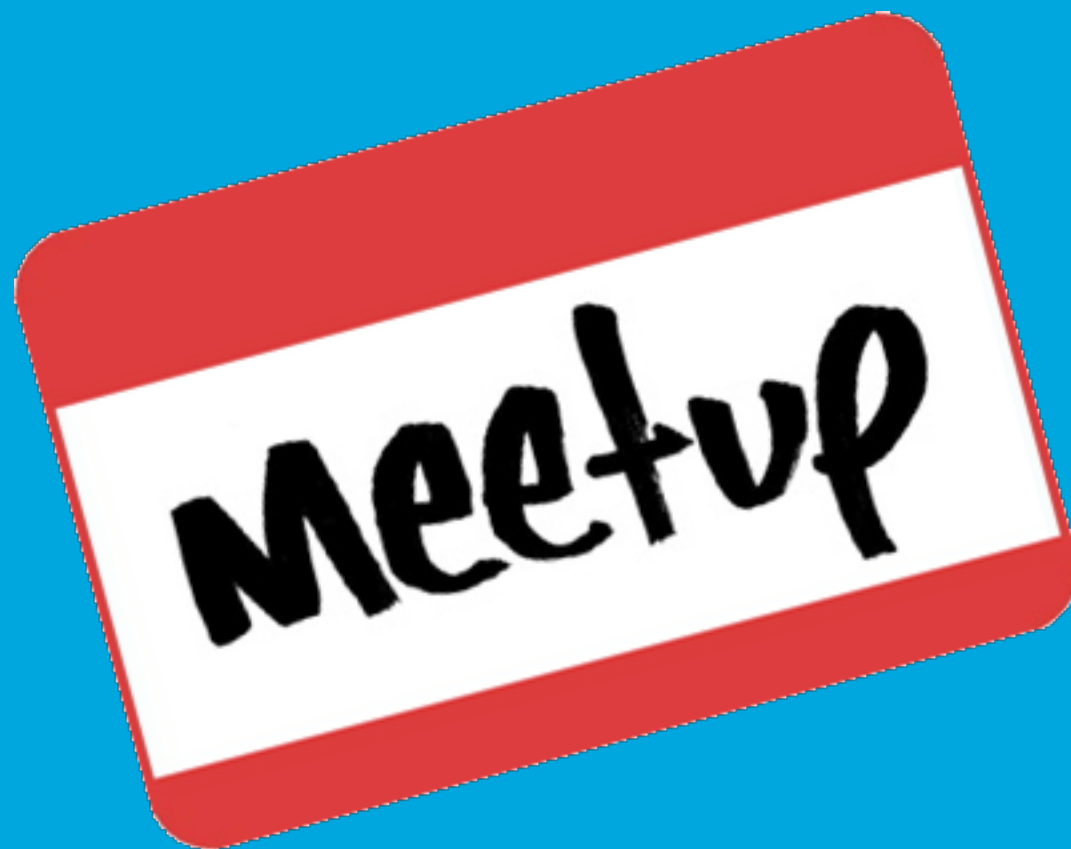


Monasca

monitoring-as-a-service (MONaaS)

