



HP
Helion



rackspace
HOSTING

IBM

Paris OpenStack® Summit

This is your cloud

© Copyright 2014 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice.





Monasca Deep Dive

Monitoring-as-a-Service (at Scale)

Roland Hochmuth, Sandy Walsh, Tong Li/ November 5, 2014

© Copyright 2014 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice.

Agenda

Problem Statement

What is Monasca?

- Architecture
- Metrics
- Events/StackTach.v3
- Anomaly Detection

Current Status

Performance

Next Steps

Demo

Q&A



Problem Statement



Monitoring-as-a-Service: Lacking multi-tenant model

Performance, scalability and data retention

Multiple uses of the data: SLA calculations, business analytics, RCA, ...

Elasticity and dynamic run-time configurability

- Metrics and Alarm management
- Spammy alerts and alert fatigue.

Real-time event stream processing

Extensibility: Integrate with other systems via API or internally

Multiple Systems: Internal/operational monitoring and external/customer-facing monitoring are separate systems. Health/Status different from metrics.

Cryptic data: Force fit metric/event names results in an impedance mismatch



What is Monasca?

Monitoring-as-a-Service solution based on a first-class REST API

- Multi-tenancy based on Keystone authentication. Supports self-service.

Highly-performant, scalable, fault-tolerant and capable of big data retention

Metrics storage/retrieval/statistics and alarm/thresholding engine

Notification system

Real-time event stream processing

Open-source and built-on open-source technologies such as:

- Kafka: Performant, scalable, fault-tolerant, durable message queue. Used by LinkedIn, Twitter, ...
- Apache Storm:
- Time-series databases: InfluxDB supported today. Elastic-search in progress.

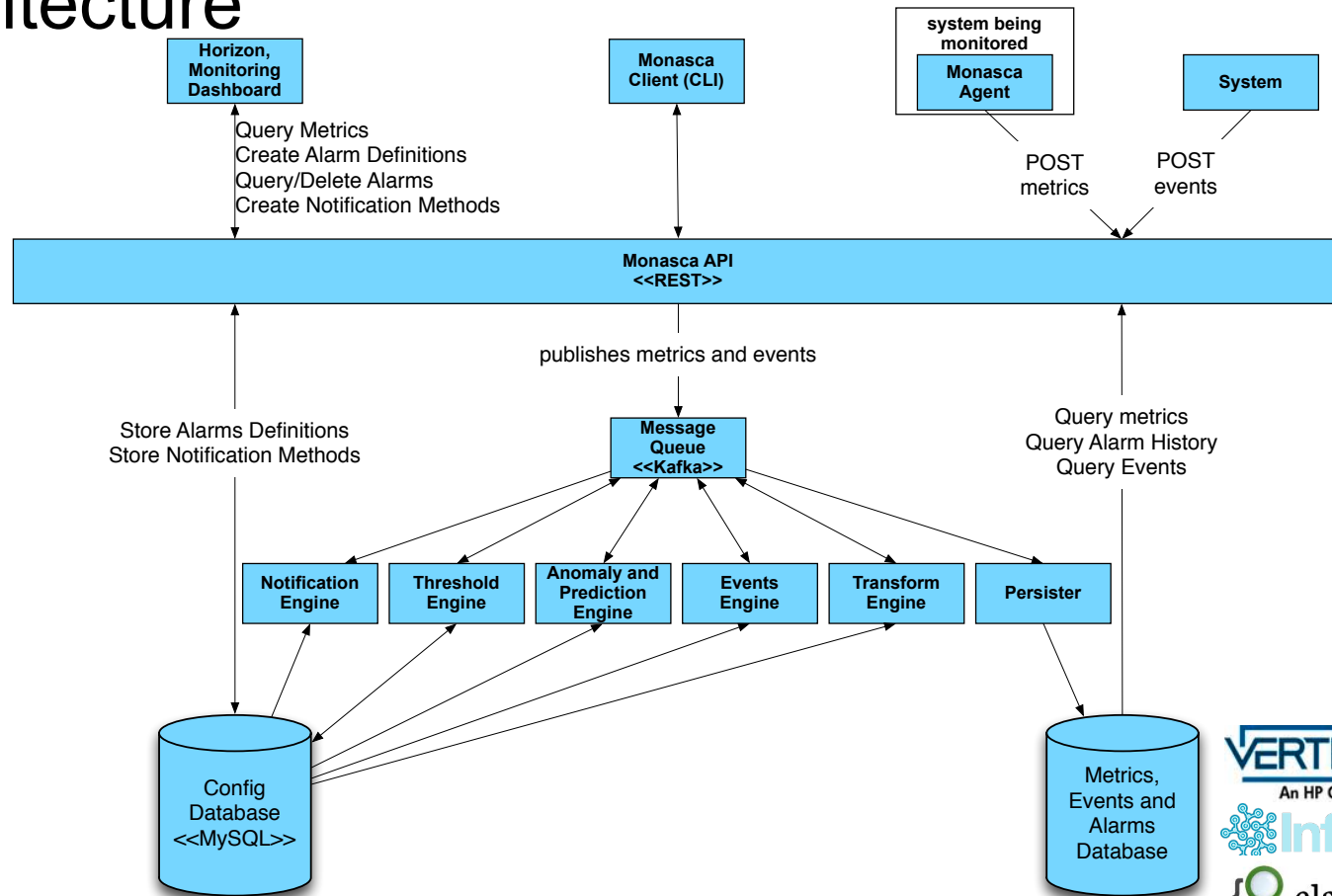
Consolidates multiple monitoring systems into a single solution

