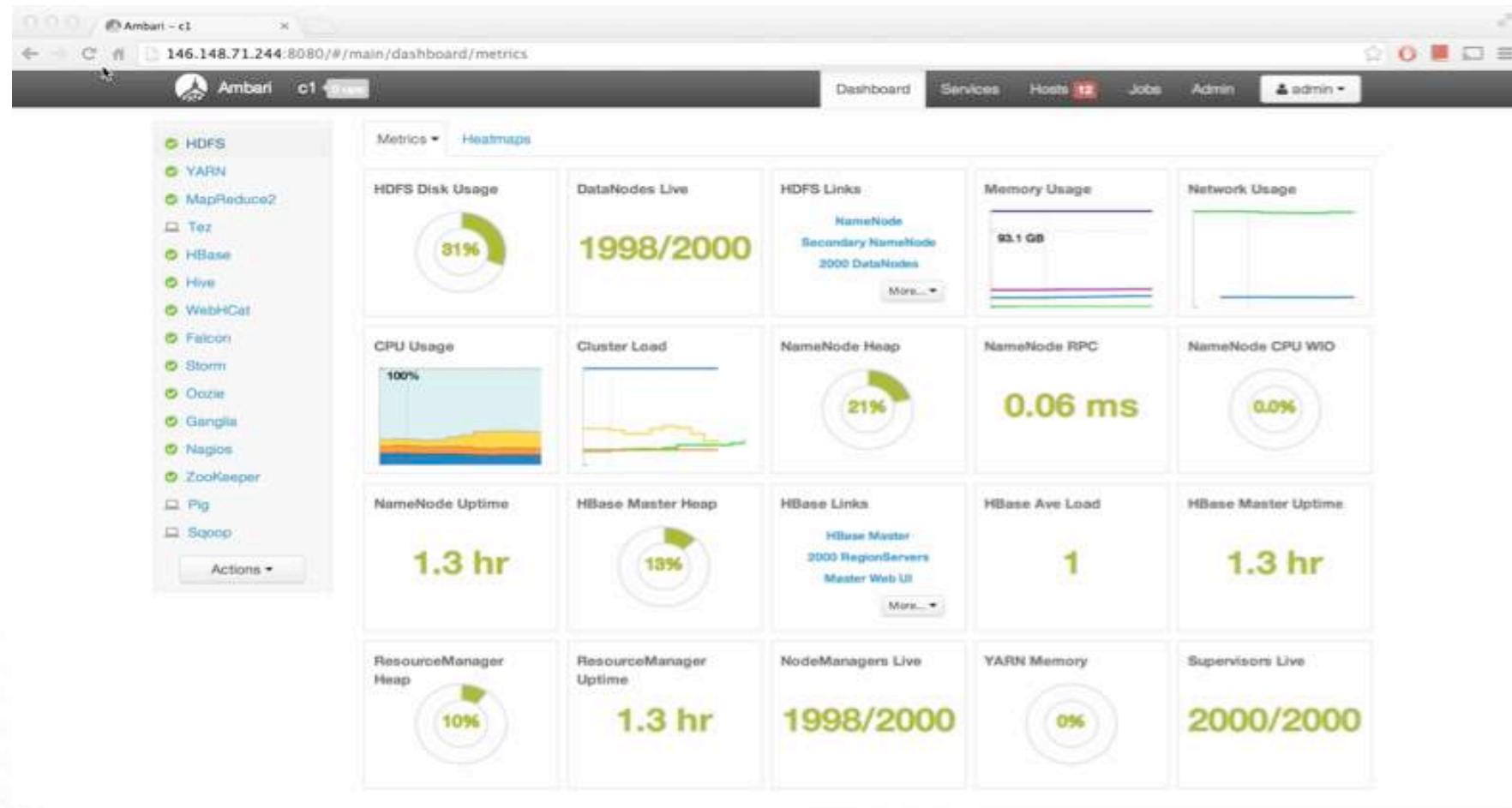
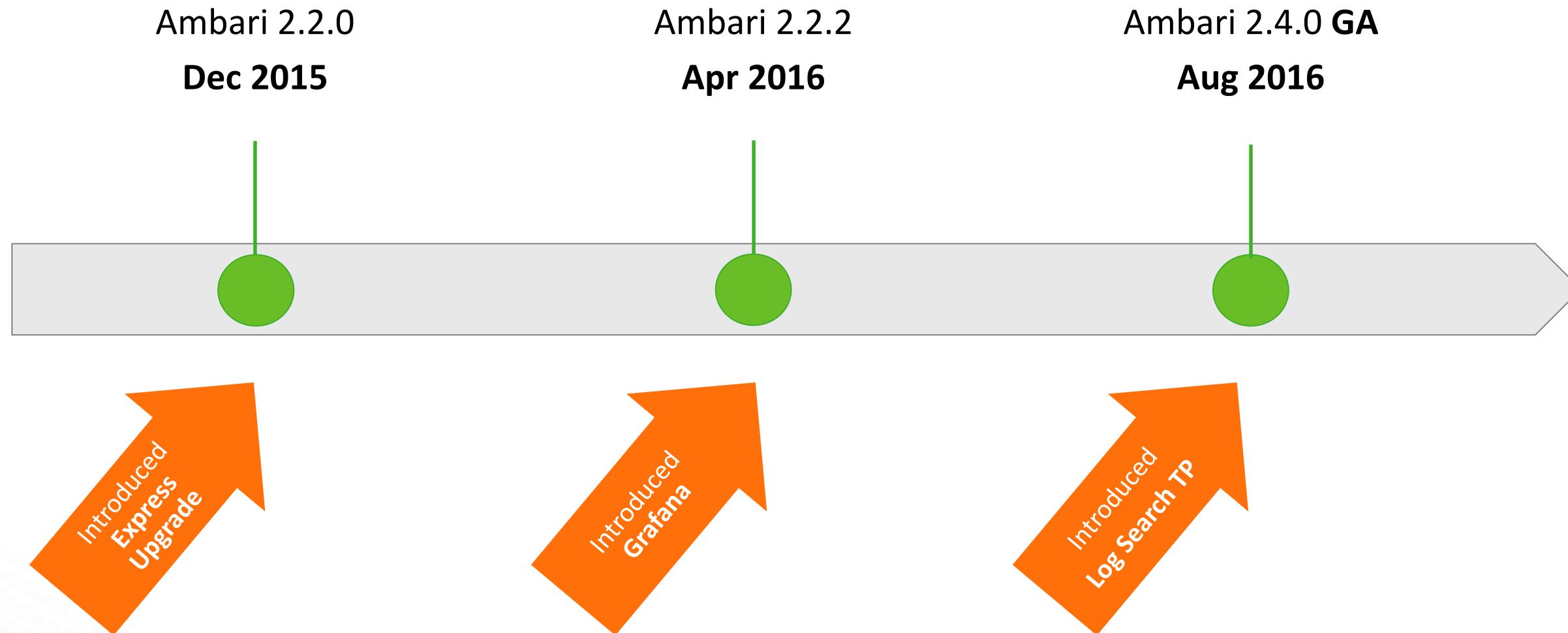


What is Apache Ambari?



A completely **open source** management platform for **provisioning, managing, monitoring and securing** **Apache Hadoop** clusters. Apache Ambari takes the guesswork out of operating Hadoop.

Ambari Recent Releases



Ambari Alerts

Full Visibility into Cluster Health

- Centralized management of Health Alerts
- Pre-defined alerts, configured by default on cluster install

The screenshot shows the Ambari UI interface for managing cluster health alerts. The top navigation bar includes the Ambari logo, the cluster name "MyCluster", and status indicators for "0 ops" and "0 alerts". The main menu has tabs for Dashboard, Services, Hosts, Alerts (which is currently selected), and Admin. A user dropdown indicates "admin".

The "Alerts" page displays a table of 37 alert definitions. The columns are: Alert Definition Name, Status, Service, Last Status Changed, and State. All alerts listed are currently "OK" (green status bar) and were last changed 4 days ago. The "State" column shows they are all "Enabled".

Alert Definition Name	Status	Service	Last Status Changed	State
DataNode Process	OK (2)	HDFS	4 days ago	Enabled
DataNode Storage	OK (2)	HDFS	4 days ago	Enabled
NodeManager Web UI	OK (2)	YARN	4 days ago	Enabled
DataNode Web UI	OK (2)	HDFS	4 days ago	Enabled
NodeManager Health	OK (2)	YARN	4 days ago	Enabled
Ambari Agent Disk Usage	OK (2)	Ambari	4 days ago	Enabled
ZooKeeper Server Process	OK	ZooKeeper	4 days ago	Enabled
History Server Process	OK	MapReduce2	4 days ago	Enabled
Secondary NameNode Process	OK	HDFS	4 days ago	Enabled
NameNode Process	OK	HDFS	4 days ago	Enabled

At the bottom, there are links for "37 of 37 definitions showing - clear filters" and pagination controls for "Show: 10" and "1 - 10 of 37".

Customizing Alerts

- Control thresholds, check intervals and response text

The screenshot shows the Ambari UI for managing alerts. The top navigation bar includes 'Ambari', 'MyCluster', '0 ops', '0 alerts', 'Dashboard', 'Services', 'Hosts', 'Alerts' (which is selected), 'Admin', and a user dropdown. The main content area is titled 'History Server Process' with a 'Back' link. On the left, there's a 'Configuration' tab with an 'Edit' button. The right side displays alert details:

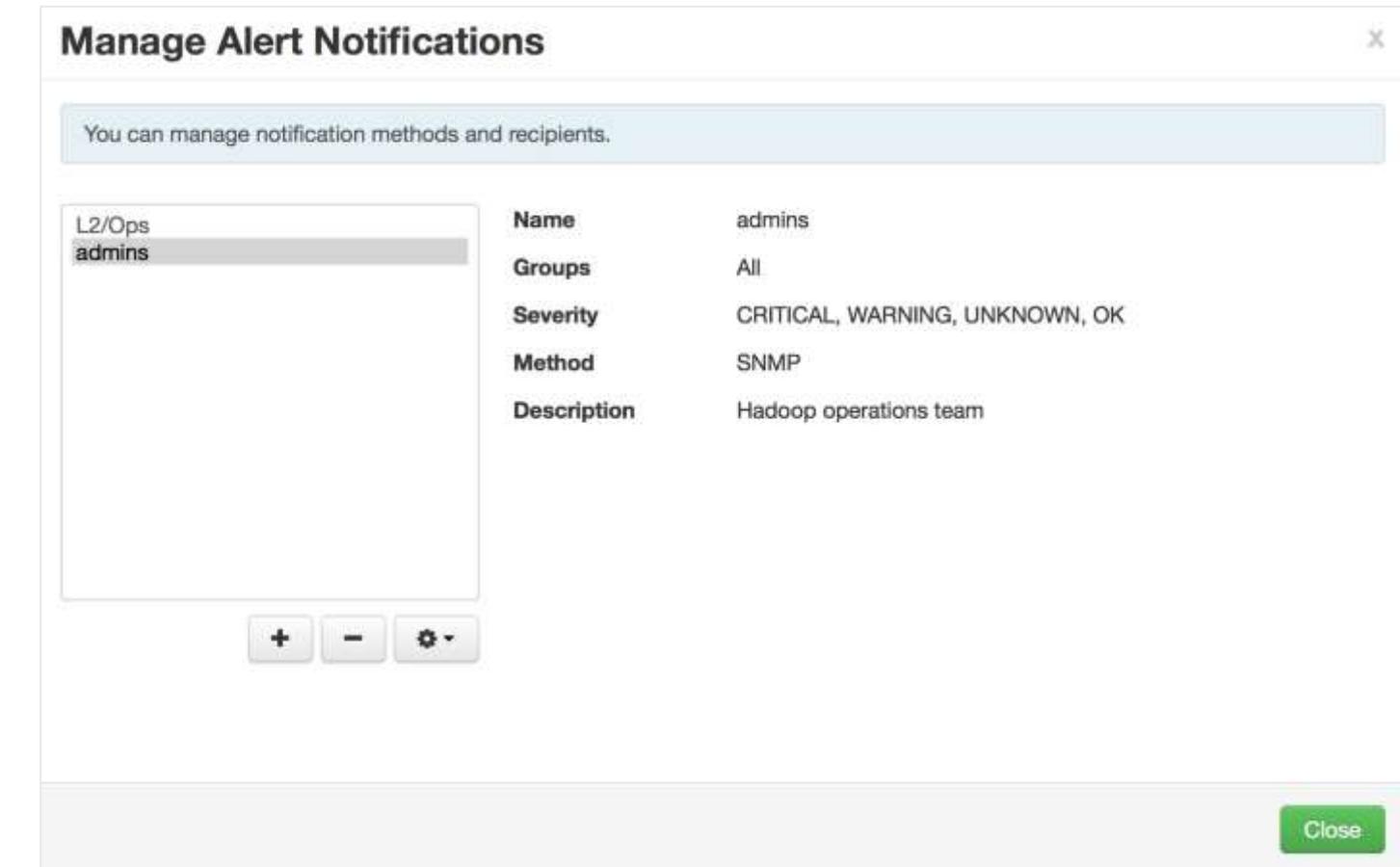
- Description:** This host-level alert is triggered if the History Server process cannot be established to be up and listening on the network.
- Interval:** 1 Minute
- Thresholds:**
 - OK:** TCP OK - {0:.3f}s response on port {1}
 - WARNING:** 1.5 Sec TCP OK - {0:.3f}s response on port {1}
 - CRITICAL:** 5 Sec Connection failed: {0} to {1}:{2}
- State:** Enabled
- Service:** MapReduce2
- Component:** History Server
- Type:** PORT
- Groups:** MAPREDUCE2 Default
- Last Changed:** Wed, Dec 17, 2014 17:52

Below the configuration, there's an 'Instances' section with a table:

Service / Host	Status	24-Hour	Check Response
MapReduce2 / c6402.ambari.apache.org	OK for 21 minutes	0	TCP OK - 0.000s response on port 19888

Alert Notifications

- **What: Create and manage multiple notification targets**
 - Control who gets notified when
- **Why: Filter by severity**
 - Send only certain notifications to certain targets based on severity
- **How: Control dispatch method**
 - Support for EMAIL + SNMP



Alert Groups

- **Create and manage groups of alerts**
- Group alerts further controls what alerts are dispatched which notifications
- **Assign group to notifications**
- Only dispatch to interested parties

Manage Alert Groups

You can manage alert groups for each service in this dialog. View the list of alert groups and the alert definitions configured in them. You can also add/remove alert definitions, and pick notification for that alert group.

AMBARI Default (1)
GANGLIA Default (5)
HDFS Default (18)
MAPREDUCE2 Default (4)
TEZ Default (0)
YARN Default (7)
ZOOKEEPER Default (2)

+ - ⚙️

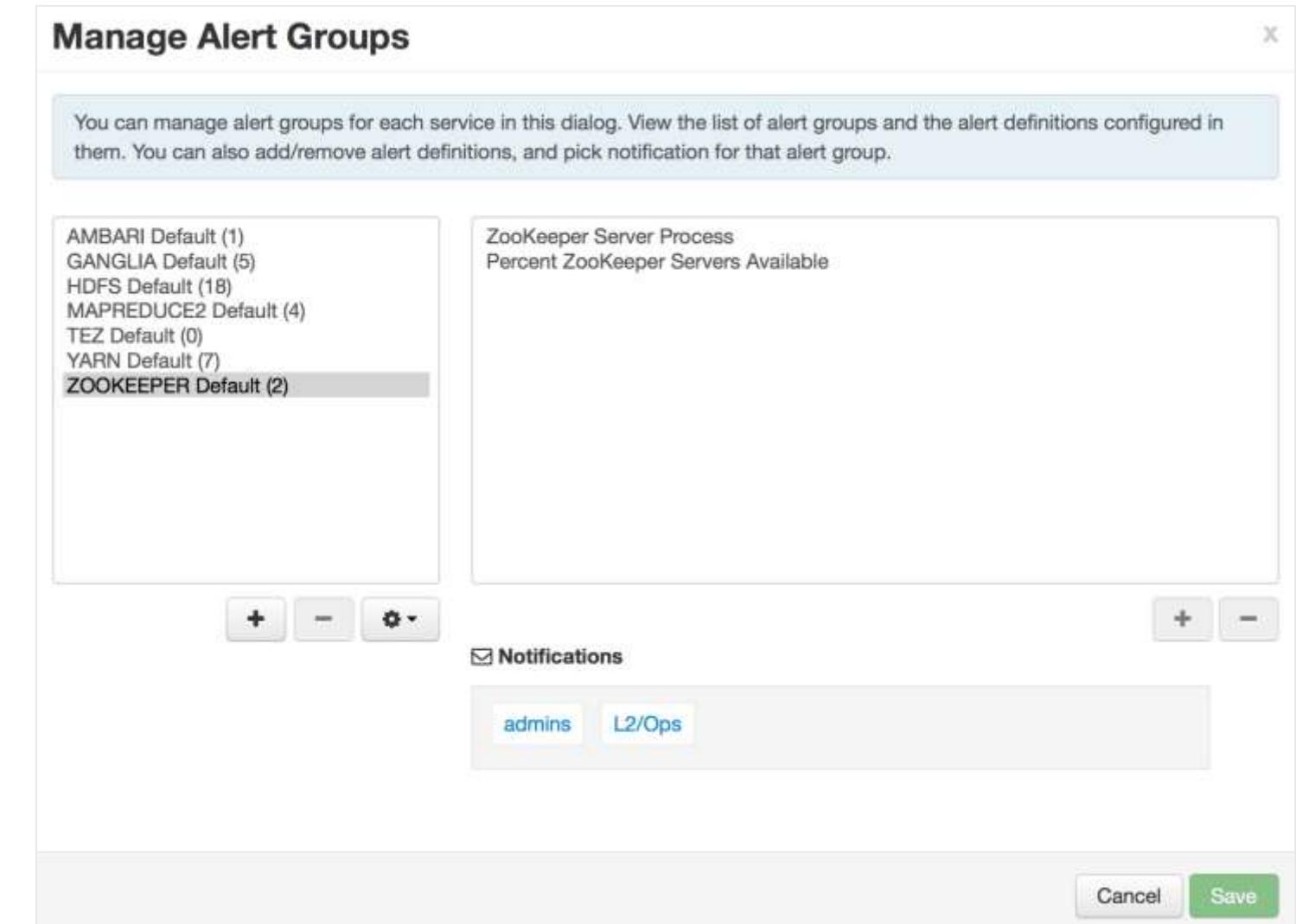
ZooKeeper Server Process
Percent ZooKeeper Servers Available