OpenStack Austin

Dexter Fryar / July 16, 2015



Outline

About me

Intro to Monasca

Architecture overview

Operational overview

Enough slideware let's see the demo

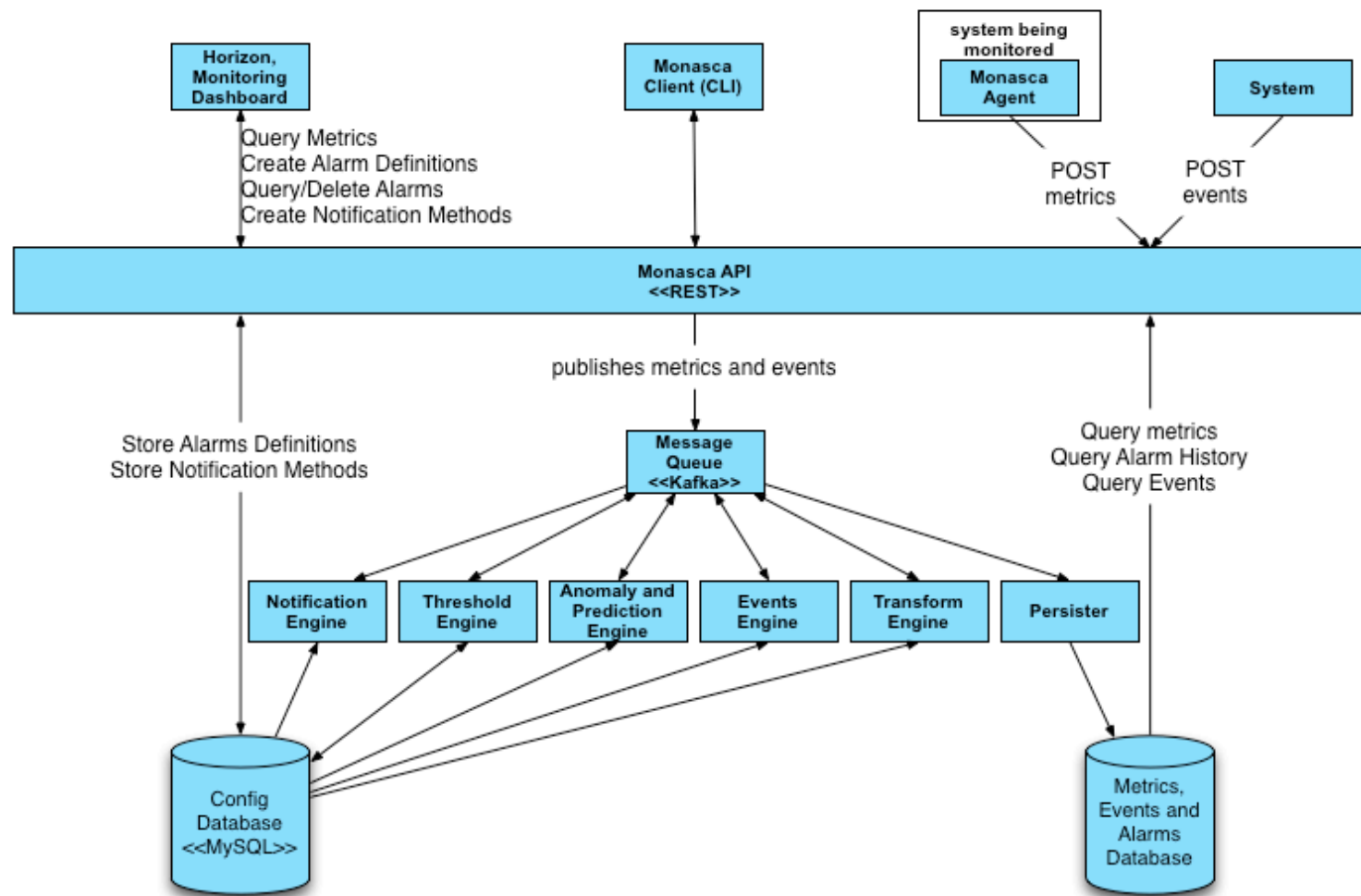
About me

- 15 years at HP in systems software engineering roles
- Worked on internal and external embedded storage subsystems
- Heterogeneous solutions engineering
- Joined HP OpenStack engineering in 2011
- Foundation services – metering and billing
- Currently working on the Monasca project