- Used for both operational and customer facing monitoring.

Extensible based on micro-services message bus architecture



Architecture





Metrics



REST API

Metrics: Create, query and get statistics for metrics

```
{
  name: cpu.user_perc,
  dimensions: {
    hostname: hostname.domain.com,
    region: uswest,
    zone: 1,
    service: compute
  }
}
```

Simple, concise beautiful flexible description
Name (string)
Dimensions: Dictionary of arbitrary (key, value) pairs

Alarm Definitions

- Alarm definitions are templates that are used to automatically create alarms based on matching metric names and dimensions
- Simple compound expression grammar: `avg(cpu.user_perc{}) > 85` or `avg(disk_read_ops{device=vda}, 120) > 1000`
- Actions associated with alarms for state transitions to ALARM, OK and UNDETERMINED
- Severity (LOW, MEDIUM, HIGH, CRITICAL).

Alarms: Query and Delete alarms and query alarm state history

Notification Methods: e.g. Email address. Associated with alarm definitions



Monasca Agent

Python monitoring agent

System metrics (cpu, memory, ...)

Service metrics

- RabbitMQ, MySQL, Kafka, and many others

Application metrics

- Built-in statsd daemon
- Python Monasca Statsd library

VM metrics

Active checks

- HTTP status checks and response times
- System up/down checks (ping and ssh)

Runs any Nagios plugin

Extensible/Pluggable: Additional services can be easily added



UI

Horizon Dashboard

- Overview/Top-level drill-down
- Create/Read/Update/Delete alarm definitions using an expression builder
- Read/Delete alarms and alarm history
- Create/Read/Update/Delete notification methods

Grafana Dashboard (<http://grafana.org/>)

- Provides visualization of metrics





Events



StackTach.v3




Anomaly Detection



Monasca Anomaly Engine implements real-time streaming anomaly detection

Two algorithms:

- Numenta Platform for Intelligent Computing (NuPIC) used by  **GROK**
 - An open-source Python/C++ implementation of Hierarchical Temporal Memory
- Kolmogorov-Smirnov (K-S) Two Sample Test

