



OPENSTACK[®] AGILITY. RED HAT[®] RELIABILITY.



openstack[®]



Operational Management

How is it really done? And what should
OpenStack do about it?

Anandee Pannu

Senior Principal Product Manager

7 November 2017

Ops Management is hard!

- Look at the number of companies that have gone to providing “managed services” in OpenStack
 - Hard to build a product that meets customers operations management needs

Ops Management - what are we looking at?

1. Not deployment, tons of focus on deployment already
2. Not updates/upgrades
3. The focus is on monitoring the cloud as it is running (which should be the majority of its life!)
 - a. How is it performing (Performance Monitoring)?
 - b. Are there any faults (Availability Monitoring)?
 - c. What has been happening (Logging)?
 - d. How do I see it all (Dashboards)?

Looking at Ops Management - solution focus

- Not component focussed
 - This isn't (just) about telemetry or logging or ElasticSearch
- Focussed on OpenStack **as cloud infrastructure** overall use cases
 - How does one get alarms that result in the right support
 - How does one do root cause analysis
- Packaging, configuration, integration of Ops Management tools and components with OpenStack infrastructure should be out-of-the-box
 - Every deployment should not be a snowflake

Customer Ops Management

1. Most customers have their own operational tools and operational management practices and want to extend them to OpenStack
2. Customers are new to operating on-premise clouds and need guidance on how to operate them
3. Customer operational management needs are at multiple levels and need flexibility from the platform

Monitoring pains - the official word*

1. Operators are generally unhappy with monitoring OpenStack in general
2. There are too many tools
3. There are too many places in OpenStack that require monitoring
4. It's not clear how and what to monitor
5. The amount of monitoring will have a performance impact on the whole cloud
6. How do you learn information about a host / service that needs monitored beforehand?

OpenStack monitoring today

Top Google hit for OpenStack operations!

The screenshot shows the OpenStack website's administrator guides page. The browser address bar displays the URL <https://docs.openstack.org/pike/admin/>. The navigation bar includes the OpenStack logo, a search bar, and menu items for SOFTWARE, USERS, COMMUNITY, MARKETPLACE, EVENTS, LEARN, DOCS, JOIN, and a LOG IN button. The main heading is "OpenStack Pike Administrator Guides", followed by a sub-heading: "This page contains documentation about administering OpenStack services." Below this, there are two columns of links. The left column is titled "Administrator Guides For OpenStack Services" and lists: Telemetry Alarming services (aodh), Key Manager service (barbican), Telemetry Data Collection service (ceilometer), Block Storage service (cinder), Rating service (cloudkitty), Governance service (congress), DNS service (designate), Image service (glance), Orchestration service (heat), Dashboard (horizon), Bare Metal service (ironic), Data Protection Orchestration Service (karbor), Identity service (keystone), Container Infrastructure Management service (magnum), Workflow service (mistral), and Application Catalog service (murano). The right column is titled "Administrator Guides For Common OpenStack Libraries" and lists: oslo.log (A logging configuration library), oslo.messaging (Inter-process communication), oslo.middleware (WSGI middleware library for web service development), and oslo.policy (Common policy enforcement).

← → ↻ 🏠 Secure | <https://docs.openstack.org/pike/admin/> ☆

openstack. 🔍 SEARCH SOFTWARE ▾ USERS ▾ COMMUNITY ▾ MARKETPLACE EVENTS ▾ LEARN ▾ DOCS JOIN ▾ **LOG IN**

OpenStack Pike Administrator Guides

This page contains documentation about administering OpenStack services.