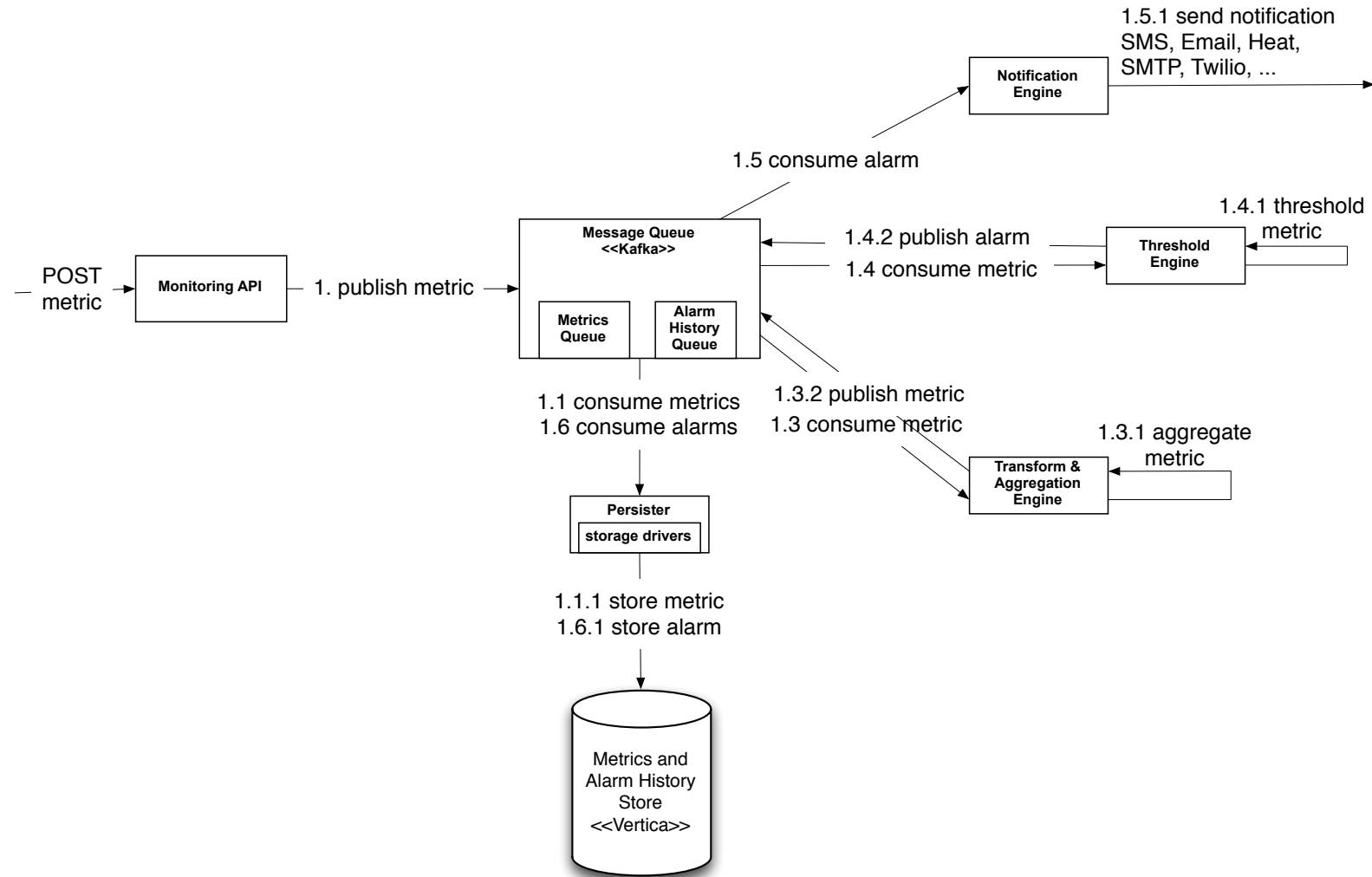
Intro

- Monasca the name
MonitoringAtScale
- How did we get here
public cloud at scale lessons, datadog, openstack
- Goals for Monasca
open-source multi-tenant, highly scalable, performant,
fault-tolerant monitoring-as-a-service solution that
integrates with OpenStack
- Uses
application, service, tenant, component
- Model
REST API for high-speed metrics processing and
querying and has a streaming alarm engine and notification engine
access via → agent, rest API, statsd

Architecture



Metric Posting Sequence



Thank you!

Q&A

<https://wiki.openstack.org/wiki/Monasca>

<https://launchpad.net/monasca>

Core code

<https://github.com/stackforge?query=monasca>

Ancillary code

<https://github.com/hpcloud-mon>

Meetings Tuesdays 10 AM CST

<https://wiki.openstack.org/wiki/Meetings/Monasca>

IRC #openstack-monasca on freenode.net

monasca-*.readthedocs.org

Backup

Stream Data Platform

<http://www.confluent.io/blog/stream-data-platform-1/>



Demo Agenda

1)

- Spin up a VM
- Create a virtualenv
- Install monasca-agent via pip
- configure other nodes to watch new instance
- restart agents
- look at metrics through the cli
- halt new instance
- watch alarms and notifications
- boot instance
- watch alarms and notifications

2)

- install influxdb
- copy plugin for influx
- run setup for plugin
- watch metrics with monasca cli

3)

- metrics posting example through statsd

```
echo "dexter.engineer.person.count.total:1|c|'{first:dexter,last:fryar}'" | nc -u -w0 127.0.0.1 8125
```



Navigation

- agent

```
ps -ef | grep monasca-agent
service monasca-agent status
/etc/monasca/agent/conf.d/*.yaml
/etc/monasca/agent/agent.yaml
/var/log/monasca/*.log
```

- mysql tables

```
mini-mon$ mysql -u root -p
use mon;
show tables;
```

- influx ui

```
192.168.10.4:8083
root / password
```

- horizon ui

```
192.168.10.5
mini-mon / password
```

- monasca file locations

```
logs
    /var/log/monasca
apps
    /opt/monasca
virtualenv
    /opt/monasca
config
    /etc/monasca
```

development

- github / gerrit / launchad
- python / java
- tagged releases
- ansible, galaxy
- pypi
- tarballs.openstack.org