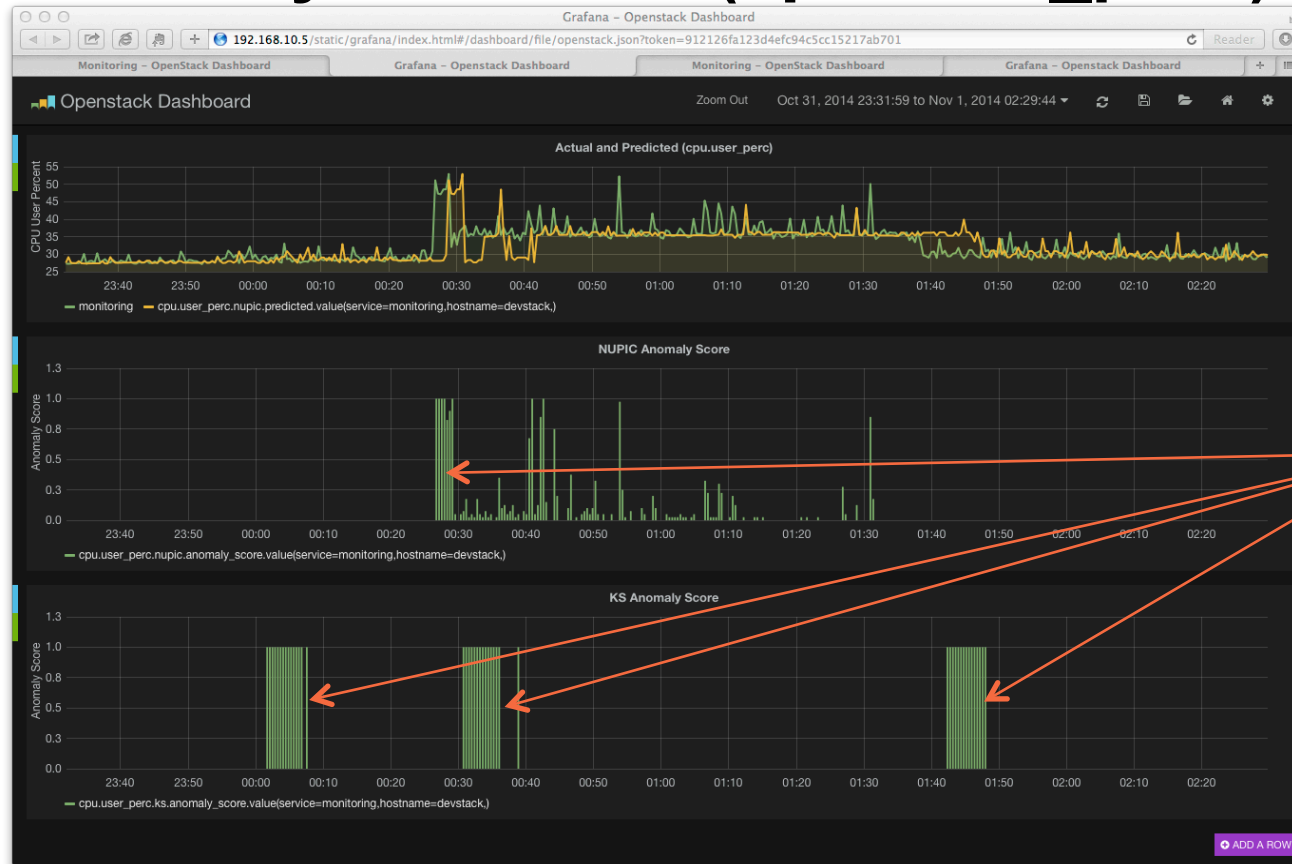
Anomaly Engine

- Consumes metrics from the Kafka metrics topic
- Calculates predicted value and anomaly score (probability of an anomaly)
- Publishes calculated values as metrics to the Kafka metrics topic

Alarms can be created for Anomaly scores



Anomaly Detection (cpu.user_perc)



Anomalies
detected



Current Status



Monasca and Stacktach.v3 is open-sourced in StackForge

Not an OpenStack incubated project, but we are targeting incubation

Metrics, Alarm Definitions, Alarms and Notification Methods completely supported/functional and ready for production deployment

Who is working on it?

