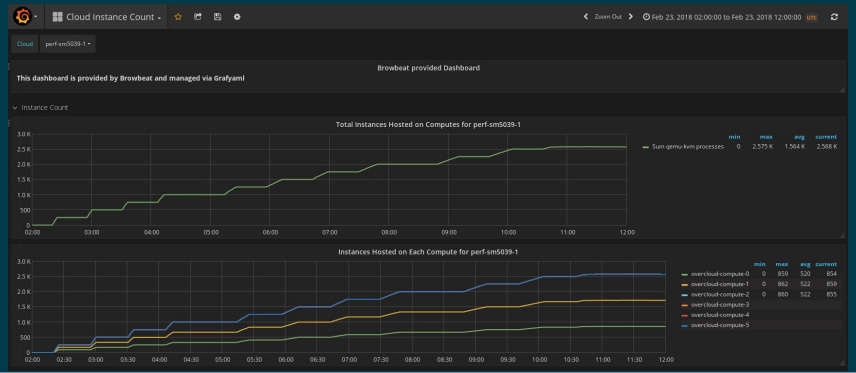
Cloud Instance Count

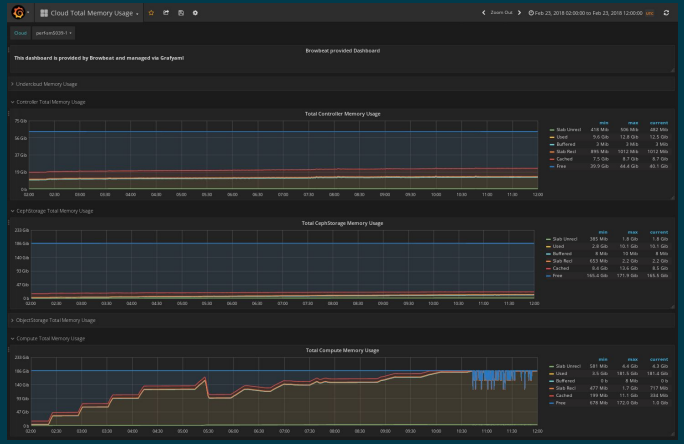
Cloud Total Memory Usage

Cloud System Performance Comparison

Cloud Keystone Token Count

Apache Request Latency





redhat.