Administrator Guides For OpenStack Services

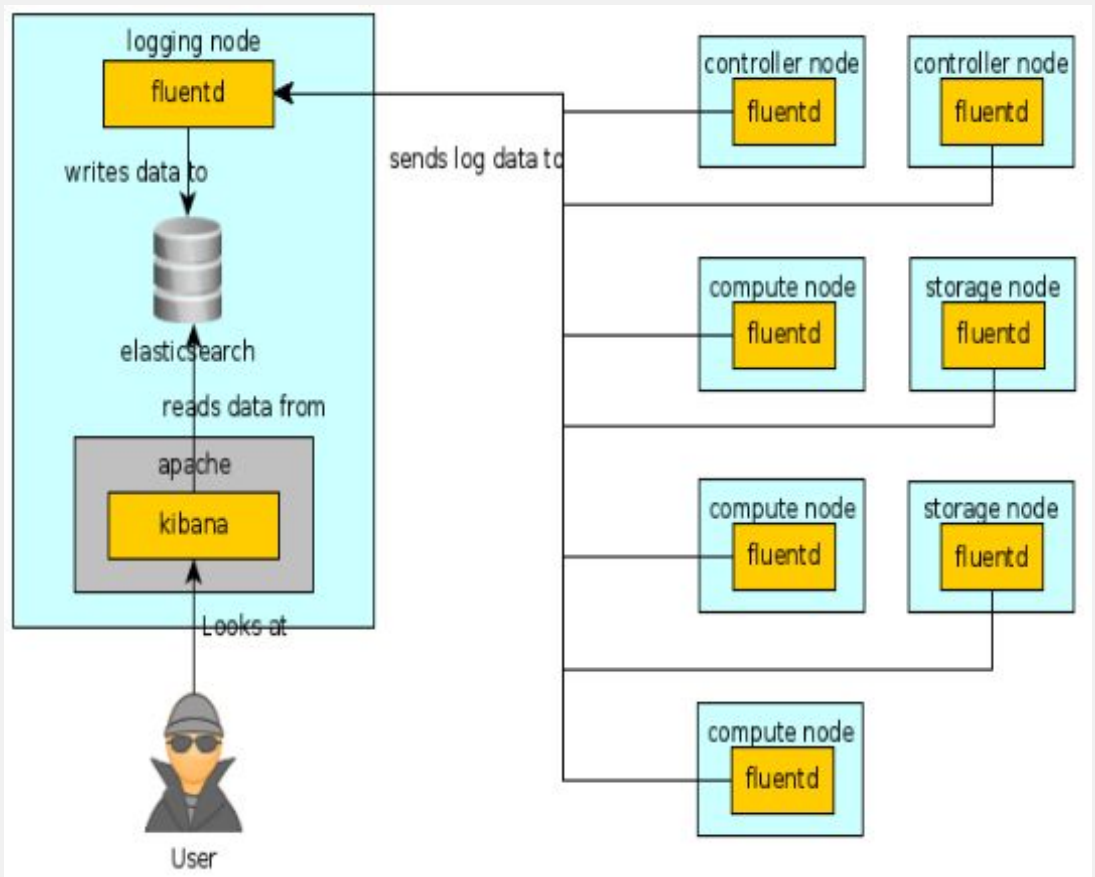
- Telemetry Alarming services (aodh)
- Key Manager service (barbican)
- Telemetry Data Collection service (ceilometer)
- Block Storage service (cinder)
- Rating service (cloudkitty)
- Governance service (congress)
- DNS service (designate)
- Image service (glance)
- Orchestration service (heat)
- Dashboard (horizon)
- Bare Metal service (ironic)
- Data Protection Orchestration Service (karbor)
- Identity service (keystone)
- Container Infrastructure Management service (magnum)
- Workflow service (mistral)
- Application Catalog service (murano)

