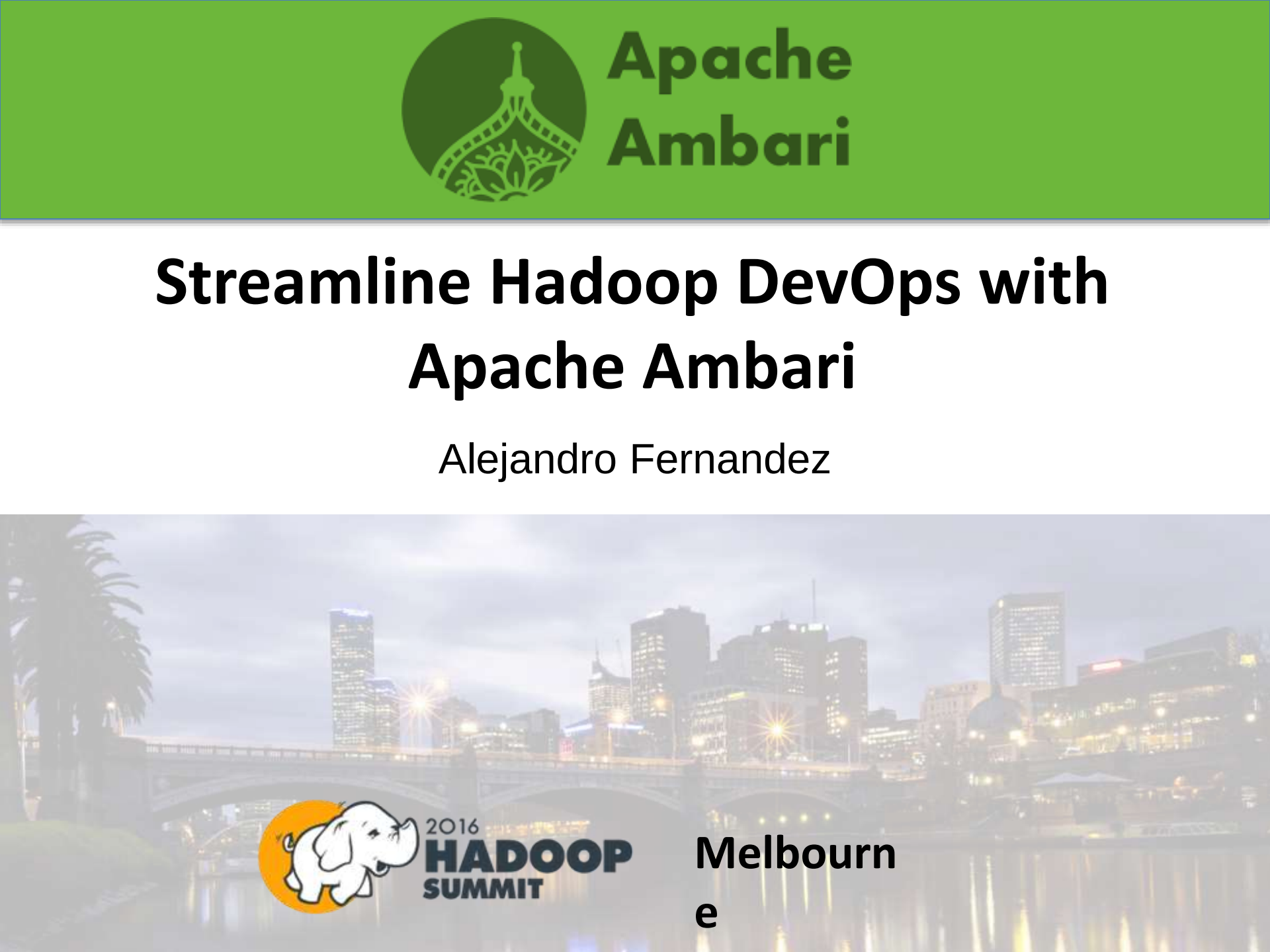


Streamline Hadoop DevOps with Apache Ambari

Alejandro Fernandez

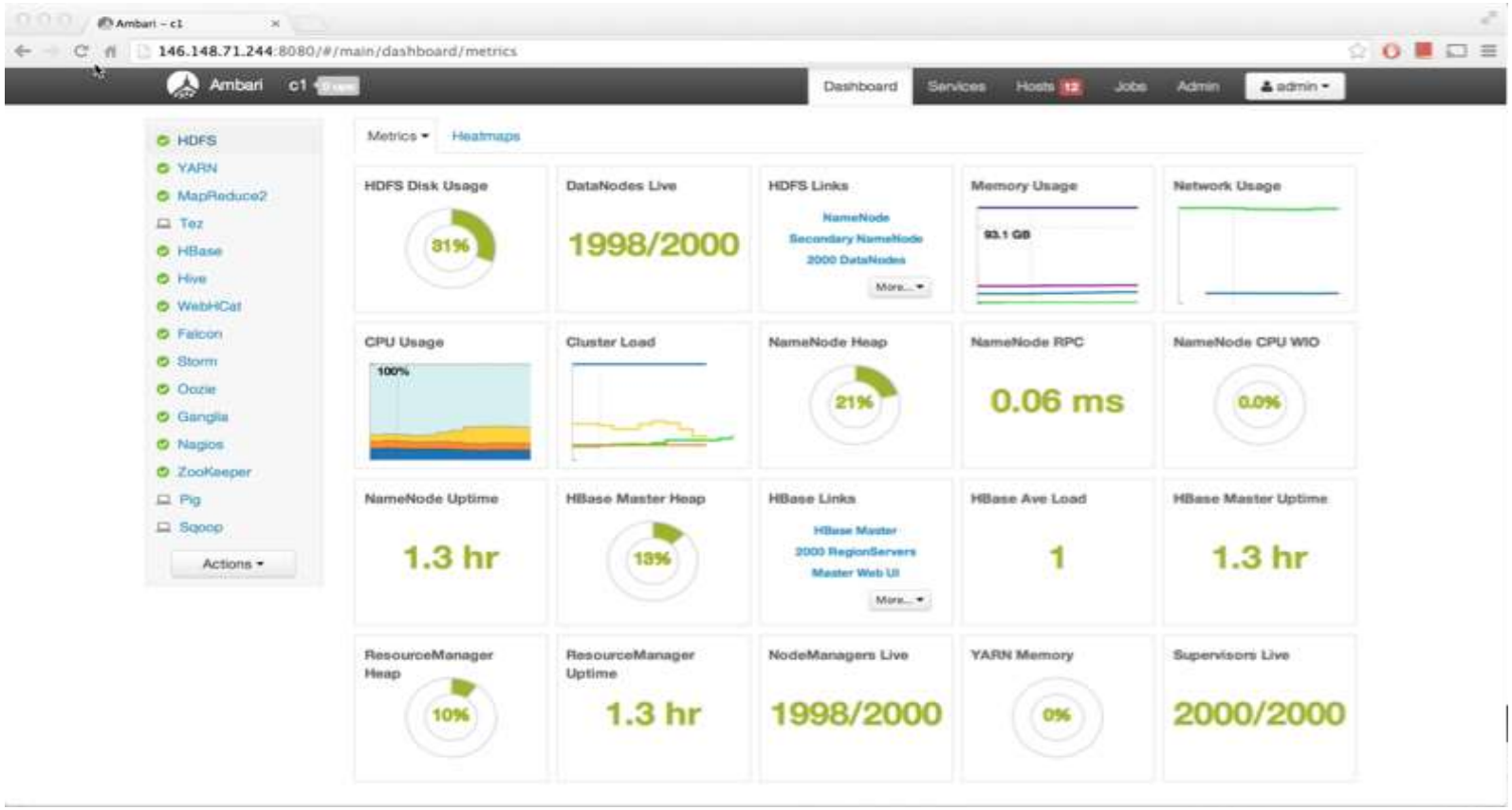
Melbourn
e

Speaker

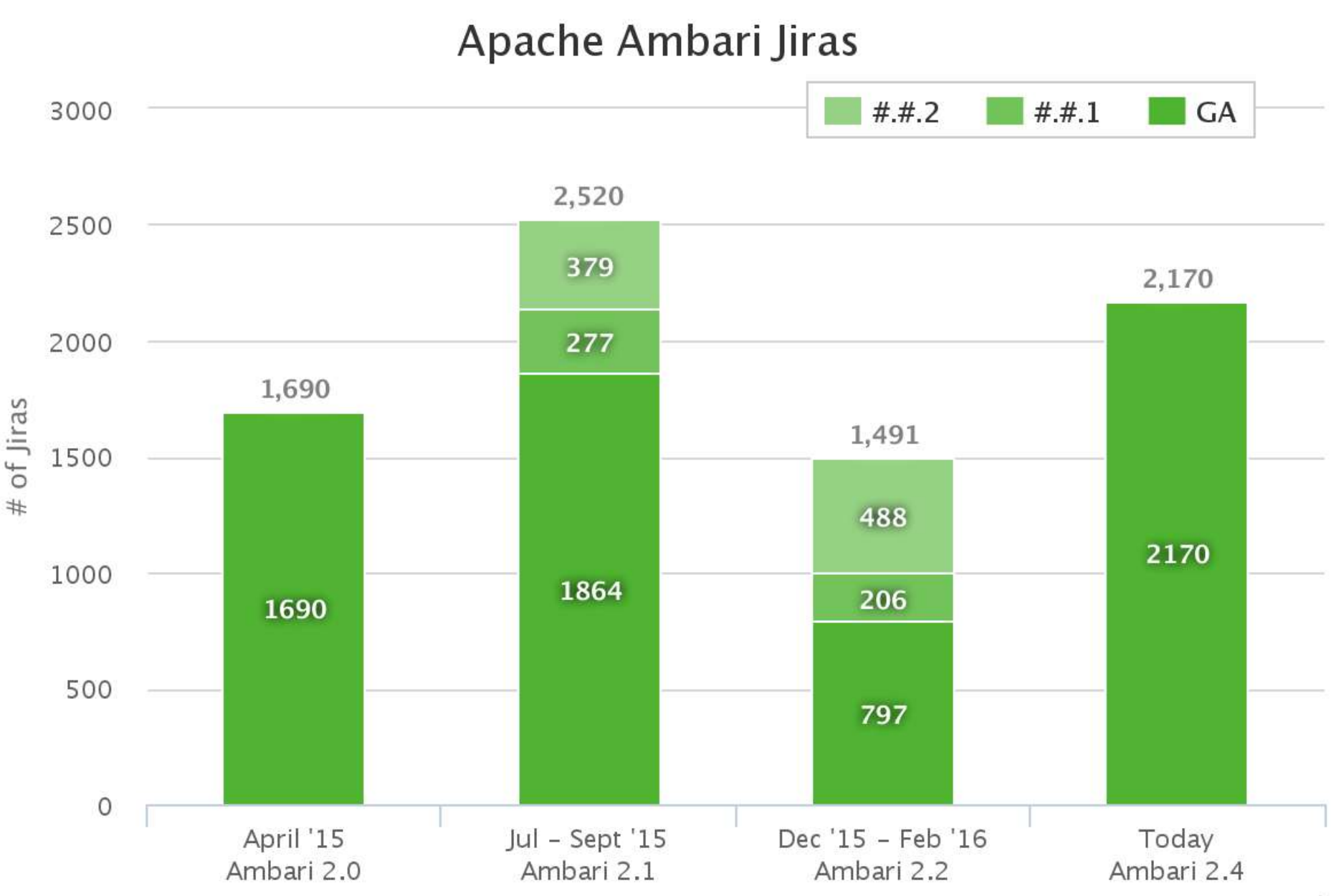
Alejandro Fernandez
Sr. Software Engineer @ Hortonworks
Apache Ambari PMC
alejandro@apache.org