Static Dashboards - Performance Comparison

Templated Dashboards

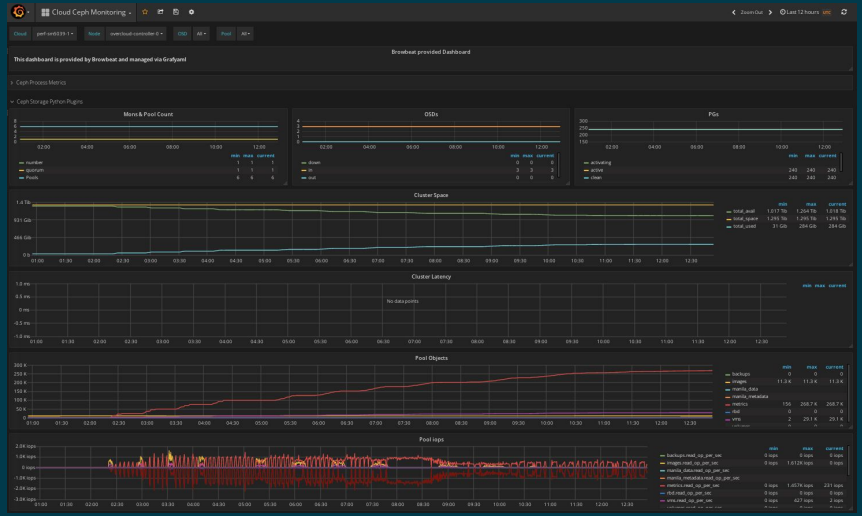
Cloud Ceph

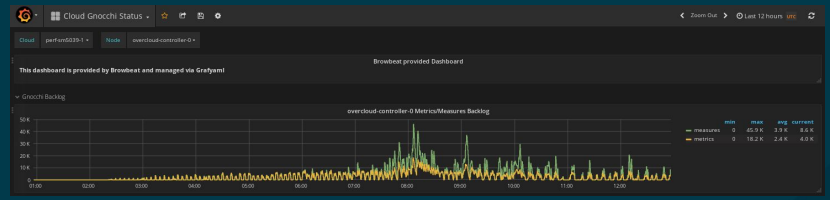
Cloud Rabbitmq

Three Node Performance

