

A Tool for Understanding OpenStack Cloud Performance using StackTach and the OpenStack Notification System

Bruno Grazioli

Outline

Zürich University
of Applied Sciences



School of
Engineering



InIT Institute of Applied
Information Technology

- The problem
- Solution for obtaining the appropriate data
 - Openstack notification system
- Working with Openstack notifications
 - StackTach
- Demo of the tool

The problem

- The standard Openstack tools do not provide a way for the operator to know what is the user experience
 - To understand Openstack API performance
- Use Cases
 - Track the average time for an operation to be performed
 - Identify possible bottlenecks
 - Know the fail rate for a certain operation
 - ...

How do we achieve this?

- By collecting operation request information and tracking its status
- We first looked at log files to track a given request 
 - distributed across multiple servers
 - sometimes not clear when a particular operation has terminated
- Openstack notification system 
 - Part of every Openstack Deployment
 - Ceilometer consumes these notifications
 - A system which provides a single endpoint (queue in the message broker) where messages are sent
 - JSON formatted messages containing granular information for a given operation
 - Quite verbose

Enabling Openstack notification system

- Add the following lines to nova - or to any other service - configuration file

```
notification_driver=messaging
notification_topics=notifications
# Specific for nova.conf
notify_on_state_change=vm_and_task_state
```

- A new queue will be created in the host if necessary with the name given in notification_topics
- Examples of Nova event types
 - compute.instance.create.{start,error,end}
 - compute.instance.delete.start/.end
 - compute.instance.rebuild.start/.end
 - compute.instance.exists
 - compute.instance.update

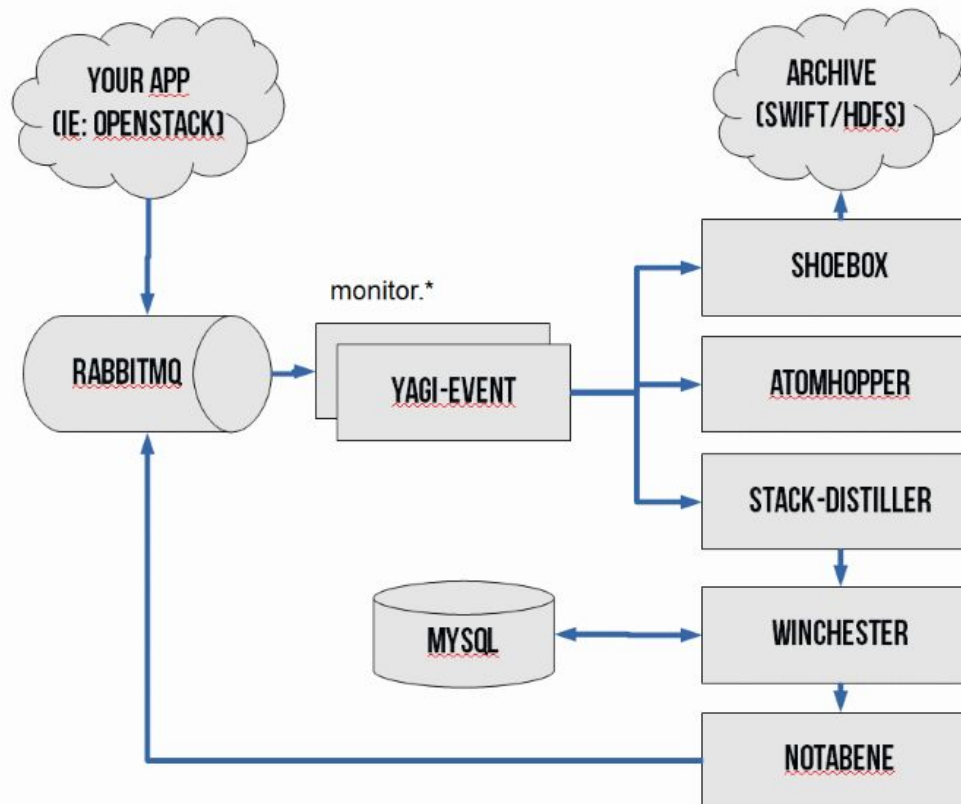
Example of a notification message

```
{
  '_context_project_id': u'7c150a59fe714e6f9263774af9688f0e',
  '_context_request_id': u'req-d68b36e0-9233-467f-9afb-d81435d64d66',
  '_context_timestamp': u'2012-05-08T20:23:41.425105',
  '_context_user_id': u'1e3ce043029547f1a61c1996d1a531a2',
  ''event_type': u'compute.instance.create.end'',
  'message_id': u'dae6f69c-00e0-41c0-b371-41ec3b7f4451',
  'priority': u'INFO',
  'timestamp': u'2012-05-08 20:23:48.028195',
  'payload': {
    'created_at': u'2012-05-08 20:23:41',
    'display_name': u'testme',
    'image_ref_url': u'http://10.0.2.15:9292/images/UUID',
    'instance_id': u'9f9d01b9-4a58-4271-9e27-398b21ab20d1',
    ...}
}
```