What is Apache Ambari?

Apache Ambari is the open-source platform to provision, manage and monitor Hadoop clusters




Apache
Ambari


Apache Ambari Jiras
#.#.2
#.#.1
GA
# of Jiras
3000
2500
2000
1500
1000
500
0
1,690
1690
2,520
379
277
1864
1,491
488
206
797
2,170
2170
April '15
Ambari 2.0
Jul – Sept '15
Ambari 2.1
Dec '15 – Feb '16
Ambari 2.2
Today
Ambari 2.4

4 years old

Exciting Enterprise Features in Ambari 2.4

- New Services: Log Search, Zeppelin, Hive LLAP
- Role Based Access Control
- Management Packs
- Grafana UI for Ambari Metrics System
- New Views: Zeppelin, Storm

More in Ambari 2.4

Core Features

- Alerts: Customizable props and thresholds (AMBARI-14898)
- Alerts: Retry tolerance (AMBARI-15686)
- Alerts: New HDFS Alerts (AMBARI-14800)
- New Host Page Filtering (AMBARI-15210)
- Remove Service from UI (AMBARI-14759)
- Support for SLES 12 (AMBARI-16007)
- Stability: Database Consistency Checking (AMBARI-16258)
- Customizable Ambari Log + PID Dirs (AMBARI-15300)
- New Version Registration Experience (AMBARI-15724)
- Log Search Technical Preview (AMBARI-14927)

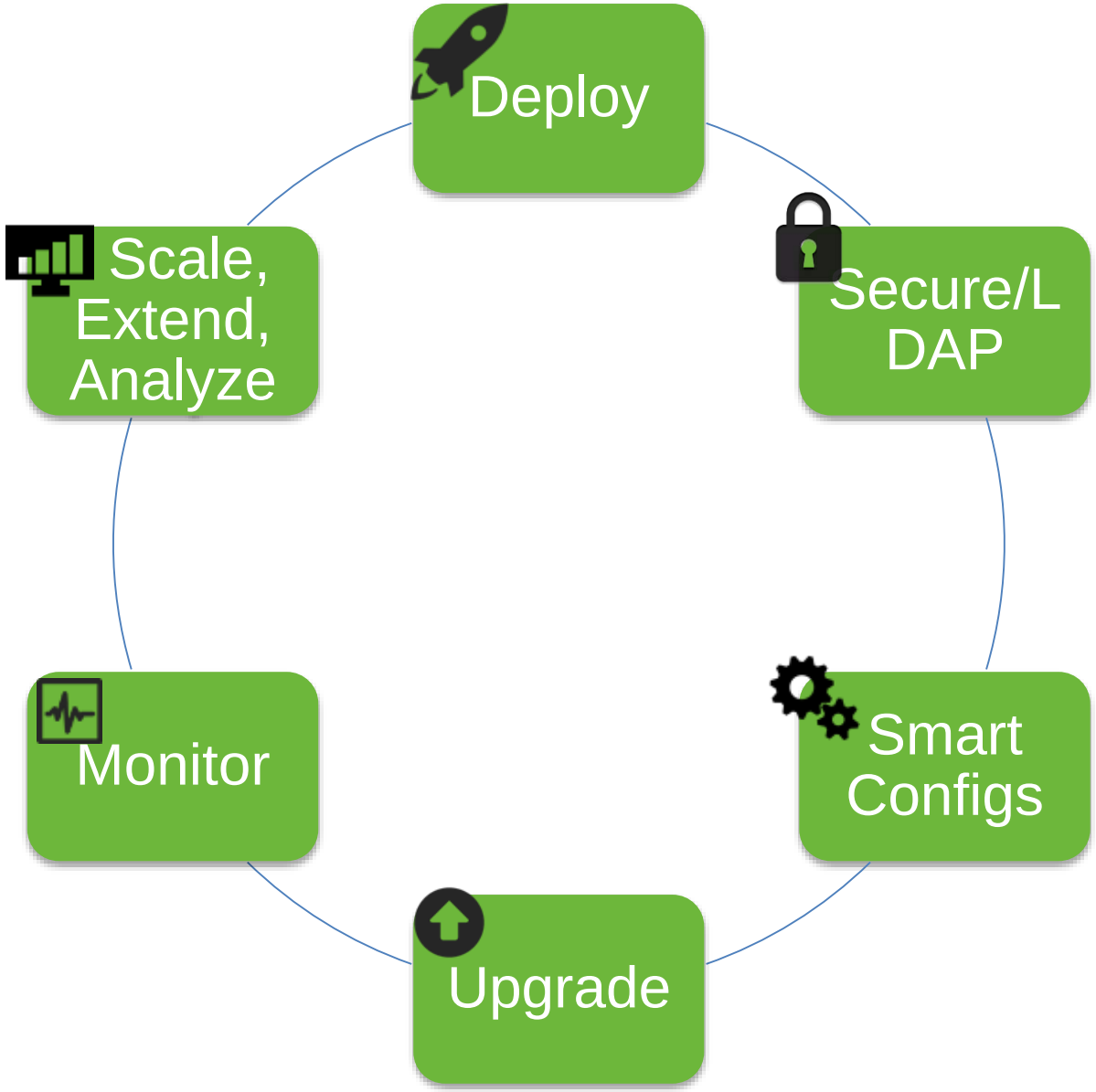
Security Features

- Operational Audit Logging (AMBARI-15241)
- Role-Based Access Control (AMBARI-13977)
- Automated Setup of Ambari Kerberos through Blueprints (AMBARI-15561)
- Automated Setup of Ambari Proxy User (AMBARI-15561)
- Customizable Host Reg. SSH Port (AMBARI-13450)

Views Framework Features

- View URLs for bookmarks (AMBARI-15821), View Refresh (AMBARI-15682)
- Inherit Cluster Permissions (AMBARI-16177)
- Remote Cluster Registration (AMBARI-16274)

Simply Operations - Lifecycle



Deploy On Premise

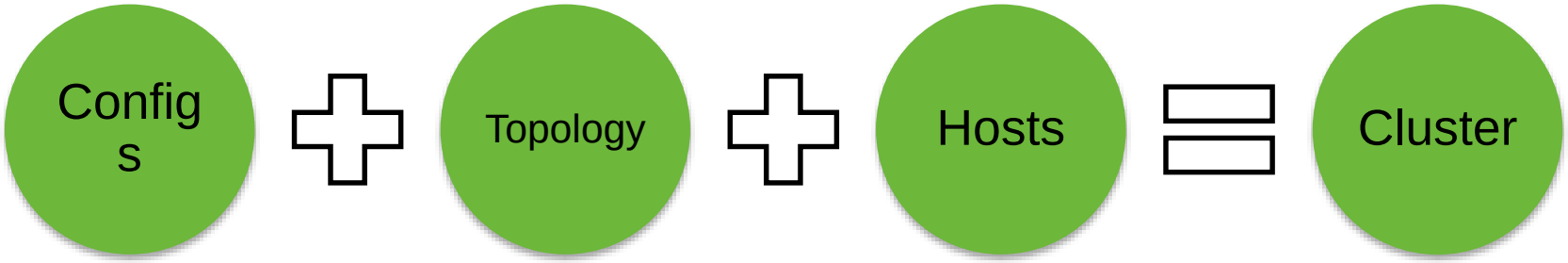
Ambari UI wizard handles all of these combinations and makes recommendations based on host specs.