Administrator Guides For Common OpenStack Libraries

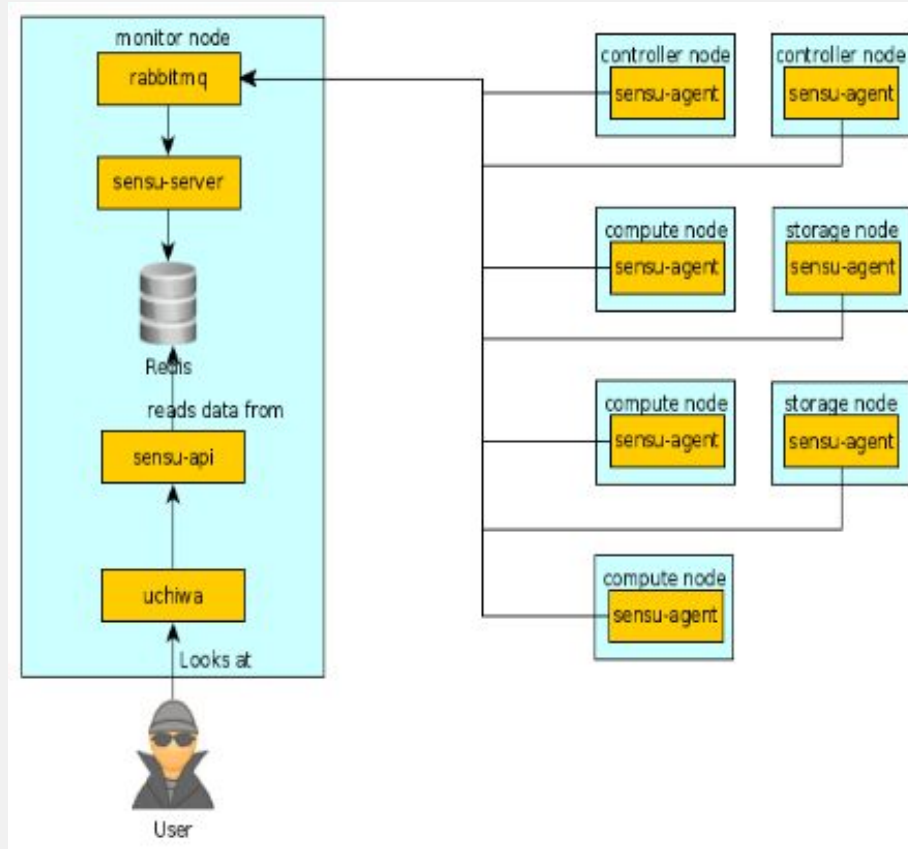
- oslo.log
A logging configuration library.
- oslo.messaging
Inter-process communication.
- oslo.middleware
WSGI middleware library for web service development.
- oslo.policy
Common policy enforcement.

What customers are doing today
(typically)

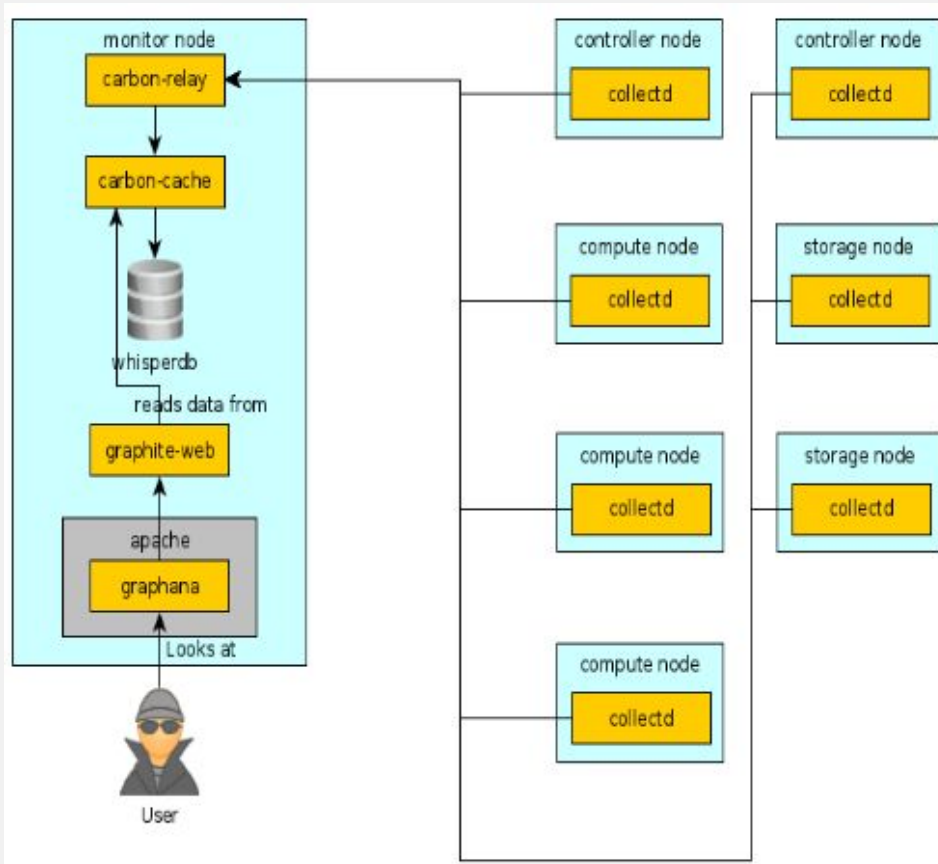
For logging



For availability monitoring



For performance metrics/ monitoring



Some salient features

1. The component solutions are well regarded Open Source projects
2. The components are NOT OpenStack projects
3. For each component there are probably a few more Open Source alternatives