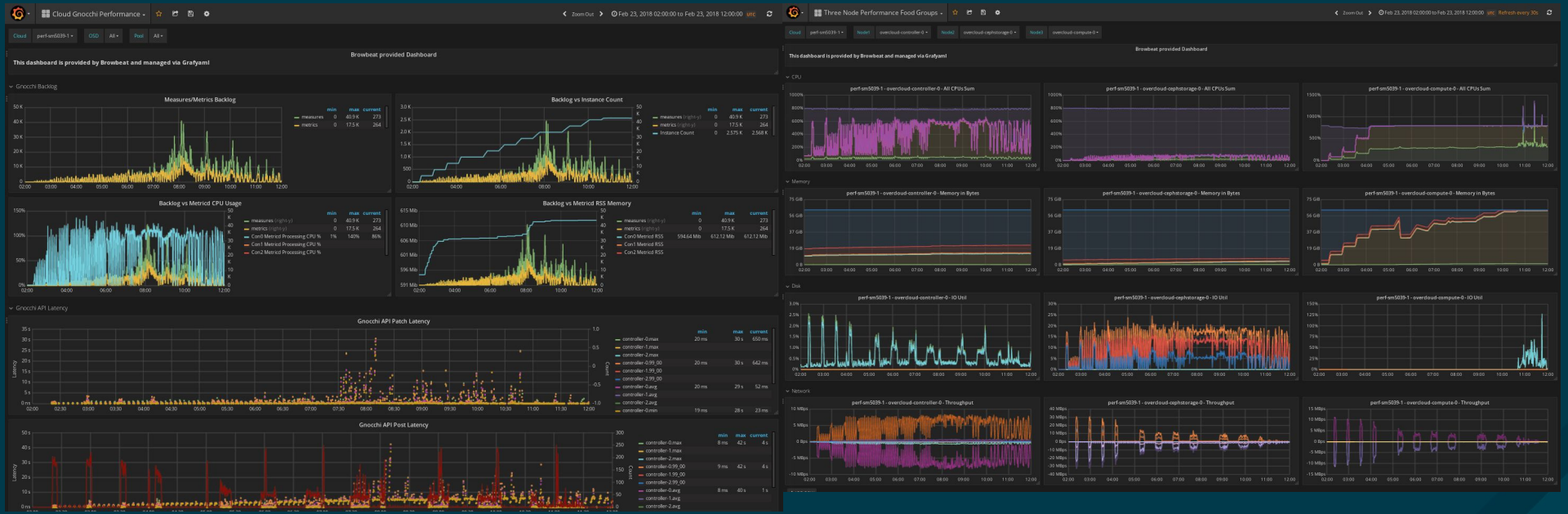
Gnocchi Status

Gnocchi Performance





Templated Dashboards

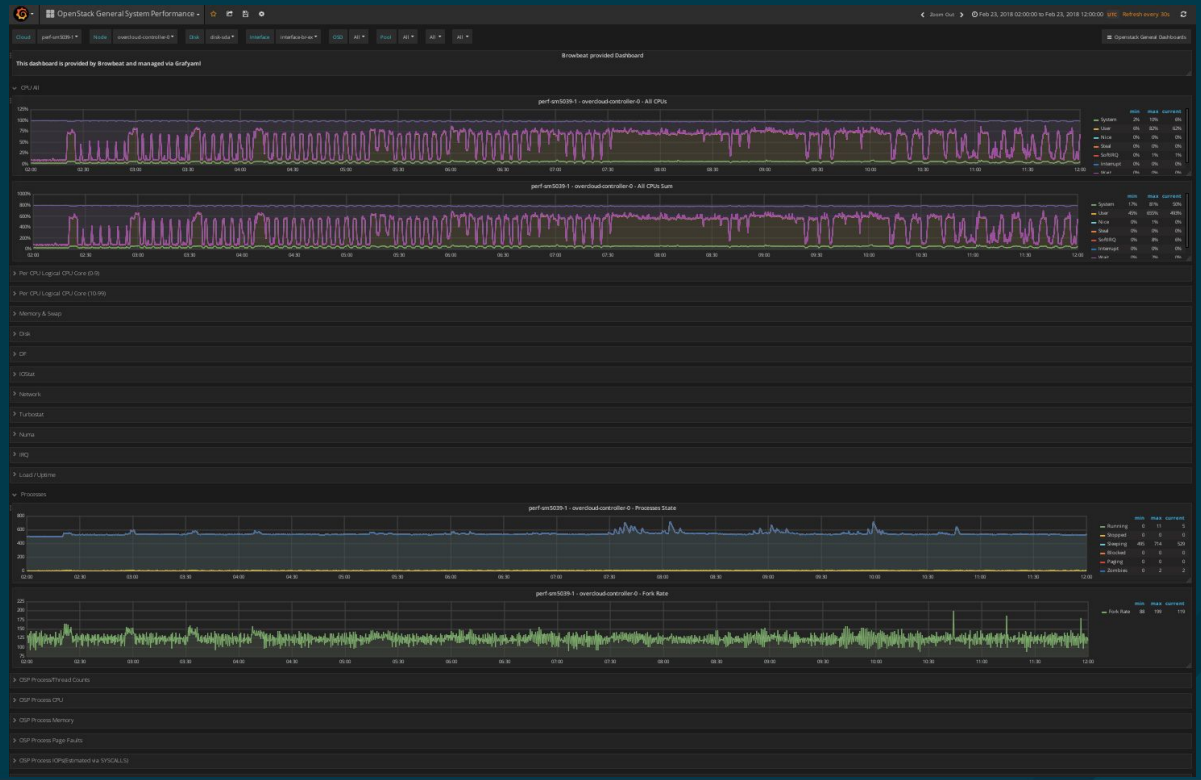


redhat.

Templated General Dashboards

Node Types:

- Undercloud
- Controller
- Ceph Storage
- Block/Object Storage
- Computes

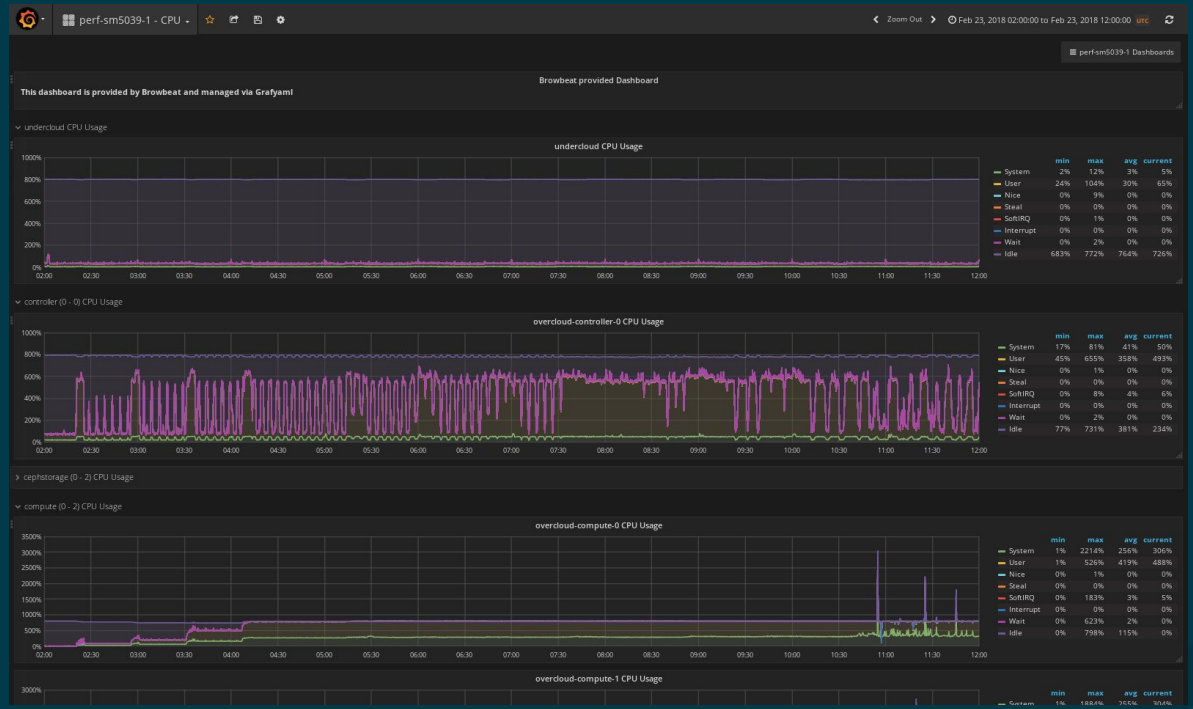


Cloud Specific Dashboards - CPU

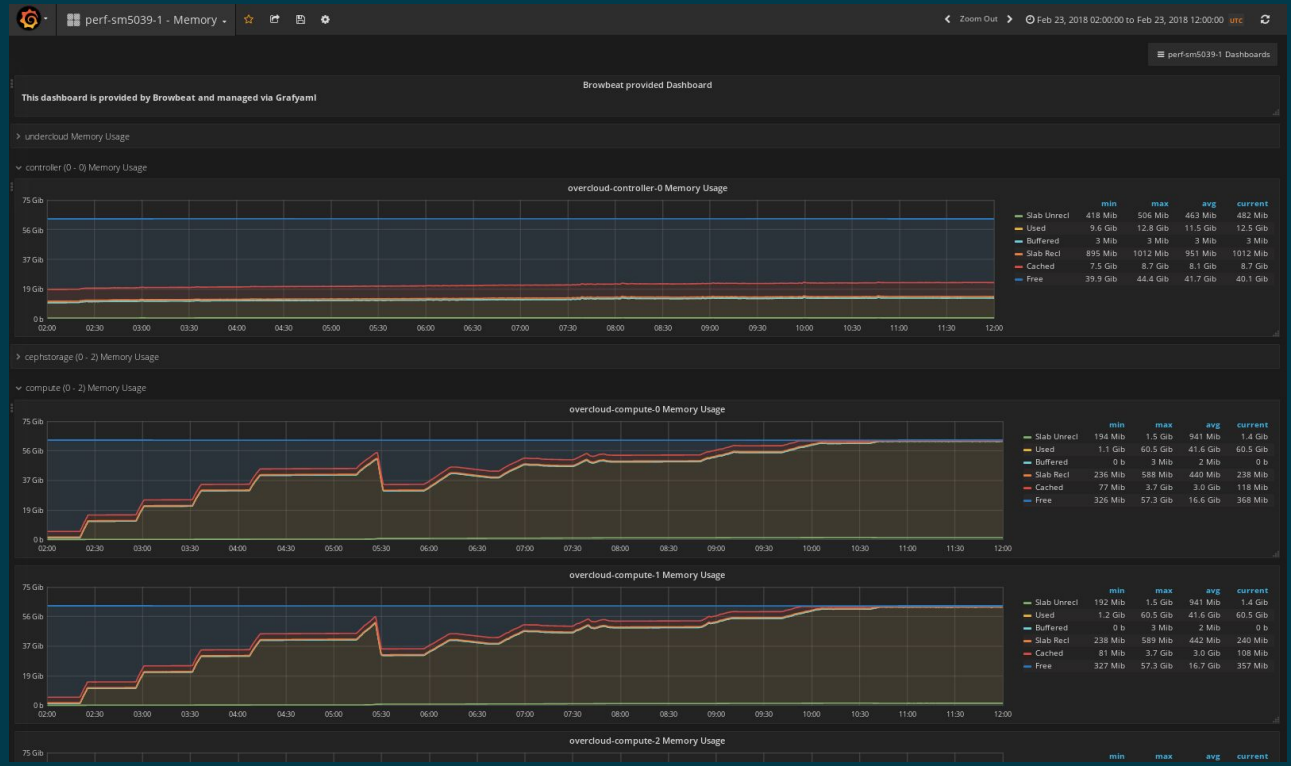
Exposes Each Node in Ansible Inventory

Includes:

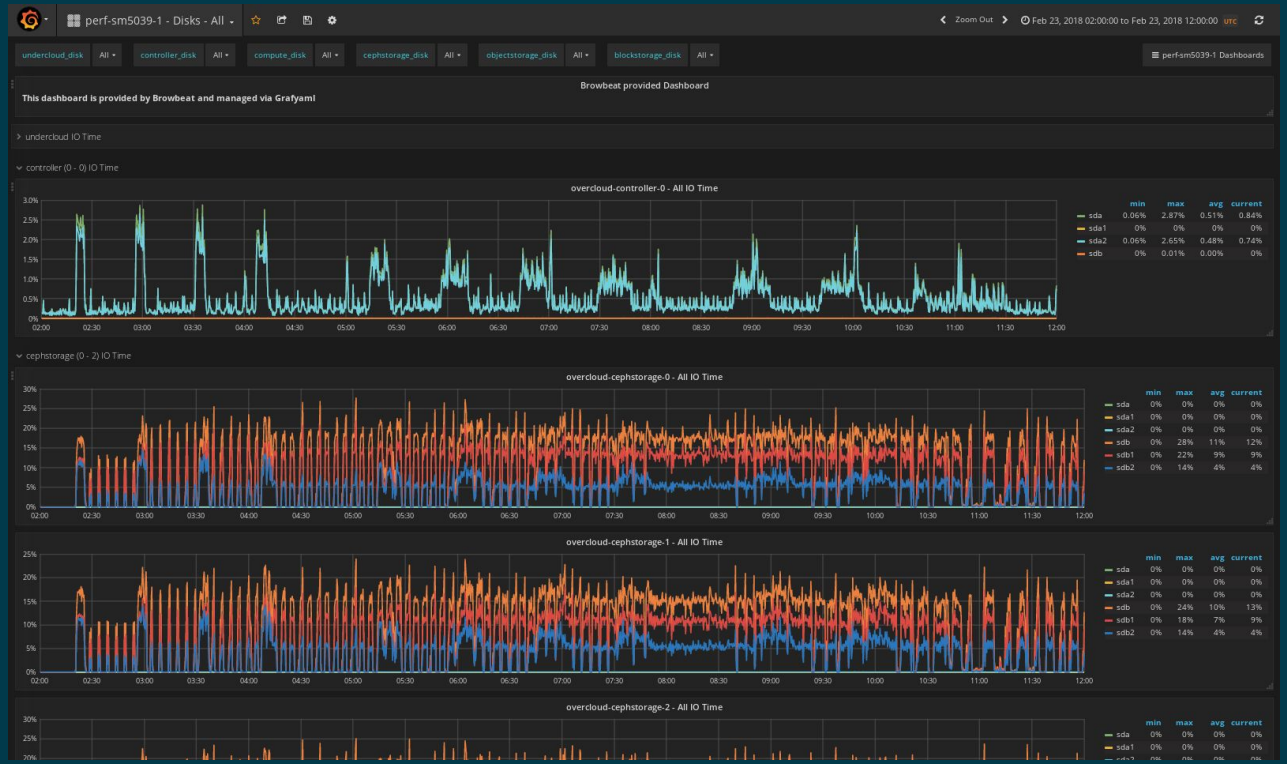
- CPU
- Memory
- Disk
- Network
- Log



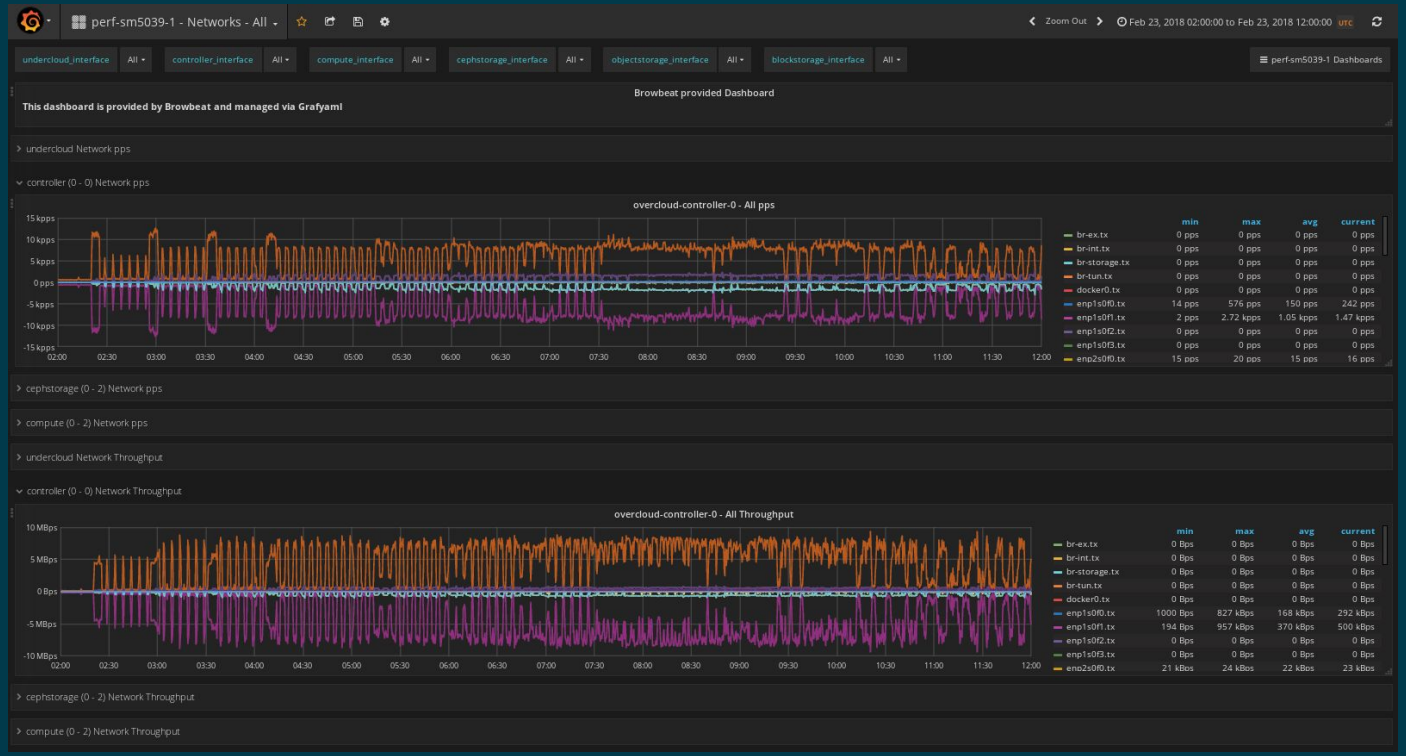
Cloud Specific Dashboards - Memory