Deploy On The Cloud

- Certified environments
- Sysprepped VMs
- Hundreds of similar clusters

Deploy with Blueprints

- Systematic way of defining a cluster



- Export existing cluster into blueprint
  - /api/v1/clusters/:clusterName?format=blueprint

Create a cluster with Blueprints

1. POST /api/v1/blueprints/my-blueprint

```
{
 "configurations" : [
    {
      "hdfs-site" : {
              "dfs.datanode.data.dir" : "/hadoop/1,
                  /hadoop/2,/hadoop/3"
      }
    }
  ],
  "host_groups" : [
    {
      "name" : "master-host",
      "components" : [
      { "name" : "NAMENODE" },
      { "name" : "RESOURCEMANAGER" },
      …
      ],
      "cardinality" : "1"
    },
    {
      "name" : "worker-host",
      "components" : [
      { "name" : "DATANODE" },
      { "name" : "NODEMANAGER" },
      …
      ],
      "cardinality" : "1+"
    },
  ],
  "Blueprints" : {
    "stack_name" : "HDP",
    "stack_version" : "2.5"
  }
}
```

2. POST /api/v1/clusters/my-cluster

```
{
  "blueprint" : "my-blueprint",
  "host_groups" :[
    {
      "name" : "master-host",
      "hosts" : [
        {
          "fqdn" : "master001.ambari.apache.org"
              }
      ]
    },
    {
      "name" : "worker-host",
      "hosts" : [
        {
          "fqdn" : "worker001.ambari.apache.org"
              },
        {
          "fqdn" : "worker002.ambari.apache.org"
              },
        …
        {
          "fqdn" : "worker099.ambari.apache.org"
              }
      ]
    }
  ]
}
```

Create a cluster with Blueprints

1. POST /api/v1/blueprints/my-blueprint

```
{
 "configurations" : [
    {
      "hdfs-site" : {
              "dfs.datanode.data.dir" : "/hadoop/1,
                  /hadoop/2,/hadoop/3"
      }
    }
  ],
  "host_groups" : [
    {
      "name" : "master-host",
      "components" : [
      { "name" : "NAMENODE" },
      { "name" : "RESOURCEMANAGER" },
      …
      ],
      "cardinality" : "1"
    },
    {
      "name" : "worker-host",
      "components" : [
      { "name" : "DATANODE" },
      { "name" : "NODEMANAGER" },
      …
      ],
      "cardinality" : "1+"
    },
  ],
  "Blueprints" : {
    "stack_name" : "HDP",
    "stack_version" : "2.5"
  }
}
```

2. POST /api/v1/clusters/my-cluster

```
{
  "blueprint" : "my-blueprint",
  "host_groups" :[
    {
      "name" : "master-host",
      "hosts" : [
        {
          "fqdn" : "master001.ambari.apache.org"
              }
      ]
    },
    {
      "name" : "worker-host",
      "hosts" : [
        {
          "fqdn" : "worker001.ambari.apache.org"
              },
        {
          "fqdn" : "worker002.ambari.apache.org"
              },
        …
        {
          "fqdn" : "worker099.ambari.apache.org"
              }
      ]
    }
  ]
}
```

Create a cluster with Blueprints

1. POST /api/v1/blueprints/my-blueprint

```
{
 "configurations" : [
    {
      "hdfs-site" : {
              "dfs.datanode.data.dir" : "/hadoop/1,
                  /hadoop/2,/hadoop/3"
      }
    }
  ],
  "host_groups" : [
    {
      "name" : "master-host",
      "components" : [
      { "name" : "NAMENODE" },
      { "name" : "RESOURCEMANAGER" },
      …
      ],
      "cardinality" : "1"
    },
    {
      "name" : "worker-host",
      "components" : [
      { "name" : "DATANODE" },
      { "name" : "NODEMANAGER" },
      …
      ],
      "cardinality" : "1+"
    },
  ],
  "Blueprints" : {
    "stack_name" : "HDP",
    "stack_version" : "2.5"
  }
}
```

2. POST /api/v1/clusters/my-cluster

```
{
  "blueprint" : "my-blueprint",
  "host_groups" :[
    {
      "name" : "master-host",
      "hosts" : [
        {
          "fqdn" : "master001.ambari.apache.org"
              }
      ]
    },
    {
      "name" : "worker-host",
      "hosts" : [
        {
          "fqdn" : "worker001.ambari.apache.org"
              },
        {
          "fqdn" : "worker002.ambari.apache.org"
              },
        …
        {
          "fqdn" : "worker099.ambari.apache.org"
              }
      ]
    }
  ]
}
```

Create a cluster with Blueprints

1. POST /api/v1/blueprints/my-blueprint

```
{
 "configurations" : [
    {
      "hdfs-site" : {
              "dfs.datanode.data.dir" : "/hadoop/1,
                  /hadoop/2,/hadoop/3"
      }
    }
  ],
  "host_groups" : [
    {
      "name" : "master-host",
      "components" : [
      { "name" : "NAMENODE" },
      { "name" : "RESOURCEMANAGER" },
      …
      ],
      "cardinality" : "1"
    },
    {
      "name" : "worker-host",
      "components" : [
      { "name" : "DATANODE" },
      { "name" : "NODEMANAGER" },
      …
      ],
      "cardinality" : "1+"
    },
  ],
  "Blueprints" : {
    "stack_name" : "HDP",
    "stack_version" : "2.5"
  }
}
```

2. POST /api/v1/clusters/my-cluster

```
{
  "blueprint" : "my-blueprint",
  "host_groups" :[
    {
      "name" : "master-host",
      "hosts" : [
        {
          "fqdn" : "master001.ambari.apache.org"
            }
      ]
    },
    {
      "name" : "worker-host",
      "hosts" : [
        {
          "fqdn" : "worker001.ambari.apache.org"
            },
        {
          "fqdn" : "worker002.ambari.apache.org"
            },
        …
        {
          "fqdn" : "worker099.ambari.apache.org"
            }
      ]
    }
  ]
}
```

Blueprints for Large Scale

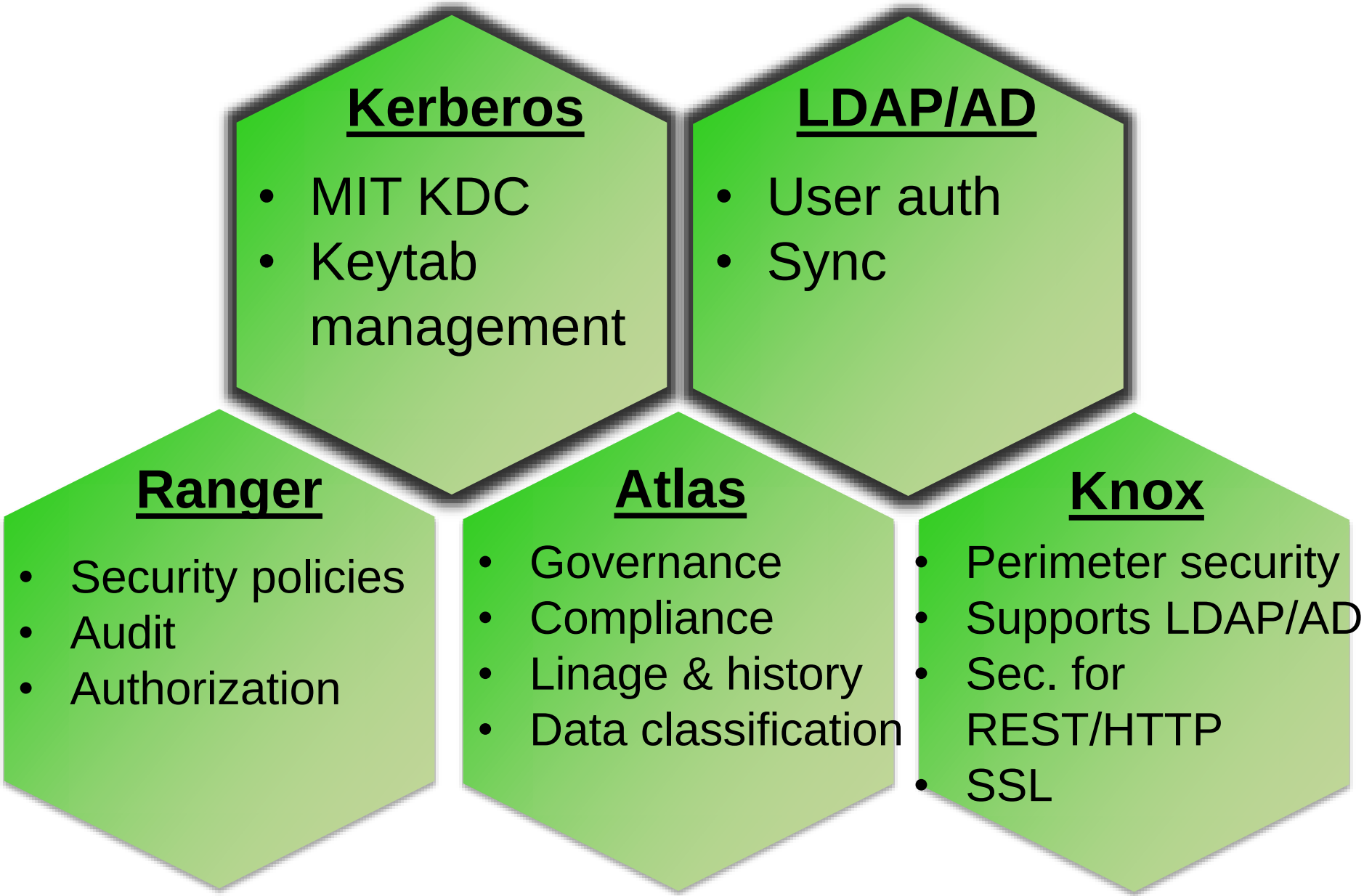
- **Kerberos**, secure out-of-the-box
- **High Availability** is setup initially for NameNode, YARN, Hive, Oozie, etc
- **Host Discovery** allows Ambari to automatically install services for a Host when it comes online
- **Stack Advisor** recommendations

Blueprint Host Discovery

```
POST /api/v1/clusters/MyCluster/hosts

[
  {
    "blueprint" : "single-node-hdfs-test2",
    "host_groups" :[
      {
      "host_group" : "slave",
      "host_count" : 3,
      "host_predicate" : "Hosts/cpu_count>1”
      }, {
      "host_group" : "super-slave",
      "host_count" : 5,
      "host_predicate" : "Hosts/cpu_count>2&
       Hosts/total_mem>3000000"
      }
    ]
  }
]
```