e.g Gnocchi instead of Carbon for collectd metrics collection

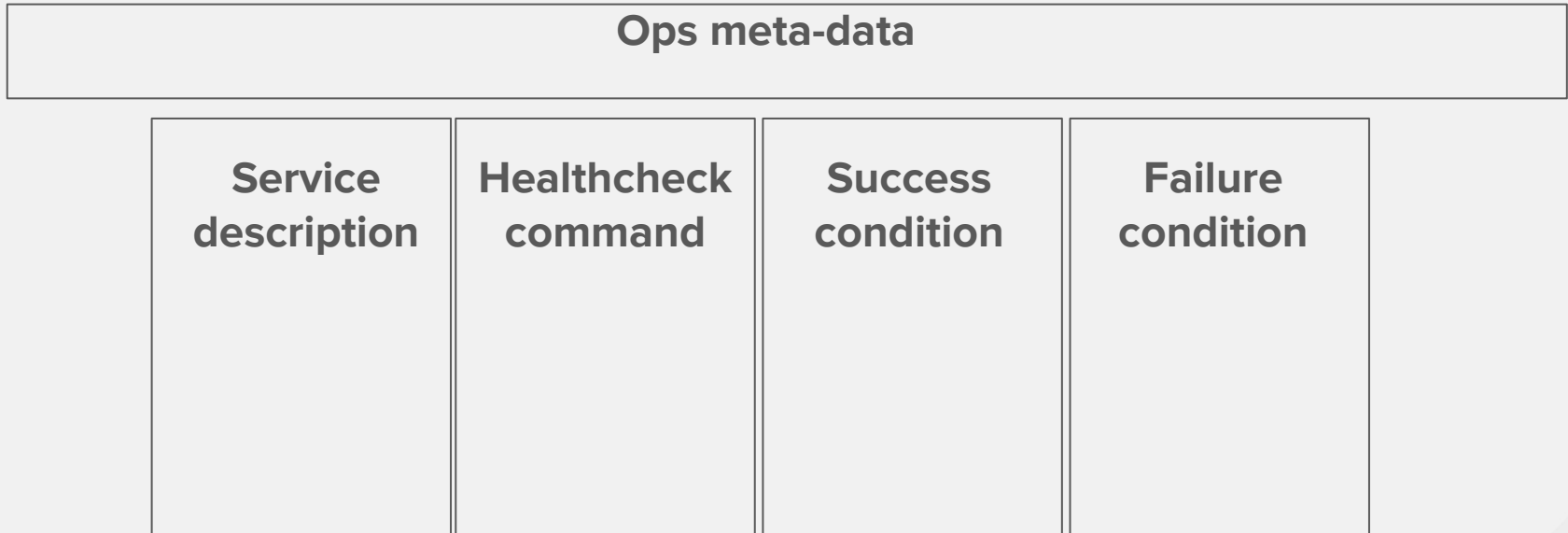
4. Every customer will have a preference for the projects they use

How does OpenStack provide a consistent framework for operations tools?

The proposed solution

1. Directory of monitoring checks in a deployed environment
 - a. Services monitored (higher level than OpenStack services)
 - b. Monitoring check procedure
 - c. Optional parameters
2. Framework and APIs for multiple levels of integration with monitoring components