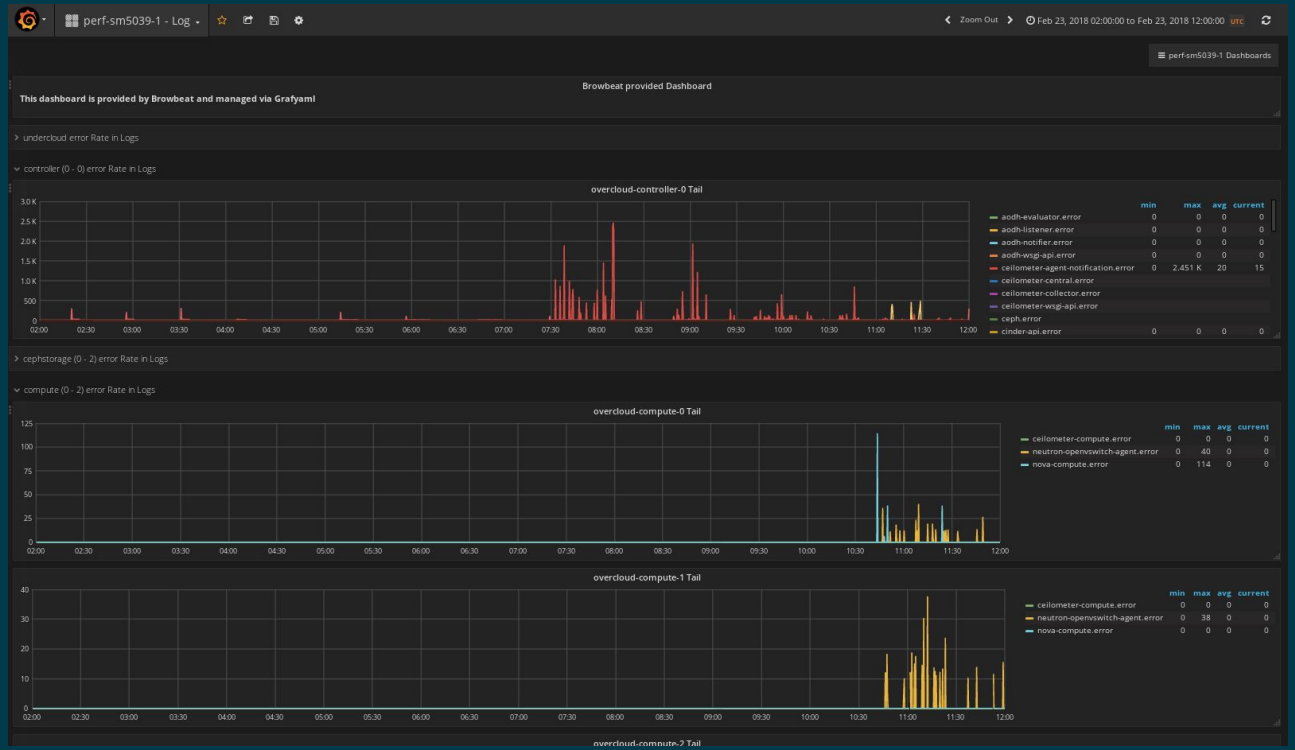
Cloud Specific Dashboards - Disk



Cloud Specific Dashboards - Network



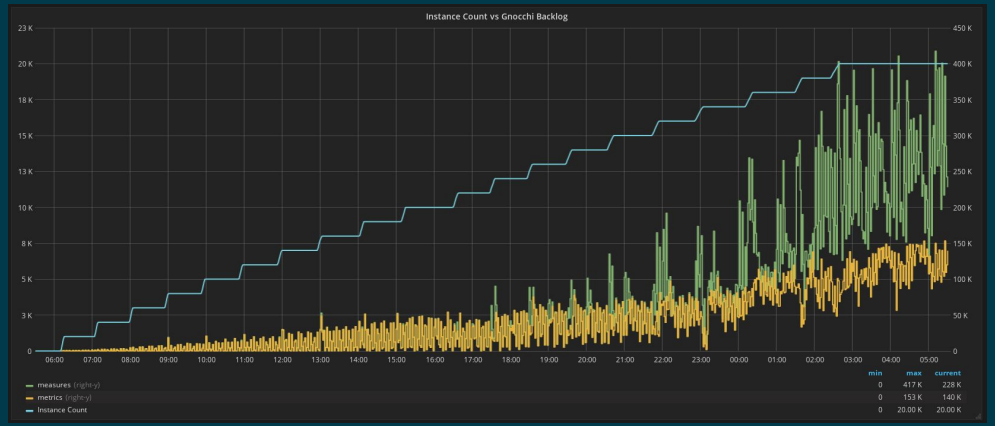
Cloud Specific Dashboards - Log