Comprehensive Security



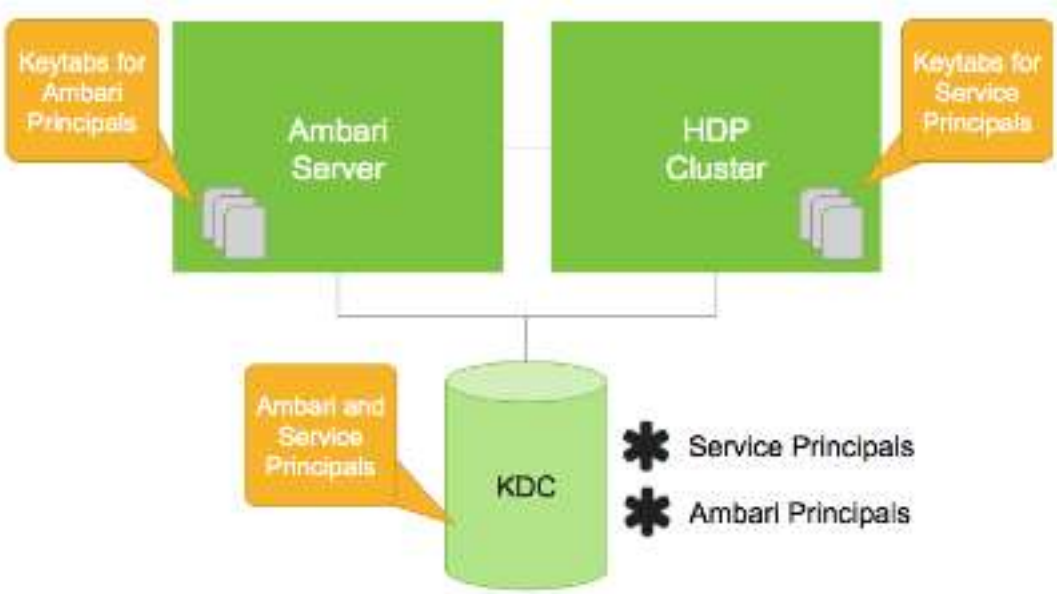
Kerberos

Ambari manages Kerberos principals and keytabs

Works with existing MIT KDC or Active Directory

Once Kerberized, handles

1. Adding hosts
2. Adding components to existing hosts
3. Adding services
4. Moving components to different hosts

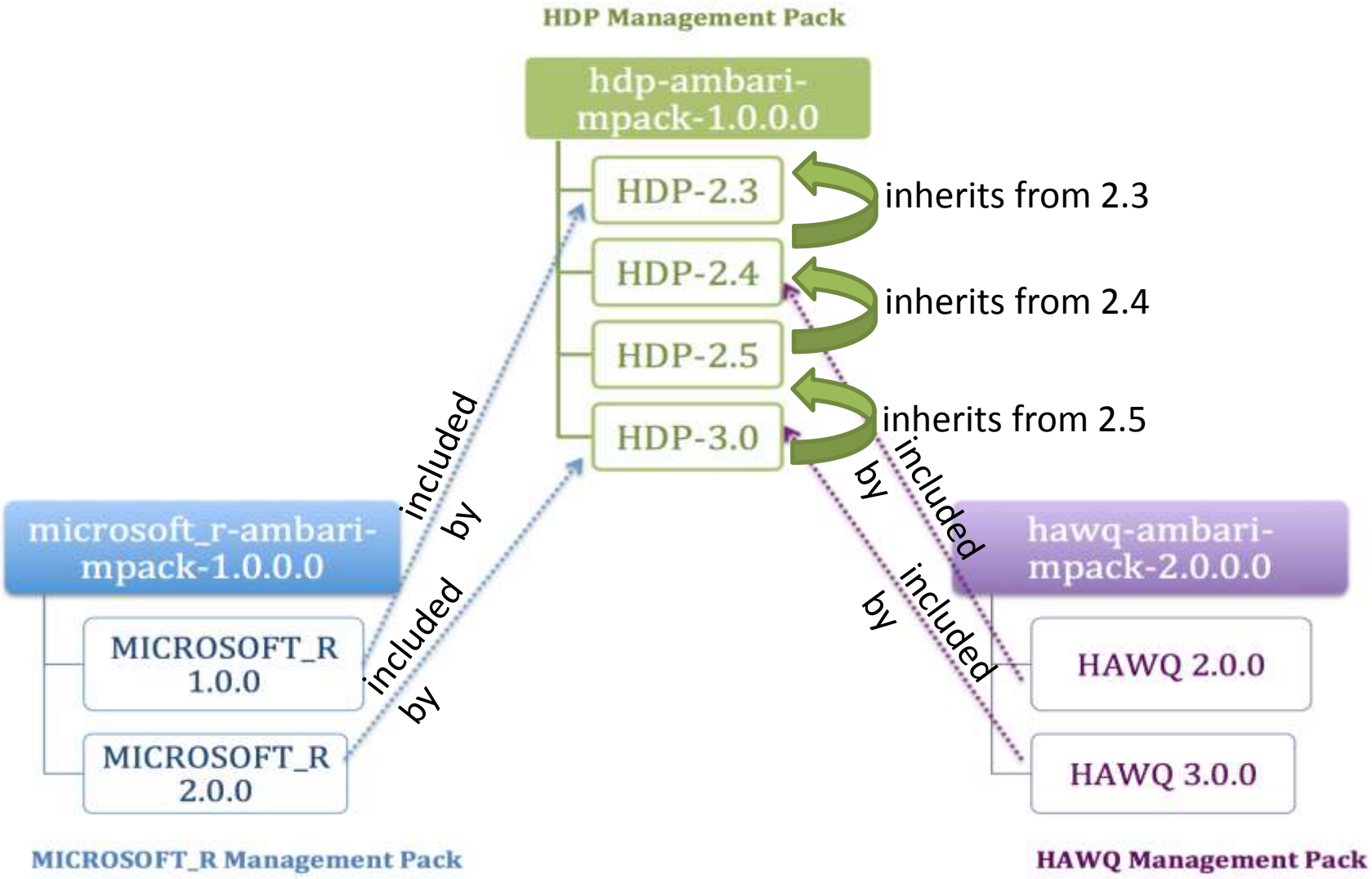


Management Packs

- Improved Release Management:
  Decouple Ambari core from stacks

- Support Add-ons:
  Release vehicle for 3rd party services, views
  Self-contained release artifacts
  Stack is an overlay of multiple management packs

Overlay of Management Packs



Management Pack++

Short Term Goals (Ambari 2.4)

- Retrofit in Stack Processing Framework
- Enable 3rd party to ship add-on services

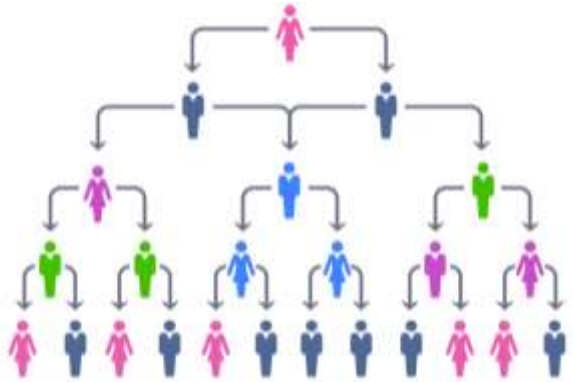
Future Goals

- Management Pack Framework
- Deliver Views

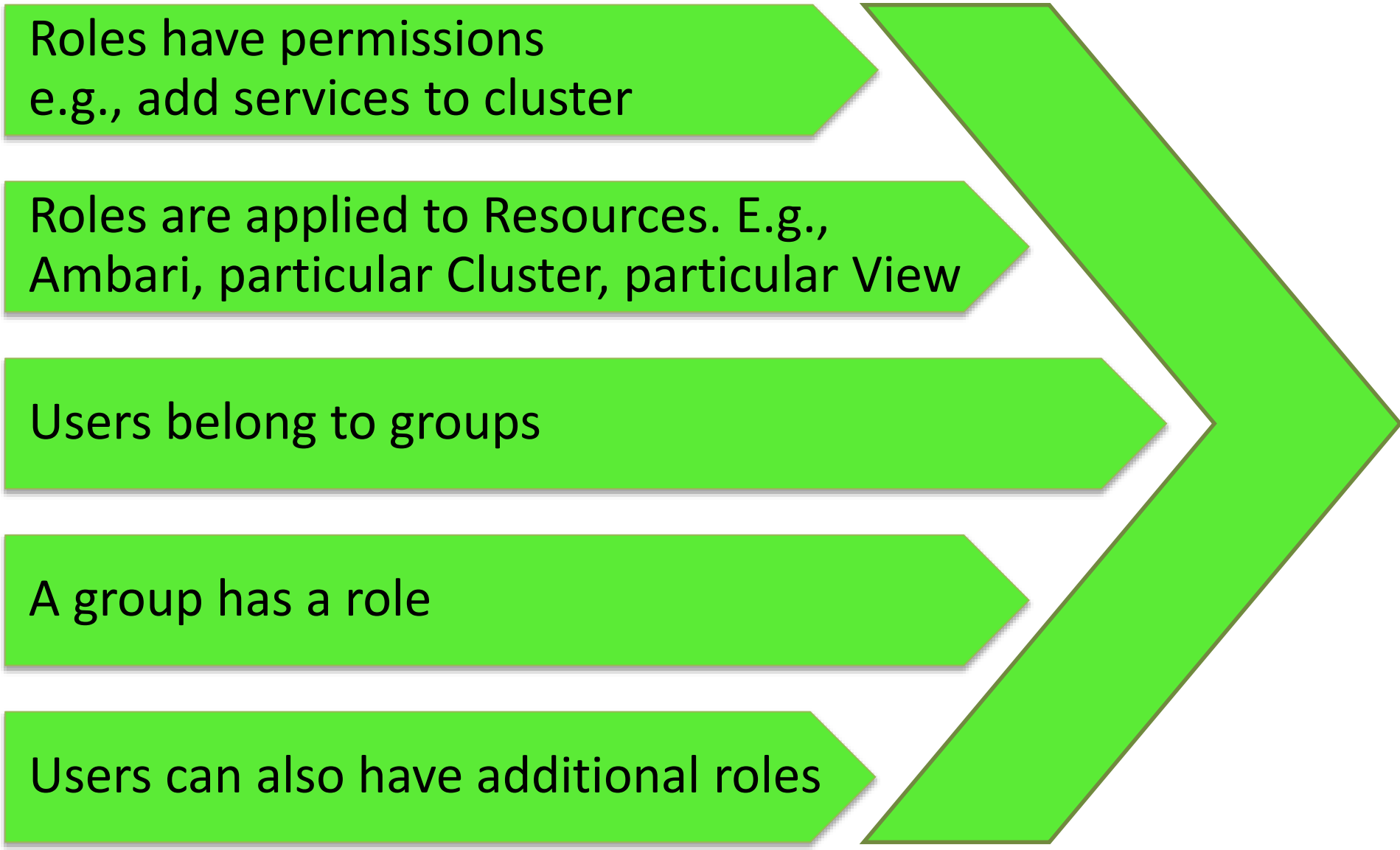
Role Based Access Control (RBAC)