Monitoring service based meta-data directory



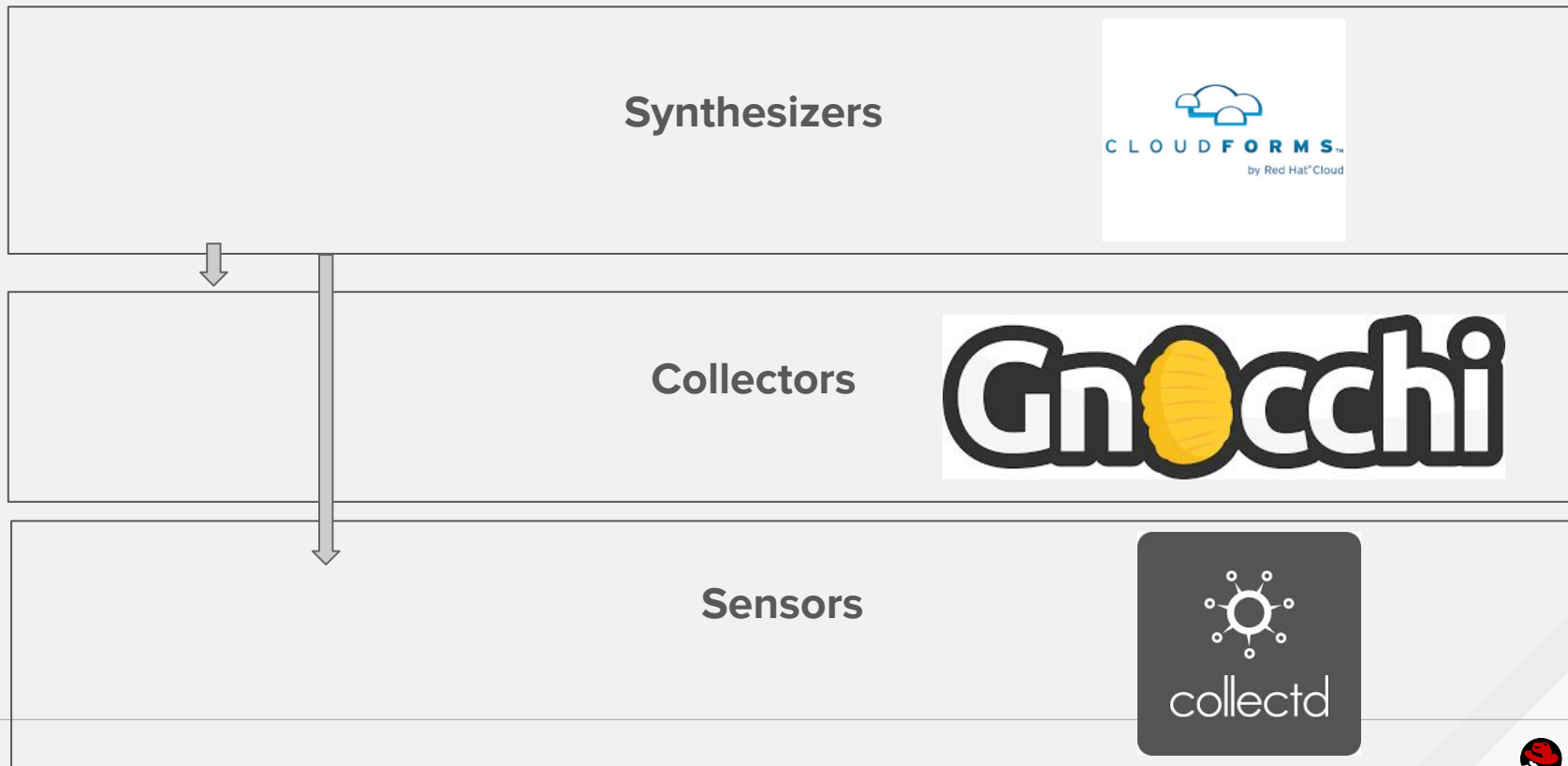
Monitoring: 3 level framework and APIs

Synthesizers

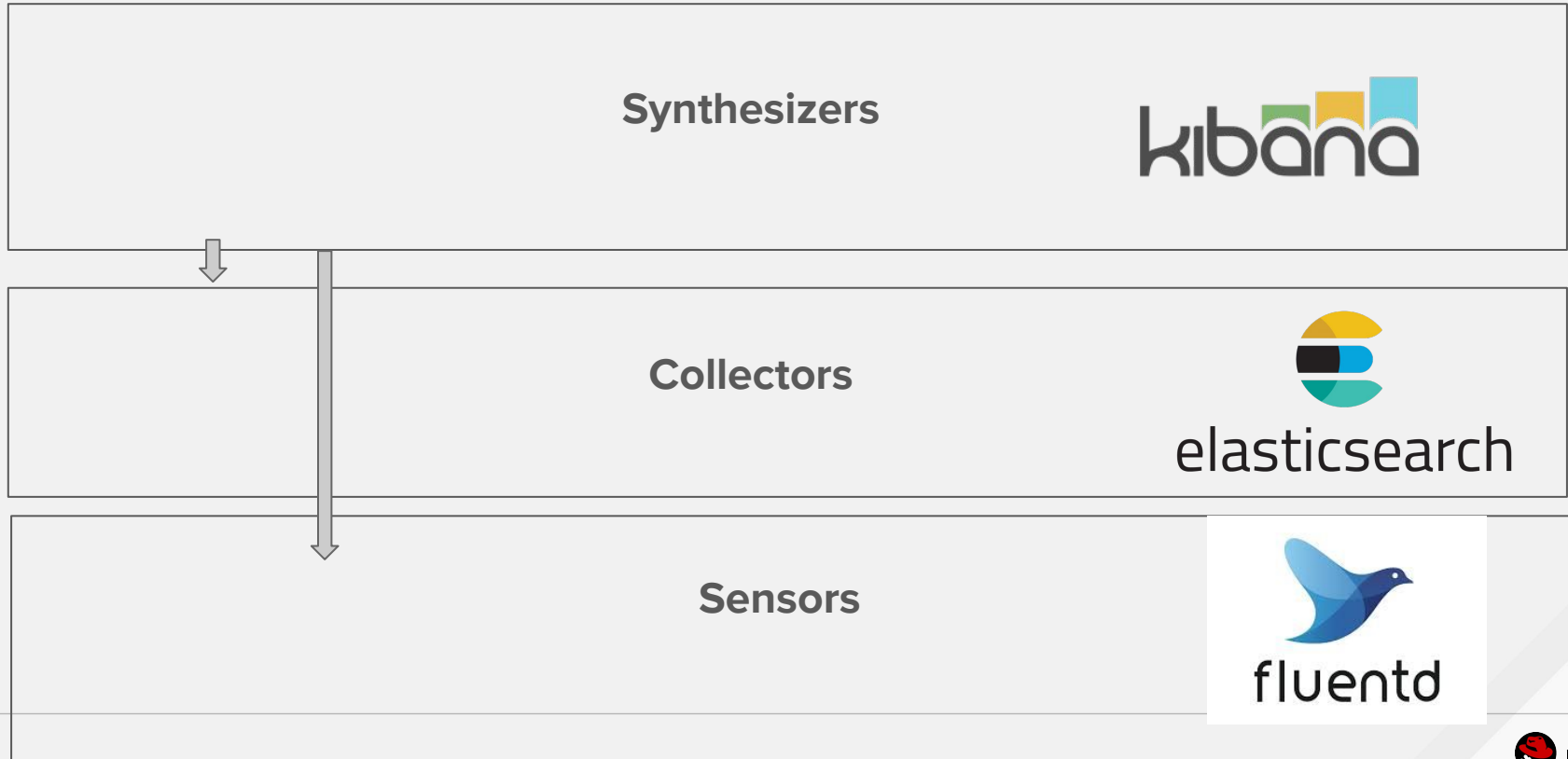
Collectors

Sensors

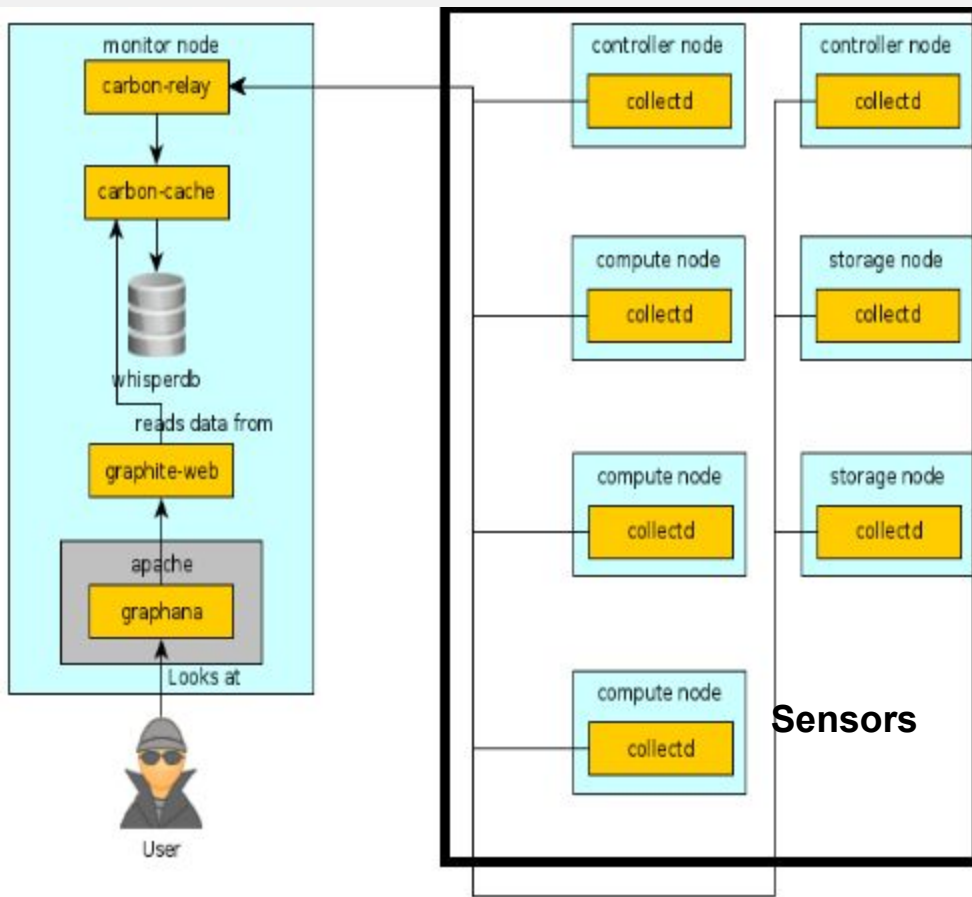
Monitoring: 3 level framework and APIs



