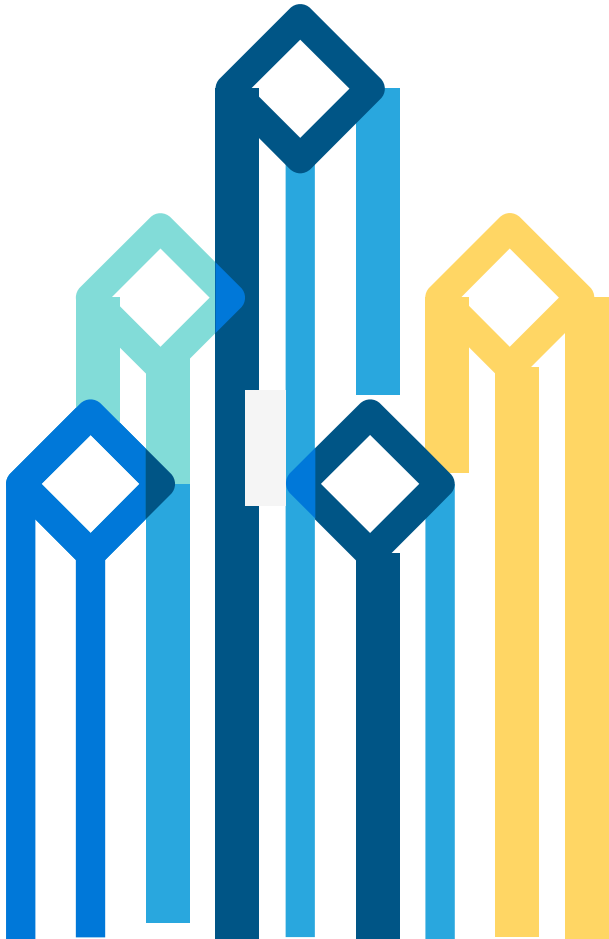
cloudera



Apache Hadoop Operations for Production Systems

Philip Zeyliger, Philip Langdale, Kathleen Ting, Miklos Christine

Strata Hadoop San Jose, 18 February 2015

Your Hosts

Philip Langdale

Kathleen Ting

Philip Zeyliger

Miklos Christine

cloudera

$ whoami @cloudera.com

Philip Langdale

- Architect of Cloudera Manager
- philipl@

Kathleen Ting

- Technical account manager for large Hadoop clusters
- kate@

Philip Zeyliger

- Architect of Cloudera Manager
- philip@

Miklos Christine

- Systems engineer for large Hadoop clusters
- mwc@

cloudera

Overall Agenda

- Intro
- Installation
- Configuration

(stretch) (official break 3-3:30)

- Troubleshooting
- Enterprise Considerations
- Q & A

Q&A at end of every section

cloudera

Hands On

We've set up a handful of clusters in various configurations.

URLs and passwords:

http://tiny.cloudera.com/strata1 (user1/strata2015) (Kerberos)

admin@54.153.123.57 pass: admin321

http://tiny.cloudera.com/strata2 (user2/strata2015)

admin@54.67.90.80 pass: admin321

http://tiny.cloudera.com/strata3 (user3/strata2015)

admin@54.153.83.11 pass: admin321

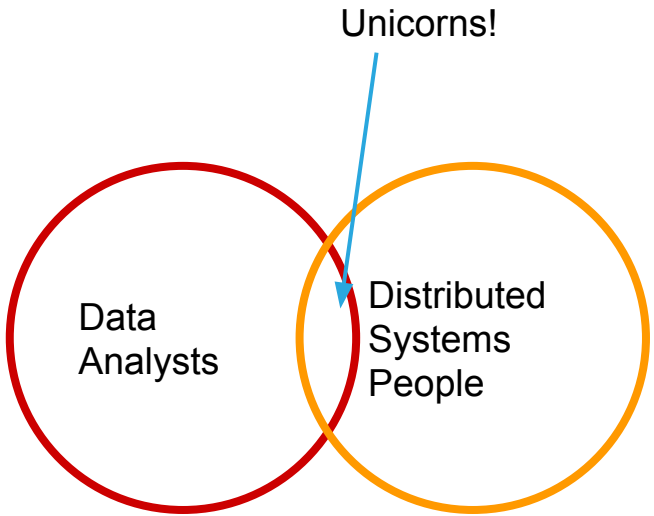
cloudera

Why Apache Hadoop?

Solves problems that don't fit on a single computer.

Doesn't require you to be a distributed systems person.

Handles failures for you.



cloudera

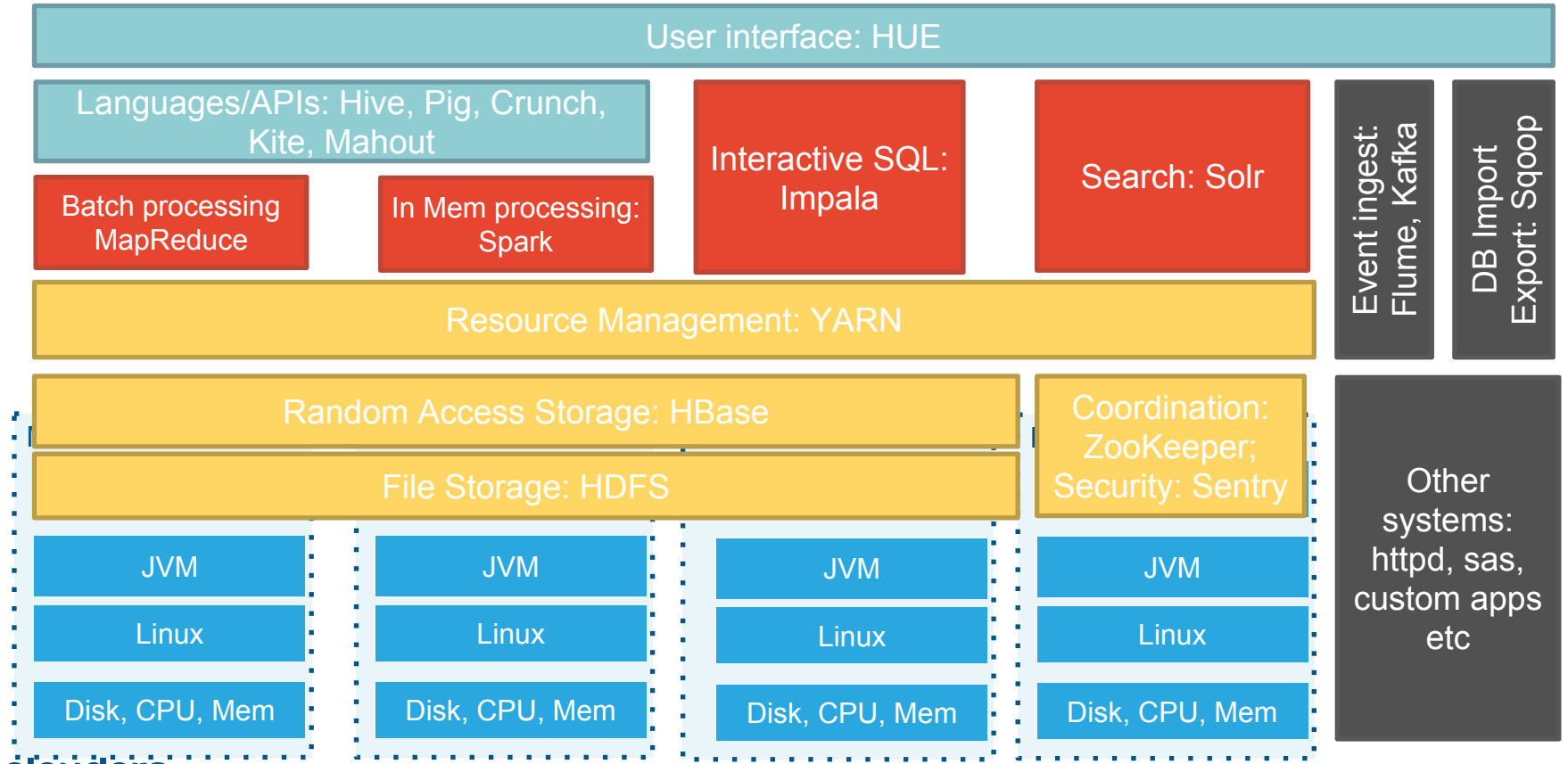
A Distributed System

Many processes, which, taken together, are trying to act like a single system.

Ideally, users deal with the system as a whole.

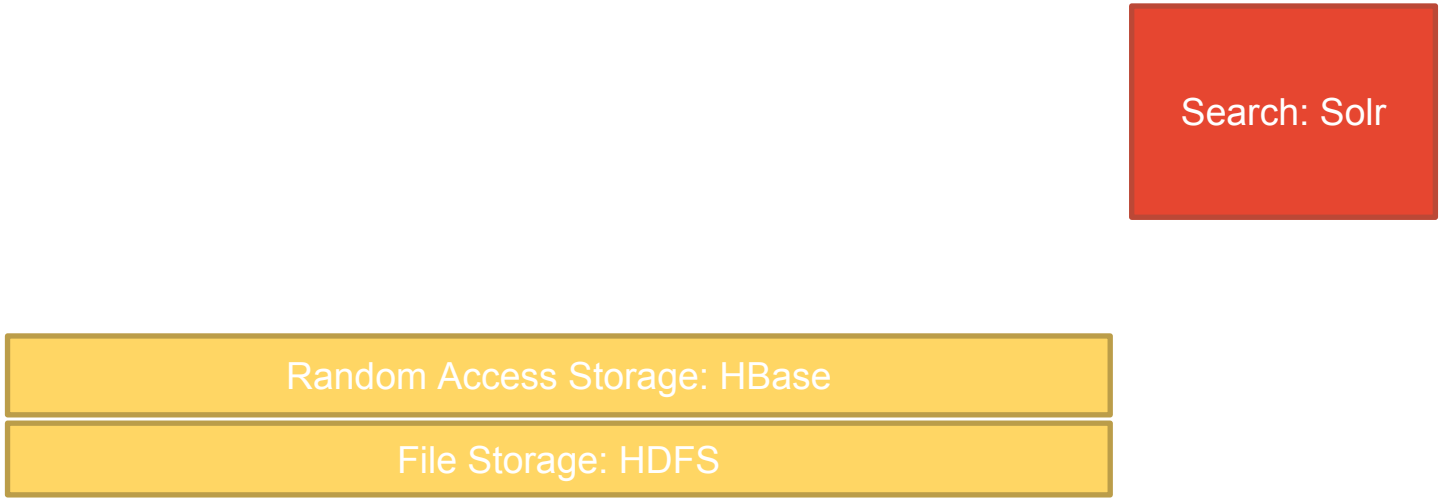
For operations, you need to understand the system by parts too.

The Hadoop Stack



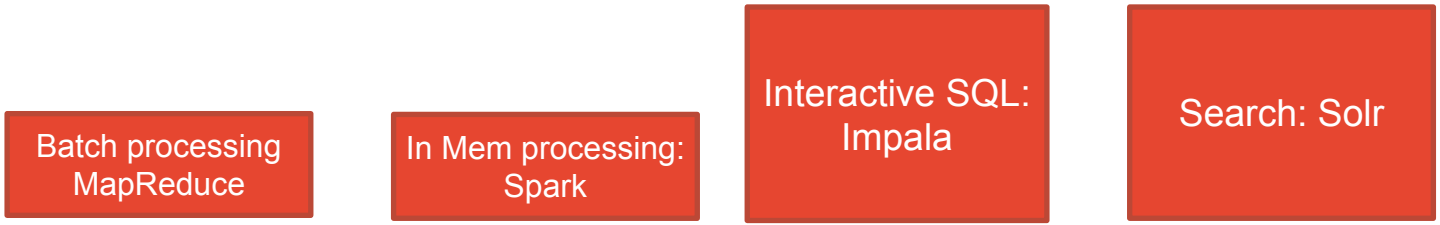
cloudera

Storage



cloudera

Execution



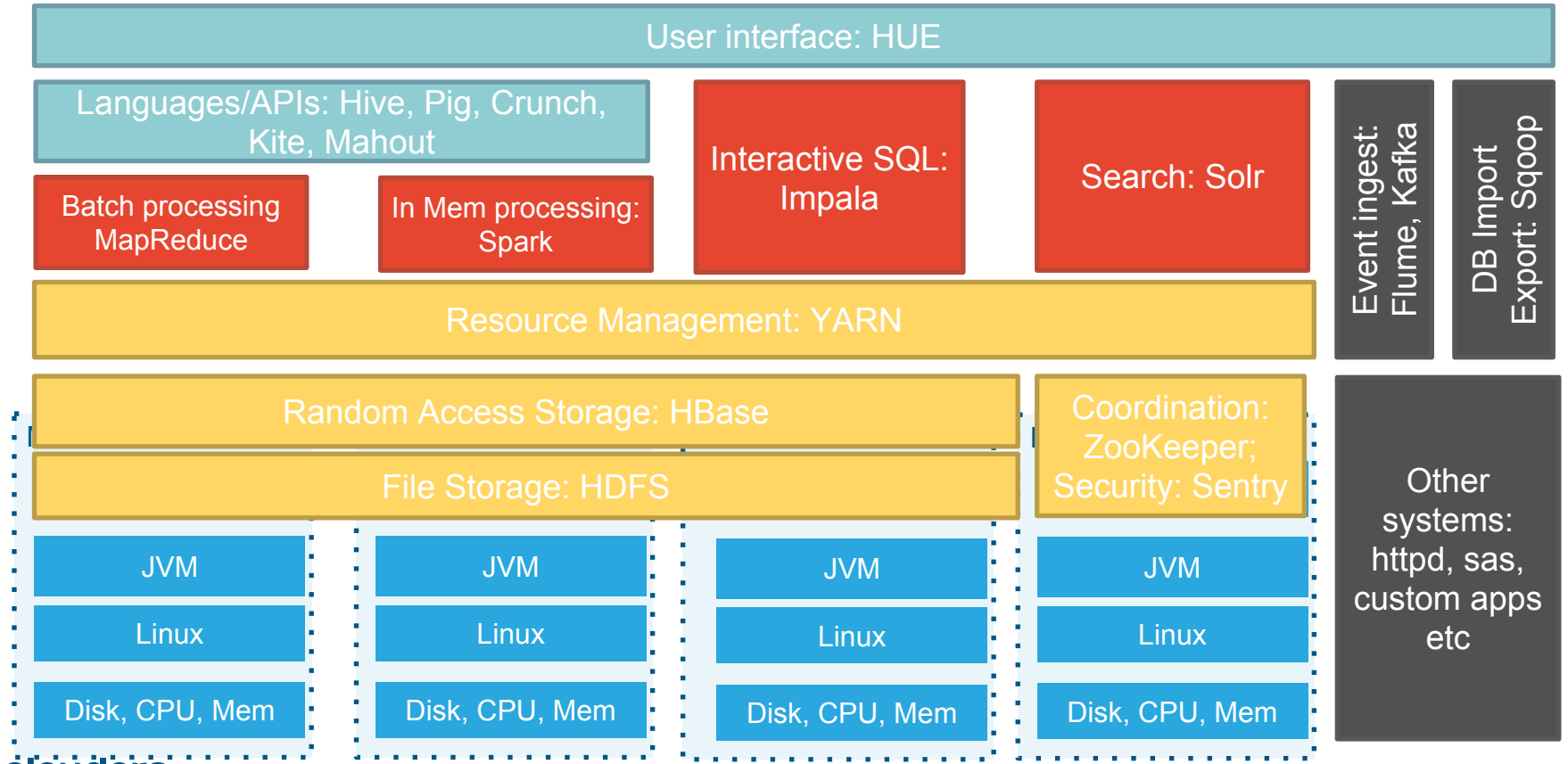
Compilers

Languages/APIs: Hive, Pig, Crunch, Kite, Mahout

Hive—SQL to MR/Spark compiler, metadata mapping between files and tables

Pig—PigLatin to MR compiler

Taken all together…



cloudera

How to learn these things...

These systems are largely defined by the state they store on disk and the defined protocols (RPCs) they use to interact amongst themselves.

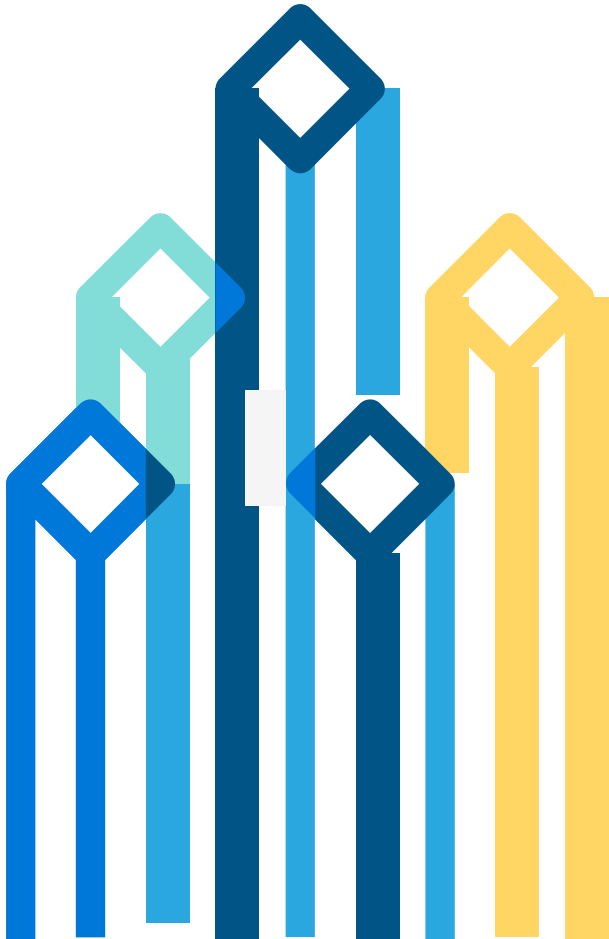
cloudera

Questions?

cloudera

Apache Hadoop Operations for Production Systems: Installation

Philip Langdale



Agenda

- Hardware Considerations
  - Node types and recommended role allocations
  - Host configuration
  - Rack configuration
- Software Installation
  - OS Prerequisites
  - Installing Hadoop and other Ecosystem components
- Launch
  - Initial Configuration
  - Sanity testing
  - Security considerations

Hardware Considerations

- As a distributed system, Hadoop is going to be deployed onto multiple interconnected hosts
- How large will the cluster be?
- What services will be deployed on the cluster?
  - Can all services effectively run together on the same hosts or is some form of physical partitioning required?
- What role will each host play in the cluster?
  - This impacts the hardware profile (CPU, Memory, Storage, etc)
- How should the hosts be networked together?

Host Roles within a Cluster

For larger clusters, roles will be spread across multiple nodes of a given type (except workers)

Master Node
HDFS NameNode
YARN ResourceManager
HBase Master
Impala StateStore
ZooKeeper

Worker Node
HDFS DataNode
YARN NodeManager
HBase RegionServer
Impalad

Utility Node
Relational Database
Management (eg: CM)
Hive Metastore
Oozie
Impala Catalog Server

Edge Node
Gateway Configuration
Client Tools
Hue
HiveServer2
Ingest (eg: Flume)

cloudera

Roles vs Cluster Size

	Master	Worker	Utility	Edge
Very Small (≤10)	1	≤10	1 shared Host	
Small (≤20)	2	≤20	1 shared Host	
Medium (≤200)	3	≤200	2	1+
Large (≤500)	5	≤500	2	1+

cloudera

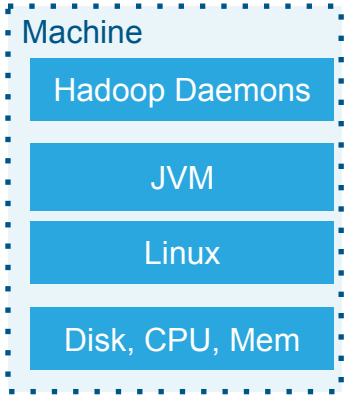
Host Hardware Configuration

- CPU
  - There's no such thing as too much CPU
  - Jobs typically do not saturate their cores, so raw clock speed is not at a premium
  - Cost and Budget are the major factors here
- Memory
  - You really don't want to overcommit and swap
  - Java heaps should fit into physical RAM with additional space for OS and non-Hadoop processes

Host Configuration (cont.)

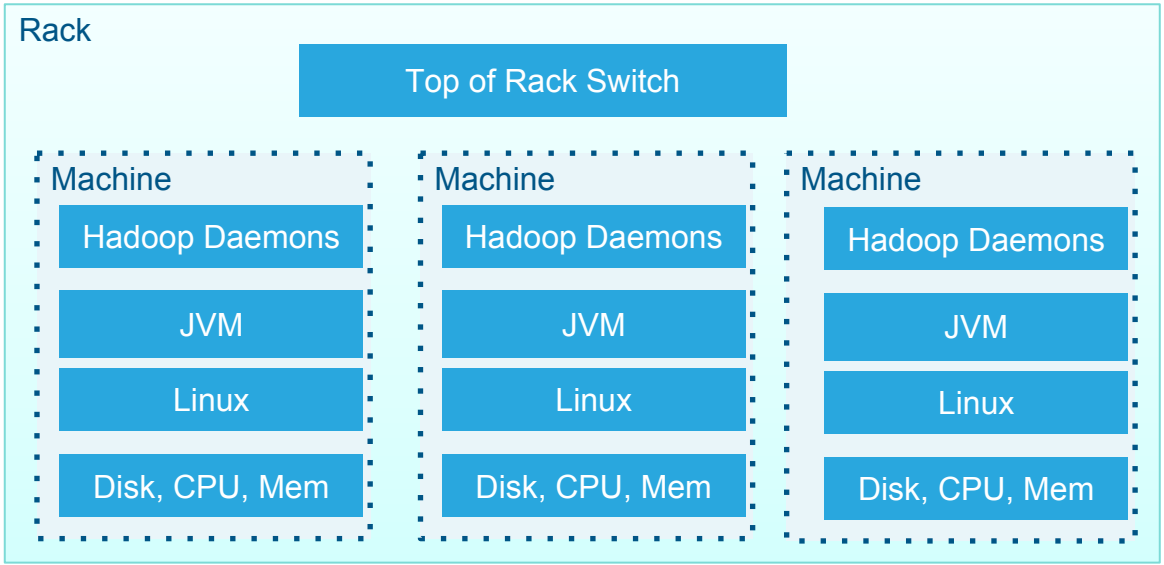
- Disk
  - More spindles == More I/O capacity
  - Larger drives == lower cost per TB
  - More hosts with less capacity increases parallelism and decreases re-replication costs when replacing a host
  - Fewer hosts with more capacity generally means lower unit cost
  - Rule of thumb: One disk per two configured YARN vcores
  - Lower latency disks are generally not a good investment, except for specific use-cases where random I/O is important

A Hadoop Machine



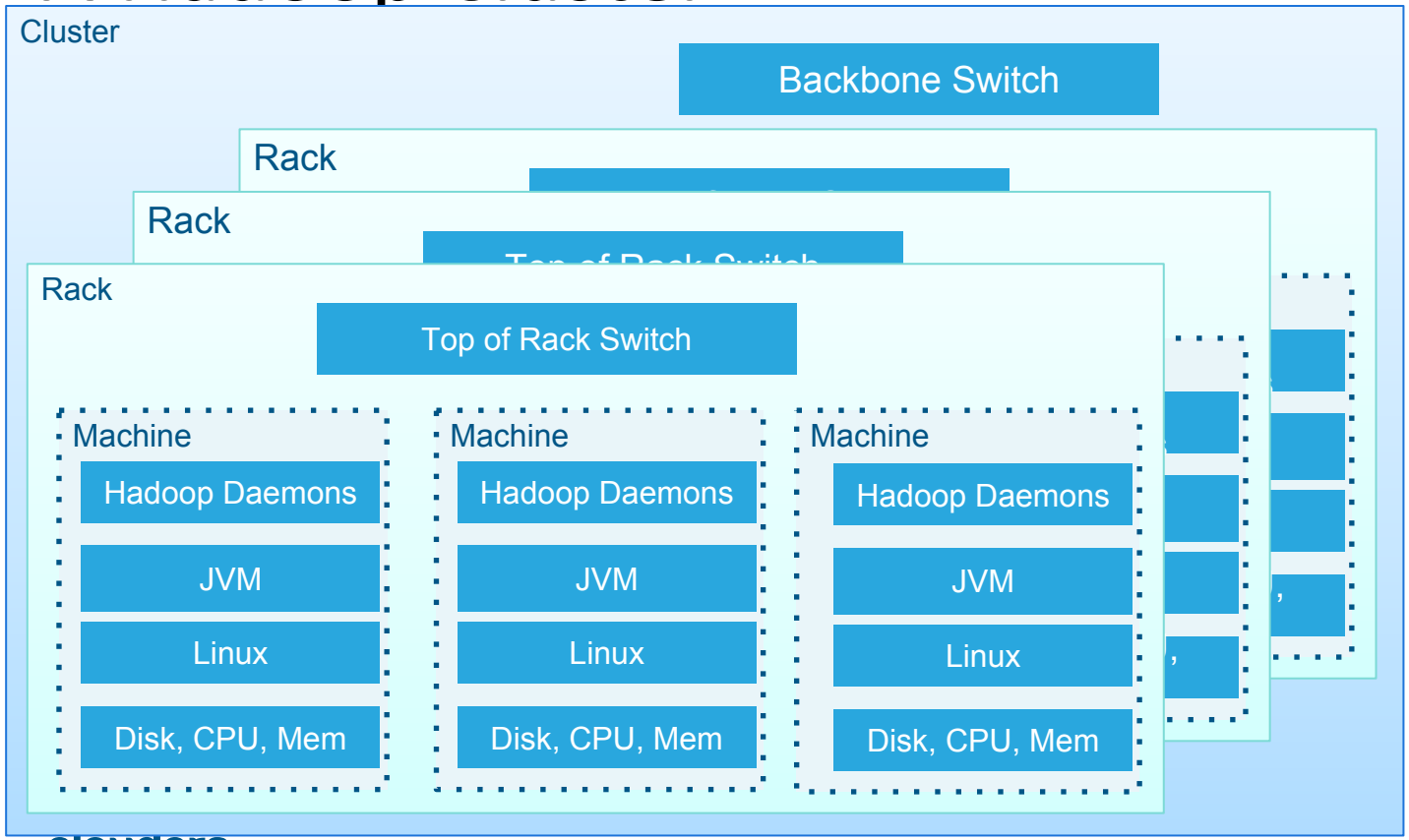
cloudera

A Hadoop Rack

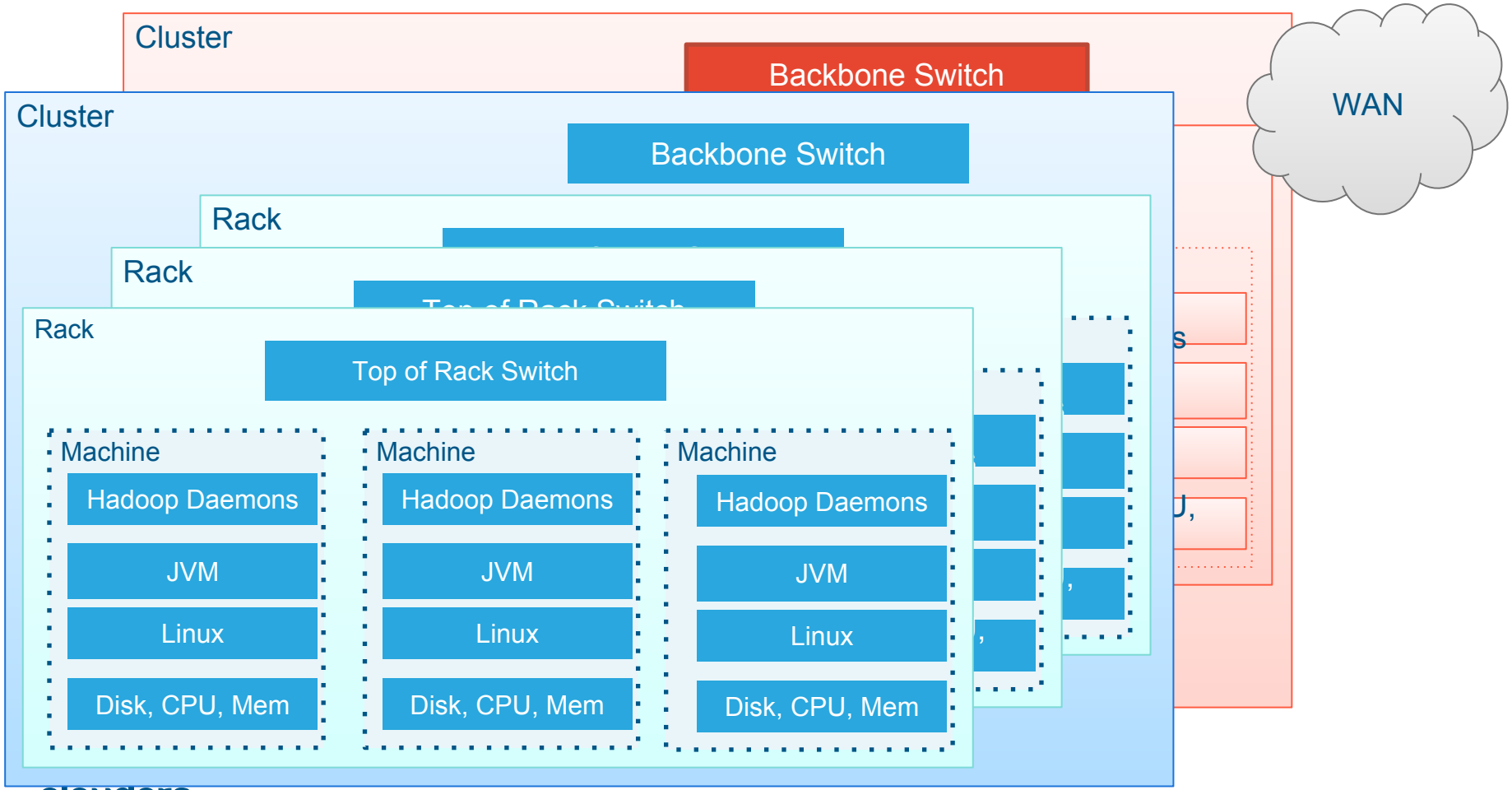


cloudera

A Hadoop Cluster



cloudera



Rack Configuration

- Common: 10Gb or (20Gb bonded) to the server, 40Gb to the spine
- Cost sensitive: 2Gb bonded to the server, 10Gb to the spine
  - This is likely to be a false economy, with the real potential of being network bottlenecked with disks idle
- Look for 25/100 in the next couple of years
- Network I/O is generally consumed by reading/writing data from disks on other nodes
  - Teragen is a useful benchmark for network capacity
  - Typically 3-9x more intensive than normal workloads

Software Installation

- Linux Distribution
- Operating System Configuration
- Hadoop Distribution
- Distribution Lifecycle Management

Linux Distributions

- All enterprise distributions are credible choices
- Do you already buy support from a vendor?
- Which distributions does your Hadoop distro support?
- What are you already familiar with
- Cloudera supports
  - RHEL 5.x, 6.x
  - Ubuntu 12.04, 14.04
  - Debian 6.x, 7.x
  - SLES 11
- RHEL 6.x is the most common

Operating System Configuration

- Turn off IPTables (or any other firewall tool)
- Turn off SELinux
- Turn down swappiness to 0/1 (depending on what kernel you have)
- Turn off Transparent Huge Page Compaction
- Use a network time source to keep all hosts in sync (also timezones!)
- Make sure forward and reverse DNS work on each host to resolve all other hosts consistently
- Use the Oracle JDK - OpenJDK is subtly different and may lead you to grief

Cloudera Manager provides a Host Inspector to check for these situations

cloudera

Inspector Results

Validations

- ✓ Inspector ran on all 8 hosts.
- ✓ The following failures were observed in checking hostnames...
- ⚠ The following errors were found while looking for conflicting init scripts. Use 'chkconfig' to disable init scripts to avoid conflicts with daemons managed by Cloudera Manager. Typically, you may be able to continue with installation, but, after a reboot, processes may fail to start because of a conflict. ›
- ✓ No errors were found while checking /etc/hosts.
- ✓ All hosts resolved localhost to 127.0.0.1.
- ✓ All hosts checked resolved each other's hostnames correctly and in a timely manner.
- ✓ Host clocks are approximately in sync (within ten minutes).
- ✓ Host time zones are consistent across the cluster.
- ✓ No users or groups are missing.
- ✓ No conflicts detected between packages and parcels.
- ✓ No kernel versions that are known to be bad are running.
- ⚠ Cloudera recommends setting /proc/sys/vm/swappiness to 0. Current setting is 10. Use the `sysctl` command to change this setting at runtime and edit `/etc/sysctl.conf` for this setting to be saved after a reboot. You may continue with installation, but you may run into issues with Cloudera Manager reporting that your hosts are unhealthy because they are swapping. The following hosts are affected: ›
- ⚠ Transparent Huge Pages is enabled and can cause significant performance problems. Kernel with release 'CentOS release 6.2 (Final)' and version '2.6.32-220.7.1.el6.x86_64' has enabled set to '[always] never' and defrag set to '[always] never'. Run "echo never > /sys/kernel/mm/redhat_transparent_hugepage/defrag" to disable this, then add the same command to an init script such as /etc/rc.local so it will be set upon system reboot. Alternatively, upgrade to RHEL 6.5 or later, which does not have this bug. The following hosts are affected: ›

Hadoop Distributions

- Well, we're obviously biased here
- CDH is a jolly good Hadoop Distro. We recommend it
- You're free to try others
- While it's technically possible to have a go at running a cluster without using any management tools, it's not going to be fun and we're not going to talk about that much

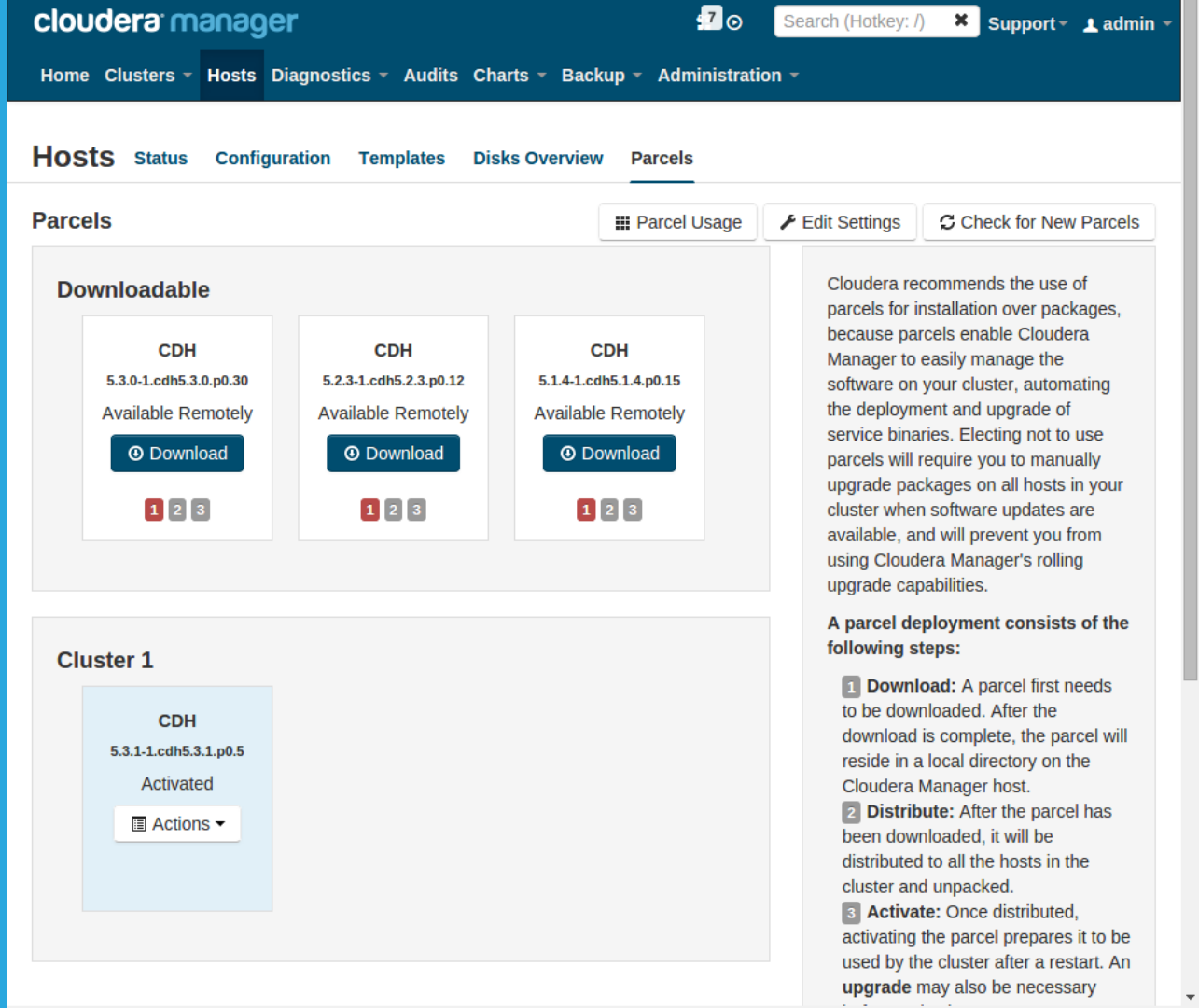
Distribution Lifecycle Management

- Theoretically, you might want to just install the binaries for services and programs you are running on a specific node
- But honestly, space is not at that much of a premium
  - Install everything everywhere and don't worry about it again
  - If you decide to alter the footprint of a service, you don't need to worry about binaries
  - You don't need different profiles for different hosts in the cluster
  - Cloudera Manager works this way

Distribution Lifecycle Management (cont.)

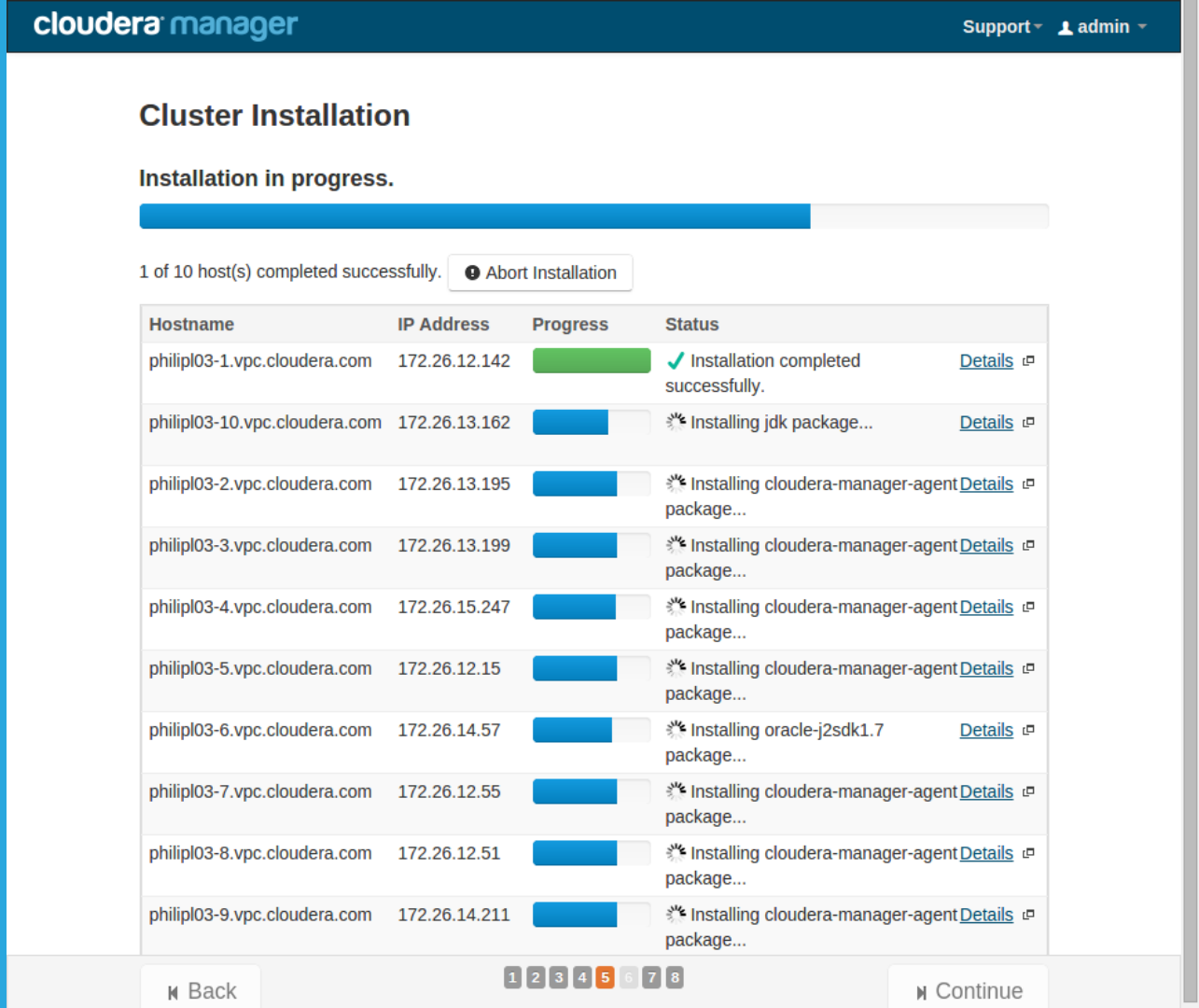
- For CDH, Cloudera Manager can handle lifecycle management through parcels
- For package based installations, there are a variety of recognised options
  - Puppet, Chef, Ansible, etc
- But please use something
  - Long term package management by hand is a recipe for disaster
  - Exposes you to having inconsistencies between hosts
    - Missing Packages
    - Un-upgraded Packages

Lifecycle Management with Parcels in Cloudera Manager



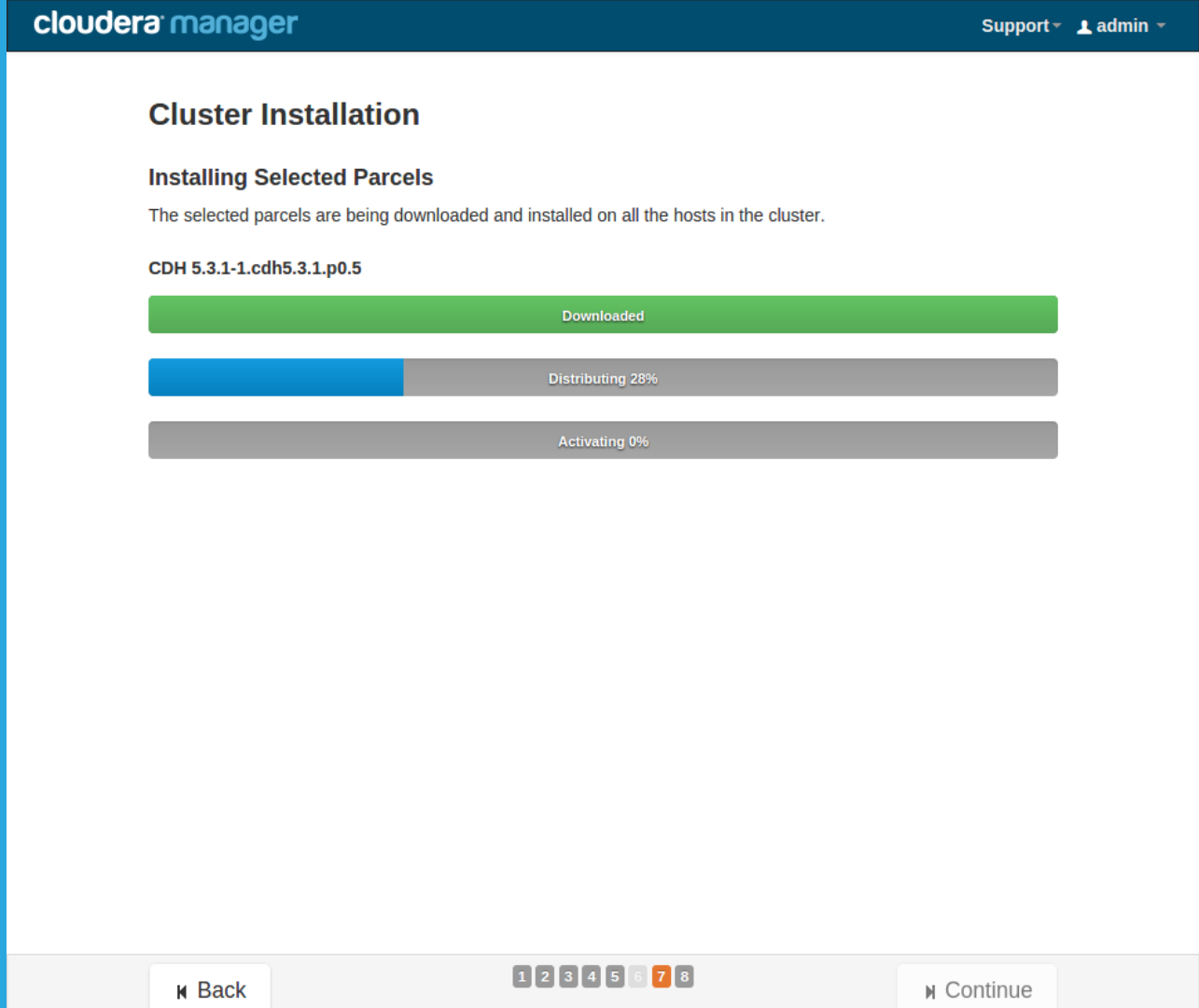
cloudera

Installation with Cloudera Manager



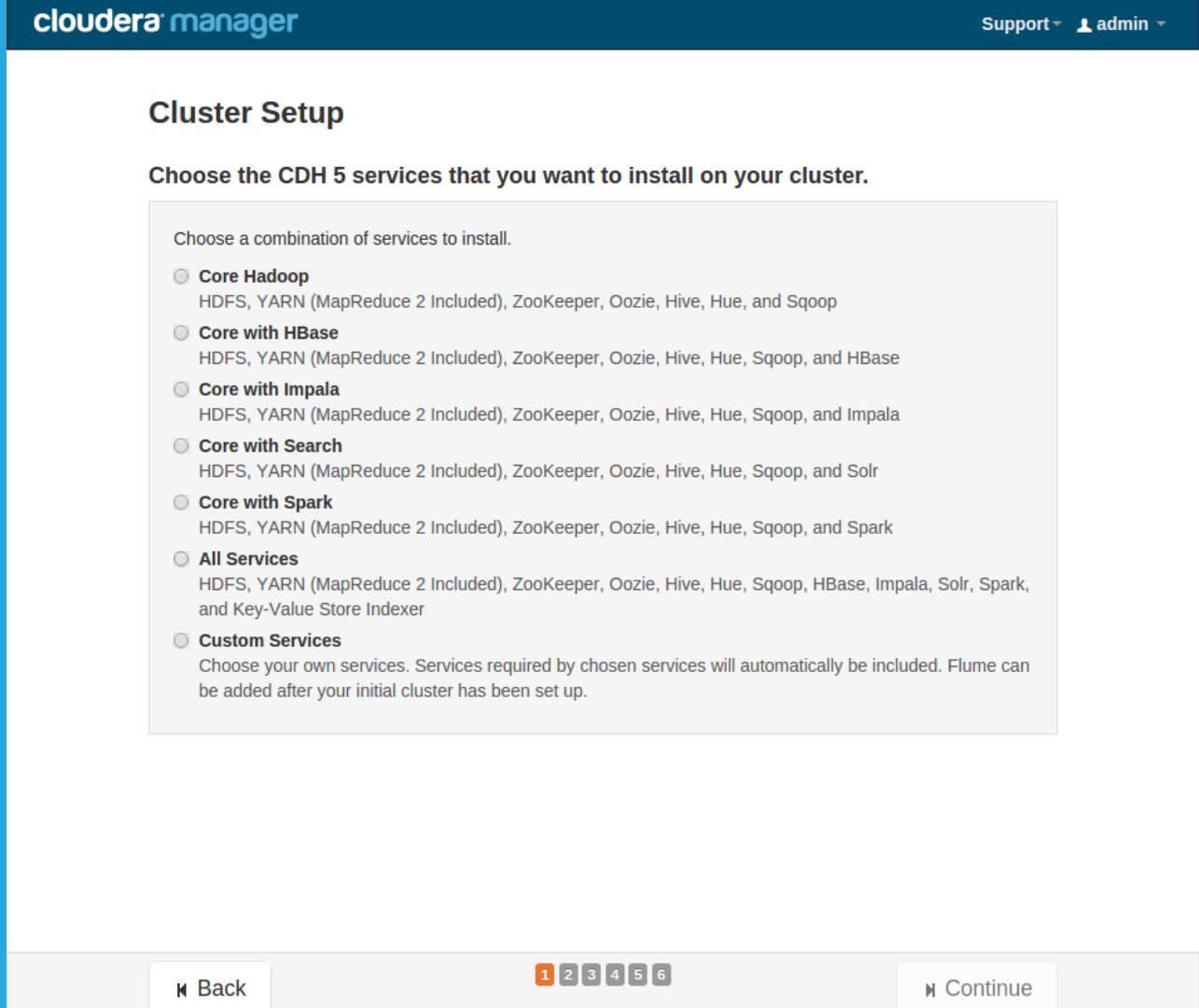
cloudera

Installation with Cloudera Manager



cloudera

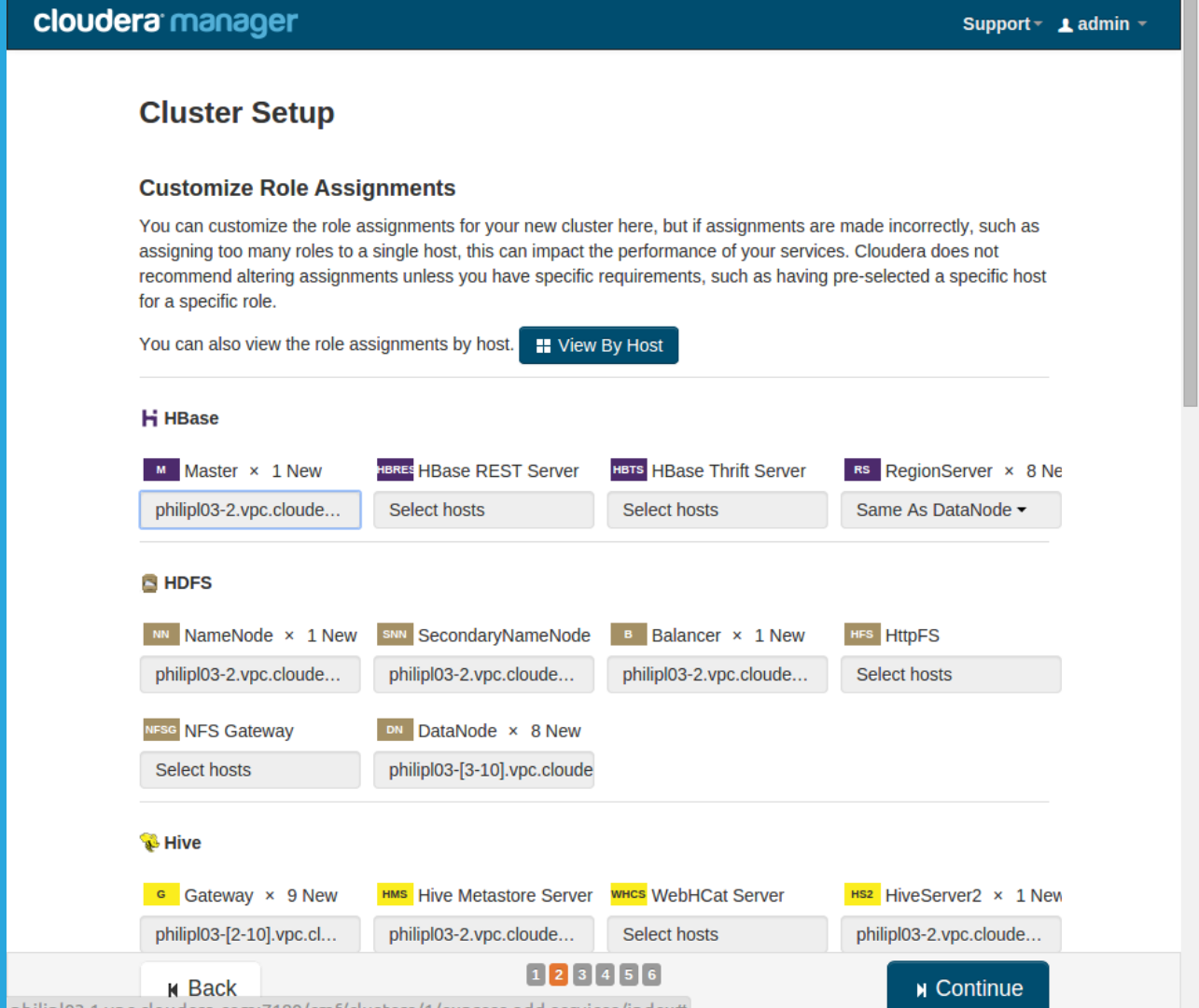
Installation with Cloudera Manager



cloudera

Installation with Cloudera Manager

cloudera



Launch

- Initial Configuration
- Sanity testing
- Security Considerations

Initial Configuration

- Recall our earlier discussion of hardware
  - YARN vcores (or MapReduceV1 slots) proportional to physical cores
    - Typically 1/1.5:1 (1.5 with hyperthreading)
- Heap sizes
  - Don’t overcommit on memory, but don’t make Java heaps too large (GC)
  - See Memory table in Appendix
- Mounting data disks
  - One partition per disk
  - No RAID
  - Use a well established filesystem (ext4)
  - Use a uniform naming scheme

Initial Configuration (cont.)

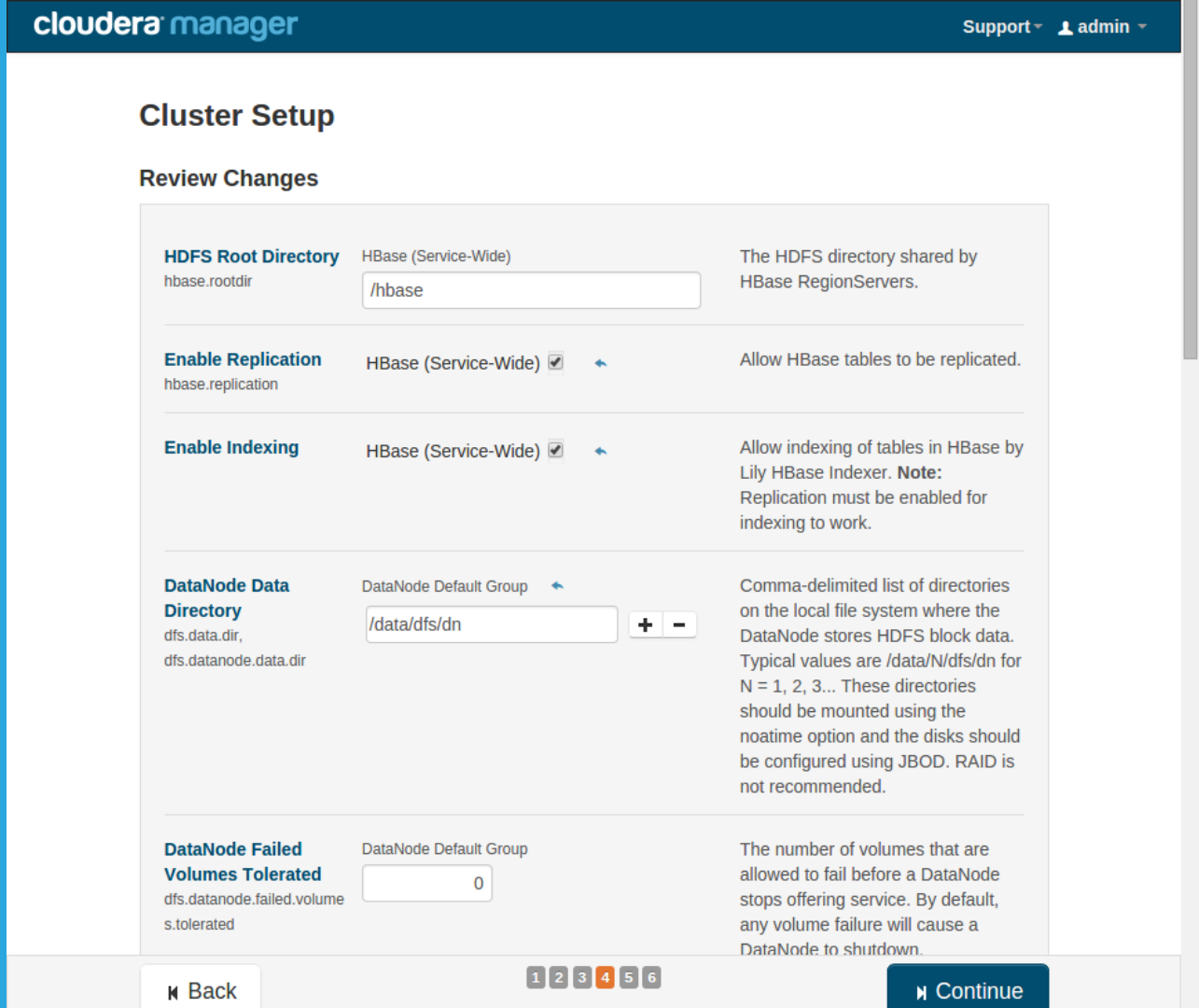
- Think about space on the OS partition(s)
  - /var is where your logs go by default
  - /opt is where Cloudera Manager stores parcels
  - By default, these are part of / on modern distros, which works out fine
  - Your IT policies may require separate partitions
    - If these areas are too small, then you'll need to change these configurations

Cloudera Manager can help

- Assigns roles based on cluster size
- Tries to detect masters based on physical capabilities
- Sets vcore count based on detected host CPUs
- Sets heap sizes to avoid overcommitting RAM
- Autodetects mounted drives and assigns them for DataNodes

Initial Service Configuration in Cloudera Manager

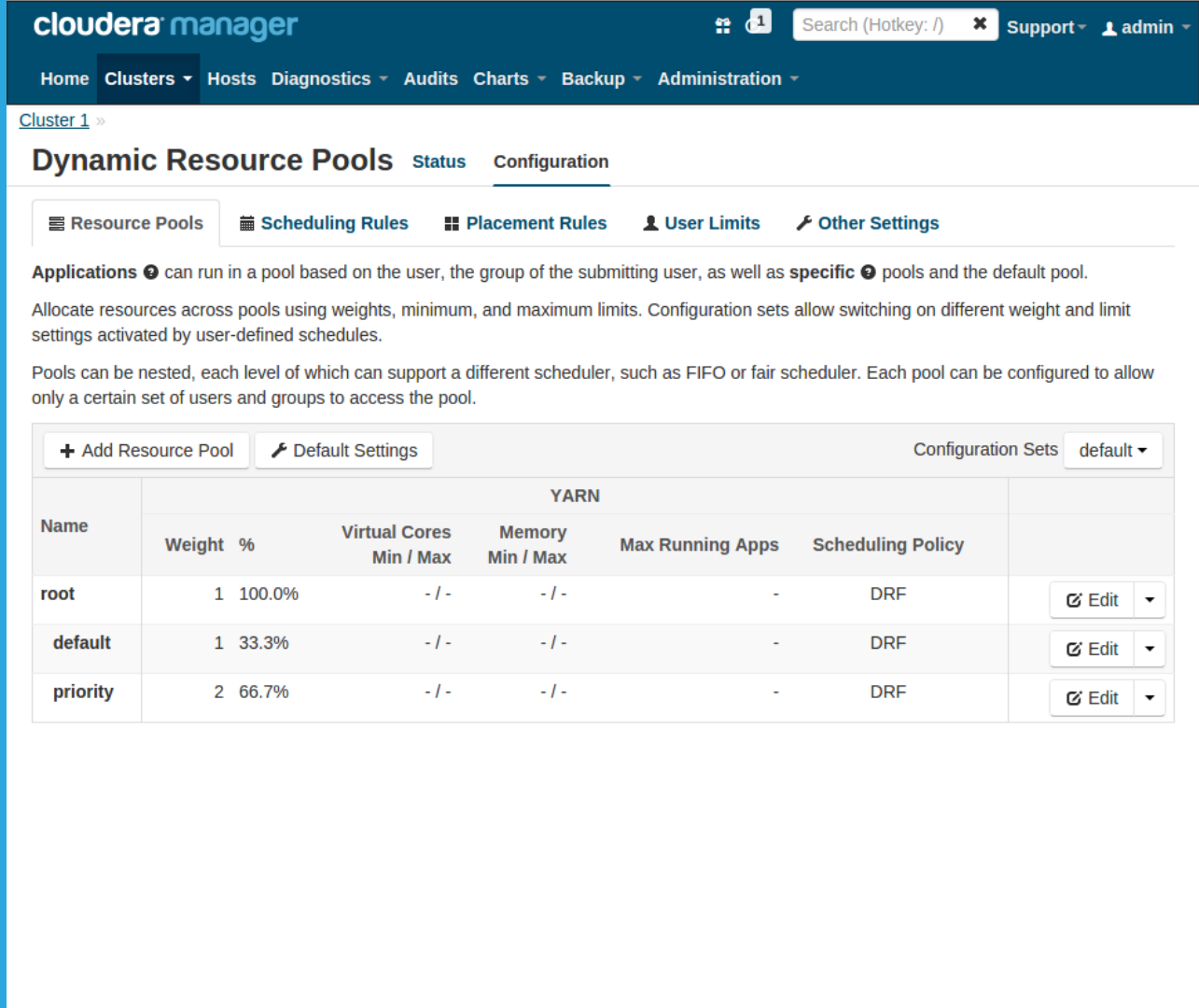
cloudera



Implications of Services in Use

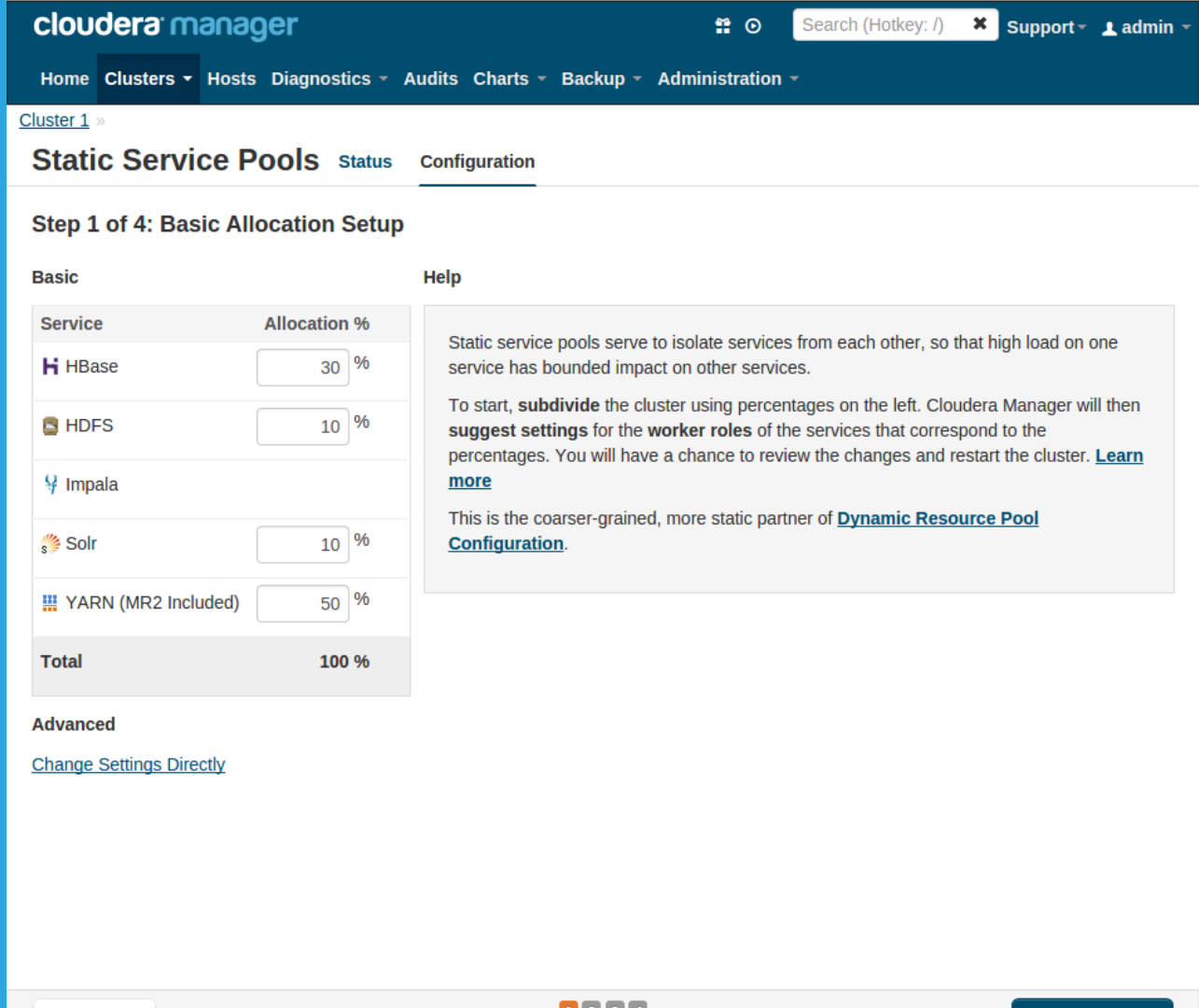
- Different services running concurrently means we need to consider how resources are shared between them
- Services that use YARN can be managed through YARN's scheduler
- But certain services do not - most visibly HBase, but also services like Accumulo or Flume
  - These services can run on a shared cluster through the use of static resource partitioning with cgroups
- Cloudera Manager can configure these

Dynamic Resource Management



cloudera

Static Resource Management



cloudera

Sanity Testing

- Use the basic sanity tests provided by each service
  - Submit example jobs: pi, sleep, teragen/terasort
  - Work with sample tables in Hive/Impala
  - Use Hue to do these things if your users will
- Repeat these tests when turning on security/authentication/authorization mechanisms
  - Make sure they succeed for the expected users and fail for others
- Cloudera Manager provides some ‘canary’ tests for certain services
  - HDFS create/read/write/delete
  - Hbase, Zookeeper, etc

HDFS Health tests

cloudera

Health Tests

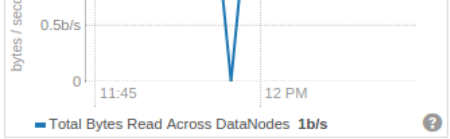
Collapse All

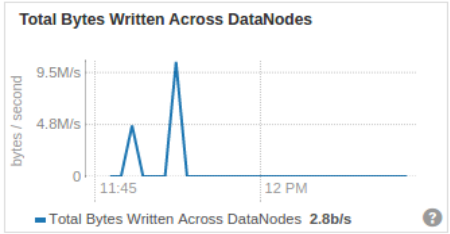
7 good.

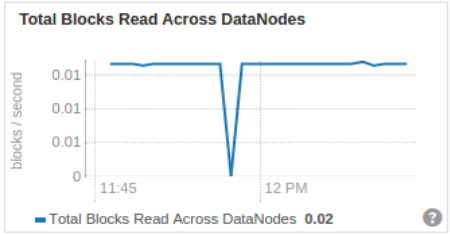
- NameNode summary: philipl03-2.vpc.cloudera.com (Availability: Active, Health: Good) — Details
- Space free in the cluster: 188.0 GiB. Capacity of the cluster: 188.9 GiB. Percentage of capacity free: 99.52%. — Details
- 0 blocks with corrupt replicas in the cluster. 299 total blocks in the cluster. Percentage blocks with corrupt replicas: 0.00%. — Details
- 0 missing blocks in the cluster. 299 total blocks in the cluster. Percentage missing blocks: 0.00%. — Details
- 0 under replicated blocks in the cluster. 299 total blocks in the cluster. Percentage under replicated blocks: 0.00%. — Details
- Canary test of file create, write, read and delete operations succeeded. — Details
- Healthy DataNode: 7. Concerning DataNode: 0. Total DataNode: 7. Percent healthy: 100.00%. Percent healthy or concerning: 100.00%. — Details

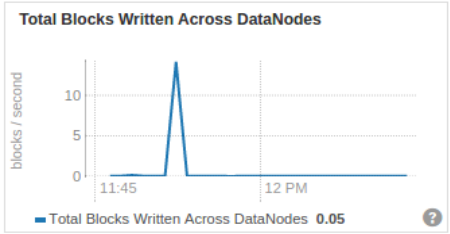
Health History

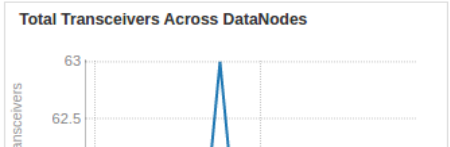
11:42:02 AM	7 Became Good	Show
11:41:42 AM	7 Became Disabled	Show



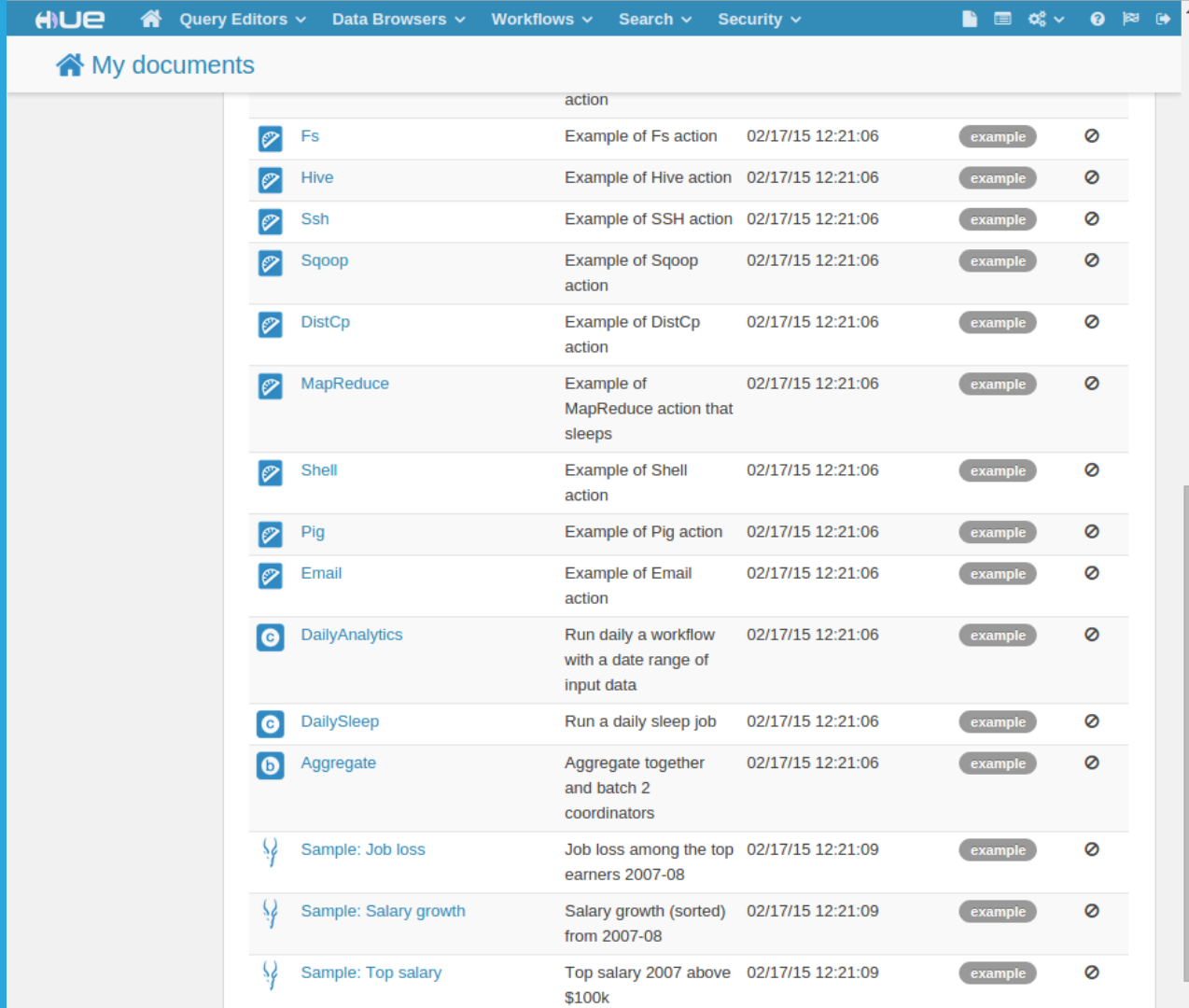








Hue Examples



cloudera

cloudera

Appendix