- As Ambari & organizations grow, so do security needs

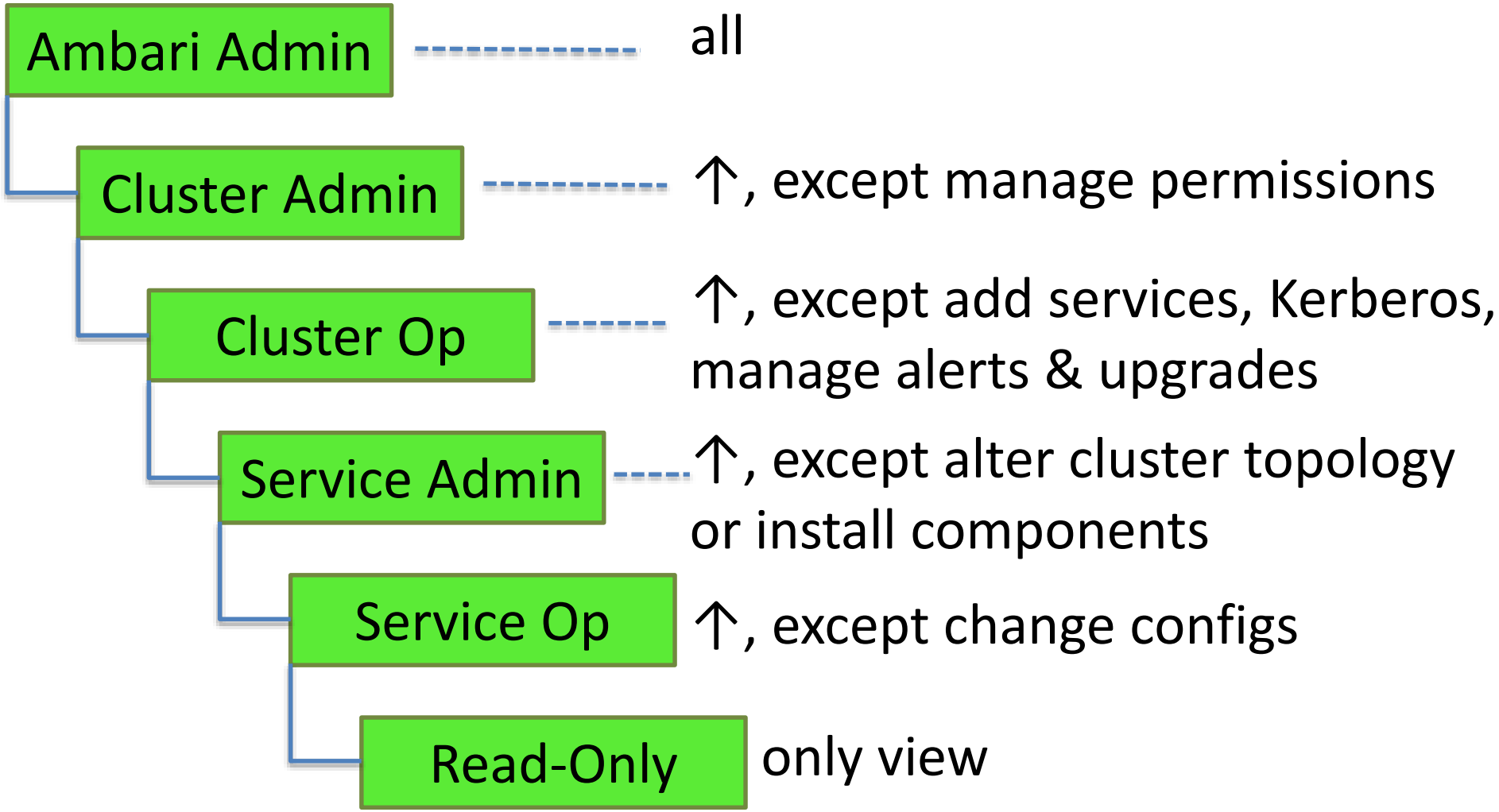
- Ambari integrates with external authentication systems & LDAP



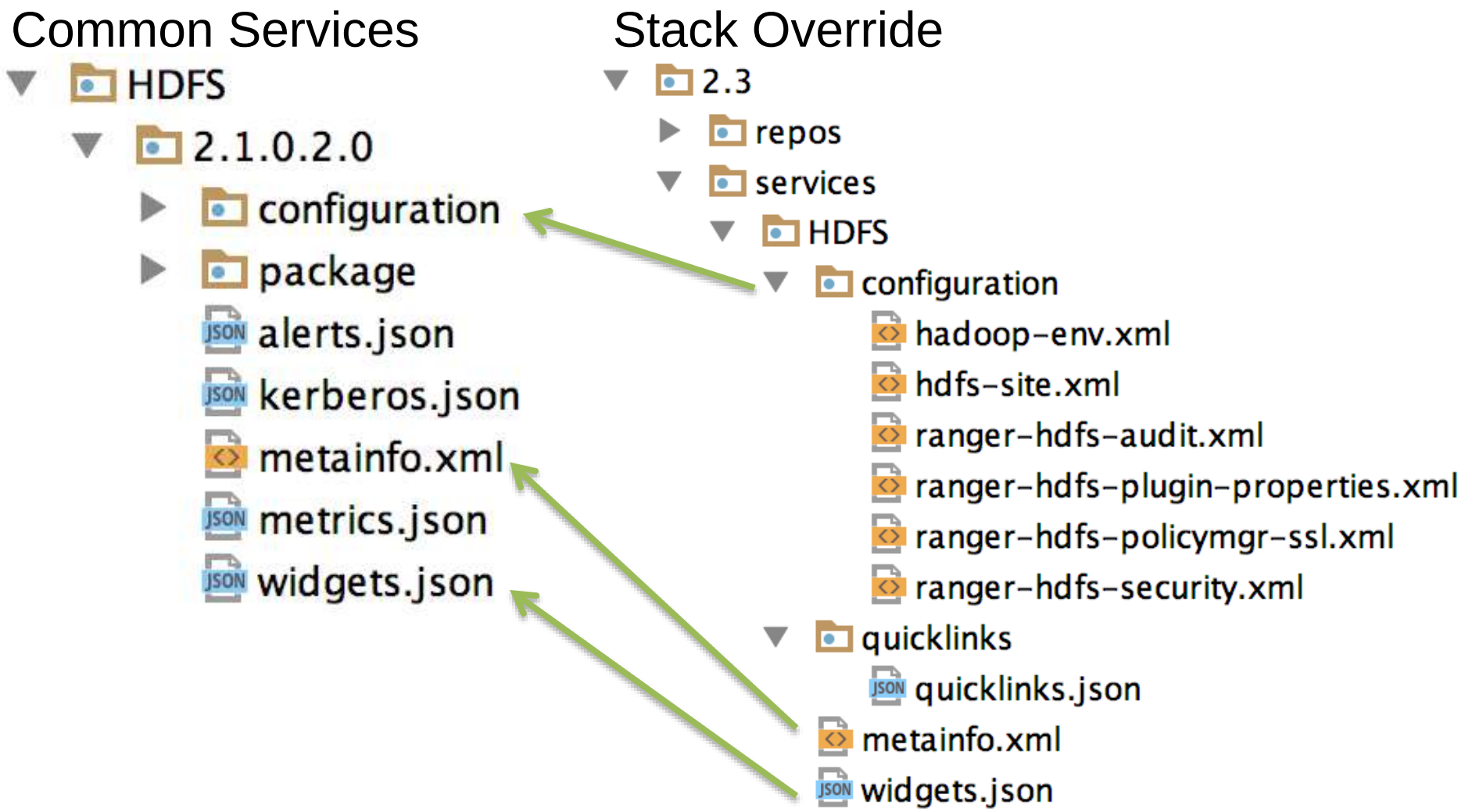
RBAC Terms


Roles have permissions
e.g., add services to cluster
Roles are applied to Resources. E.g.,
Ambari, particular Cluster, particular View
Users belong to groups
A group has a role
Users can also have additional roles

New RBAC Roles



Service Layout



Stack Advisor

Configuration

Kerberos
HTTPS
Zookeeper Servers
Memory Settings
…
High Availability

Example

```
# Atlas Servers
atlas.enabletTLS = true|false
atlas.server.http.port = 21000
atlas.server.https.port = 21443
```

atlas.rest.address = http(s)://host:port

Background: Upgrade Terminology

Manual Upgrade

- The user follows instructions to upgrade the stack
- Incurs downtime

Background: Upgrade Terminology

Rolling Upgrade

- ✓ Automated
- ✗ Upgrades one component per host at a time
- ✓ Preserves cluster operation and minimizes service impact