- HP
- RackSpace
- IBM

Who is deploying it?

- HP: Public Cloud and Helion distribution
- Time Warner Cable (TWC)
- Workday



Performance (Metrics inserts/sec)



Test Deployment (HP R&D Cluster):

- Three HP Proliant SL390s G7 servers
- InfluxDB cluster

Performance:

- Total end-to-end performance including storage in InfluxDB: Approximately 25K to 30K metrics/second.
- monasca-api: > 50K metrics/sec per single API server.
- monasca-api > 150K metrics/sec for a three node cluster with a load-balancing VIP.

If you need more database performance?

-  is supported. Scales to hundreds of thousands of metrics per second.



Next Steps

Events/StackTach.v3 integration is in progress

Anomaly detection is in progress

Formalize micro-services architecture

- Define message formats
- Define how services are published and registered

Python port is in progress:

- All components Python except for API and Threshold Engine
 - API is 75% ported to Python. Note, Java API is 100% functional
 - Threshold Engine is the only remaining Java component



Call to Action



Looking for contributors

- Monasca Service, StackTach.v3, Events, Anomaly Detection
- Monasca Agent: Help extend with additional services. E.g. Sensor data
- Help Integrate, Deploy, Test and Performance benchmarking

More info:

- Launchpad: <https://launchpad.net/monasca>
- Wiki: <https://wiki.openstack.org/wiki/Monasca>
- IRC: #openstack-monasca

Monasca development environment:

- monasca-vagrant (<https://github.com/stackforge/monasca-vagrant>): A turn-key development environment that installs Monasca and a Devstack VM
- Newly upgraded to use Ansible





Demo



StackTach.v3



Demo Recap



What did we just show?

1. An OpenStack Notification is sent to the Monasca Events API
2. The API publishes the notification to the Kafka raw events topic
3. The Transform Engine consumes, transforms (using StackTach Distiller) and publishes the event to the transformed event topic
4. The Events Engine consumes, adds to the StackTach Winchester Pipeline.
5. If the notification is a “compute.instance.create.end” event the Winchester pipeline handler fires, calculates the delta, and publishes to the Kafka metrics topic
6. The metric can then be alarmed on in the metrics pipeline or visualized