Example Performance and Scale Analysis

Example - OpenStack Telemetry Scaling

OpenStack Pike



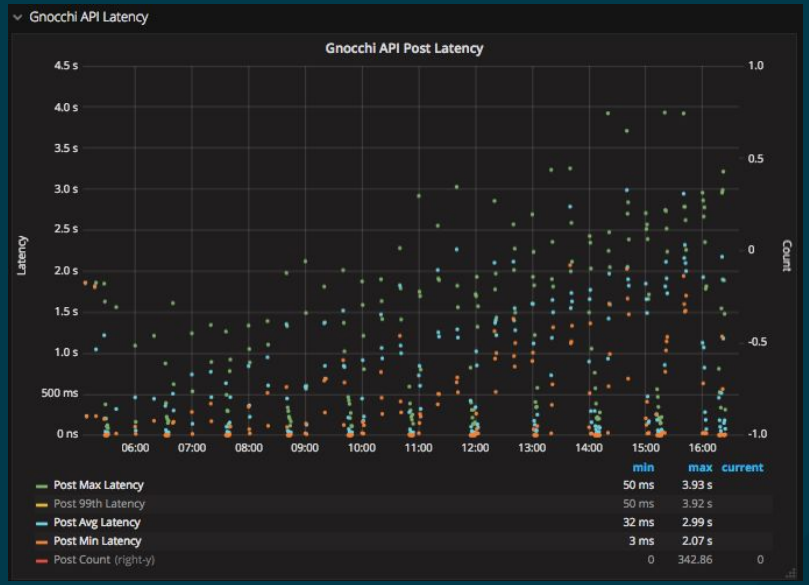
OpenStack Ocata

redhat.