Manual Upgrade

- ✗ The user follows instructions to upgrade the stack
- ✗ Incurs downtime

Background: Upgrade Terminology

Express Upgrade

- Automated
- Runs in parallel across hosts
- Incurs downtime

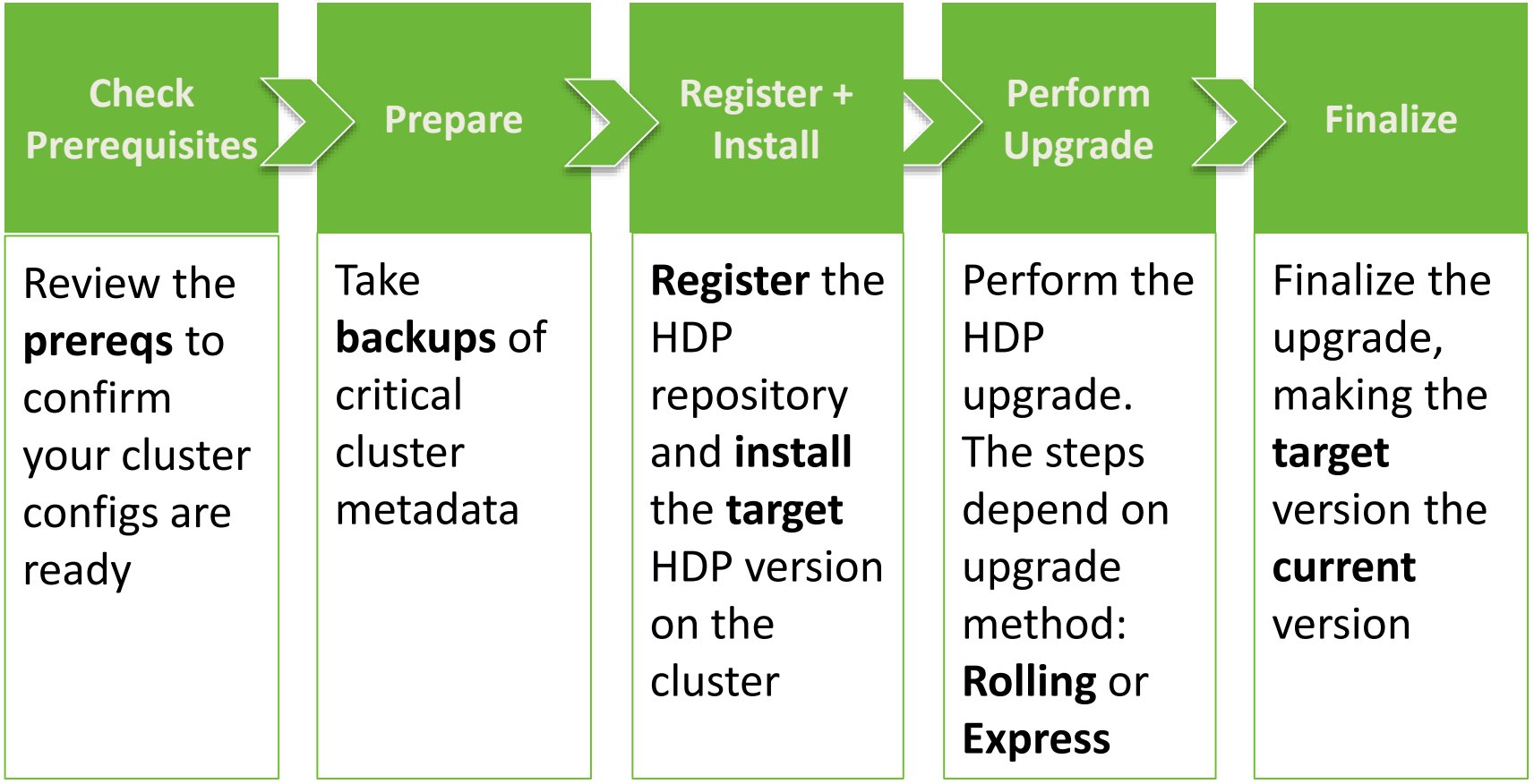
Rolling Upgrade

- Automated
- Upgrades one component per host at a time
- Preserves cluster operation and minimizes service impact

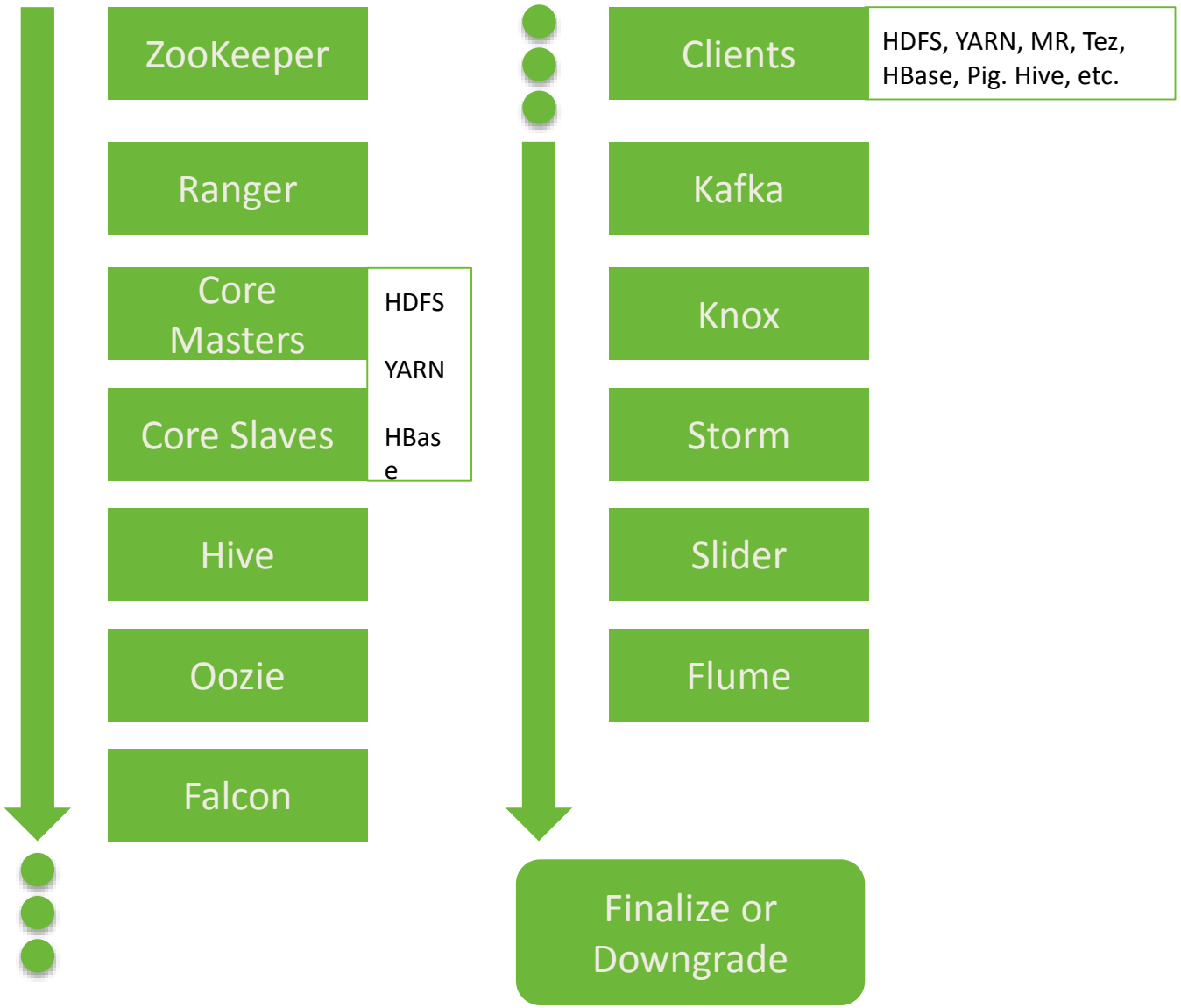
Manual Upgrade

- The user follows instructions to upgrade the stack
- Incurs downtime

Automated Upgrade: Rolling or Express



Process: Rolling Upgrade



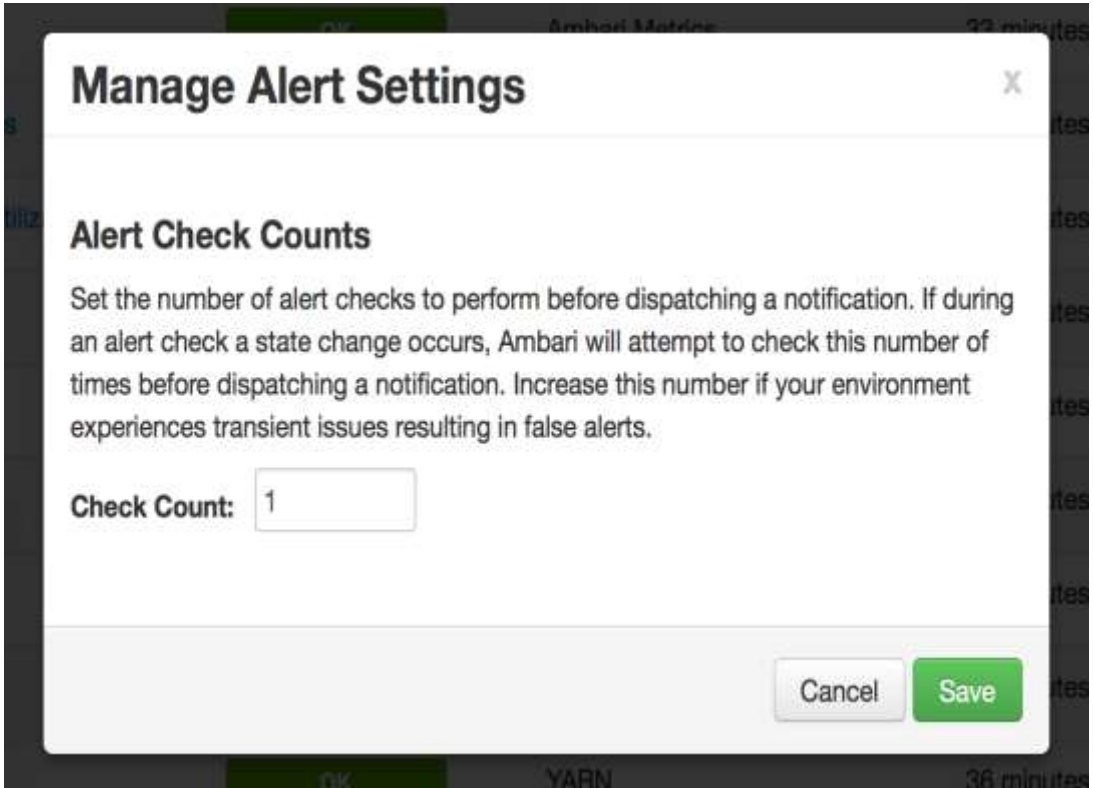
Alerting Framework

Alert Type	Description	Thresholds (units)
WEB	Connects to a Web URL. Alert status is based on the HTTP response code	Response Code (n/a) Connection Timeout (seconds)
PORT	Connects to a port. Alert status is based on response time	Response (seconds)
METRIC	Checks the value of a service metric. Units vary, based on the metric being checked	Metric Value (units vary) Connection Timeout (seconds) 
AGGREGATE	Aggregates the status for another alert	% Affected (percentage)
SCRIPT	Executes a script to handle the alert check	Varies 
SERVER	Executes a server-side runnable class to handle the alert check	Varies 

Alert Check Counts

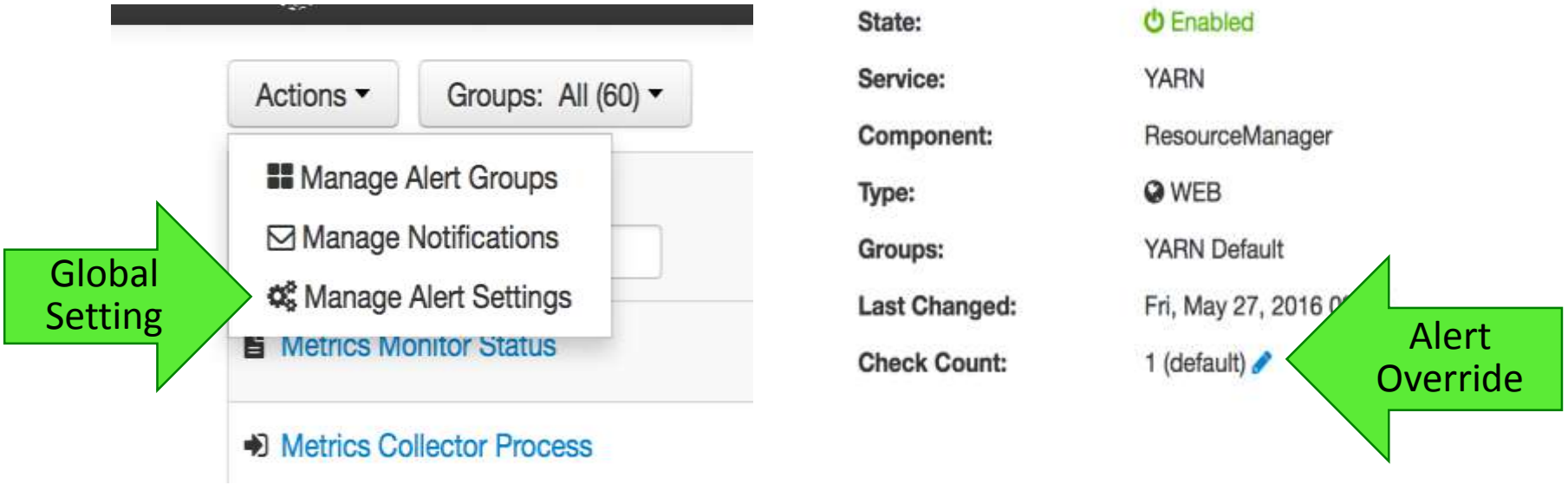
- Customize the number of times an alert is checked before dispatching a notification

- Avoid dispatching an alert notification (email, snmp) in case of transient issues

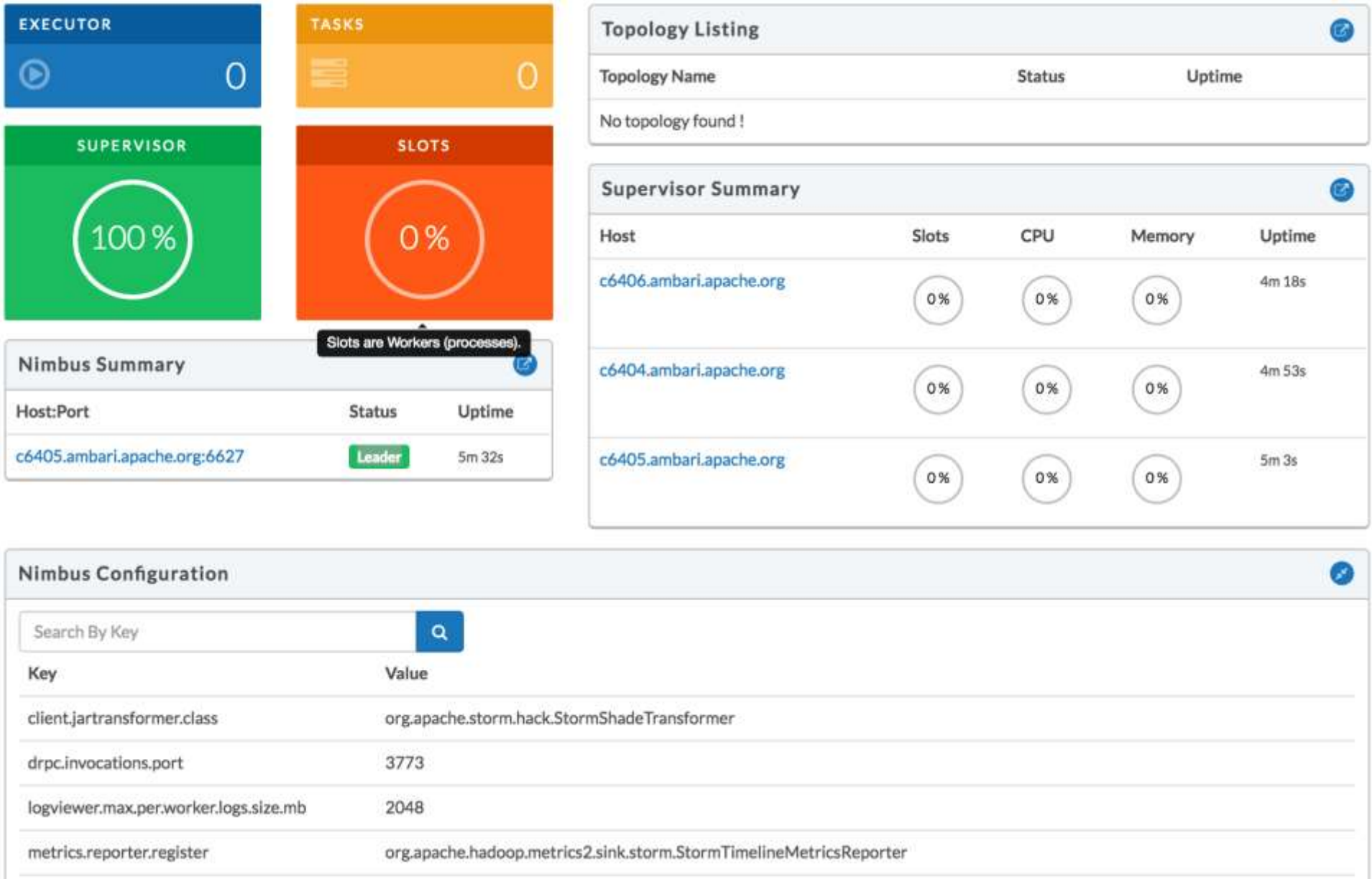


Alerts - Configuring the Check Count

Set globally for all alerts, or override for a specific alert



Storm Monitoring View



Grafana for Ambari Metrics

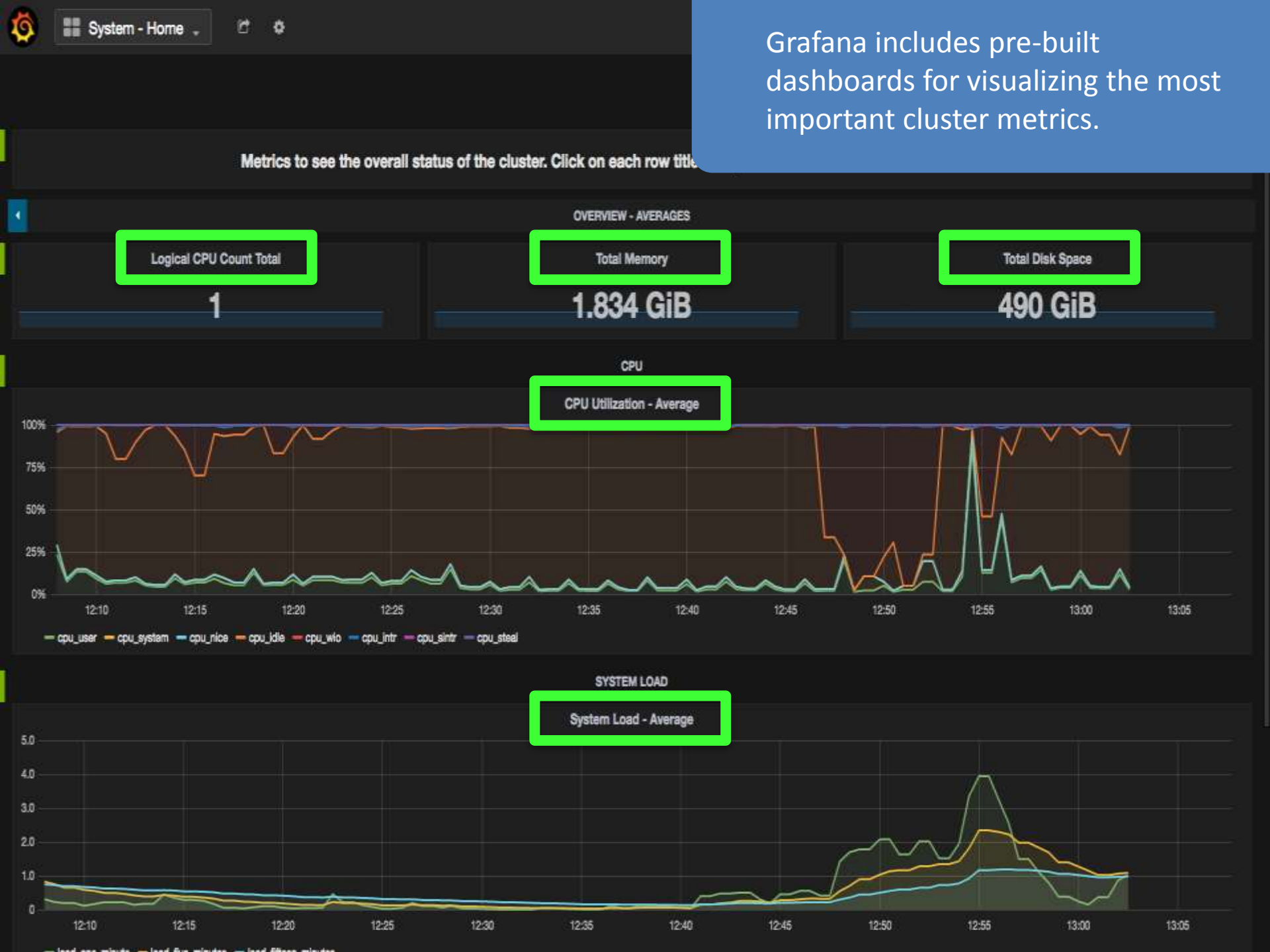
FEATURES

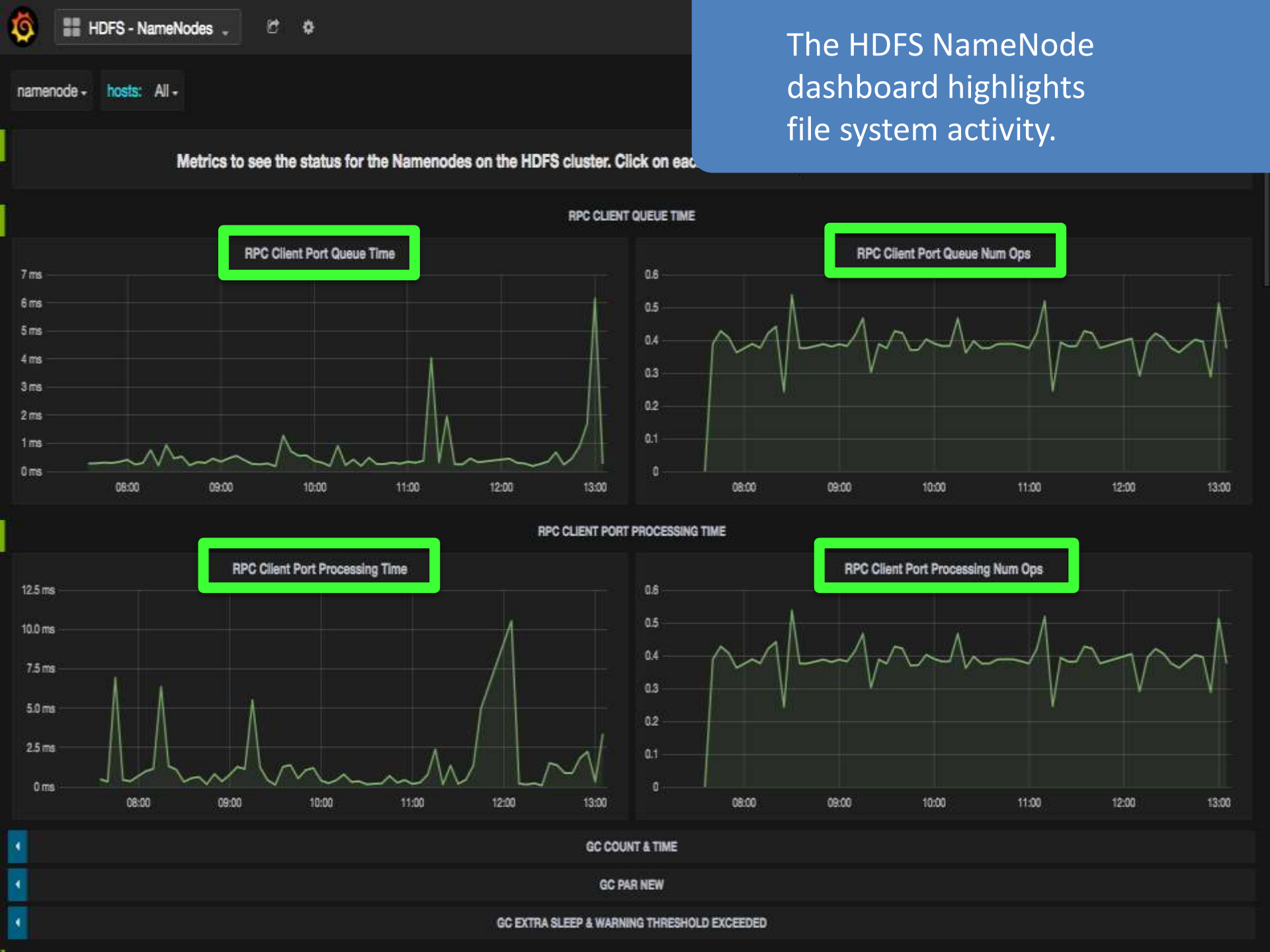
- Grafana as a “Native UI” for Ambari Metrics
- Pre-built Dashboards Host-level, Service-level
- Supports HTTPS

DASHBOARDS

- System Home, Servers
- HDFS Home, NameNodes, DataNodes
- YARN Home, Applications, Job History Server
- HBase Home, Performance

Grafana includes pre-built dashboards for visualizing the most important cluster metrics.

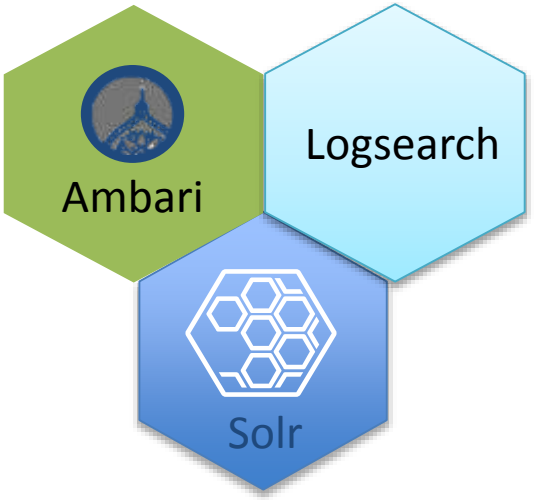




The HDFS NameNode dashboard highlights file system activity.

Log Search

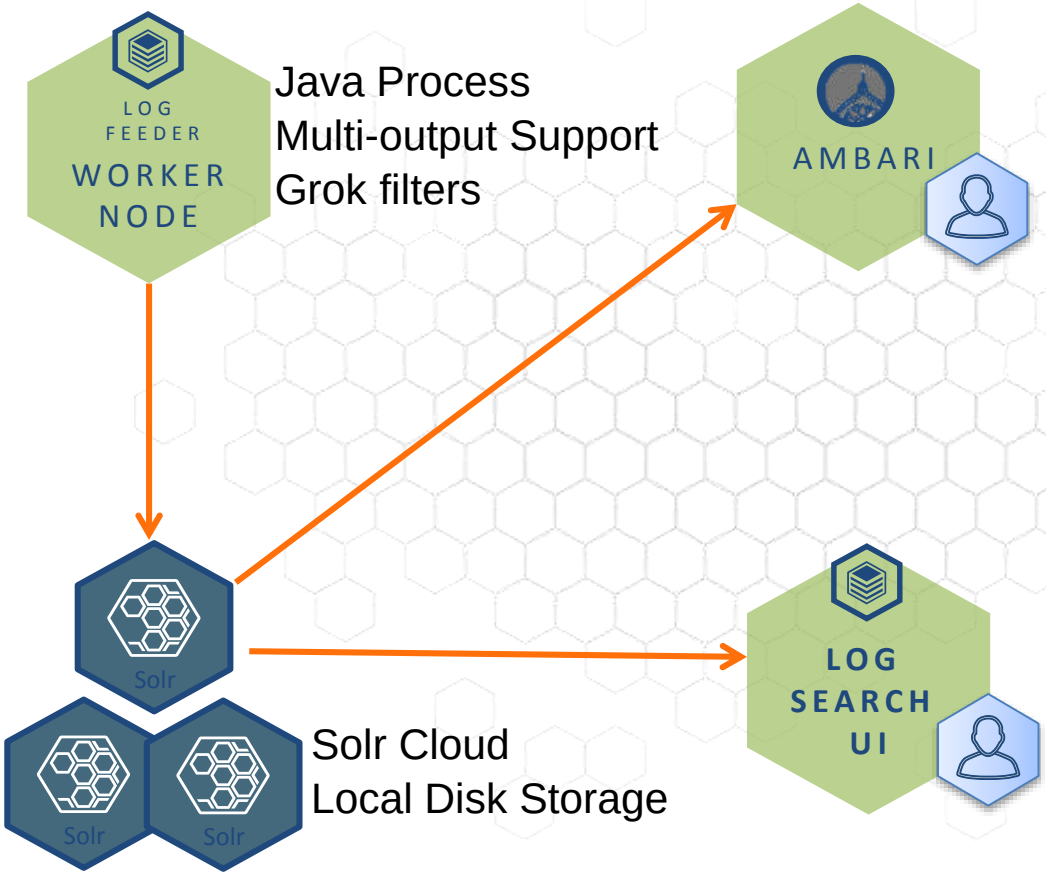
Search and index HDP logs!



Capabilities

- Rapid Search of all HDP component logs
- Search across time ranges, log levels, and for keywords

Log Search


LOG
FEEDER
WORKER
NODE
Java Process
Multi-output Support
Grok filters
AMBARI
Solr
Solr
Solr
Solr Cloud
Local Disk Storage
LOG
SEARCH
UI


revo5.hortonworks.local

Future of Ambari

- Cloud features
- Service multi-instance (two ZK quorums)
- Service multi-versions (Spark 1.6 & Spark 2.0)
- YARN assemblies
- Patch Upgrades: upgrade individual components in the same stack version, e.g., just DN and RM in HDP 2.5.*.* with zero downtime
- Ambari High Availability

As good as