+ - ⚙️

Notifications

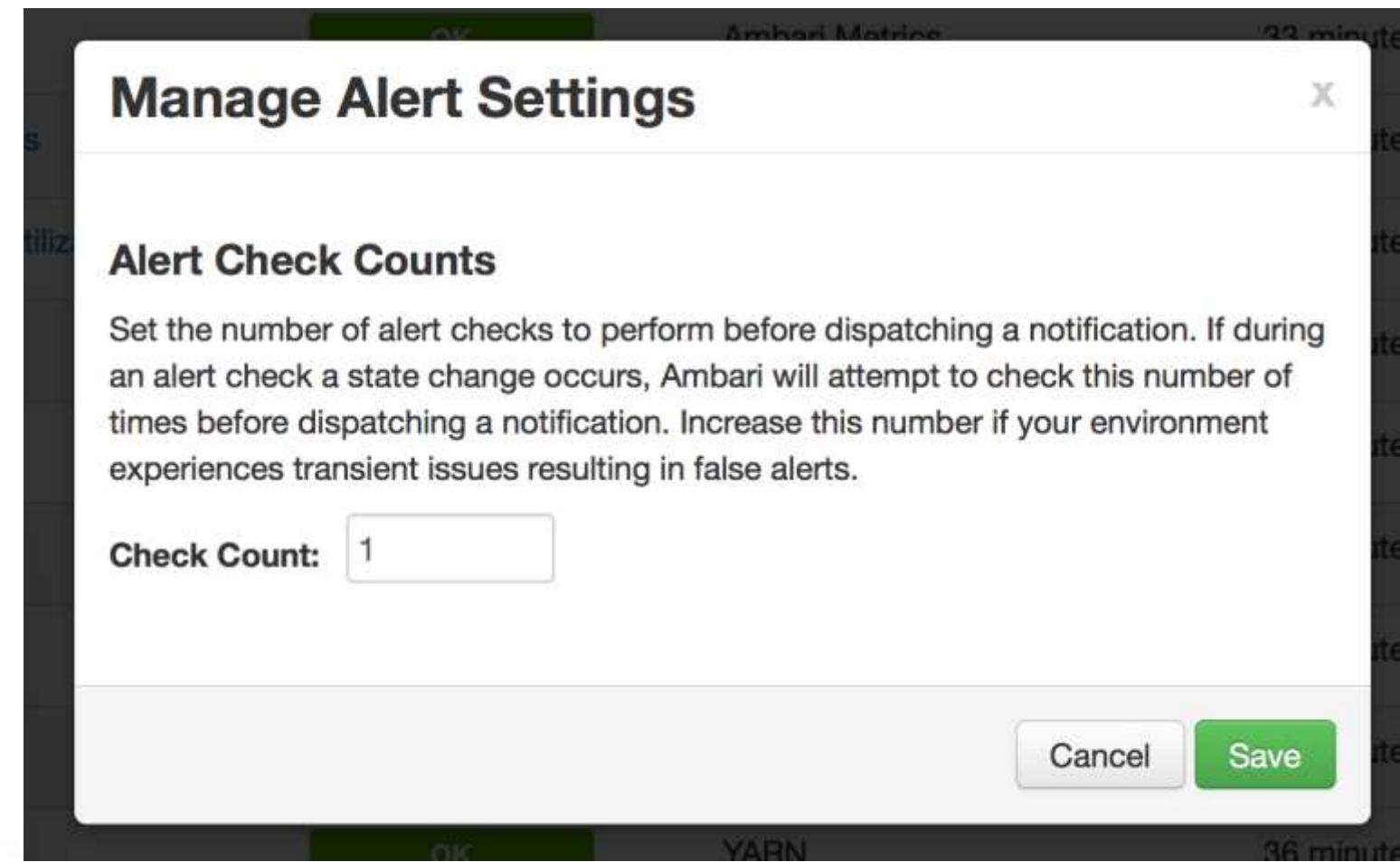
[admins](#) [L2/Ops](#)

[Cancel](#) [Save](#)



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



Configuring the Check Count

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

The screenshot shows the Ambari Metrics Monitor Status interface. On the left, there's a sidebar with 'Actions' and 'Groups: All (60)' dropdowns, and a list of links: 'Manage Alert Groups', 'Manage Notifications', 'Manage Alert Settings', 'Metrics Monitor Status' (which is highlighted in blue), and 'Metrics Collector Process'. An orange arrow labeled 'Global Setting' points to the 'Manage Alert Settings' link. On the right, a detailed view of an alert is shown with the following fields:

State:	Enabled
Service:	YARN
Component:	ResourceManager
Type:	WEB
Groups:	YARN Default
Last Changed:	Fri, May 27, 2016 01:52
Check Count:	1 (default)

An orange arrow labeled 'Alert Override' points to the 'Check Count' field.

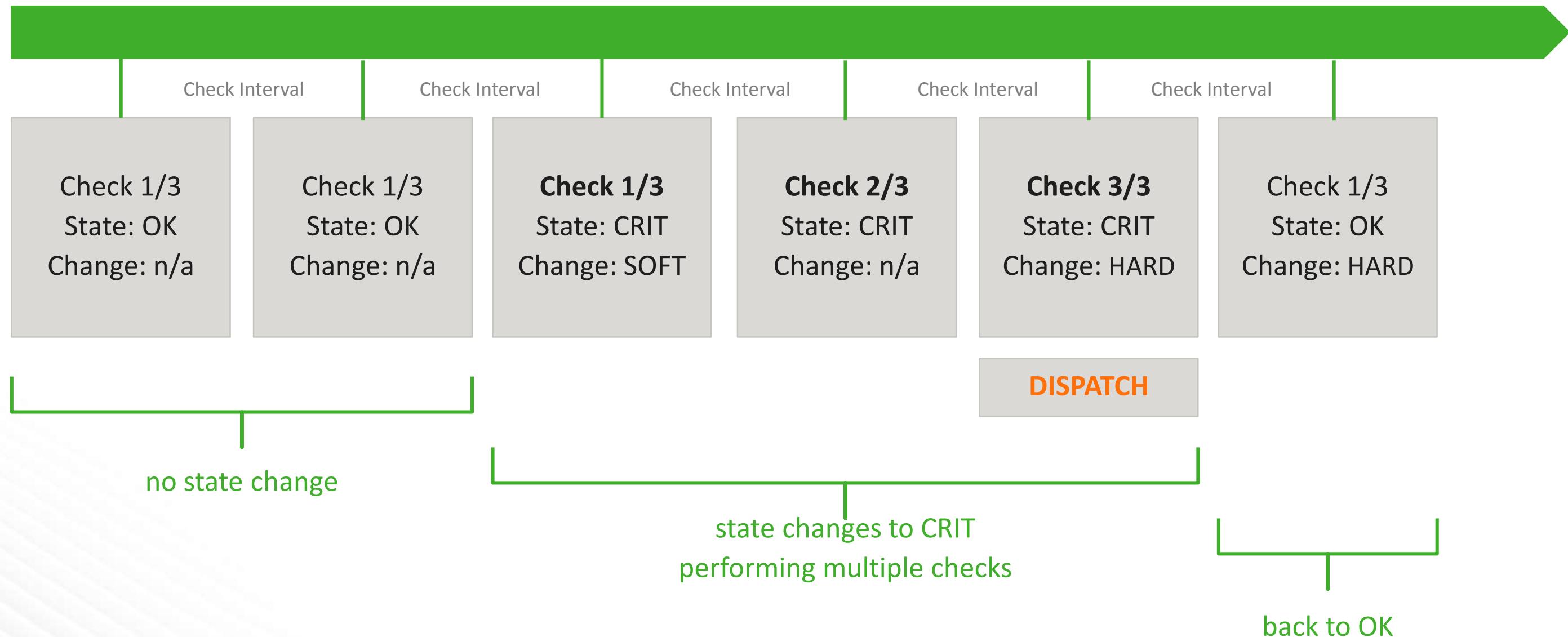
State Change Types

- ◆ SOFT state changes do not perform a dispatch
- ◆ HARD state changes (to non-OK) perform dispatch
- ◆ Regardless of change:
 - The Ambari Web UI will show the current state (OK/WARN/CRIT)
 - The state change is written to `ambari-alerts.log`

```
2016-05-31 13:20:52,294 [CRITICAL] [SOFT] [AMBARI_METRICS]
[grafana_webui] (Grafana Web UI) Connection failed to
http://c6401.ambari.apache.org:3000 (<urlopen error [Errno
111] Connection refused>)
2016-05-31 13:22:52,290 [CRITICAL] [HARD] [AMBARI_METRICS]
[grafana_webui] (Grafana Web UI) Connection failed to
http://c6401.ambari.apache.org:3000 (<urlopen error [Errno
111] Connection refused>)
```



Example: Check Count = 3



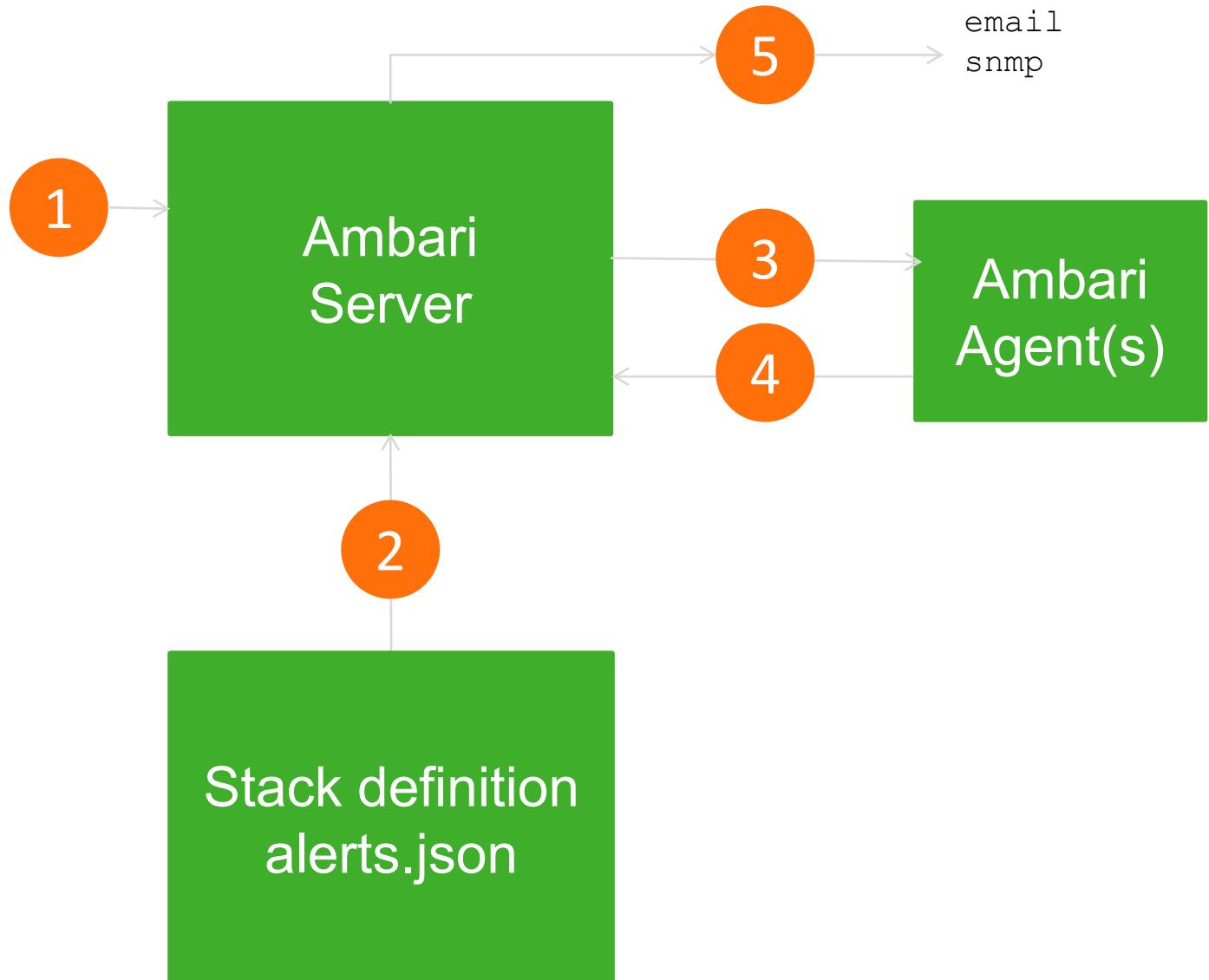
Other Items

- ◆ Alerts Log (AMBARI-10249)
 - Alert state changes are written to /var/log/ambari-server/ambari-alerts.log

```
2015-07-13 14:58:03,744 [OK] [ZOOKEEPER] [zookeeper_server_process] (ZooKeeper Server Process) TCP OK - 0.000s response on port 2181
2015-07-13 14:58:03,768 [OK] [HDFS] [datanode_process_percent] (Percent DataNodes Available) affected: [0], total: [1]
```
- ◆ Script-based Alert Notifications (AMBARI-9919)
 - Define a custom script-based notification dispatcher
 - Executed on alert state changes

Ambari Alerting System

1. User creates or modifies cluster
2. Ambari reads alert definitions from Stack
3. Ambari sends alert definitions to Agents and Agent schedules instance checks
4. Agents reports alert instance status in the heartbeat
5. Ambari responds to alert instance status changes and dispatches notifications (if applicable)



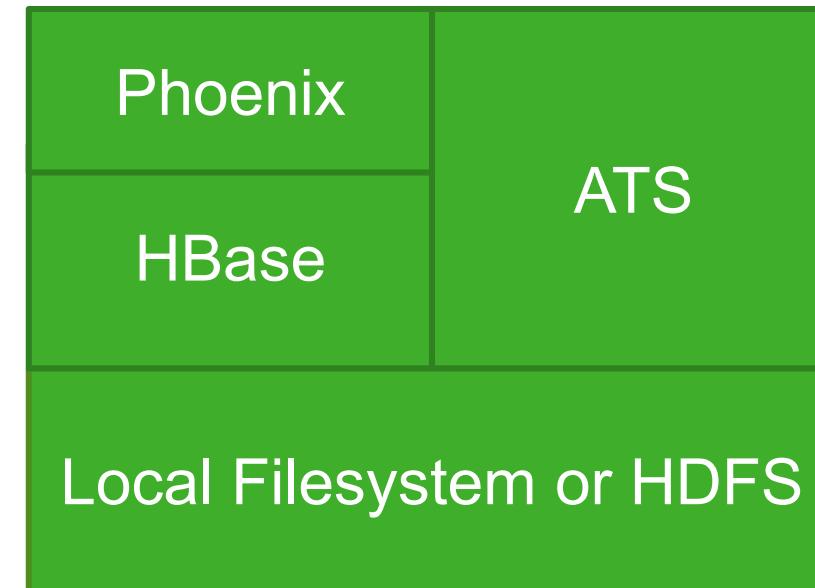
Alert REST APIs

REST Endpoint	Description
/api/v1/clusters/:clusterName/alert_definitions	The list of alert definitions for the cluster.
/api/v1/clusters/:clusterName/alerts	<p>The list of alert instances for the cluster. Use partial response syntax to filter.</p> <p>Example: find all alert instances that are CRITICAL /api/v1/clusters/c1/alerts?Alert/state.in(CRITICAL)</p> <p>Example: find all alert instances for “ZooKeeper Process” alert def /api/v1/clusters/c1/alerts?Alert/definition_name=zookeeper_server_process</p>
/api/v1/clusters/:clusterName/alert_groups	The list of alert groups.
/api/v1/clusters/:clusterName/alert_history	The list of alert instance status changes.
/api/v1/alert_targets/	The list of configured alert notification targets for Ambari.

Ambari Metrics

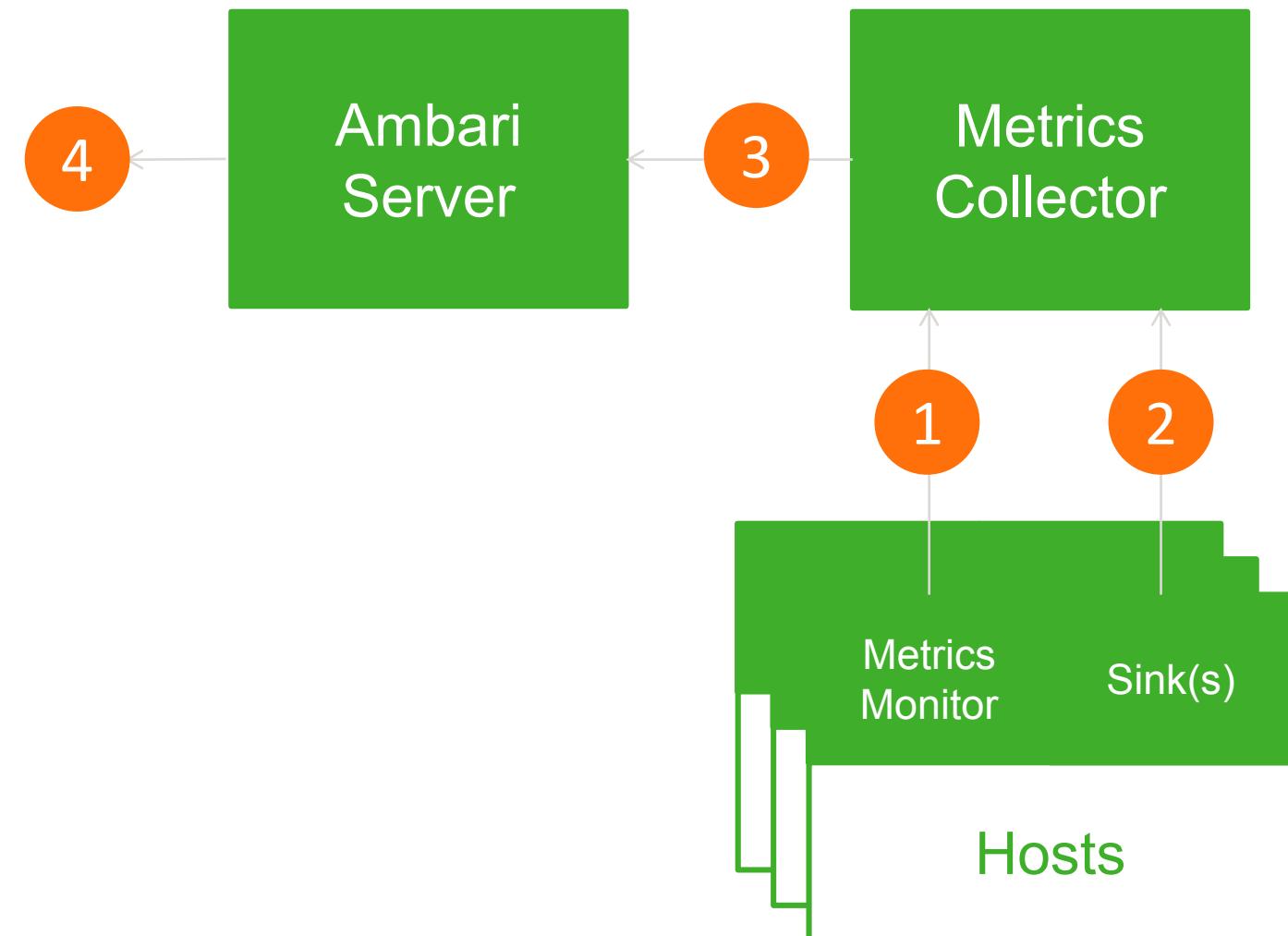
Ambari Metrics

- ◆ Built using Hadoop technologies
- ◆ Three components: Metrics Collector, Metrics Monitors, Grafana



Metrics Collection

1. Metric Monitors send system-level metrics to Collector
2. Sinks send Hadoop-level metrics to Collector
3. Metrics Collector service stores and aggregates metrics
4. Ambari exposes REST API for metrics retrieval



Ambari Metrics + Grafana: Introduced with Ambari 2.2

Highlights

- ◆ Including Grafana as a “Native UI” for Ambari Metrics
- ◆ Including pre-build Dashboards
- ◆ Supports configuring for HTTPS

List of Dashboards

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



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

Metrics to see the overall status of the cluster. Click on each row title.

OVERVIEW - AVERAGES

Logical CPU Count Total

1

Total Memory

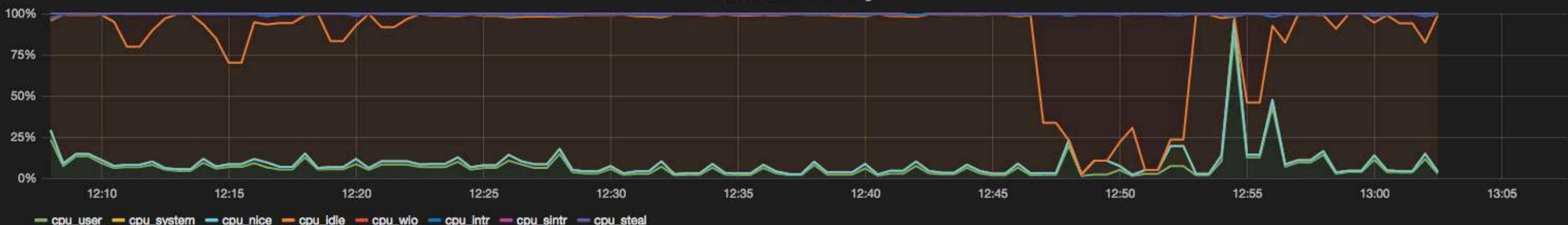
1.834 GiB

Total Disk Space

490 GiB

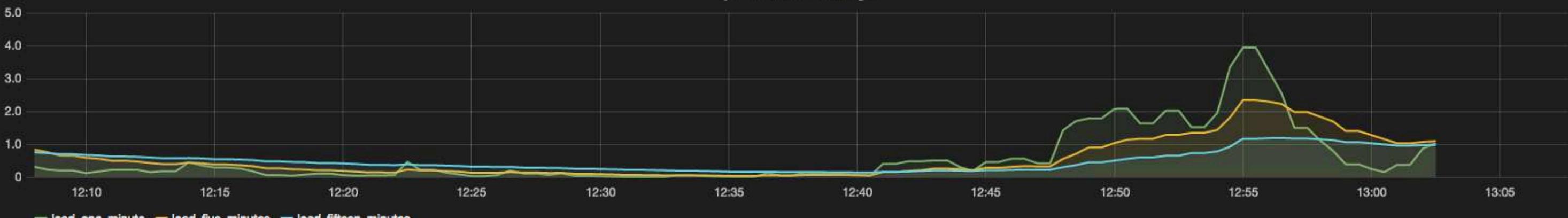
CPU

CPU Utilization - Average



SYSTEM LOAD

System Load - Average

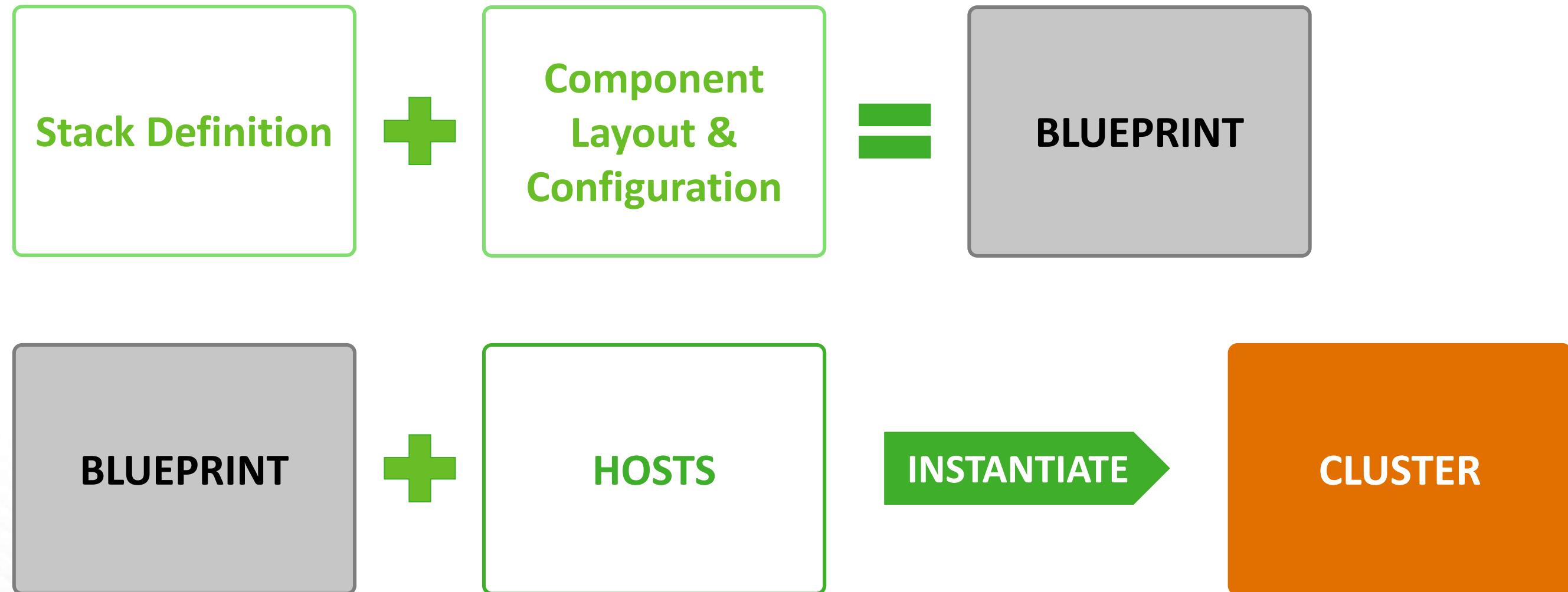


Ambari Blueprints

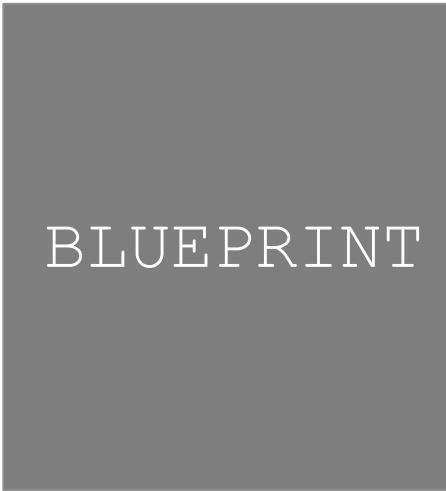
Ambari Blueprints

- ◆ Two primary goals of Ambari Blueprints:
 - Provide API-driven deployments based on self-contained cluster description
 - Ability to export a complete cluster description of a running cluster
- ◆ Blueprints contain cluster topology and configuration information
- ◆ Enables interesting use cases between physical and virtual environments

Ambari Stacks + Blueprints Together



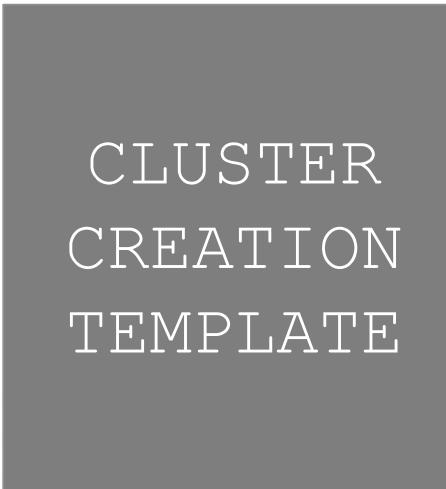
Blueprints API



BLUEPRINT

1

POST /blueprints/my-blueprint



CLUSTER
CREATION
TEMPLATE

2

POST /clusters/MyCluster

Example: Single-Node Cluster

BLUEPRINT

```
{  
  "configurations" : [  
    {  
      "dfs-site" : {  
        "dfs.namenode.name.dir" : "/hadoop/nn"  
      }  
    }  
  ],  
  "host_groups" : [  
    {  
      "name" : "uber-host",  
      "components" : [  
        { "name" : "NAMENODE" },  
        { "name" : "SECONDARY_NAMENODE" },  
        { "name" : "DATANODE" },  
        { "name" : "HDFS_CLIENT" },  
        { "name" : "RESOURCEMANAGER" },  
        { "name" : "NODEMANAGER" },  
        { "name" : "YARN_CLIENT" },  
        { "name" : "HISTORYSERVER" },  
        { "name" : "MAPREDUCE2_CLIENT" }  
      ],  
      "cardinality" : "1"  
    }  
  ],  
  "Blueprints" : {  
    "blueprint_name" : "single-node-hdfs-yarn",  
    "stack_name" : "HDP",  
    "stack_version" : "2.2"  
  }  
}
```

Description

- Single-node cluster
- Use HDP 2.2 Stack
- HDFS + YARN + MR2
- Everything on c6401

CLUSTER CREATION TEMPLATE

```
{  
  "blueprint" : "single-node-hdfs-yarn",  
  "host_groups" : [  
    {  
      "name" : "uber-host",  
      "hosts" : [  
        {  
          "fqdn" : "c6401.ambari.apache.org"  
        }  
      ]  
    }  
  ]  
}
```

Blueprints Add Host

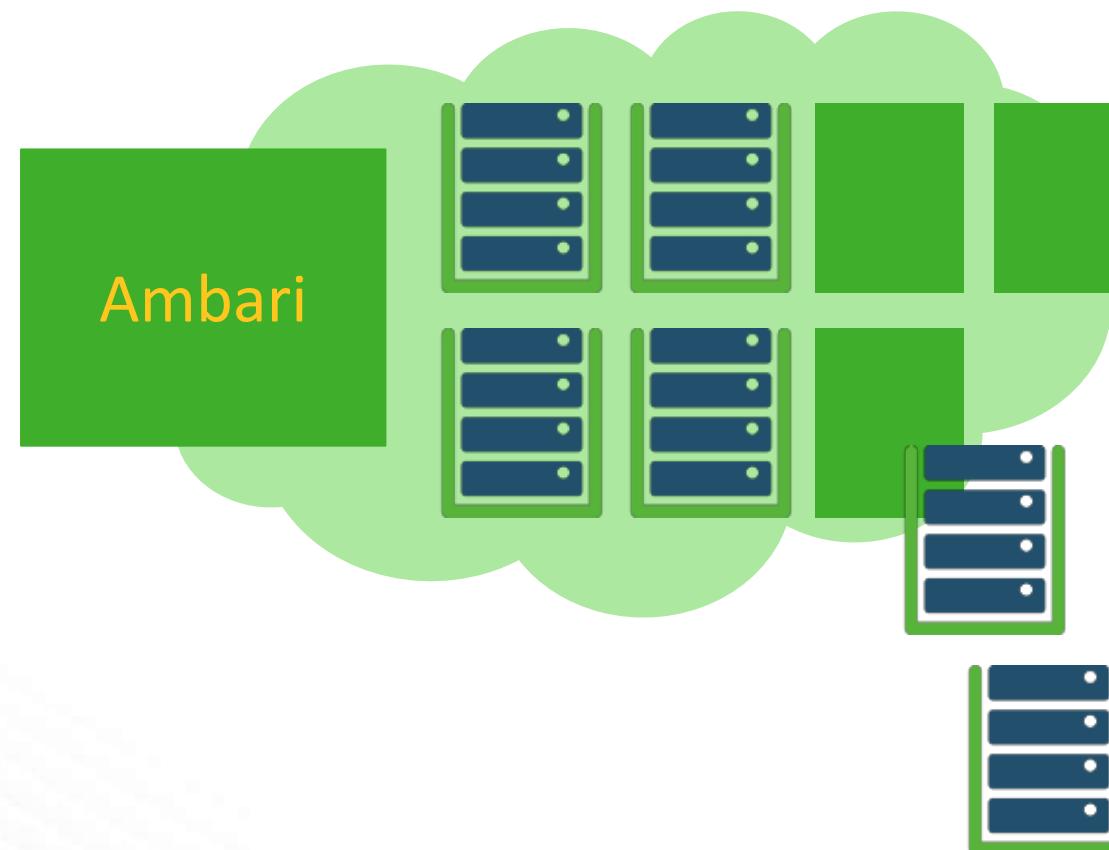
- Add hosts to a cluster based on a host group from a Blueprint
- Add one or more hosts with a single call

```
POST /api/v1/clusters/MyCluster/hosts
{
    "blueprint" : "myblueprint",
    "host_group" : "workers",
    "host_name" : "c6403.ambari.apache.org"
}
```

```
POST /api/v1/clusters/MyCluster/hosts
[
    {
        "blueprint" : "myblueprint",
        "host_group" : "workers",
        "host_name" : "c6403.ambari.apache.org"
    },
    {
        "blueprint" : "myblueprint",
        "host_group" : "edge",
        "host_name" : "c6404.ambari.apache.org"
    }
]
```

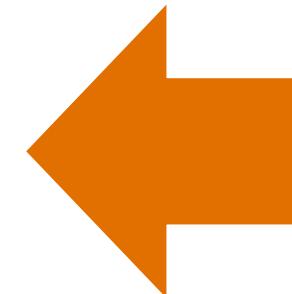
<https://issues.apache.org/jira/browse/AMBARI-8458>

Blueprints Host Discovery (AMBARI-10750)



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"  
      }  
    ]  
  }  
]
```



Where Can I Learn More About Blueprints?

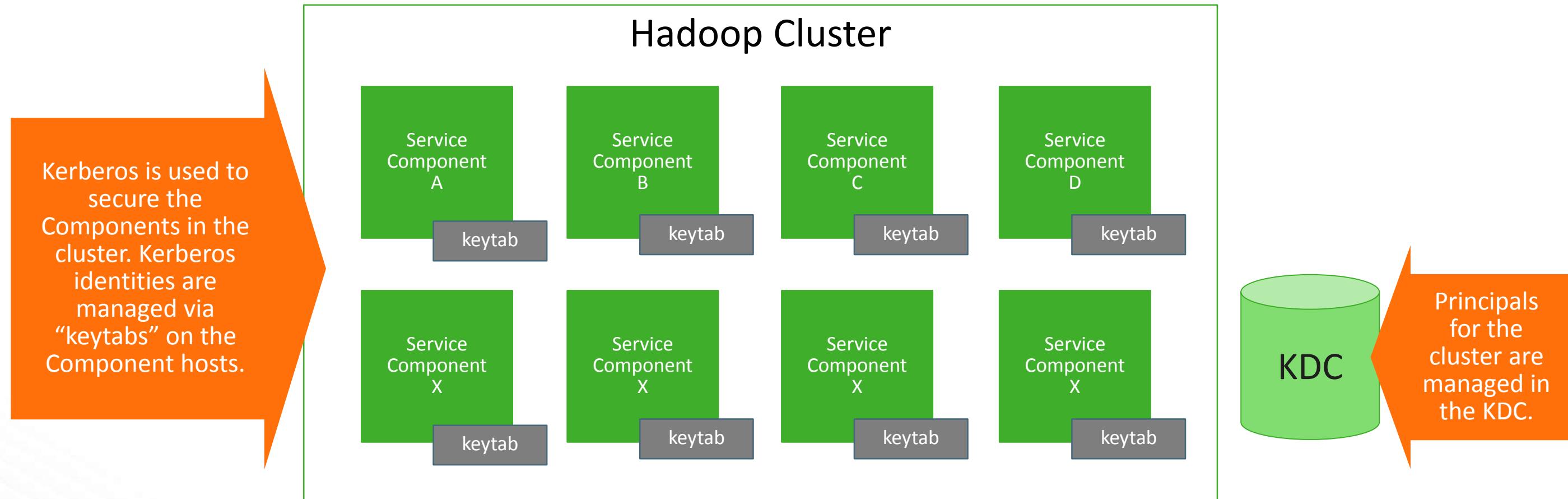
- ◆ <https://cwiki.apache.org/confluence/display/AMBARI/Blueprints>

Kerberos Setup

Background: Hadoop + Kerberos

- Strongly authenticating and establishing a user's identity is the basis for secure access in Hadoop. Users need to be able to reliably "identify" themselves and then have that identity propagated throughout the Hadoop cluster.
- Once this is done, those users can access resources (such as files or directories) or interact with the cluster (like running MapReduce jobs).
- Besides users, Hadoop cluster resources themselves (such as Hosts and Services) need to authenticate with each other to avoid potential malicious systems or daemon's "posing as" trusted components of the cluster to gain access to data.

Background: Hadoop + Kerberos



Automated Kerberos Security Setup

Wizard Driven

Simplify the experience of enabling and managing Kerberos security

Automated and Manual Options

Critical for today's enterprise

Works with Existing Kerberos Infrastructure

Such as Active Directory and MIT KDC

Get Started

Welcome to the Ambari Security Wizard. Use this wizard to enable kerberos security in your cluster. Let's get started.

Note: This process requires services to be restarted and cluster downtime. As well, depending on the options you select, might require support from your Security administrators. Please plan accordingly.

What type of KDC do you plan on using?

- Existing MIT KDC
- Existing Active Directory
- Manage Kerberos principals and keytabs manually

Manage Kerberos principals and keytabs manually:

Following prerequisites needs to be checked to progress ahead in the wizard.

- Cluster hosts have network access to the KDC
- Kerberos client utilities (such as kinit) have been installed on every cluster host
- The Java Cryptography Extensions (JCE) have been setup on the Ambari Server host and all hosts in the cluster
- The Service and Ambari Principals will be manually created in the KDC before completing this wizard
- The keytabs for the Service and Ambari Principals will be manually created and distributed to cluster hosts before completing this wizard

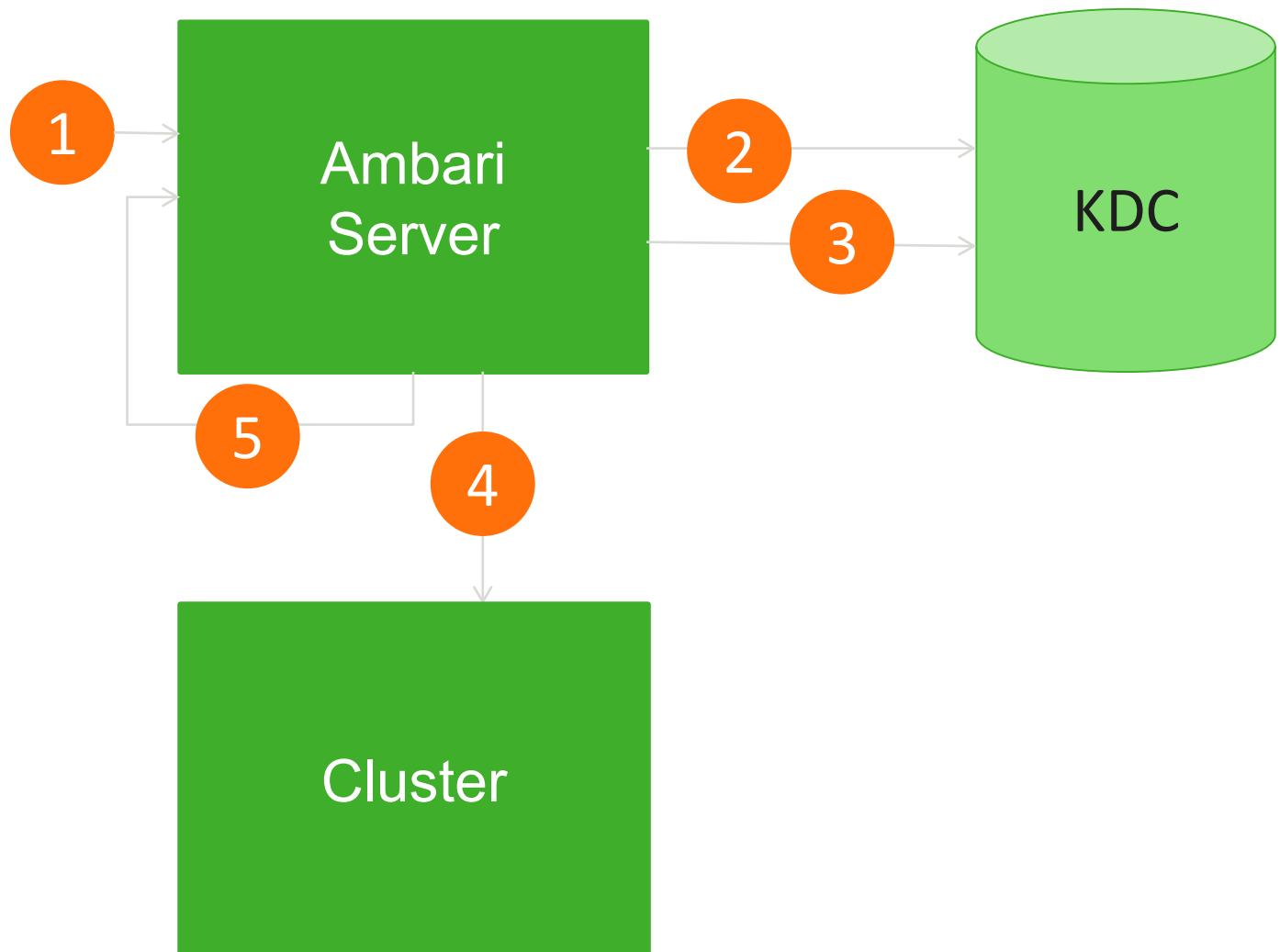
Next →

Automated vs. Manual Kerberos Options

	Automated	Manual
KDC Infrastructure	MIT, Active Directory	MIT, Active Directory, FreeIPA
Requires KDC administrative credentials	Yes	No
Installation of Kerberos clients	Yes, optional	No
Management of Kerberos client krb5.conf	Yes, optional	No
Creation of principals	Yes	No
Creation of keytabs	Yes	No
Distribution of keytabs	Yes	No
Cluster configuration	Yes	Yes

Automated: Principals and Keytabs

1. User provides KDC Admin Account credentials to Ambari
2. Ambari connects to KDC, creates principals (Service and Ambari) needed for cluster
3. Ambari generates keytabs for the principals
4. Ambari distributes keytabs to Ambari Server and cluster hosts
5. Ambari discards the KDC Admin Account credentials



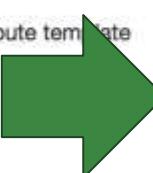
Automated: Customizable Principal Attributes

- Customize the directory principal attributes for your environment
- Particularly useful when using Active Directory for Kerberos

Advanced kerberos-env

Encryption Types: aes des3-cbc-sha1 rc4 des-cbc-md5

Attribute template:



```
{  
  "objectClass": ["top", "person", "organizationalPerson", "user"],  
  "cn": "$principal_name",  
  #if $is_service  
  "servicePrincipalName": "$principal_name",  
  #end  
  "userPrincipalName": "$normalized_principal",  
  "unicodePwd": "$password",  
  "accountExpires": "0",  
  "userAccountControl": "66048"  
}
```

Attribute Variables	Example
\$normalized_principal	nn/c6401.ambari.apache.org@EXAMPLE.COM
\$principal_name	nn/c6401.ambari.apache.org
\$principal_primary	nn
\$principal_digest	[[MD5 hash of the \$normalized_principal]]
\$principal_instance	c6401.ambari.apache.org
\$realm	EXAMPLE.COM
\$password	[[password]]

Automated: Kerberos Clients

- Option to manage krb5.conf
- Option to install clients

Advanced krb5-conf

- Manage Kerberos client
krb5.conf
- Install OS-specific
Kerberos client
package(s)

Install and Test Kerberos Client

Please wait while the Kerberos clients are being installed and tested.

✓ Install Kerberos Client

⚙ Test Kerberos Client



← Back

Next→

OS	Client
RHEL/CentOS/OEL	krb5-workstation
SLES 11	krb5-client
Ubuntu 12	krb5-user, krb5-config

Automated: Kerberos Operations

- Once enabled, Ambari works directly with the Kerberos infrastructure to automate these common tasks, removing the burden from the operator:
- Add/Delete Host
- Add Service
- Add/Delete Component
- Regenerate Keytabs
- Disable Kerberos

Kerberos security is enabled

[Disable Kerberos](#)

[C Regenerate Keytabs](#)

Manual: Specify Realm, Client Utilities Path

Configure Kerberos

Please configure kerberos related properties.

Kerberos

▼ KDC

KDC type

Manage Kerberos principals and keytabs
manually

Realm name

EXAMPLE.COM



▼ Advanced kerberos-env

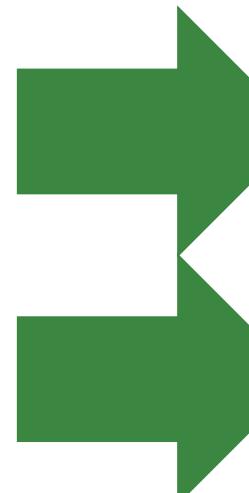
Executable Search Paths

/usr/bin, /usr/kerberos/bin, /usr/sbin, /usr/lib/mit/bin, /usr/lib/mit

All configurations have been addressed.

← Back

Next →



Manual: Principals and Keytabs

- Configure Identities
- Download CSV

The screenshot shows a Microsoft Excel spreadsheet titled "kerberos-1.csv". The table has 13 columns labeled A through M. Column A contains host names, column B contains descriptions, column C contains principal names, and column D contains types. Columns E through M contain keytab file paths and other metadata. The data includes entries for various Ambari services like http, ambari-smokeuser, ambari-ams-hbase, ambari-dn, ambari-hdfs, ambari-history-server, ambari-nodemanager, ambari-secondary-nm, ambari-resource-man, ambari-app-timeline-yarn, and ambari-ams-zookeeper.

A	B	C	D	E	F	G	H	I	J	K	L	M
host	description	principal name	principal type	local user	keytab file path	keytab file owner	keytab file group	keytab file permissions	keytab file installed			
c6402.ambari	/spnego	HTTP/c6402.ambari.apache.org@EXAMPLE.COM	SERVICE		/etc/security/keytabs/spnego.service.keytab	root	r	hadoopX	r	440	unknown	
c6402.ambari	/smokeuser	ambari-qax@EXAMPLE.COM	USER	ambari-qax	/etc/security/keytabs/smokeuser.headless.keytab	ambari-qaxX	r	hadoopX	r	440	unknown	
c6402.ambari_ams_hbase_m	amshbase/c6402.ambari.apache.org@EXAMPLE.COM	amsX	SERVICE		/etc/security/keytabs/ams-hbase.master.keytab	amsX	r	hadoopX		400	unknown	
c6402.ambari_datanode_dn	dn/c6402.ambari.apache.org@EXAMPLE.COM	hdfsX	SERVICE		/etc/security/keytabs/dn.service.keytab	hdfsX	r	hadoopX		400	unknown	
c6402.ambari_hdfs	hdfs@EXAMPLE.COM	USER		hdfsX	/etc/security/keytabs/hdfs.headless.keytab	hdfsX	r	hadoopX	r	440	unknown	
c6402.ambari_history_server_jhs	jhs/c6402.ambari.apache.org@EXAMPLE.COM	mapredX	SERVICE		/etc/security/keytabs/jhs.service.keytab	mapredX	r	hadoopX		400	unknown	
c6402.ambari_nodemanager_nm	nm/c6402.ambari.apache.org@EXAMPLE.COM	yarnX	SERVICE		/etc/security/keytabs/nm.service.keytab	yarnX	r	hadoopX		400	unknown	
c6402.ambari_secondary_nm	nn/c6402.ambari.apache.org@EXAMPLE.COM	hdfsX	SERVICE		/etc/security/keytabs/nn.service.keytab	hdfsX	r	hadoopX		400	unknown	
c6402.ambari_resource_man_rm	rm/c6402.ambari.apache.org@EXAMPLE.COM	yarnX	SERVICE		/etc/security/keytabs/rm.service.keytab	yarnX	r	hadoopX		400	unknown	
c6402.ambari_app_timeline_yarn	yarn/c6402.ambari.apache.org@EXAMPLE.COM	yarnX	SERVICE		/etc/security/keytabs/yarn.service.keytab	yarnX	r	hadoopX		400	unknown	
c6402.ambari_ams_zookeeper	zkservice.ams.keytab	amsX	SERVICE		/etc/security/keytabs/zk.service.ams.keytab	amsX	r	hadoopX		400	unknown	

Confirm Configuration

Please review the configuration before continuing the setup process

Important: Use the Download CSV button to obtain a list of the required principals and keytabs that are needed by Ambari to enable Kerberos in the cluster. Do not proceed until you have manually created and distributed the principals and keytabs to the cluster hosts.

Cluster Name: MyCluster

Manage Kerberos principals and keytabs manually: true

```
content: [libdefaults] renew_lifetime = 7d forwardable = true default_realm = {{realm|upper()}} ticket_lifetime = 24h
dns_lookup_realm = false dns_lookup_kdc = false #default_tgs_enctypes = {{encryption_types}}
#default_tkt_enctypes = {{encryption_types}} {% if domains %} [domain_realm] {% for domain in domains.split(',') %} {{domain}} = {{realm|upper()}} {% endfor %} {% endif %} [logging] default = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log kdc = FILE:/var/log/krb5kdc.log [realms] {{realm}} = { admin_server =
{{admin_server_host|default(kdc_host, True)}} } kdc = {{kdc_host}} } # Append additional realm declarations below #
```

kdc_type: Manage Kerberos principals and keytabs manually

realm: EXAMPLE.COM

manage_identities: false

install_packages: false

executable_search_paths: /usr/bin, /usr/kerberos/bin, /usr/sbin, /usr/lib/mit/bin, /usr/lib/mit/sbin

encryption_types: aes des3-cbc-sha1 rc4 des-cbc-md5

Exit Wizard

Download CSV

← Back

Next →

Role-Based Access Control

Role-Based Access Control

- ◆ Use “roles” for more granular division of control for cluster operations

Old Permission	New Role	Notable Permissions
Operator	Cluster Administrator	Full operational control, including upgrades. Ambari Admins are implicitly granted this Role.
	Cluster Operator	Adding and removing hosts.
	Service Administrator	Manage configurations, move components.
	Service Operator	Service stop and start and service-specific operations such as HDFS Rebalance.
Read-Only	Cluster User	View cluster service and host information.

Note: Users flagged as “Ambari Admins” are implicitly granted Cluster Administrator permission.

Managing Cluster Roles

The screenshot shows the 'MyCluster Roles' management interface. On the left, a sidebar menu includes 'Clusters' (selected), 'MyCluster' (selected), 'Roles' (selected), 'Go to Dashboard', 'Versions', and 'Remote Clusters'. Below these are 'Views' (selected), 'Views', and 'View URLs'. At the bottom is 'User + Group Management' (selected), with 'Users' and 'Groups' options.

The main area, titled 'MyCluster Roles', displays five roles: 'Cluster Administrator', 'Cluster Operator', 'Service Administrator', 'Service Operator', and 'Cluster User'. Each role has two sections: 'Assign roles to these users' (containing user names like 'clusteradmin', 'clusterop', etc.) and 'Assign roles to these groups' (containing group names like 'clusteradmins', 'Add Group', etc.). A 'BLOCK LIST' button is located at the top right of the main area.

Annotations with orange arrows point to specific features:

- An arrow points from the text 'Assign roles to users or groups' to the 'Assign roles to these users' section of the first role row.
- An arrow points from the text 'Manage roles in Block or List View layouts' to the 'BLOCK LIST' button at the top right.

Managing Cluster Roles

The screenshot shows the Ambari User + Group Management interface for the 'MyCluster' cluster. The left sidebar includes links for Clusters, Views, and User + Group Management. The main area displays 'MyCluster Roles' with tabs for 'Users' (selected) and 'Groups'. A search bar allows filtering by 'Name' (set to 'Any') and 'Role' (set to 'All'). The table lists users with their assigned roles:

Name	Role
admin	Ambari Administrator
clusteradmin	Cluster Administrator
clusterop	Cluster Operator
clusteruser	None
joe	None
mike	None
pat	None
serviceadmin	None
serviceop	Service Operator

Annotations with orange arrows point to the 'Users' tab, the 'Role' dropdown, and the role assignment for the 'clusteradmin' user.

View users or groups

Change current role assignment

Name	Role
admin	Ambari Administrator
clusteradmin	Cluster Administrator
clusterop	Cluster Operator
clusteruser	None
joe	None
mike	None
pat	None
serviceadmin	None
serviceop	Service Operator

9 of 9 users showing - clear filters

10 ▾ Previous 1 Next

TECH PREVIEW

Log Search

Log Search

Search cluster Component Logs from within Ambari!



Goal: *When issues arise, be able to quickly find issues across all cluster components*

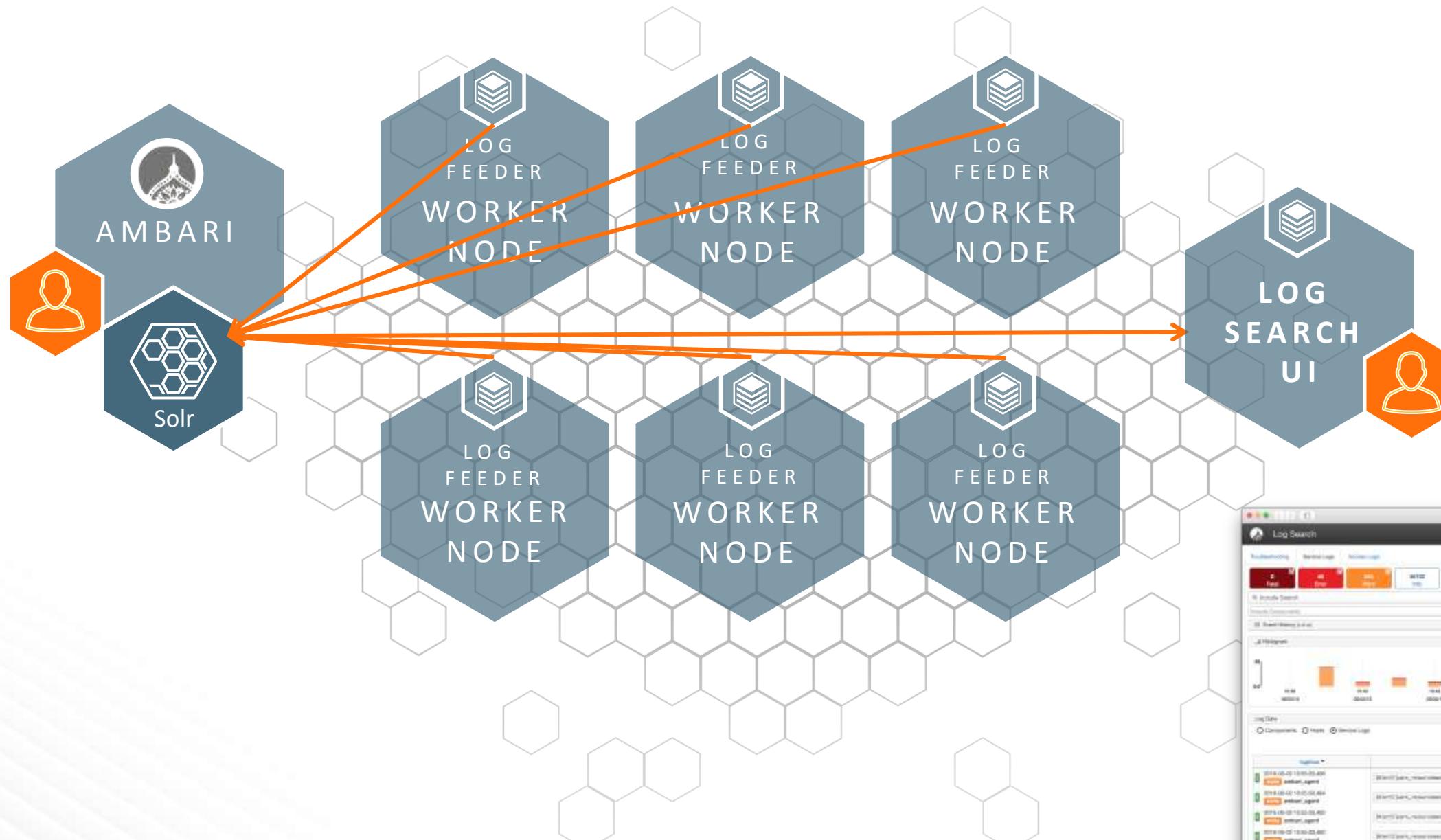
- ◆ **Capabilities**

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

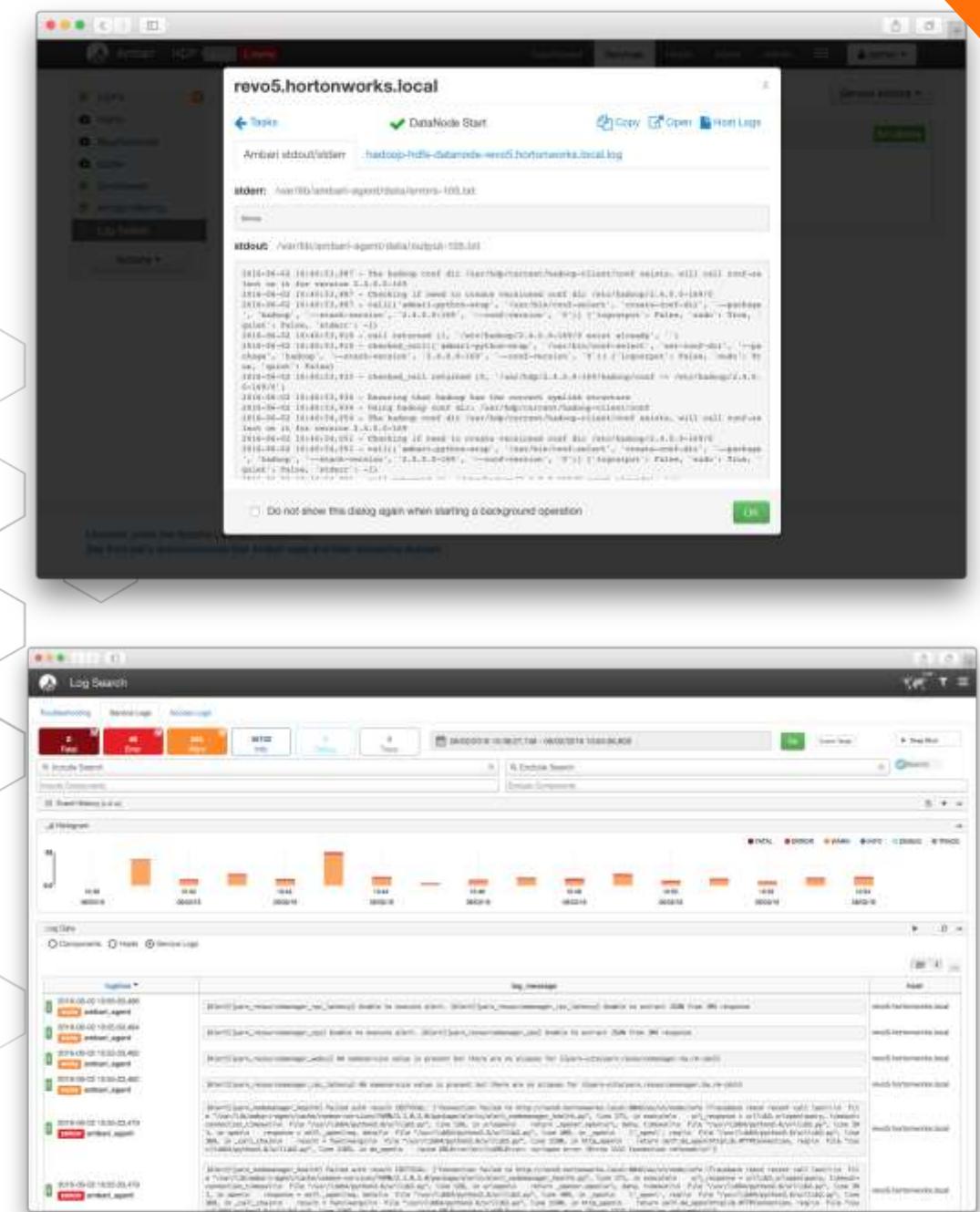
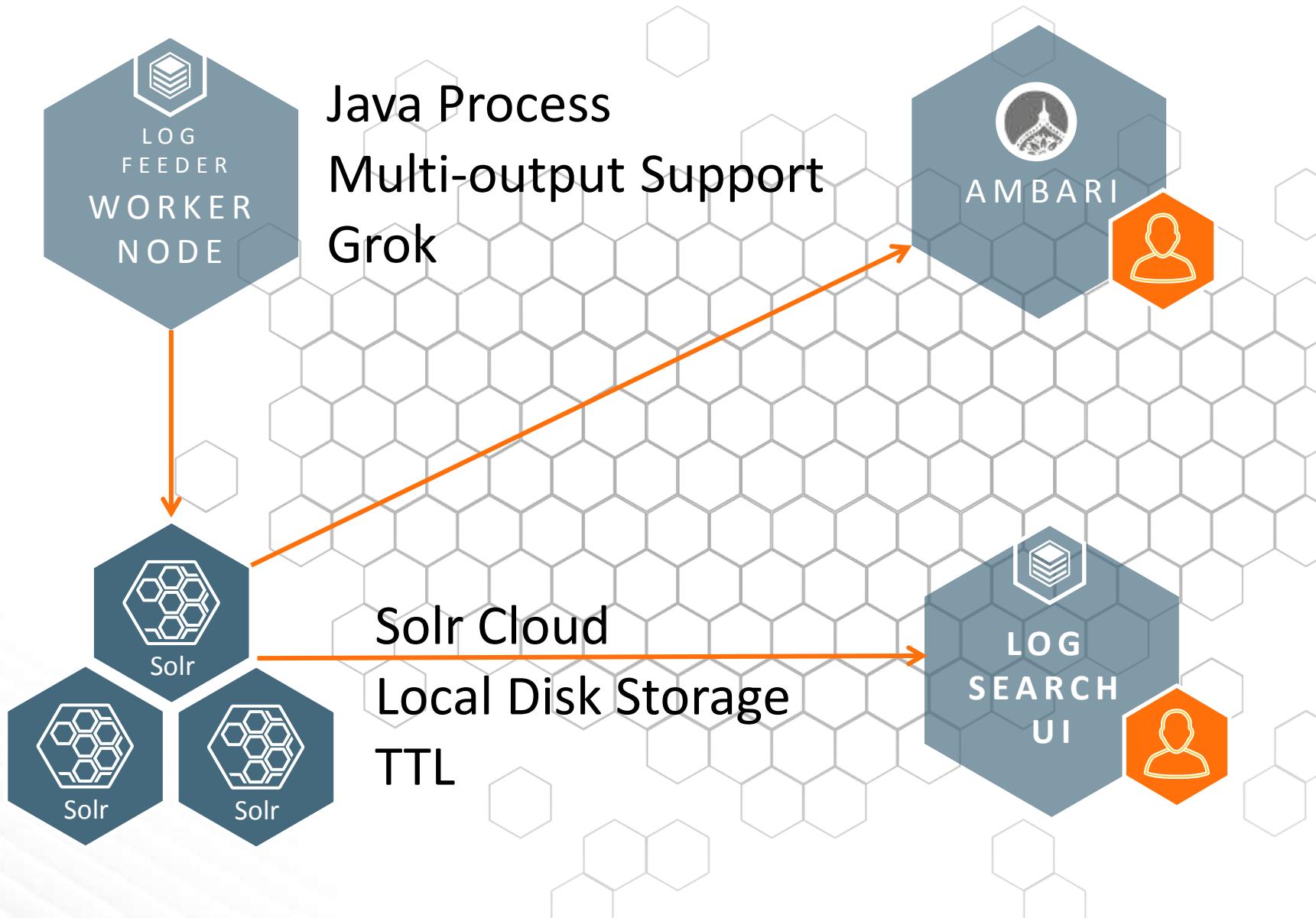
- ◆ **Core Technologies:**

- Apache Ambari
- Apache Solr
- Apache Ambari Log Search

Log Search Architecture



Log Search Details



Automated Cluster Upgrade

Automated Cluster Upgrade Terminology

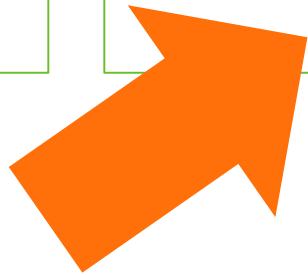
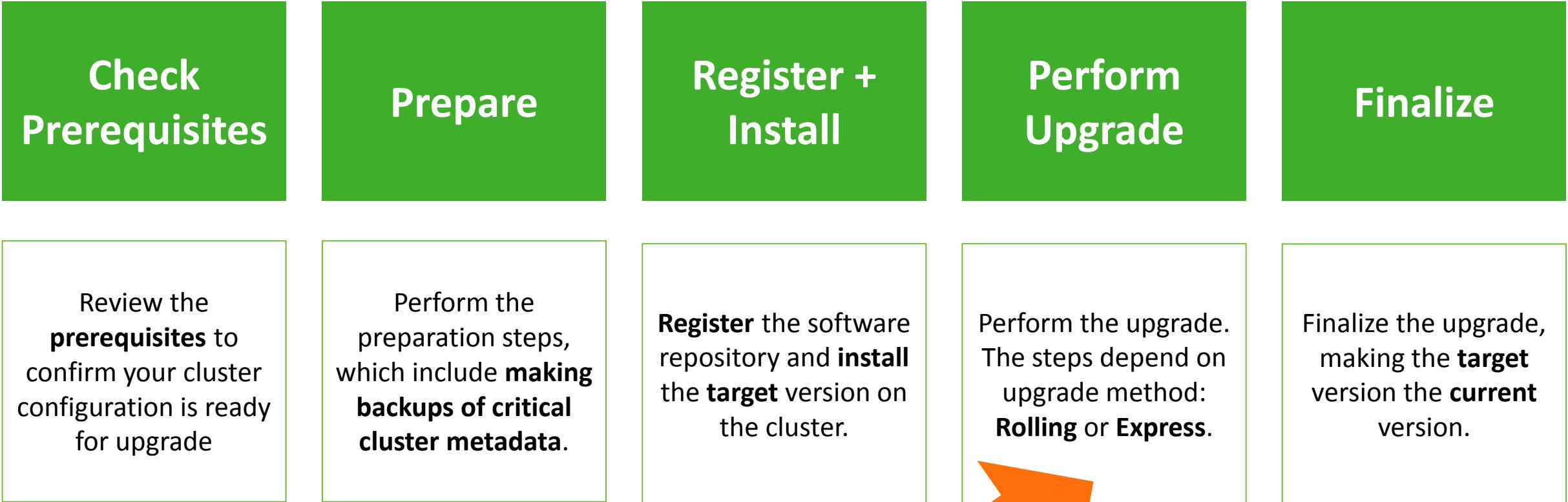
Rolling Upgrade

Orchestrates the cluster upgrade in an order that is meant to preserve cluster operation and minimize service impact. The tradeoff is more stringent prerequisites and a longer upgrade time.

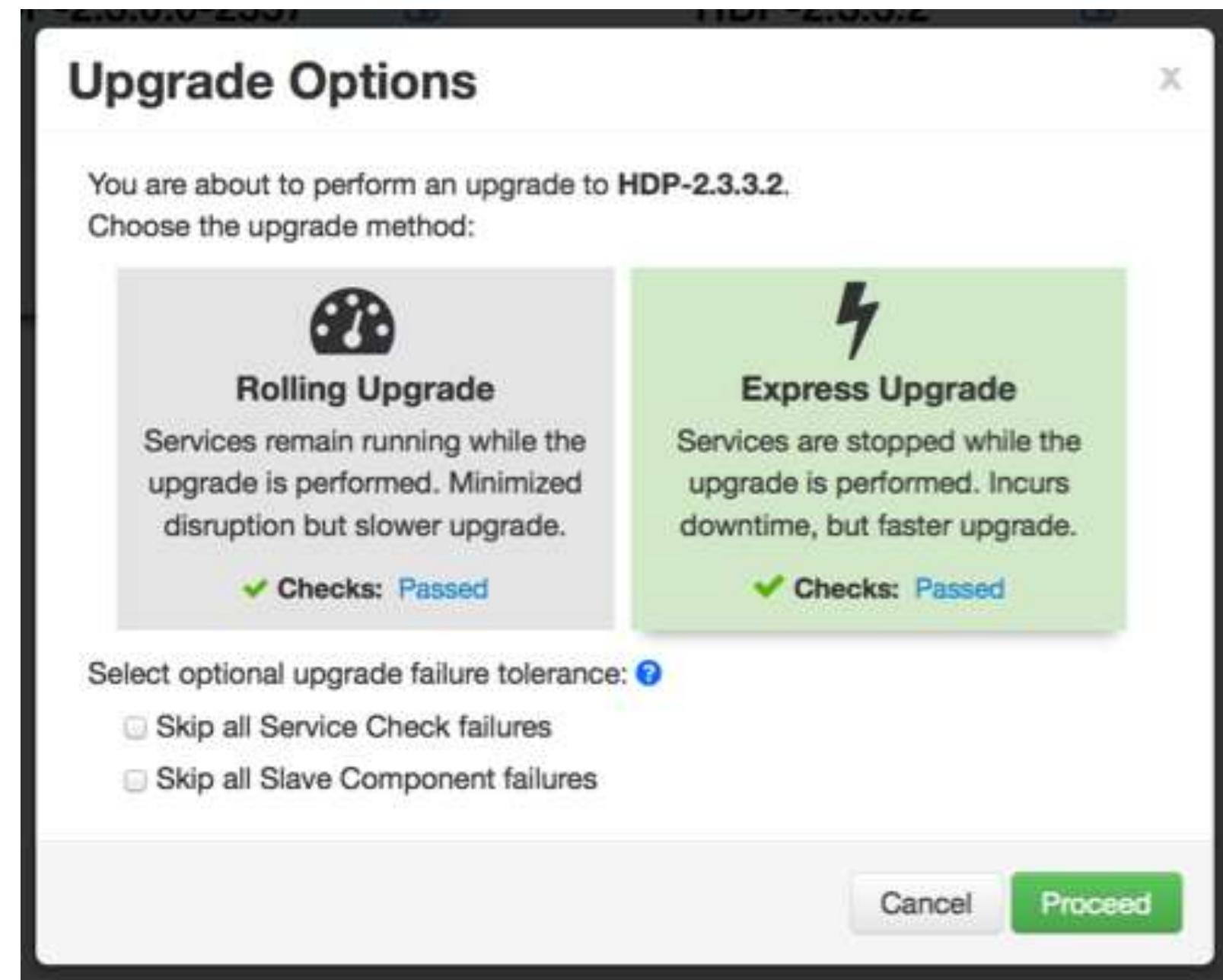
Express Upgrade

Orchestrates the cluster upgrade in an order that will incur cluster downtime. This method has less stringent prerequisites but completes in a faster upgrade time.

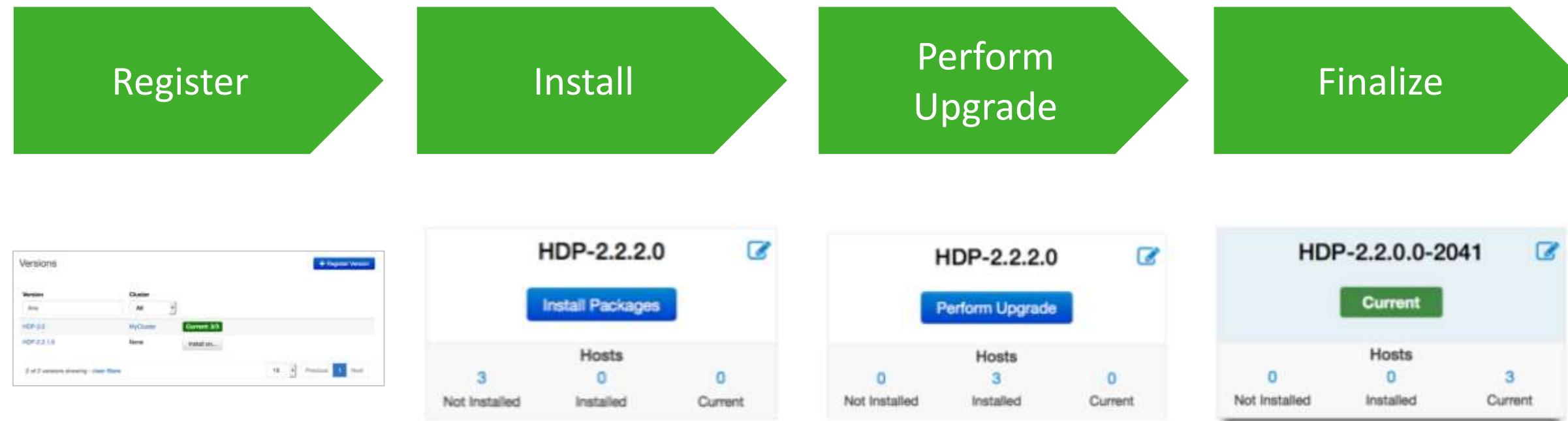
Automated Upgrade: Rolling or Express Choice



Automated Upgrade Choice: Rolling or Express



Automated Upgrade: High-Level Process



Register New Version

Register

Install

Perform Upgrade

Finalize

- Register new version with Ambari
- Provide the Repository Base URLs for new version

 Manage Versions

 Register Version

Versions / Register Version

Details

Name

HDP-2.2

2.0

Repositories

Provide Base URLs for the Operating Systems you are configuring. Uncheck all other Operating Systems.

OS	Name	Base URL
----	------	----------

<input type="checkbox"/> redhat5	HDP	
	HDP-UTILS	

<input checked="" type="checkbox"/> redhat6	HDP	http://c6404.ambari.apache.org/hdp/HDP-2.2.1.0/
	HDP-UTILS	http://c6404.ambari.apache.org/hdp/centos6/HDP-UTILS-1.1.0.20/

<input type="checkbox"/> suse11	HDP	
	HDP-UTILS	

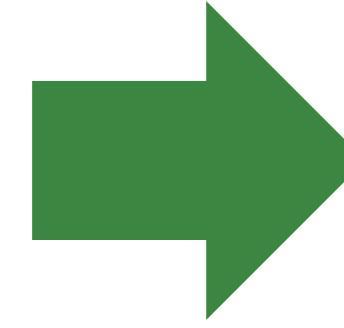
<input type="checkbox"/> ubuntu12	HDP	
	HDP-UTILS	

Skip Repository Base URL validation (Advanced) 

Cancel

Save

Install Packages



Install version

Operations Hosts Show: All (3)

Host	Status	Progress
c6401.ambari.apache.org	Success	100%
c6402.ambari.apache.org	Success	100%
c6403.ambari.apache.org	Success	100%

Show: 10 1 - 3 of 3 OK

Do not show this dialog again when starting a background operation

- Click “Install Packages” to start the install of the version across hosts

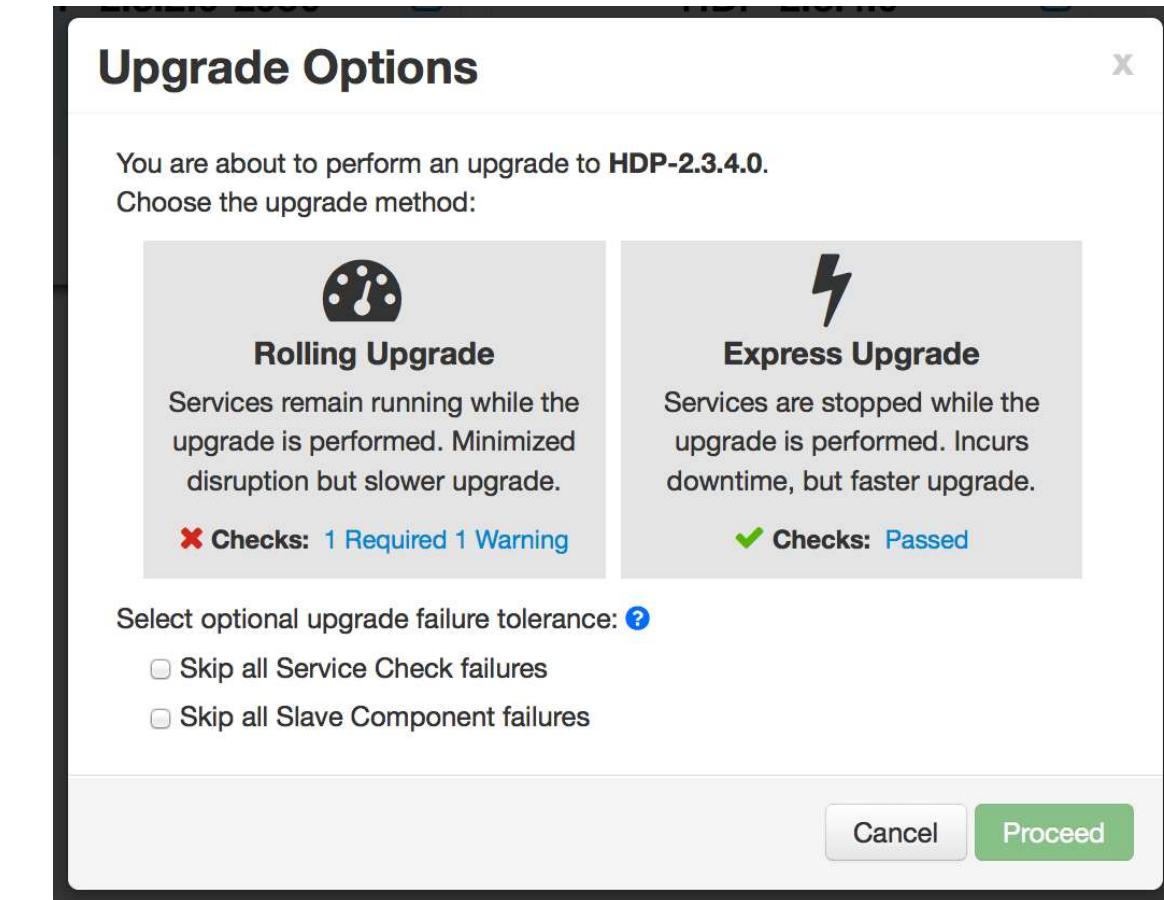
Upgrade Choice

Register

Install

Perform Upgrade

Finalize



- ◆ Upgrade Options dialog gives you a choice of method: **Rolling or Express**

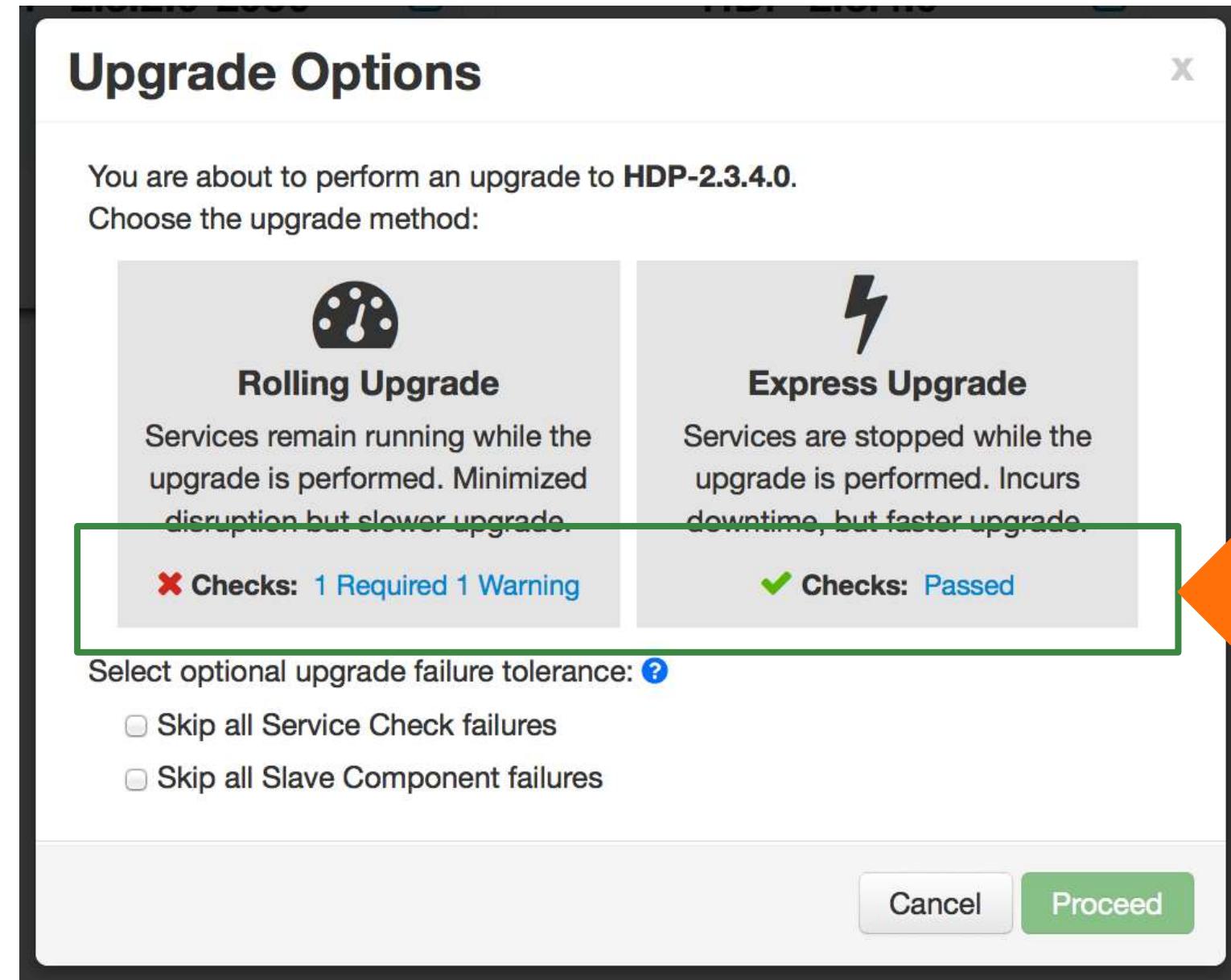
Upgrade Choice: Rolling or Express

Register

Install

Perform Upgrade

Finalize



Checks indicate if an option is available

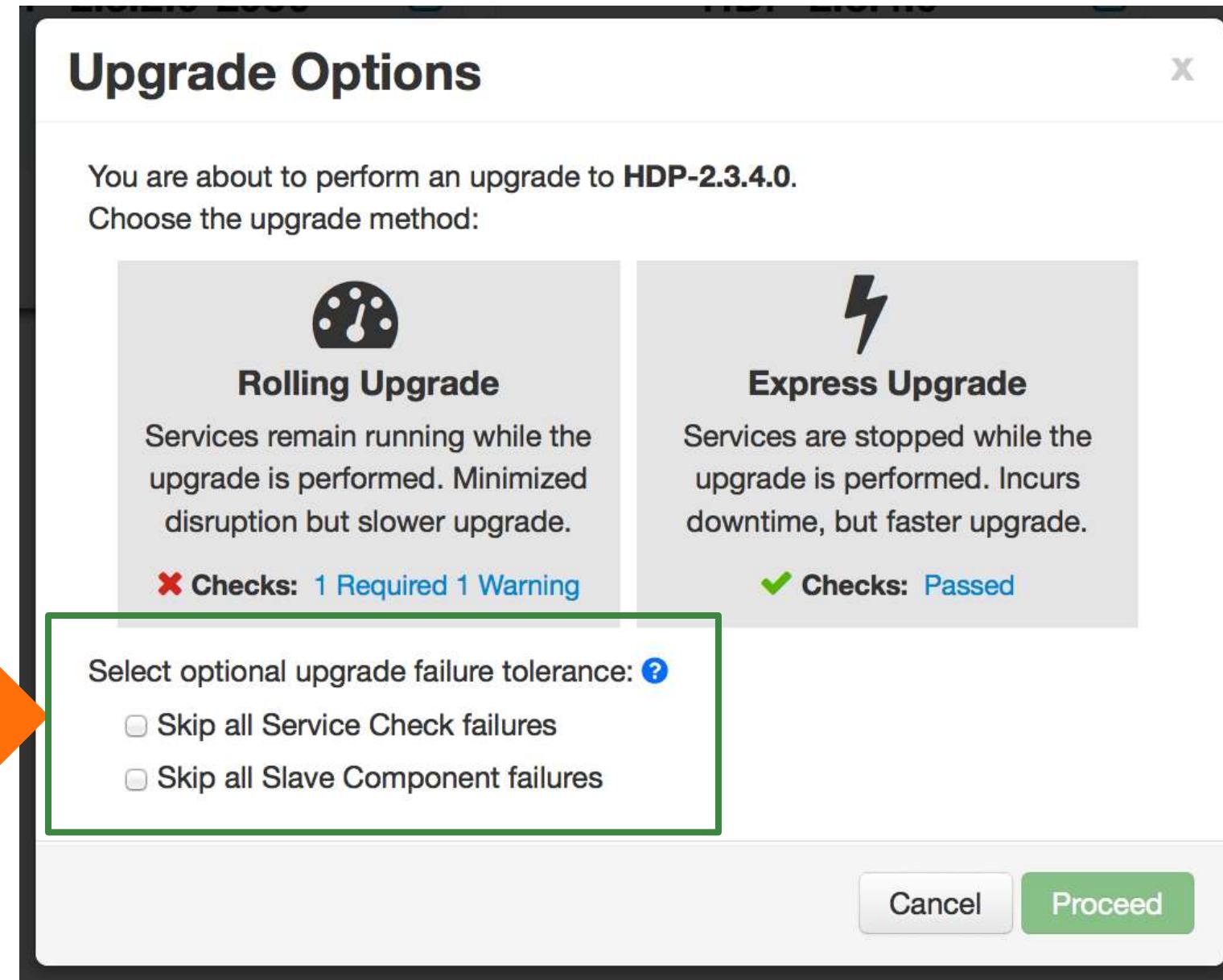
Upgrade Choice: Rolling or Express

Register

Install

Perform Upgrade

Finalize



Upgrade Tolerance Options

Register

Install

Perform Upgrade

Finalize

Tolerance Option	Description
Skip all Service Check failures	Ambari will automatically skip any Service Check failures and complete the task without requiring user intervention to continue. After all the Service Checks have run in a task, you will be presented with summary of the failures and an option to continue the upgrade or pause.
Skip all Slave Component failures	Ambari will automatically skip any Slave Component failures and complete the task of upgrading Slave components without requiring user intervention to continue. After all Slave Components have been upgraded, you will be presented with a summary of the failures and an option to continue the upgrade or pause.

Wizard Driven Experience

Register

Install

Perform Upgrade

Finalize

Upgrade to HDP-2.2.1.0

II Upgrade Paused

55%

Manual steps required

Core Services have been upgraded. You are advised to perform tests against your cluster to ensure proper operation before proceeding with upgrade of remaining services.

I have performed the manual steps above.

Downgrade

Do This Later

Proceed

With verification
and validation

II Core Slaves

II Verification Required

- ✓ Restarting NODEMANAGER on c6403.ambari.apache.org
- ✓ Restarting DATANODE on c6403.ambari.apache.org
- ✓ Full Service Check
- ✓ Core Masters
- ✓ Zookeeper

Dismiss

Rolling Upgrade – Success!

Register

Install

Perform
Upgrade

Finalize

Ambari MyCluster 0 ops 0 alerts

Dashboard Services Hosts Alerts Admin admin

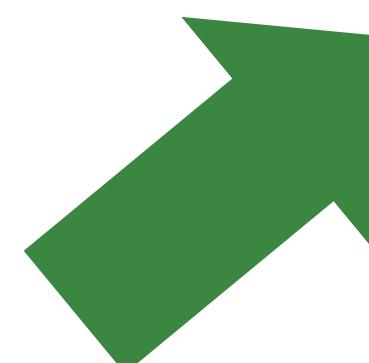
Stack and Versions Stack Versions Filter: All (2) Manage Versions

HDP-2.2.0.0-2041 Installed

	Hosts	
Not Installed	0	
Installed	3	
Current	0	

HDP-2.2.2.0 Current

	Hosts	
Not Installed	0	
Installed	0	
Current	3	



Ambari Extensibility

Extensibility Features

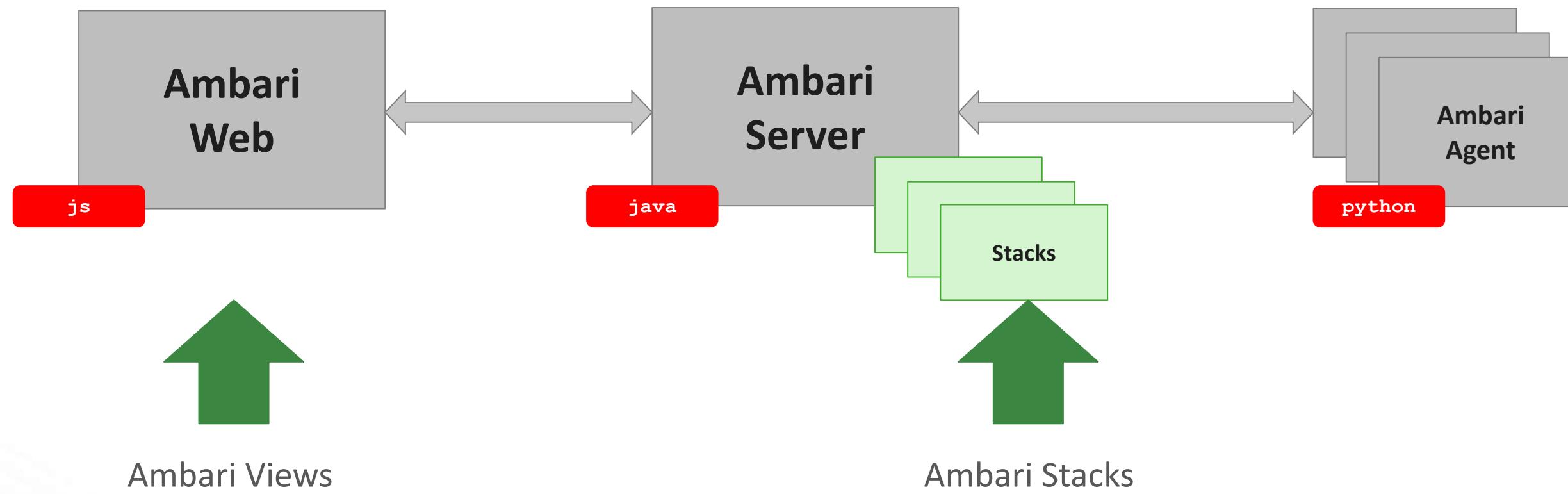
Stacks

- To add new Services (ISV or otherwise) beyond Stack
- To customize a Stack for customer specific environments

Views

- To extend and customize the Ambari Web UI
- Add new capabilities, customize existing capabilities

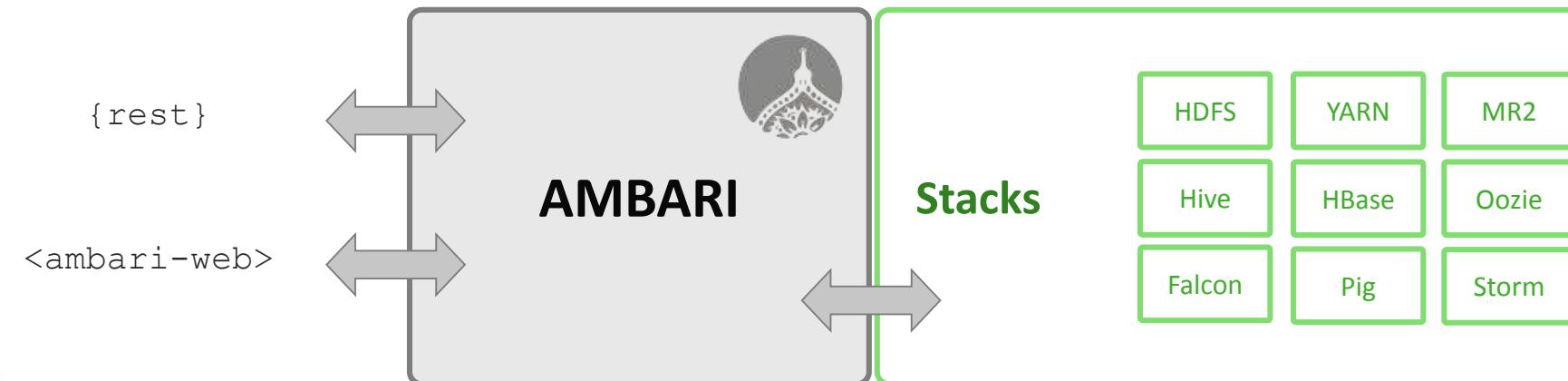
Anatomy of Ambari Extension Points



Ambari Stacks

Ambari Stacks

- ◆ Defines a consistent Stack lifecycle interface that can be extended
- ◆ Encapsulates Stack Versions, Services, Components, Dependencies, Cardinality, Configurations, Commands
- ◆ Dynamically add Stack + Service definitions

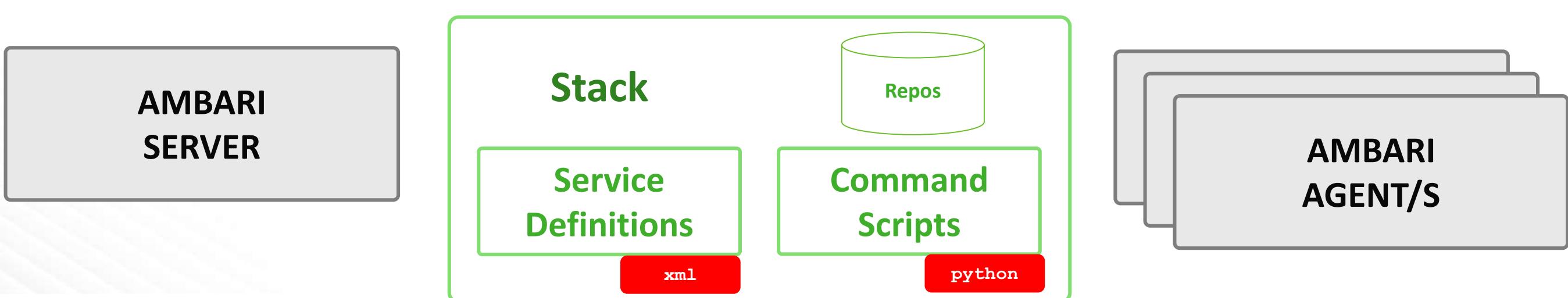


Stack Terminology

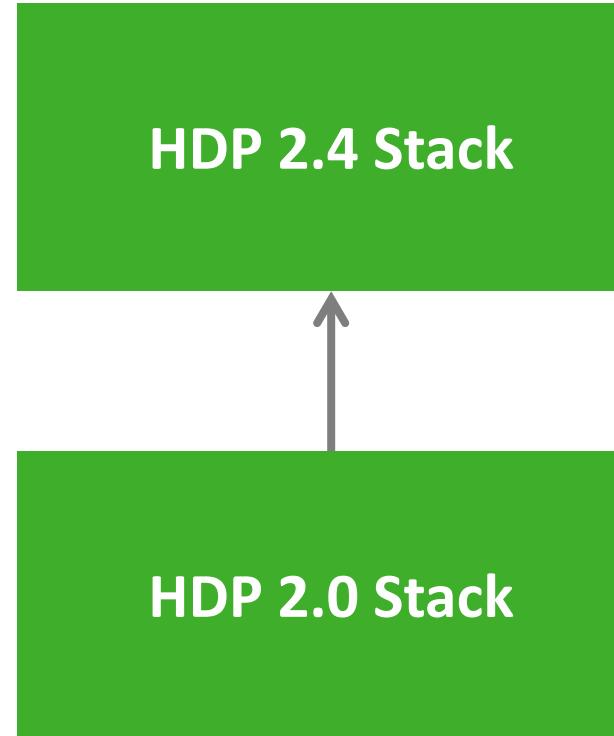
Term	Definition	Examples
STACK	Defines a set of Services, where to obtain the software packages and how to manage the lifecycle.	HDP-2.4, HDP-2.5
SERVICE	Defines the Components that make-up the Service.	HDFS, YARN, HIVE, SPARK
COMPONENT	The building-blocks of a Service, that adhere to a certain lifecycle.	NAMENODE, DATANODE, OOZIE_SERVER
CATEGORY	The category of Component.	MASTER, SLAVE, CLIENT

Stack Mechanics

- ◆ Stacks define Services + Repos
 - What is in the Stack, and where to get the bits
- ◆ Each Service has a definition
 - What Components are part of the Service
- ◆ Each Service has defined lifecycle commands
 - start, stop, status, install, configure
- ◆ Lifecycle is controlled via command scripts
- ◆ Ability to define “custom” commands



Stacks Support Inheritance



- Overrides any Service definitions, commands and configurations
 - Adds new Services specific to this Stack
-
- Defines a set of Service definitions
 - Default service configurations and command scripts

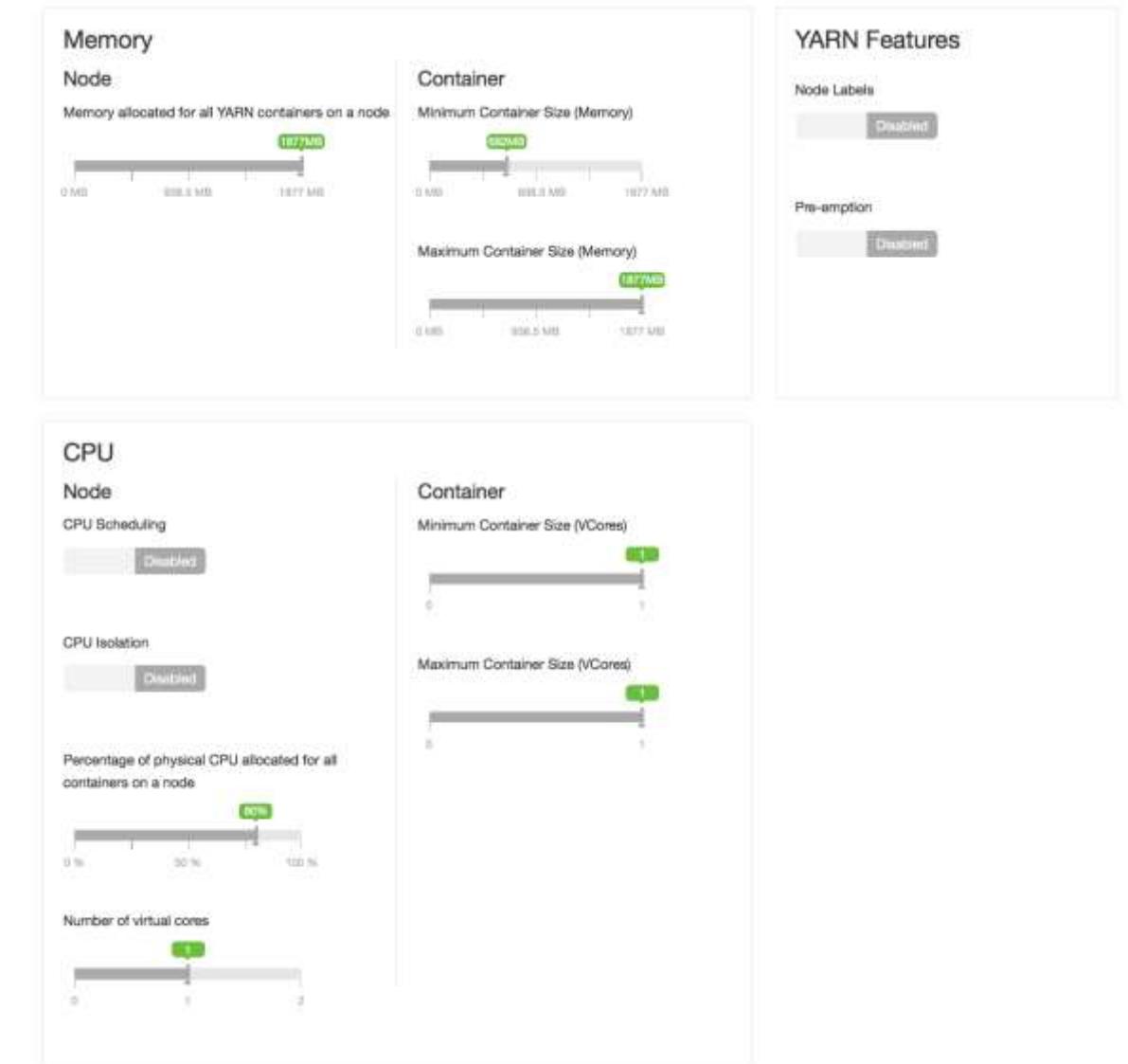
Where Can I Learn More About Stacks?

- ◆ <https://cwiki.apache.org/confluence/pages/viewpage.action?pageId=38571133>
- ◆ <https://github.com/apache/ambari/tree/trunk/ambari-server/src/main/resources/stacks/HDP/>

Guided Configs

Guided Configurations

- ◆ Take the guesswork out of cluster configuration
- ◆ Intuitive layout and grouping of configurations
- ◆ Smart UI controls to make it easier to set values
- ◆ Recommendations and cross-service dependency checks
- ◆ Take the guesswork out of cluster configuration
- ◆ Driven by Stack definition



Layout and Grouping

[Summary](#)[Heatmaps](#)[Configs](#)[Quick Links ▾](#)[Service Actions ▾](#)

Group

YARN Default (1)

Manage Config Groups

Filter...



V10 admin authored on Tue, Jul 14, 2015 19:37

[Discard](#)[Save](#)[Settings](#)[Advanced](#)

Memory

Node

Memory allocated for all YARN containers on a node



Container

Minimum Container Size (Memory)



Maximum Container Size (Memory)



YARN Features

Node Labels

[Disabled](#)

Pre-emption

[Disabled](#)

Subtabs

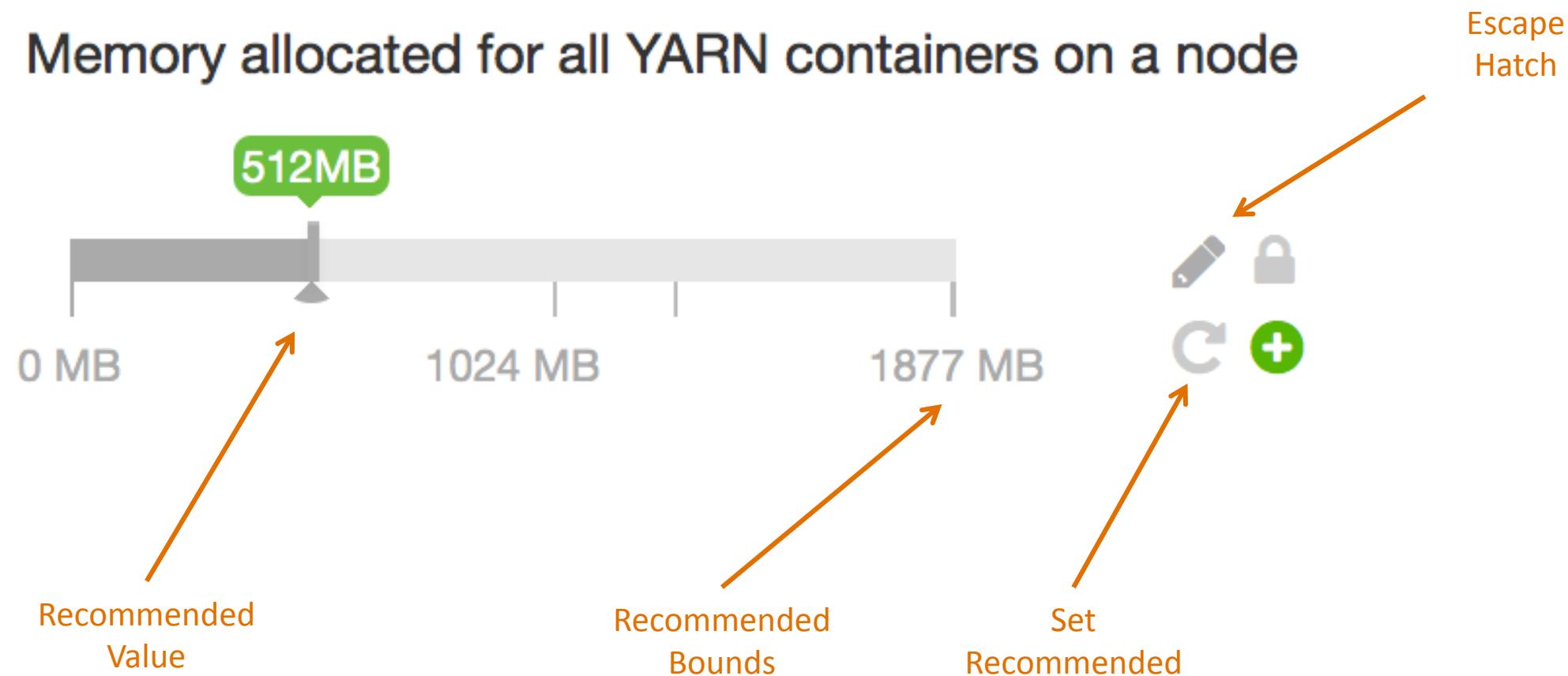
Groups

Groups

New Controls

New Controls

UI Controls



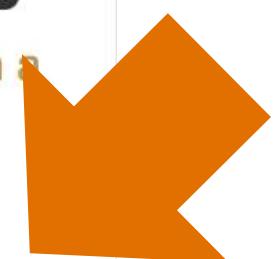
Control “Escape Hatch”

Memory allocated for all YARN containers on a node

 MB

N Values greater than 1877MB are not recommended

A Memory allocated for all YARN containers on a node

 MB

Dependency Checking

The screenshot shows the Ambari interface for managing Hadoop services. A large orange arrow points from the top left towards the configuration changes section. Another orange arrow points from the bottom right towards the dependent configurations dialog.

Configuration Changes:

- V10 (blue button)
- admin authored on Tue, Jul 14, 2015 19:37
- There are 7 configuration changes in 1 service [Show Details](#)

Advanced Settings:

- Settings
- Advanced (selected)

Memory Configuration:

- Node:** Memory allocated for all YARN containers on a node. Current value: 1024MB.
- Container:** Minimum Container Size (Memory). Current value: 768MB.
- Container:** Maximum Container Size (Memory). Current value: 1024MB.

YARN Features:

- Node Labels:** Disabled
- Pre-emption:** Disabled

Dependent Configurations:

Based on your configuration changes, Ambari is recommending the following dependent configuration changes. Ambari will update all checked configuration changes to the Recommended Value. Uncheck any configuration to retain the Current Value.

Property	Service	Config Group	File Name	Current Value	Recommended Value
mapreduce.map.memory.mb	MapReduce2	MapReduce2 Default	mapred-site.xml	170	512
mapreduce.reduce.memory.mb	MapReduce2	MapReduce2 Default	mapred-site.xml	340	512
yarn.app.mapreduce.am.command-opts	MapReduce2	MapReduce2 Default	mapred-site.xml	-Xmx136m -Dhdp.version=\$[hdp.version]	-Xmx614m -Dhdp.version=\$[hdp.version]
mapreduce.reduce.java.opts	MapReduce2	MapReduce2 Default	mapred-site.xml	-Xmx272m	-Xmx408m
yarn.app.mapreduce.am.resource.mb	MapReduce2	MapReduce2 Default	mapred-site.xml	170	768
mapreduce.map.java.opts	MapReduce2	MapReduce2 Default	mapred-site.xml	-Xmx136m	-Xmx408m
mapreduce.task.io.sort.mb	MapReduce2	MapReduce2 Default	mapred-site.xml	95	286

Driven By Stack Definition: Themes

/ resources / stacks / HDP / 2.2 / services / YARN / themes / **theme.json**

Settings Advanced

Memory

Node

Memory allocated for all YARN containers on a node

1024MB



0 MB 1024 MB 1877 MB

Container

Minimum Container Size (Memory)

170MB



0 MB 512 MB 1024 MB

Maximum Container Size (Memory)

1024MB



0 MB 512 MB 1024 MB

YARN Features

Node Labels

Disabled

Pre-emption

Disabled

Tabs

Tab1 Tab2

Settings Advanced

Memory

Node

Memory allocated for all YARN containers on a node



A horizontal slider with a green input field labeled "1024MB". The slider scale has major ticks at 0 MB, 1024 MB, and 1877 MB.

0 MB 1024 MB 1877 MB

Container

Minimum Container Size (Memory)



A horizontal slider with a green input field labeled "170MB". The slider scale has major ticks at 0 MB, 512 MB, and 1024 MB.

0 MB 512 MB 1024 MB

Maximum Container Size (Memory)



A horizontal slider with a green input field labeled "1024MB". The slider scale has major ticks at 0 MB, 512 MB, and 1024 MB.

0 MB 512 MB 1024 MB

YARN Features

Node Labels

Disabled

Pre-emption

Disabled

Sections

Settings Advanced

Memory

Node

Memory allocated for all YARN containers on a node



A horizontal slider with a green track and a blue thumb. The label '1024MB' is at the right end of the track. Below the track are tick marks for '0 MB', '1024 MB', and '1877 MB'. A small green arrow points to the left from the slider.

Container

Minimum Container Size (Memory)



A horizontal slider with a green track and a blue thumb. The label '170MB' is at the right end of the track. Below the track are tick marks for '0 MB', '512 MB', and '1024 MB'. A small green arrow points to the left from the slider.

Section1

Maximum Container Size (Memory)



A horizontal slider with a green track and a blue thumb. The label '1024MB' is at the right end of the track. Below the track are tick marks for '0 MB', '512 MB', and '1024 MB'. A small green arrow points to the left from the slider.

YARN Features

Node Labels

Disabled

Pre-emption

Disabled

Section2

SubSections

Settings Advanced

Memory

Node

Memory allocated for all YARN containers on a node



SubSection1

Container

Minimum Container Size (Memory)



SubSection2

Maximum Container Size (Memory)



YARN Features

Node Labels

Disabled

Pre-emption

SubSection1
Disabled

UI Controls and Placement

Settings Advanced

Memory

Node

Memory allocated for all YARN containers on a node

Control + Placement

0 MB 1024 MB 1877 MB

Container

Minimum Container Size (Memory)

Control + Placement

0 MB 512 MB 1024 MB

Maximum Container Size (Memory)

Control + Placement

0 MB 512 MB 1024 MB

YARN Features

Node Labels

Control + Placement

Disabled

Pre-emption

Control + Placement

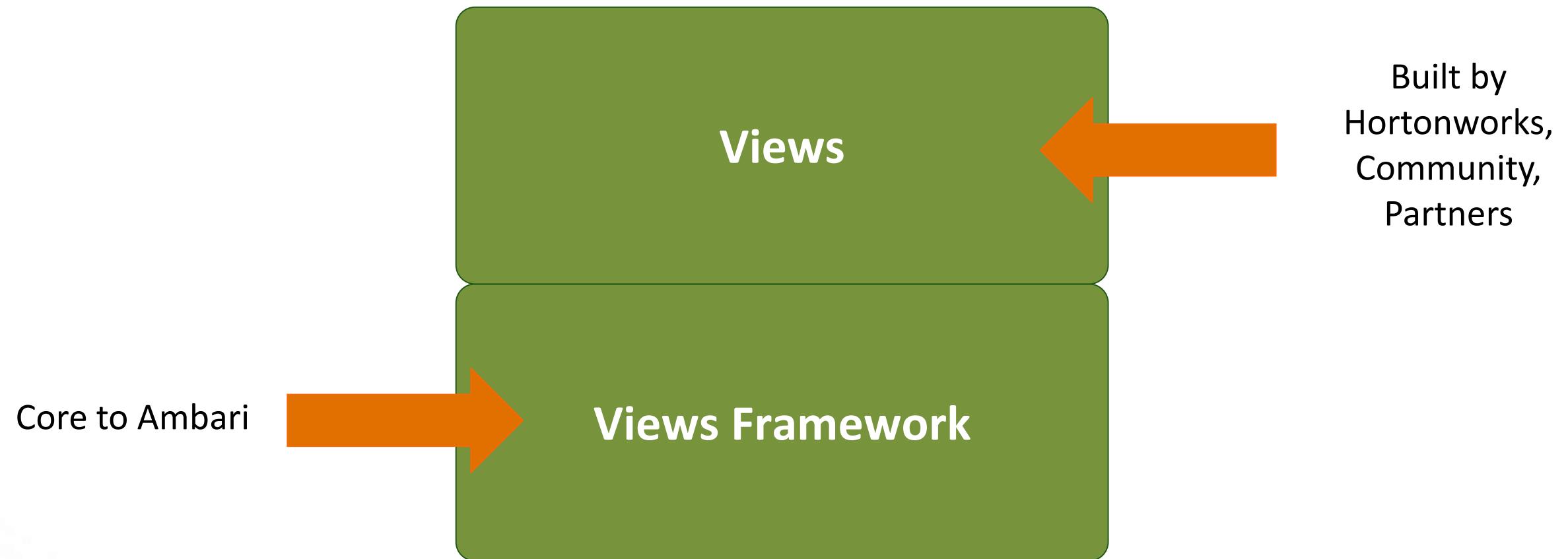
Disabled

Ambari Views

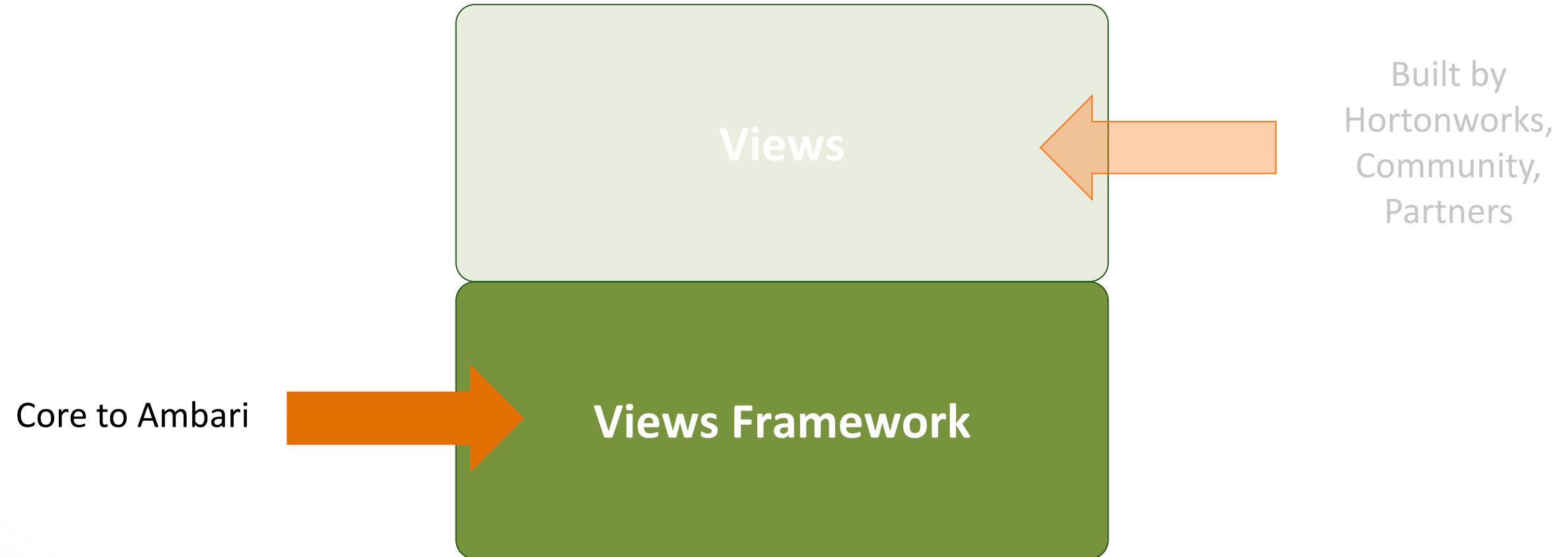
Ambari Views Framework

- ◆ Developers can extend the Ambari Web interface
 - Views expose custom UI features for Hadoop Services
- ◆ Ambari Admins can entitle Views to Ambari Web users
 - Entitlements framework for controlling access to Views

Views Framework vs. Views



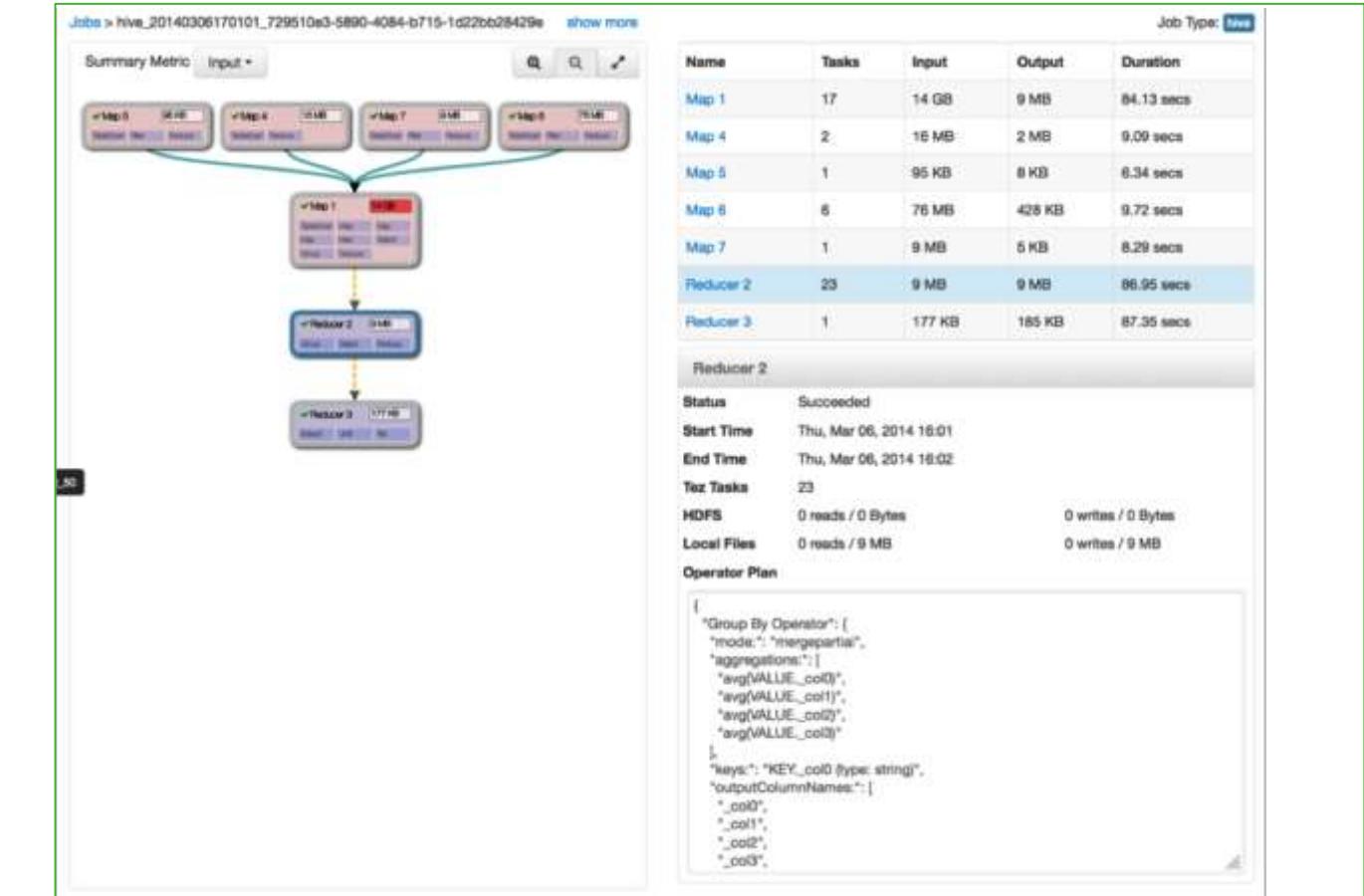
Views Framework vs. Views



Example Views

The screenshot shows the "Development" queue configuration in the YARN Queue Manager. The "Capacity" section indicates the queue is at 20% capacity. The "Access Control and Status" section shows the queue is running, with a User Limit Factor of 1, Minimum User Limit of 15%, Maximum Applications of 10000, and Maximum AM Resource of 35%. The "Resources" section displays resource usage metrics.

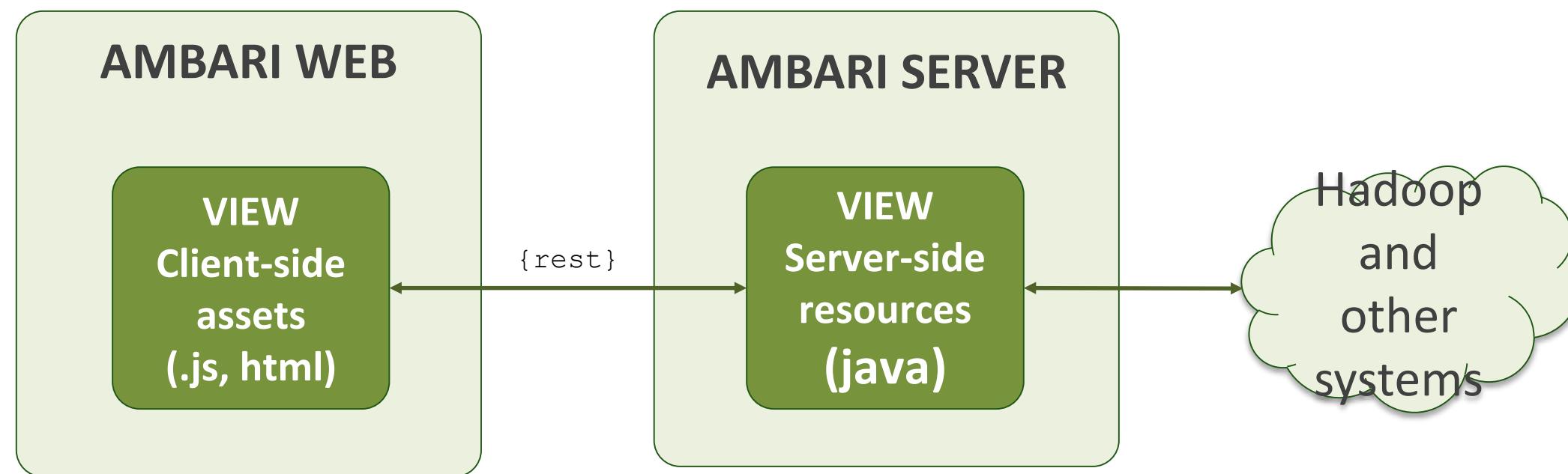
Capacity Scheduler
“Queue Manager” View



Tez
“Jobs” View

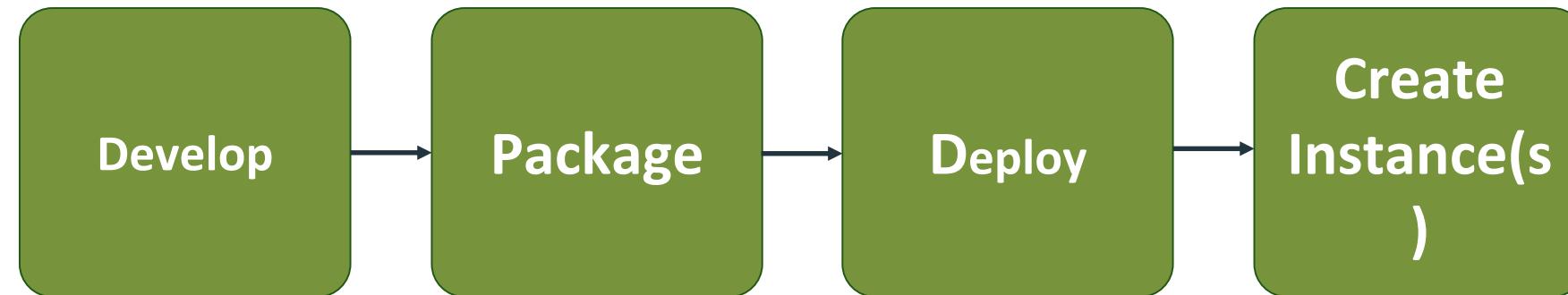
View Components

- ◆ Serve client-side assets (such as HTML + JavaScript)
- ◆ Expose server-side resources (such as REST endpoints)



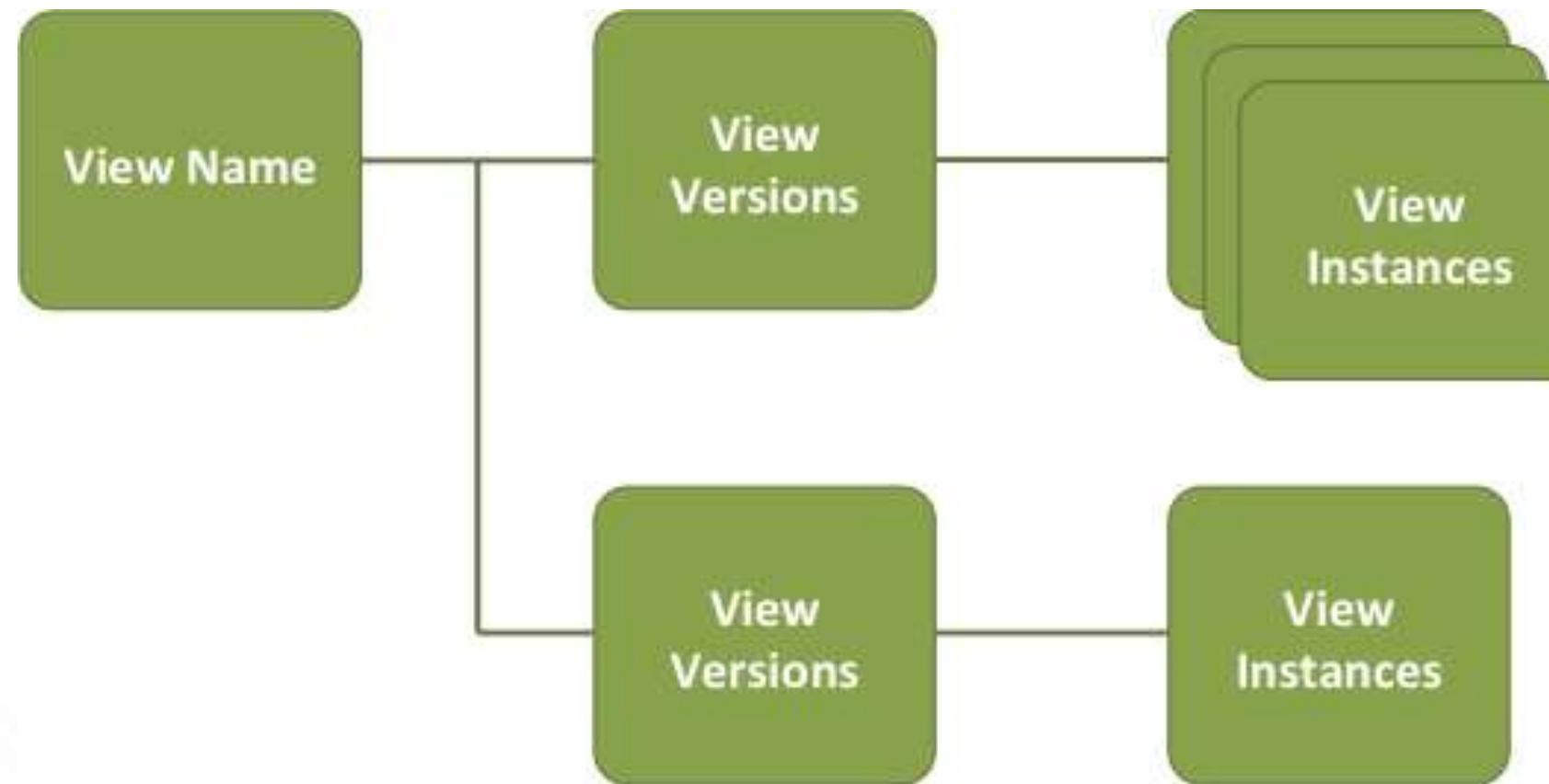
View Delivery

1. Develop the View (just like you would for a Web App)
2. Package as a View (basically a WAR)
3. Deploy the View into Ambari
4. Ambari Admins create + configuration view instance(s) and give access to users + groups



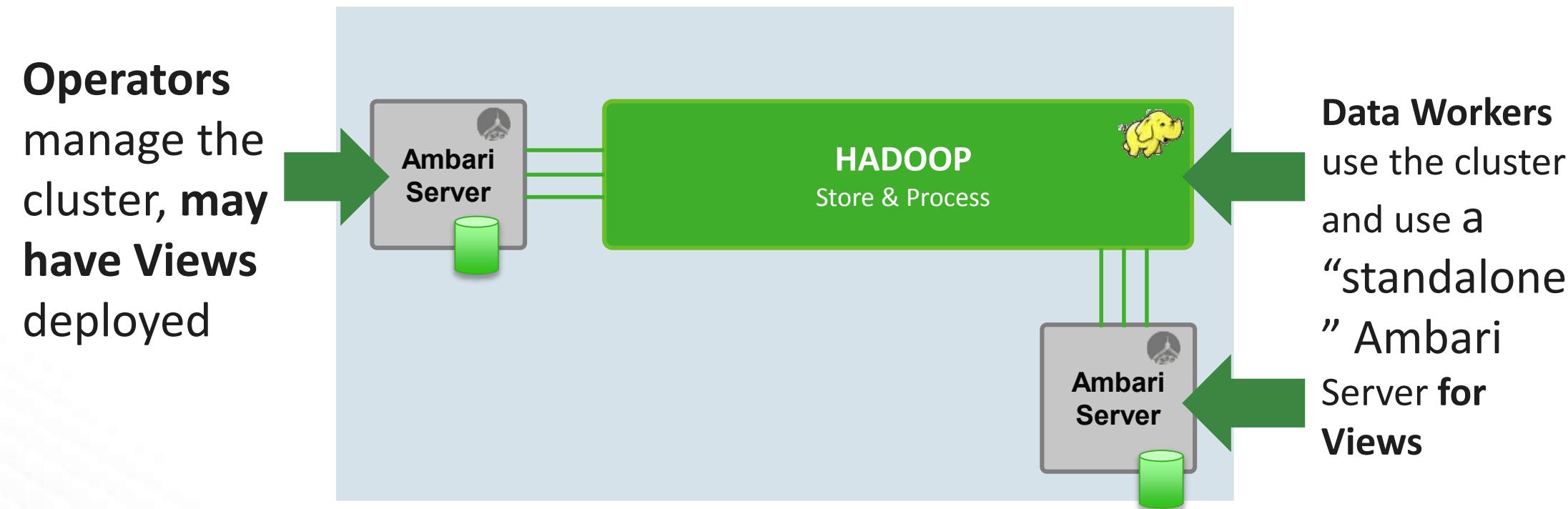
Versions and Instances

- ◆ Deploy multiple versions and create multiple instances of a view
- ◆ Manage accessibility and usage



Choice of Deployment Model

- ◆ For Hadoop Operators:
 - Deploy Views in an Ambari Server that is managing a Hadoop cluster
- ◆ For Data Workers:
 - Run Views in a “standalone” Ambari Server



Where Can I Learn More about Views?

- ◆ <https://cwiki.apache.org/confluence/display/AMBARI/Views>
- ◆ <https://github.com/apache/ambari/blob/trunk/ambari-views/docs/index.md>
- ◆ <https://github.com/apache/ambari/tree/trunk/ambari-views/examples>
- ◆ <https://github.com/apache/ambari/tree/trunk/contrib/views>

Learn More About Apache Ambari

Resource	Location
Apache Ambari Project Page	http://ambari.apache.org
Ambari Project Wiki	https://cwiki.apache.org/confluence/display/AMBARI
Ambari Project JIRA	https://issues.apache.org/jira/browse/AMBARI
Stacks	https://cwiki.apache.org/confluence/pages/viewpage.action?pageId=38571133
Blueprints	https://cwiki.apache.org/confluence/display/AMBARI/Blueprints
Views	https://cwiki.apache.org/confluence/display/AMBARI/Views