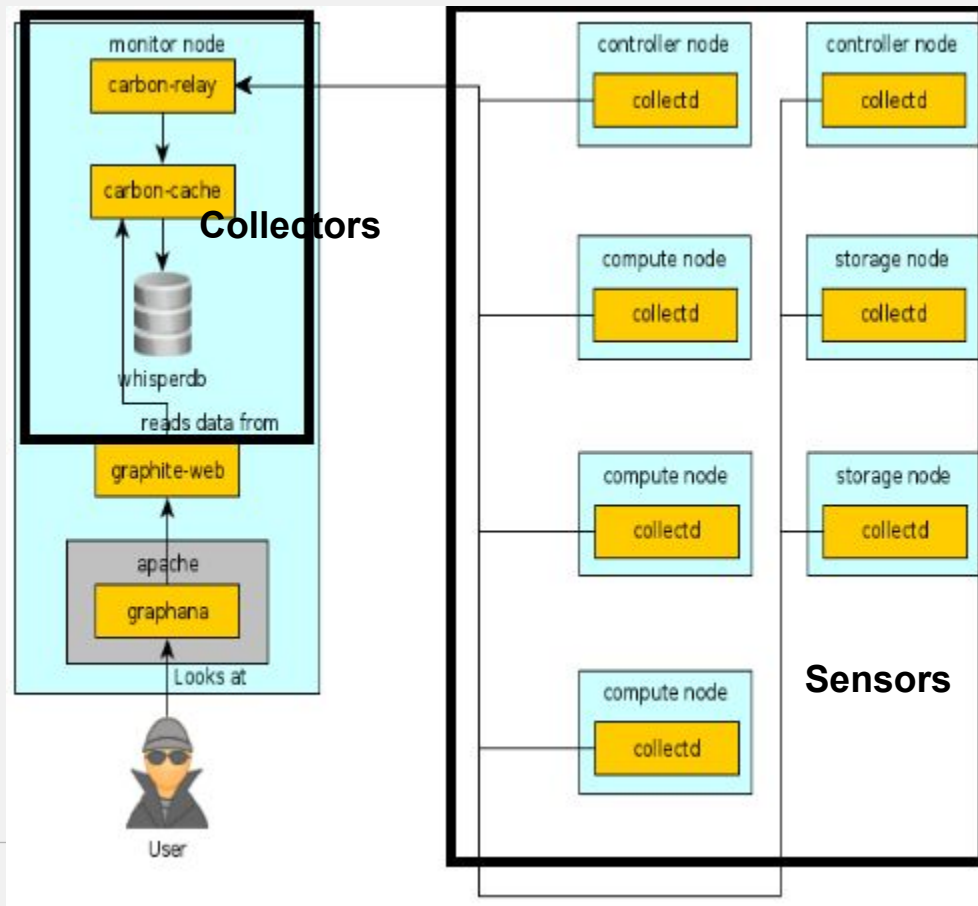
Monitoring: 3 level framework and APIs



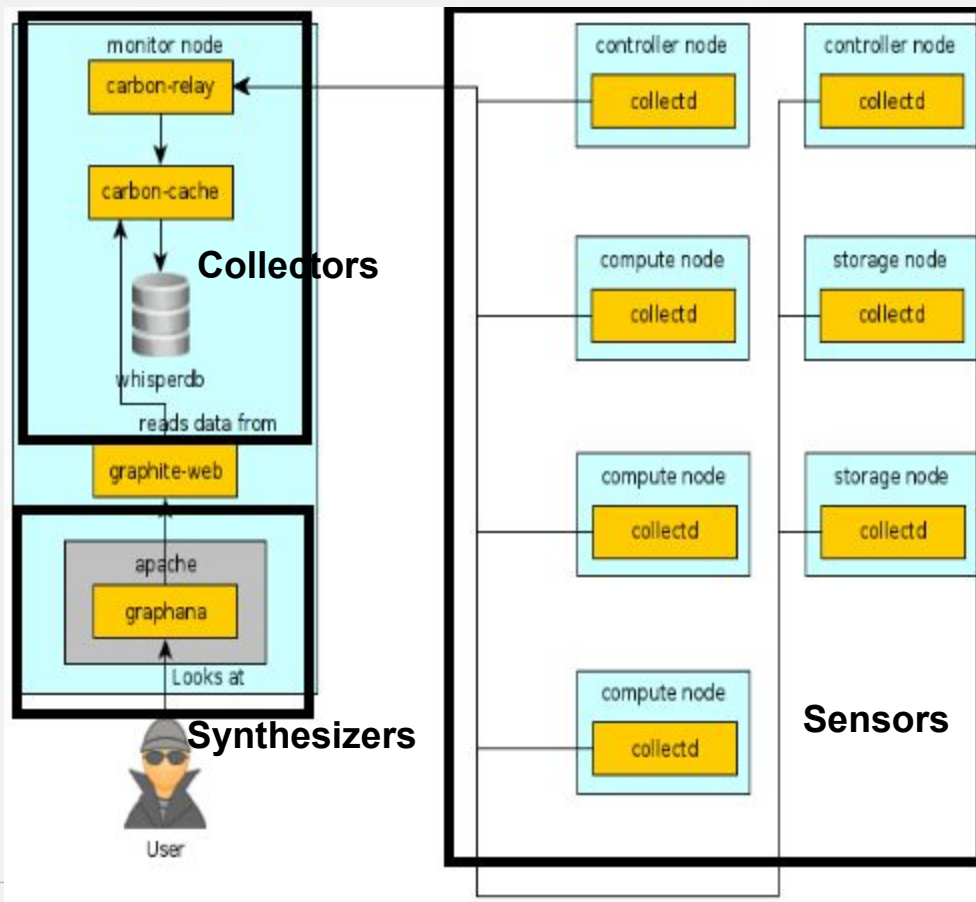
Monitoring: 3 level framework and APIs



Monitoring: 3 level framework and APIs



Monitoring: 3 level framework and APIs



Advantages

1. For each deployment, what is being monitored can be accessed from a directory
 - a. at the level the user desires
 - b. orchestrated through HEAT/ other orchestration from lower level primitives
2. Monitoring components can now be treated like plug-ins
3. Uniform interface for monitoring
4. User tools can either use intermediate tools like time-series databases or go directly to the source
5. Operational tools can be deployed as a “role”



THANK YOU



plus.google.com/+RedHat



linkedin.com/company/red-hat



youtube.com/user/RedHatVideos



facebook.com/redhatinc



twitter.com/RedHatNews