Thank you

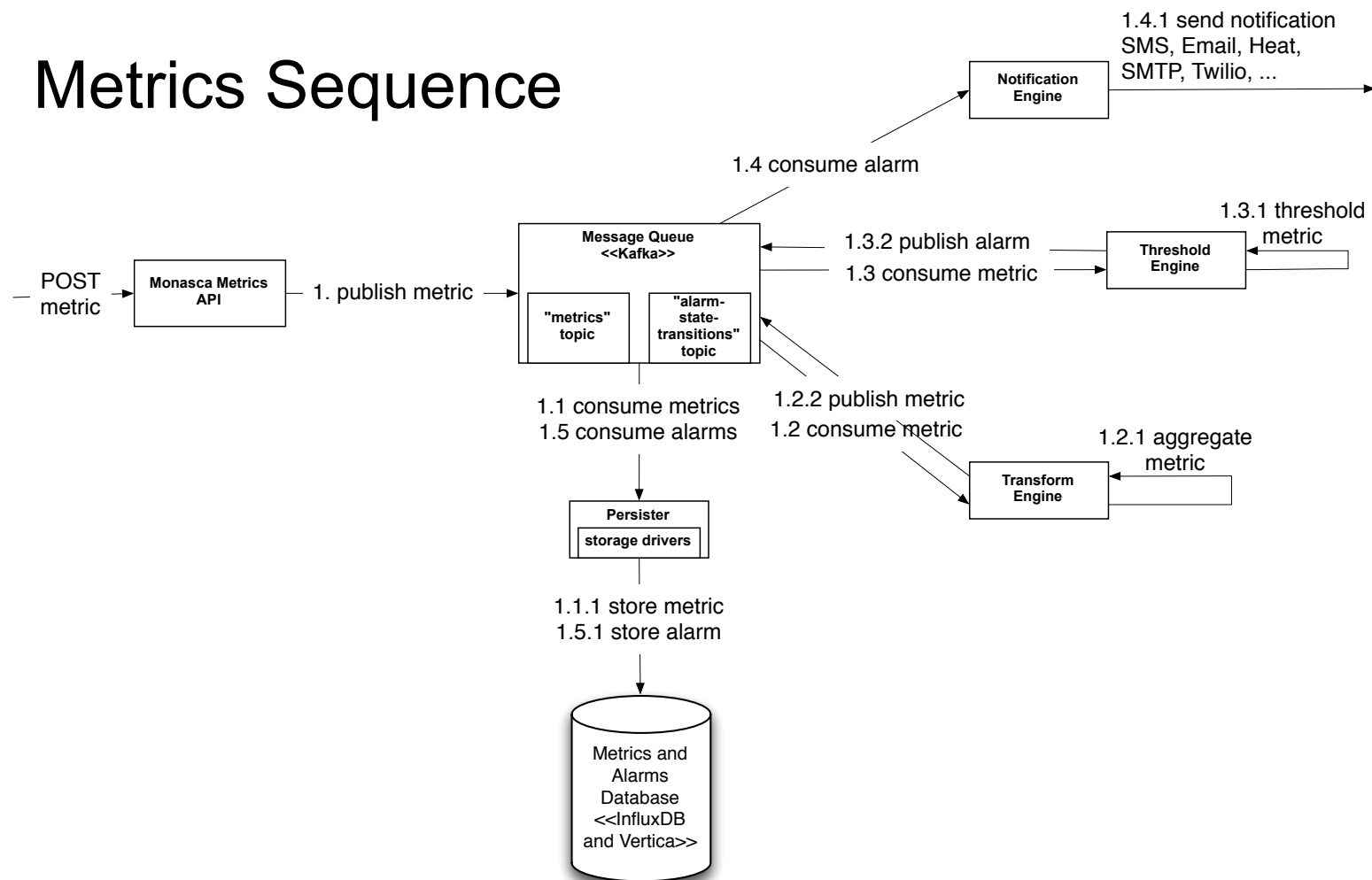


The OpenStack word mark and the Square O Design, together or apart, are trademarks or registered trademarks of OpenStack Foundation in the United States and other countries, and are used with the OpenStack Foundation's permission.

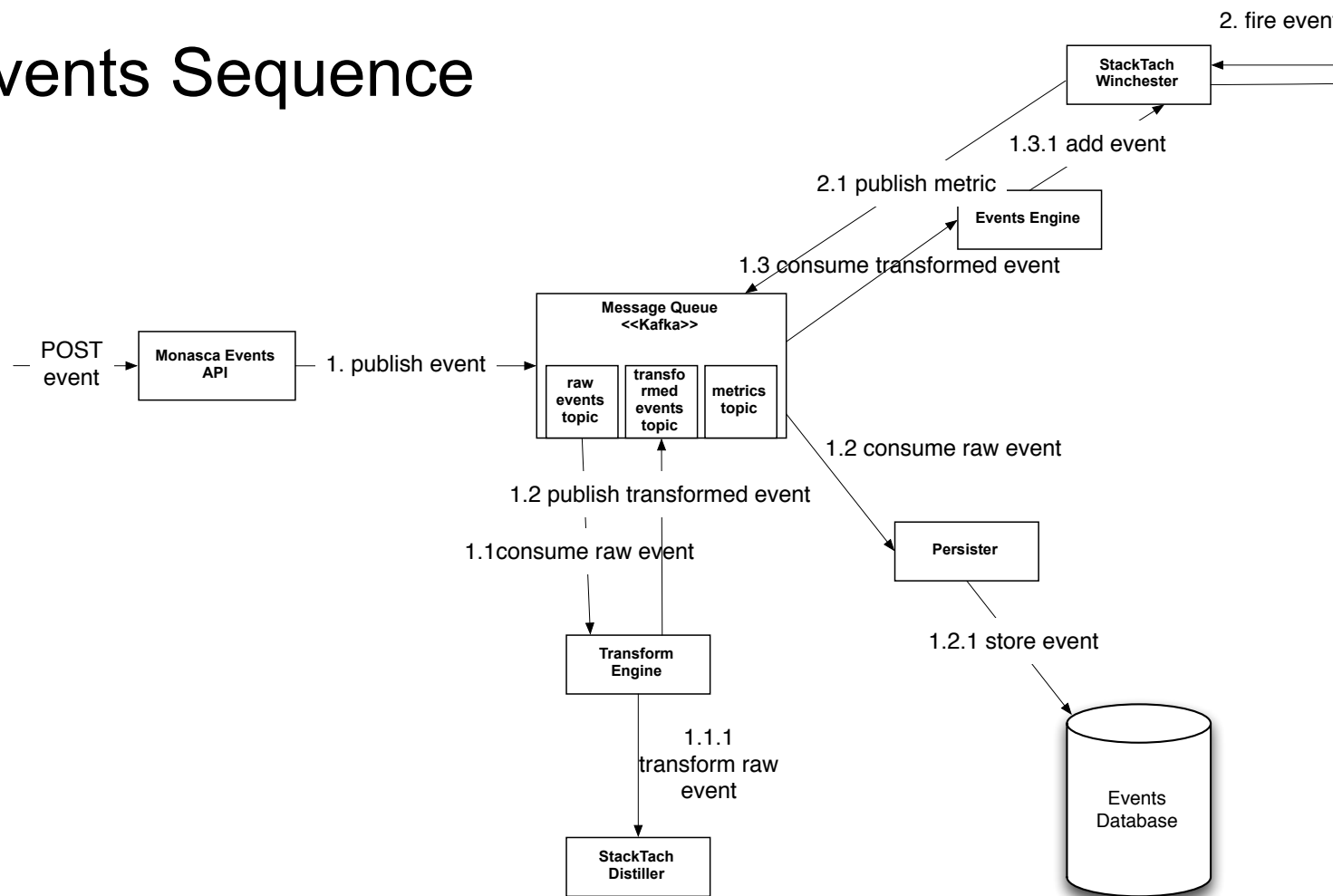
© Copyright 2014 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice.



Metrics Sequence



Events Sequence



Monitoring Overview (All Services Healthy)



The screenshot displays the OpenStack Monitoring Dashboard in a web browser. The browser's address bar shows the URL `192.168.10.5/monitoring/`. The dashboard header includes the OpenStack logo, a user menu for 'mini-mon', and a 'Sign Out' button. A left-hand navigation menu lists 'Project', 'Identity', and 'Monitoring', with 'Monitoring' expanded to show 'Overview', 'Alarm Definitions', 'Alarms', and 'Notifications'. The main content area is titled 'Monitoring' and contains two tabs: 'All Alarms' and 'Dashboard'. Under the 'Dashboard' tab, there are two sections: 'OpenStack Services' and 'Servers'. The 'OpenStack Services' section displays five service status boxes, each with a green checkmark indicating a healthy state: 'cinder_api', 'nova_api', 'monitoring', 'swift_api', and 'glance_api'. The 'Servers' section displays two server status boxes, also with green checkmarks: 'devstack' and 'mini-mon'.



Monitoring Overview (nova-api down)



The screenshot shows the OpenStack Monitoring Dashboard in a web browser. The browser's address bar displays the URL `192.168.10.5/monitoring/`. The dashboard header includes the OpenStack logo, a 'mini-mon' dropdown menu, and a 'Sign Out' button. On the left, a sidebar menu lists 'Project', 'Identity', and 'Monitoring' (which is expanded to show 'Overview', 'Alarm Definitions', 'Alarms', and 'Notifications'). The main content area is titled 'Monitoring' and features two tabs: 'All Alarms' and 'Dashboard'. Under the 'OpenStack Services' section, five service status boxes are shown: 'cinder_api' (green checkmark), 'nova_api' (red minus sign, highlighted with a red border), 'monitoring' (green checkmark), 'swift_api' (green checkmark), and 'glance_api' (green checkmark). Below this, the 'Servers' section shows two server status boxes: 'devstack' (red minus sign, highlighted with a red border) and 'mini-mon' (green checkmark).



List Alarm Definitions



Alarm Definitions - OpenStack Dashboard

192.168.10.5/monitoring/alarmdefs/

Reader

openstack

mini-mon

mini-mon Sign Out

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

Alarm Definitions

Filter

Filter

Create Alarm Definition

Delete Alarm Definitions

<input type="checkbox"/>	Name	Description	Notifications Enabled	Actions
<input type="checkbox"/>	API Status	API Status	True	Edit Alarm Definition
<input type="checkbox"/>	Disk Space	Disk space utilization percentage	True	Edit Alarm Definition
<input type="checkbox"/>	Compute Instance Create Time	Compute Instance Create Time	True	Edit Alarm Definition
<input type="checkbox"/>	CPU User Percent	CPU User Percent	True	Edit Alarm Definition

Displaying 4 items



Create Alarm Definition (Disk Space)



Alarm Definitions - OpenStack Dashboard

192.168.10.5/monitoring/alarmdefs/

openstack mini-mon

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

Create Alarm Definition

Description:

The Name field is used to identify the notification method.

The Expression field which if true, triggers a notification to be sent. See [Alarm Expressions](#) for how to write an expression.

The Alarm Actions field contains the list of Notification that should be sent when transitioning to an ALARM state.

Name *

Disk Space Utilization

Expression *

avg disk_space_utilization_perc > 85

Add a dimension

Matching Metrics

name	dimensions
disk_space_utilization_perc	("device":"/dev/sda1","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/dev/sda1","hostname":"mini-mon","service":"monitoring")
disk_space_utilization_perc	("device":"/tmpfs","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/tmpfs","hostname":"mini-mon","service":"monitoring")
disk_space_utilization_perc	("device":"/udev","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/udev","hostname":"mini-mon","service":"monitoring")
disk_space_utilization_perc	("device":"/v-csc-1","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/v-csc-3","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/v-csr-2","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant","hostname":"mini-mon","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant-cache","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant-cache","hostname":"mini-mon","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant_home","hostname":"devstack","service":"monitoring")
disk_space_utilization_perc	("device":"/vagrant_home","hostname":"mini-mon","service":"monitoring")

Apply function to metrics

☒ individually

← Matching metrics



Edit Alarm Definition (API Status)

The screenshot shows a web browser window titled "Alarm Definitions - OpenStack Dashboard" with the URL "192.168.10.5/monitoring/alarmsdefs/". A modal window titled "Edit Alarm Definition" is open, displaying the configuration for an alarm named "API Status".

Description:

- The Name field is used to identify the notification method.
- The Expression field which if true, triggers a notification to be sent. See [Alarm Expressions](#) for how to write an expression.
- The Alarm Actions field contains the list of Notification that should be sent when transitioning to an ALARM state.

Name *

API Status

Expression *

avg(http_status()) > 0

Apply function to metrics

☒ individually
☐ as a group

Description

API Status

Severity

Critical

☒ Notifications Enabled

Notifications

Name

Roland Hochmuth

+ Add more



List All Alarms



Alarms – OpenStack Dashboard

192.168.10.5/monitoring/alarms/alarm/all/

Reader

openstack

mini-mon

mini-mon Sign Out

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

All Alarms

Alarms

Filter

Filter

Delete Alarms

	Status	Metric Name	Metric Dimensions	Definition	Actions
<input type="checkbox"/>	✓	http_status	url=http://localhost:8081/healthcheck,hostname=mini-mon,service=monitoring	API Status	Graph Metric
<input type="checkbox"/>	✗	http_status	url=http://localhost:8774/v2.0,hostname=devstack,service=nova_api	API Status	Graph Metric
<input type="checkbox"/>	✓	http_status	url=http://localhost:8776/v2.0,hostname=devstack,service=cinder_api	API Status	Graph Metric
<input type="checkbox"/>	✓	http_status	url=http://localhost:8080/healthcheck,hostname=devstack,service=swift_api	API Status	Graph Metric
<input type="checkbox"/>	✓	http_status	url=http://localhost:9292,hostname=devstack,service=glance_api	API Status	Graph Metric
<input type="checkbox"/>	✓	disk_space_utilization_perc	device=v-csr-2,hostname=devstack,service=monitoring	Disk Space	Graph Metric
<input type="checkbox"/>	✓	disk_space_utilization_perc	device=udev,hostname=devstack,service=monitoring	Disk Space	Graph Metric



List Alarm History



Alarm History - OpenStack Dashboard

192.168.10.5/monitoring/alarms/history/036cba6b-ad9d-424c-abfc-854184a8d466/9a11447c-feaf-40be-8b71-cdb573cd179b

Reader

openstack

mini-mon

Sign Out

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

Alarm History

Name	Old State	New State	Timestamp	Reason
036cba6b-ad9d-424c-abfc-854184a8d466	ALARM	OK	2014-11-04T17:46:30.000Z	The alarm threshold(s) have not been exceeded
036cba6b-ad9d-424c-abfc-854184a8d466	OK	ALARM	2014-11-04T17:44:30.000Z	Thresholds were exceeded for the sub-alarms: [avg(http_status) > 0.0]
036cba6b-ad9d-424c-abfc-854184a8d466	ALARM	OK	2014-11-04T17:42:30.000Z	The alarm threshold(s) have not been exceeded
036cba6b-ad9d-424c-abfc-854184a8d466	OK	ALARM	2014-11-04T17:40:30.000Z	Thresholds were exceeded for the sub-alarms: [avg(http_status) > 0.0]
036cba6b-ad9d-424c-abfc-854184a8d466	ALARM	OK	2014-11-04T17:35:30.000Z	The alarm threshold(s) have not been exceeded
036cba6b-ad9d-424c-abfc-854184a8d466	OK	ALARM	2014-11-04T17:31:30.000Z	Thresholds were exceeded for the sub-alarms: [avg(http_status) > 0.0]
036cba6b-ad9d-424c-abfc-854184a8d466	ALARM	OK	2014-11-04T17:19:30.000Z	The alarm threshold(s) have not been exceeded
036cba6b-ad9d-424c-abfc-854184a8d466	OK	ALARM	2014-11-04T16:50:30.000Z	Thresholds were exceeded for the sub-alarms: [avg(http_status) > 0.0]
036cba6b-ad9d-424c-abfc-854184a8d466	UNDETERMINED	OK	2014-11-04T08:07:45.000Z	The alarm threshold(s) have not been exceeded

Show Alarm Definition



openstack

mini-mon

mini-mon Sign Out

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

Alarm Definition Details

Info

Name

API Status

Description

API Status

Expression

avg(http_status{}) > 0

Severity

CRITICAL

Notifications Enabled

True

Notifications

Name	Type	Address
------	------	---------



List Notifications



Monitoring - OpenStack Dashboard

192.168.10.5/monitoring/notifications/

openstack mini-mon Sign Out

Project

Identity

Monitoring

Overview

Alarm Definitions

Alarms

Notifications

Notifications

Filter Filter + Create Notification

<input type="checkbox"/>	Name	Type	Address	Actions
<input type="checkbox"/>	Roland Hochmuth	EMAIL	roland.hochmuth@hp.com	Edit Notification
<input type="checkbox"/>	Sandy Walsh	EMAIL	sandy.walsh@rackspace.com	Edit Notification
<input type="checkbox"/>	Tong Li	EMAIL	litong01@us.ibm.com	Edit Notification

Displaying 3 items



Grafana



Grafana (compute_instance_create_time)