Source: <http://stacktach.com/about.html>

- Comprises of several services focused on analysis or notification
 - Focused on stream processing
 - consuming
 - filtering
 - processing
 - storing
- Decoupled from Openstack but design within Openstack context
 - it can be used for any application
- Lightweight

StackTach architecture



Source: <http://stacktach.com/about.html>

The Tool

Zürich University
of Applied Sciences



School of
Engineering

InIT Institute of Applied
Information Technology

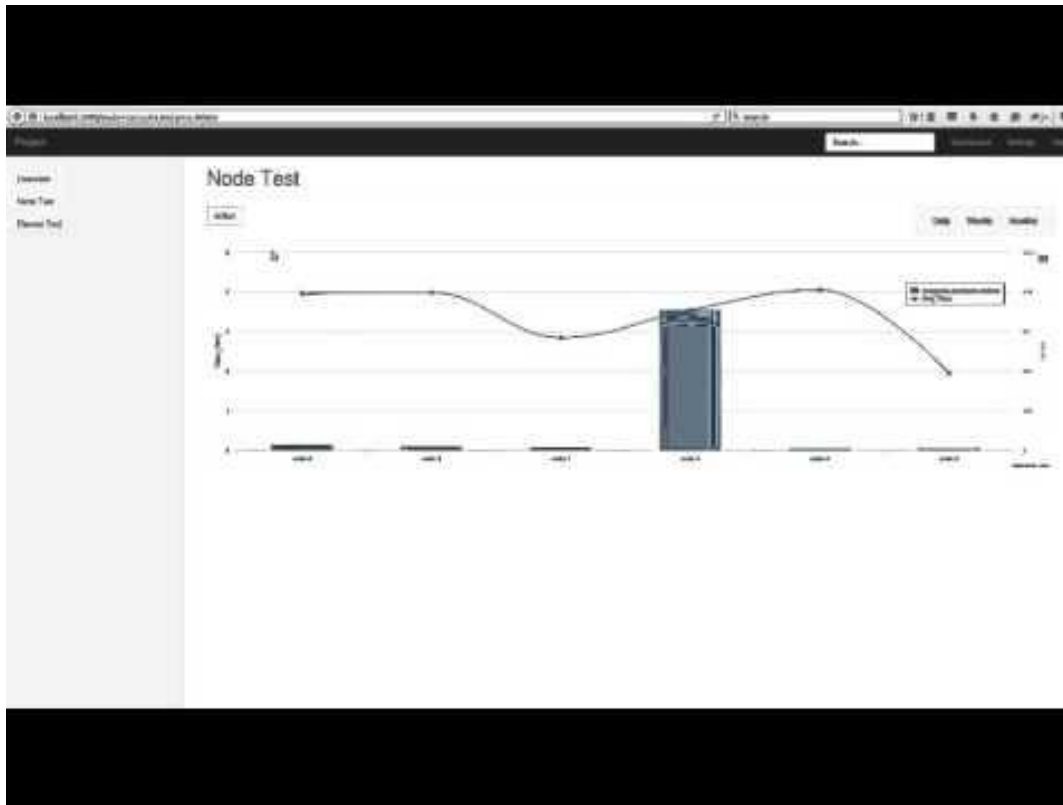
- A web-based application which consumes information stored by StackTach
- Aggregates data in a daily, weekly or monthly time basis
- Breaks down averaged data over the entire cluster
 - By flavor
 - By node
- Presents it in a much more convenient way - charts!

Demo

Zürich University
of Applied Sciences



School of
Engineering
InIT Institute of Applied
Information Technology



Any Questions?

More information at <https://blog.zhaw.ch/icclab/>