Example - Slow API Post

Threaded

Batch

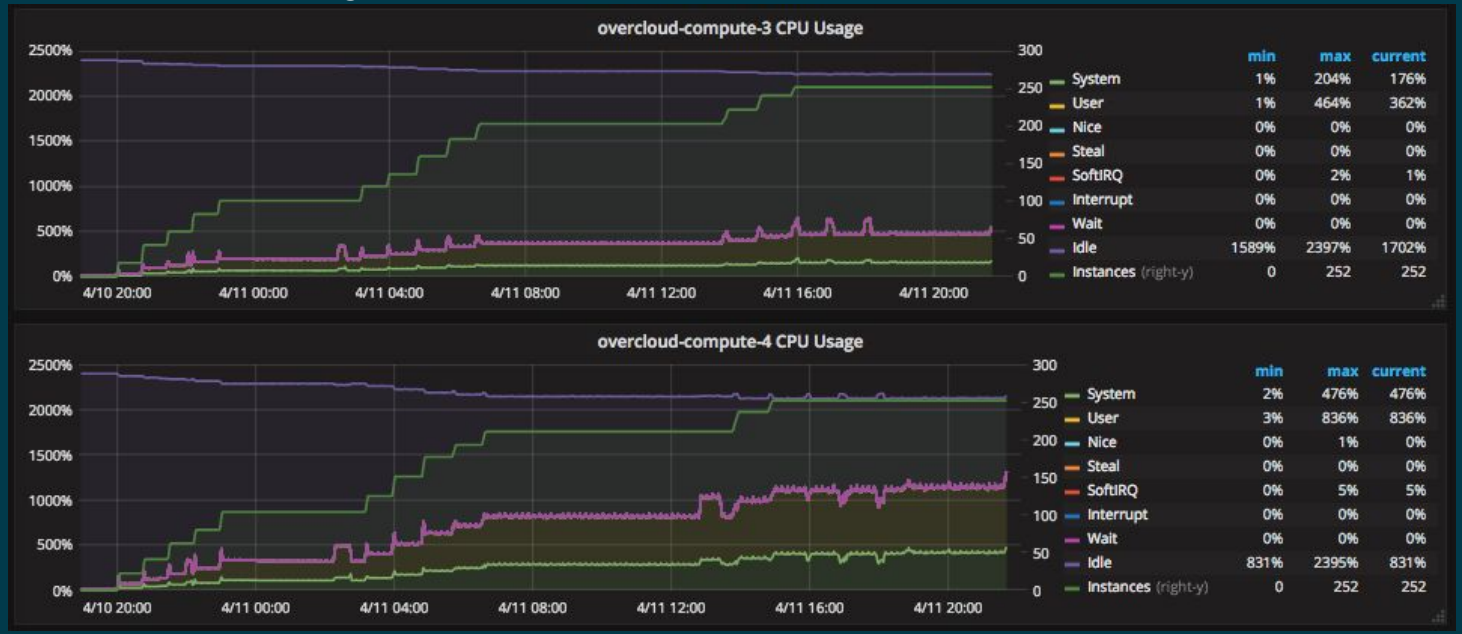


Example - Coordination Loss

Daemon lost contact with coordination service

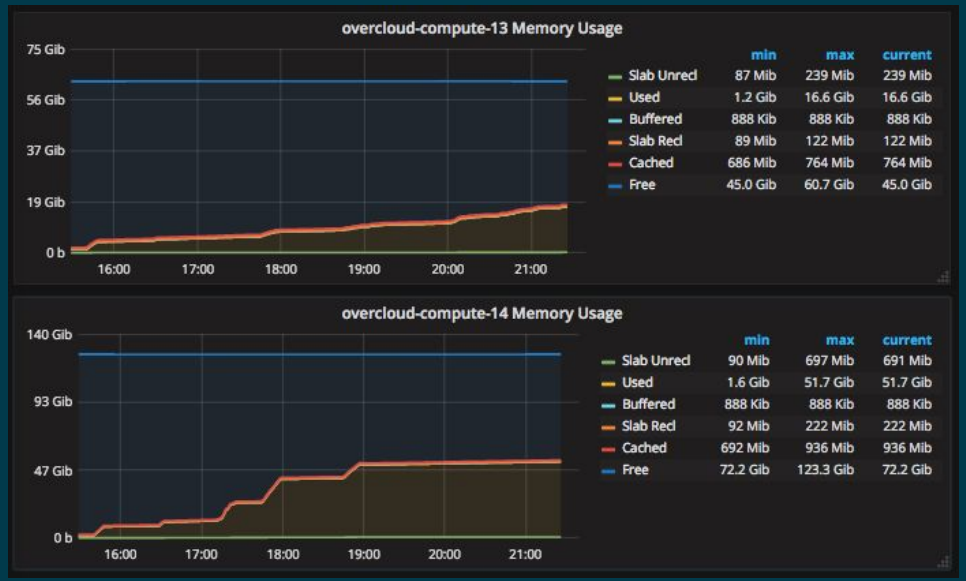
Example - SMIs using more CPU

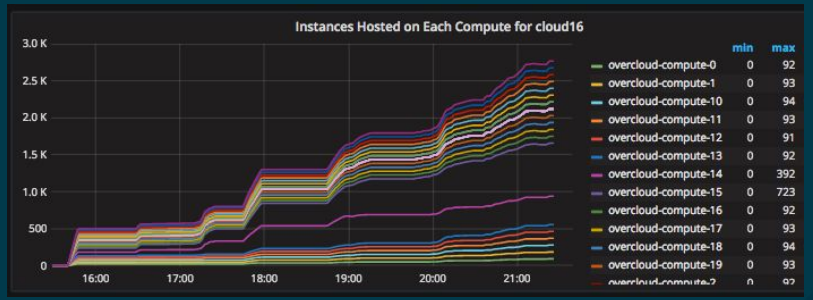
Overcloud-compute-4 has 480 SMIs every 10s resulting in higher CPU util, Set “OS Control” in your BIOS power settings…



Example - Uneven Memory

Visualize distribution of instances in uneven compute memory environment





Example - Boot Timings

Unpatched vs Patched Boot Scenario

What can be improved?

Dashboard Storage / Reuse

Queries for different Data Sources

Preserving Data (Snapshotting)

Time Series Data Units

Standard Time Series Benchmarks

Metrics interval


Questions?

References / Links

Browbeat - https://github.com/openstack/browbeat

GrafYaml - https://github.com/openstack-infra/grafyaml

Ansible - https://github.com/ansible/ansible

OpenStack - https://www.openstack.org/

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos