

Big Data Processing Using Cloudera Quickstart with a Docker Container

July 2016

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

Modify from Original Version by Danairat T.
Certified Java Programmer, TOGAF – Silver
danairat@gmail.com

Outline

- Launch AWS EC2 Instance
- Install Docker on Ubuntu
- Pull Cloudera QuickStart to the docker
- HDFS
- HBase
- MapReduce
- Hive
- Pig
- Impala
- Sqoop

Cloudera VM

This lab will use a EC2 virtual server on AWS to install Cloudera. However, you can also use Cloudera QuickStart VM which can be downloaded from:

<http://www.cloudera.com/content/www/en-us/downloads.html>

The screenshot shows a landing page for downloading Cloudera software. The title is "Download Cloudera Enterprise" with the subtitle "Local, On Premise, or Cloud-based Apache Hadoop Management". Below this are three large blue cards:

- QuickStart VM**: Features an icon of two monitors. Text: "Get Started on your local machine using a QuickStart VM." Buttons: "DOWNLOAD NOW" and "Learn More".
- Cloudera Manager**: Features an icon of a gear. Text: "A unified interface to manage your enterprise data hub. Express and Enterprise editions available." Buttons: "DOWNLOAD NOW".
- Cloudera Director**: Features an icon of a cloud. Text: "Self-service, reliable experience for CDH and Cloudera Enterprise in the cloud" Buttons: "DOWNLOAD NOW".

Hands-On: Launch a virtual server on EC2 Amazon Web Services

**(Note: You can skip this session if you use your own
computer or another cloud service)**

Amazon Web Services

Compute

 **EC2**
Virtual Servers in the Cloud

 **Lambda** PREVIEW
Run Code in Response to Events

Storage & Content Delivery

 **S3**
Scalable Storage in the Cloud

 **Storage Gateway**
Integrates On-Premises IT Environments with Cloud Storage

 **Glacier**
Archive Storage in the Cloud

 **CloudFront**
Global Content Delivery Network

Database

 **RDS**
MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora

 **DynamoDB**
Predictable and Scalable NoSQL Data Store

 **ElastiCache**
In-Memory Cache

 **Redshift**
Managed Petabyte-Scale Data Warehouse Service

Administration & Security

 **Directory Service**
Managed Directories in the Cloud

 **Identity & Access Management**
Access Control and Key Management

 **Trusted Advisor**
AWS Cloud Optimization Expert

 **CloudTrail**
User Activity and Change Tracking

 **Config**
Resource Configurations and Inventory

 **CloudWatch**
Resource and Application Monitoring

Deployment & Management

 **Elastic Beanstalk**
AWS Application Container

 **OpsWorks**
DevOps Application Management Service

 **CloudFormation**
Templated AWS Resource Creation

 **CodeDeploy**
Automated Deployments

Analytics

 **EMR**
Managed Hadoop Framework

Application Services

 **SQS**
Message Queue Service

 **SWF**
Workflow Service for Coordinating Application Components

 **AppStream**
Low Latency Application Streaming

 **Elastic Transcoder**
Easy-to-use Scalable Media Transcoding

 **SES**
Email Sending Service

 **CloudSearch**
Managed Search Service

Mobile Services

 **Cognito**
User Identity and App Data Synchronization

 **Mobile Analytics**
Understand App Usage Data at Scale

 **SNS**
Push Notification Service

Enterprise Applications

 **WorkSpaces**
Desktops in the Cloud

 **WorkDocs**
Secure Enterprise Storage and Sharing

Resource Groups

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

[Create a Group](#)

[Tag Editor](#)

Additional Resources

Getting Started

See our documentation to get started and learn more about how to use our services.

AWS Console Mobile App

View your resources on the go with our AWS Console mobile app, available from [Amazon Appstore](#), [Google Play](#), or [iTunes](#).

AWS Marketplace

Find and buy software, launch with 1-Click and pay by the hour.

Service Health

Amazon Web Services

Compute

 **EC2**
Virtual Servers in the Cloud

 **Lambda** PREVIEW
Run Code in Response to Events

Storage & Content Delivery

 **S3**
Scalable Storage in the Cloud

 **Storage Gateway**
Integrates On-Premises IT Environments with Cloud Storage

 **Glacier**
Archive Storage in the Cloud

 **CloudFront**
Global Content Delivery Network

Database

 **RDS**
MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora

 **DynamoDB**
Predictable and Scalable NoSQL Data Store

 **ElastiCache**
In-Memory Cache

 **Redshift**
Managed Petabyte-Scale Data Warehouse Service

Administration & Security

 **Directory Service**
Managed Directories in the Cloud

 **Identity & Access Management**
Access Control and Key Management

 **Trusted Advisor**
AWS Cloud Optimization Expert

 **CloudTrail**
User Activity and Change Tracking

 **Config**
Resource Configurations and Inventory

 **CloudWatch**
Resource and Application Monitoring

Deployment & Management

 **Elastic Beanstalk**
AWS Application Container

 **OpsWorks**
DevOps Application Management Service

 **CloudFormation**
Templated AWS Resource Creation

 **CodeDeploy**
Automated Deployments

Analytics

 **EMR**
Managed Hadoop Framework

Application Services

 **SQS**
Message Queue Service

 **SWF**
Workflow Service for Coordinating Application Components

 **AppStream**
Low Latency Application Streaming

 **Elastic Transcoder**
Easy-to-use Scalable Media Transcoding

 **SES**
Email Sending Service

 **CloudSearch**
Managed Search Service

Mobile Services

 **Cognito**
User Identity and App Data Synchronization

 **Mobile Analytics**
Understand App Usage Data at Scale

 **SNS**
Push Notification Service

Enterprise Applications

 **WorkSpaces**
Desktops in the Cloud

 **WorkDocs**
Secure Enterprise Storage and Sharing

Resource Groups

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

[Create a Group](#)

[Tag Editor](#)

Additional Resources

Getting Started

See our documentation to get started and learn more about how to use our services.

AWS Console Mobile App

View your resources on the go with our AWS Console mobile app, available from [Amazon Appstore](#), [Google Play](#), or [iTunes](#).

AWS Marketplace

Find and buy software, launch with 1-Click and pay by the hour.

Service Health

Virtual Server

This lab will use a EC2 virtual server to install a Cloudera Cluster using the following features:

Ubuntu Server 14.04 LTS

Four m3.xLarge 4vCPU, 15 GB memory, 80 GB SSD

Security group: default

Keypair: imchadoop

Select a EC2 service and click on Launch Instance

AWS | Services | Edit | IMC Institute | Oregon | Support

EC2 Dashboard

- Events
- Tags
- Reports
- Limits

INSTANCES

- Instances
- Spot Requests
- Reserved Instances

IMAGES

- AMIs
- Bundle Tasks

ELASTIC BLOCK STORE

- Volumes
- Snapshots

NETWORK & SECURITY

- Security Groups
- Elastic IPs
- Placement Groups

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

0 Running Instances	0 Elastic IPs
1 Volumes	1 Snapshots
8 Key Pairs	0 Load Balancers
0 Placement Groups	11 Security Groups

Easily deploy Ruby, PHP, Java, .NET, Python, Node.js & Docker applications with [Elastic Beanstalk](#). [Hide](#)

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance

Note: Your instances will launch in the US West (Oregon) region

Service Health

Scheduled Events

Service Status: US West (Oregon): [View details](#)

AWS Marketplace

Find free software trial products in the AWS Marketplace from the [EC2 Launch Wizard](#). Or try these popular AMIs: [Vyatta Virtual Router/Firewall/VPN](#)

Feedback



Select an Amazon Machine Image (AMI) and Ubuntu Server 14.04 LTS (PV)

AWS Services Edit IMC Institute Oregon Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

Amazon Linux AMI 2014.09.2 (PV) - ami-9fc29baf

Amazon Linux Free tier eligible

The Amazon Linux AMI is an EBS backed image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Apache HTTPD, Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: paravirtual

SUSE Linux Enterprise Server 11 SP3 (PV), SSD Volume Type - ami-5df2ab6d

SUSE Linux Enterprise Server 11 Service Pack 3 (PV), EBS General Purpose (SSD) Volume Type. Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.5, PHP 5.3, and Ruby 1.8.7 available.

Root device type: ebs Virtualization type: paravirtual

Ubuntu Server 14.04 LTS (PV), SSD Volume Type - ami-23ebb513

Ubuntu Server 14.04 LTS (PV), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: paravirtual

Select 64-bit

Select 64-bit

Select 64-bit



Choose m3.xlarge Type virtual server

AWS Services Edit IMC Institute Oregon Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 2: Choose an Instance Type

					Available	
<input type="checkbox"/>	Micro instances	t1.micro Free tier eligible	1	0.613	EBS only	-
<input type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-
<input type="checkbox"/>	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-
<input type="checkbox"/>	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-
<input checked="" type="checkbox"/>	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes
<input type="checkbox"/>	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes

Cancel Previous **Review and Launch** Next: Configure Instance Details



AWS

Services

Edit

IMC Institute

N. Virginia

Support

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Tag Instance
6. Configure Security Group
7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances

1

[Launch into Auto Scaling Group](#)**Purchasing option** Request Spot instances**Network**

vpc-ccdf24a9 (172.31.0.0/16) (default)

[Create new VPC](#)**Subnet**

No preference (default subnet in any Availability Zone)

[Create new subnet](#)**Auto-assign Public IP**

Use subnet setting (Enable)

**IAM role**

None

[Create new IAM role](#)**Shutdown behavior**

Stop

**Enable termination protection** Protect against accidental termination[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Add Storage](#)

Add Storage: 80 GB

AWS Services Edit IMC Institute Oregon Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination	Encrypted
Root	/dev/sda1	snap-306df873	80	General Purpose S	240 / 3000	<input checked="" type="checkbox"/>	Not Encrypted
Instance Store 0	/dev/sdb	N/A	N/A	N/A	N/A	N/A	Not Encrypted 
Instance Store 1	/dev/sdc	N/A	N/A	N/A	N/A	N/A	Not Encrypted 

Add New Volume

Cancel Previous **Review and Launch** Next: Tag Instance

Name the instance

The screenshot shows the AWS EC2 instance creation process at Step 5: Tag Instance. The top navigation bar includes links for AWS, Services, Edit, IMC Institute, Oregon, and Support. Below the navigation is a progress bar with steps 1 through 7. Step 5, "Tag Instance", is highlighted with an orange underline. The main area is titled "Step 5: Tag Instance" and contains instructions about tagging EC2 resources. A table allows defining key-value pairs. One row is shown with the key "Name" and value "Cloudera-Demo". A "Create Tag" button is available for adding more. At the bottom are buttons for Cancel, Previous, Review and Launch (which is blue and bold), and Next: Configure Security Group.

Key	(127 characters maximum)	Value	(255 characters maximum)
Name	Cloudera-Demo	X	

Create Tag (Up to 10 tags maximum)

Cancel Previous **Review and Launch** Next: Configure Security Group

Select Create an existing security group > Default

AWS Services Edit IMC Institute Oregon Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group

Step 6: Configure Security Group

<input type="checkbox"/> sg-f5468393 awseb-e-yus3d2g5qk-stack-AWSEBSecurityGroup-YKZSMZVQZY90	SecurityGroup for ElasticBeanstalk environment
<input type="checkbox"/> sg-e4271581Cassandra Security	Security Group for Cassandra
<input type="checkbox"/> sg-1adbf77d cloudera-sgp	launch-wizard-35 created 2016-04-23T09:00:00Z
<input type="checkbox"/> sg-36fe2d50 cluster2-2-ClusterNodeSecurityGroup-H1QQUXYP4C2E	Allow access from web and bastion as well as
<input type="checkbox"/> sg-2f23b84b Danairat_SecureGroup	launch-wizard-5 created 2015-10-07T04:40:54Z
<input type="checkbox"/> sg-793ef81f DBServerSG	Security
<input checked="" type="checkbox"/> sg-2e1cff41 default	default VPC security group
<input type="checkbox"/> sg-46638029ElasticMapReduce-master	Master group for Elastic MapReduce

Inbound rules for sg-2e1cff41 (Selected security groups: sg-2e1cff41)

Type	Protocol	Port Range	Source
HTTP	TCP	80	0.0.0.0/0

Cancel Previous Review and Launch

Click Launch and choose imchadoop as a key pair

Select an existing key pair or create a new key pair X

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair

Select a key pair

imchadoop

I acknowledge that I have access to the selected private key file (imchadoop.pem), and that without this file, I won't be able to log into my instance.

[Cancel](#) [Launch Instances](#)

Review an instance and rename one instance as a master / click **Connect** for an instruction to connect to the instance

The screenshot shows the AWS EC2 Instances page. The top navigation bar includes 'AWS', 'Services', 'Edit', 'IMC Institute', 'Oregon', and 'Support'. Below the navigation is a sidebar with links for EC2 Dashboard, Events, Tags, Reports, Limits, and sections for INSTANCES, Instances, Spot Requests, Reserved Instances, Scheduled Instances, Commands, and Dedicated Hosts. The main content area displays a table of five instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status
Cloudera-Demo	i-783431a2	m3.xlarge	us-west-2c	running	2/
Cloudera-Demo	i-7e3431a4	m3.xlarge	us-west-2c	running	2/
Cloudera-Demo-Master	i-7f3431a5	m3.xlarge	us-west-2c	running	2/
Cloudera-Demo	i-793431a3	m3.xlarge	us-west-2c	running	2/

Below the table, it says 'Instance: i-7f3431a5 (Cloudera-Demo-Master)' and 'Public DNS: ec2-54-201-147-59.us-west-2.compute.amazonaws.com'. The 'Actions' dropdown menu is open, and a red arrow points from the 'Connect' button in the top navigation bar to this menu.

Connect to an instance from Mac/Linux

Connect To Your Instance

I would like to connect with A standalone SSH client A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (imchadoop.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
`chmod 400 imchadoop.pem`
4. Connect to your instance using its Public DNS:
`ec2-54-201-147-59.us-west-2.compute.amazonaws.com`

Example:

`ssh -i "imchadoop.pem" ubuntu@ec2-54-201-147-59.us-west-2.compute.amazonaws.com`

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

Can also view details of the instance such as Public IP and Private IP

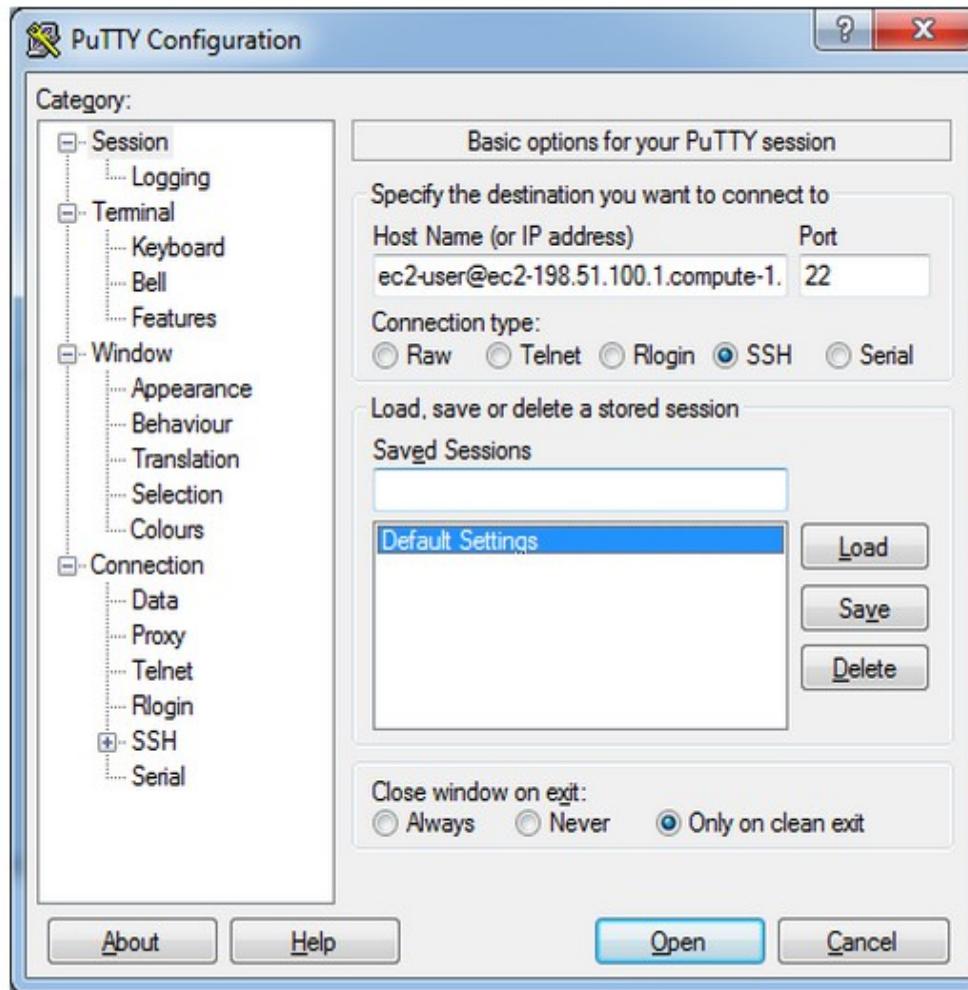
The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (with Instances selected), Spot Requests, Reserved Instances, Scheduled Instances, Commands, Dedicated Hosts, and IMAGES (with AMIs selected). The main area has tabs for Launch Instance, Connect, and Actions. Below that is a search bar and a table with columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, and Status. One row is selected for 'Cloudera-Demo-Master'. The table provides detailed information for this instance, including its state, type, DNS names, and network details. Two specific fields are highlighted with red circles: 'Private IPs' (172.31.10.53) and 'Public IP' (54.201.147.59).

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status
Cloudera-Demo-Master	i-7f3431a5	m3.xlarge	us-west-2c	running	2/2

Instance state: running
Instance type: m3.xlarge
Private DNS: ip-172-31-10-53.us-west-2.compute.internal
Public IP: 54.201.147.59
Elastic IP: -
Availability zone: us-west-2c
Security groups: default, view rules
Scheduled events: No scheduled events
AMI ID: ubuntu-trusty-14.04-amd64-server-20160114.5 (ami-)

Private IPs: 172.31.10.53

Connect to an instance from Windows using Putty



Connect to the instance

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

WARNING! Your environment specifies an invalid locale.

This can affect your user experience significantly, including the ability to manage packages. You may install the locales by running:

```
sudo apt-get install language-pack-UTF-8
or
sudo locale-gen UTF-8
```

To see all available language packs, run:

```
apt-cache search "^language-pack-[a-z][a-z]$"
```

To disable this message for all users, run:

```
sudo touch /var/lib/cloud/instance/locale-check.skip
```

```
ubuntu@ip-172-31-1-242:~$
```

Hands-On: Installing Cloudera Quickstart on Docker Container

Installation Steps

- Update OS
- Install Docker
- Pull Cloudera Quickstart
- Run Cloudera Quickstart
- Run Cloudera Manager

Update OS (Ubuntu)

- Command: sudo apt-get update

```
ubuntu@ip-172-31-30-238:~$ sudo apt-get update
Ign http://us-east-1.ec2.archive.ubuntu.com trusty InRelease
Get:1 http://us-east-1.ec2.archive.ubuntu.com trusty-updates InRelease [65.9 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com trusty-backports InRelease [65.9 kB]
Hit http://us-east-1.ec2.archive.ubuntu.com trusty Release.gpg
Hit http://us-east-1.ec2.archive.ubuntu.com trusty Release
Get:3 http://security.ubuntu.com trusty-security InRelease [65.9 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com trusty-updates/main Sources [277 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com trusty-updates/restricted Sources [535
2 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com trusty-updates/universe Sources [156 k
B]
Get:7 http://us-east-1.ec2.archive.ubuntu.com trusty-updates/multiverse Sources [593
9 B]
Get:8 http://us-east-1.ec2.archive.ubuntu.com trusty-updates/main amd64 Packages [78
1 kB]
```

Docker Installation

- Command: sudo apt-get install docker.io

```
ubuntu@ip-172-31-30-238:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  aufs-tools cgroup-lite git git-man liberror-perl
Suggested packages:
  btrfs-tools debootstrap lxc rinse git-daemon-run git-daemon-sysvinit git-doc
  git-el git-email git-gui gitk gitweb git-arch git-bzr git-cvs git-mediawiki
  git-svn
The following NEW packages will be installed:
  aufs-tools cgroup-lite docker.io git git-man liberror-perl
0 upgraded, 6 newly installed, 0 to remove and 84 not upgraded.
Need to get 8150 kB of archives.
After this operation, 51.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/ trusty/universe aufs-tools amd
64 1:3.2+20130722-1.1 [92.3 kB]
```

Pull Cloudera Quickstart

- Command: sudo docker pull cloudera/quickstart:latest

```
ubuntu@ip-172-31-30-238:~$ sudo docker pull cloudera/quickstart:latest
latest: Pulling from cloudera/quickstart
2cda82941cb7: Already exists
Digest: sha256:f91bee4cdfa2c92ea3652929a22f729d4d13fc838b00f120e630f91c941acb63
Status: Downloaded newer image for cloudera/quickstart:latest
ubuntu@ip-172-31-30-238:~$ █
```

Show docker images

- Command: sudo docker images

```
ubuntu@ip-172-31-30-238:~$ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED
 VIRTUAL SIZE
cloudera/quickstart  latest   2cda82941cb7  9 weeks ago
 6.336 GB
```

Run Cloudera quickstart

- Command: sudo docker run
--hostname=quickstart.cloudera --privileged=true -t -i
[OPTIONS] [IMAGE] /usr/bin/docker-quickstart

Example: sudo docker run
--hostname=quickstart.cloudera --privileged=true -t -i -p
8888:8888 cloudera/quickstart /usr/bin/docker-quickstart

```
ubuntu@ip-172-31-30-238:~$ sudo docker run --hostname=quickstart.cloudera --privileged=true -t -i -p 8888:8888 -p 7180:7180 cloudera/quickstart /usr/bin/docker-quickstart
Starting mysqld:                                         [ OK ]  
  
if [ "$1" == "start" ] ; then
  if [ "${EC2}" == 'true' ] ; then
    FIRST_BOOT_FLAG=/var/lib/cloudera-quickstart/.ec2-key-installed
    if [ ! -f "${FIRST_BOOT_FLAG}" ] ; then
      METADATA_API=http://169.254.169.254/latest/meta-data
      KEY_URL=${METADATA API}/public-keys/0/openssh-key
```

Finding the EC2 instance's DNS

Connect To Your Instance X

I would like to connect with A standalone SSH client A Java SSH Client directly from my browser (Java required)

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (cloudera.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
`chmod 400 cloudera.pem`
4. Connect to your instance using its Public DNS:
`ec2-54-173-154-79.compute-1.amazonaws.com`

Example:

```
ssh -i "cloudera.pem" ubuntu@ec2-54-173-154-79.compute-1.amazonaws.com
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Login to Hue

http://ec2-54-173-154-79.compute-1.amazonaws.com:8888

Welcome to Hue
Sign in to continue to your dashboard



Username

Password

Sign in

Hue and the Hue logo are trademarks of Cloudera, Inc.

Quick Start Wizard - Hue™ 3.9.0 - The Hadoop UI

Step 1: Check Configuration

Step 2: Examples

Step 3: Users

Step 4: Go!

Checking current configuration

Configuration files located in </etc/hue/conf.empty>

All OK. Configuration check passed.

Back

Next

Hue and the Hue logo are trademarks of Cloudera, Inc.

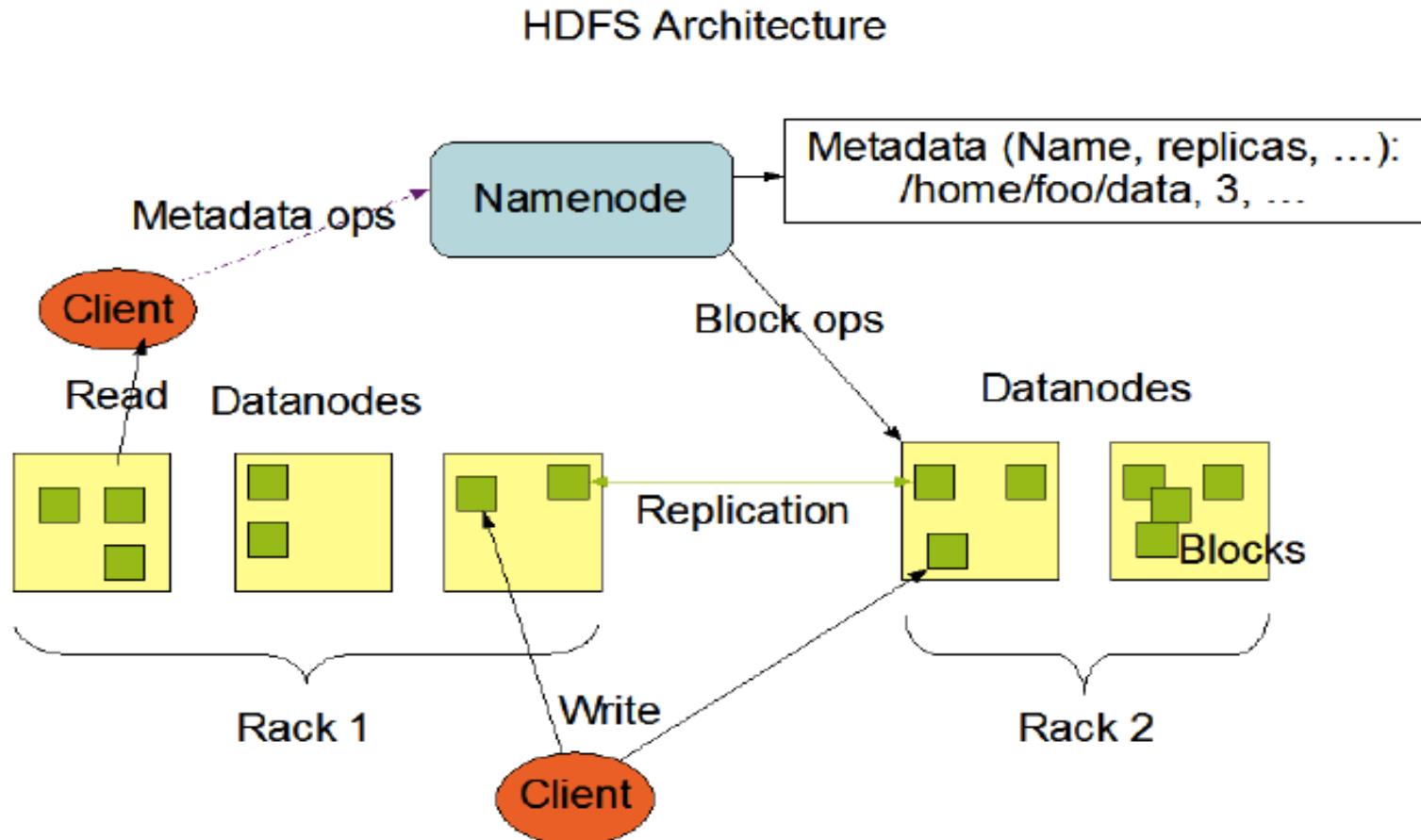
Hadoop File System (HDFS)

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

HDFS

- Default storage for the Hadoop cluster
- Data is distributed and replicated over multiple machines
- Designed to handle very large files with streaming data access patterns.
- NameNode/DataNode
- Master/slave architecture (1 master 'n' slaves)
- Designed for large files (64 MB default, but configurable) across all the nodes

HDFS Architecture



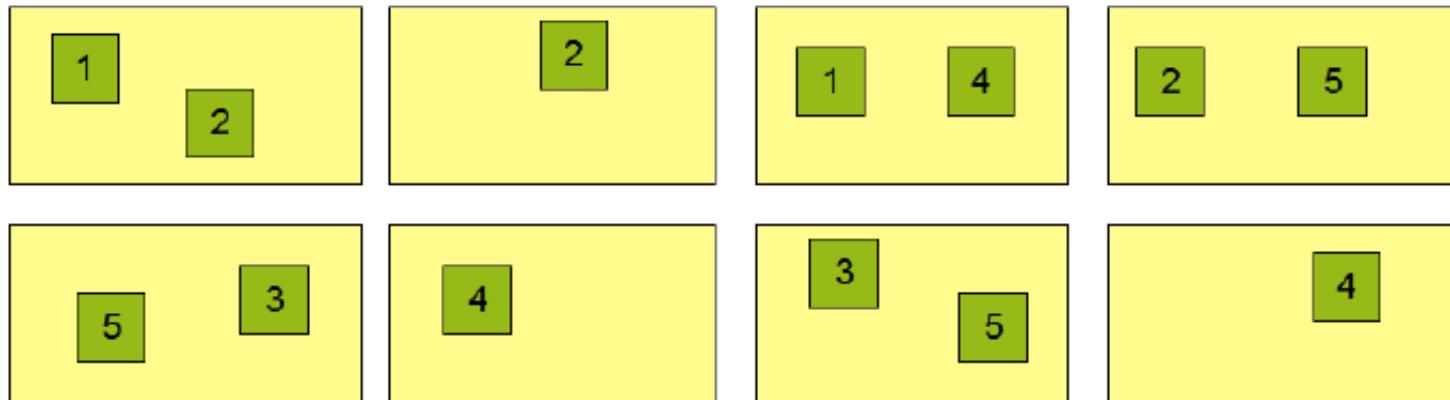
Source Hadoop: Shashwat Shriparv

Data Replication in HDFS

Block Replication

```
Namenode (Filename, numReplicas, block-ids, ...)  
/users/sameerp/data/part-0, r:2, {1,3}, ...  
/users/sameerp/data/part-1, r:3, {2,4,5}, ...
```

Datanodes



Source Hadoop: Shashwat Shriparv

How does HDFS work?

A file we want to store on HDFS ...

600 MB

We're raising the question because no one else wants to, because no one else wants to say what needs to be said.

And let's be real, it's the two-ton elephant in the room with nearly every other star's name on the trade rumor radar these days.

We've read over and over again about Nash refusing to ask for a trade, refusing to play the game that so many others have late in their careers.

Source Introduction to Apache Hadoop-Pig: PrashantKommireddi

How does HDFS work?

HDFS Splits file into **blocks** ...

256 MB

We're raising the question because no one else wants to, because no one else wants to say what needs to be said.

256 MB

And let's be real, it's the two-ton elephant in the room with nearly every other star's name on the trade rumor radar these days.

88 MB

We've read over and over again about Nash refusing to play the game that so many others have late in their careers.

Source Introduction to Apache Hadoop-Pig: PrashantKommireddi

How does HDFS work?

HDFS will create **3replicas** of each block ...

3 copies

We're raising the question because no one else wants to, because no one else wants to say what needs to be said.

3 copies

And let's be real, it's the two-ton elephant in the room with nearly every other star's name on the trade rumor radar these days.

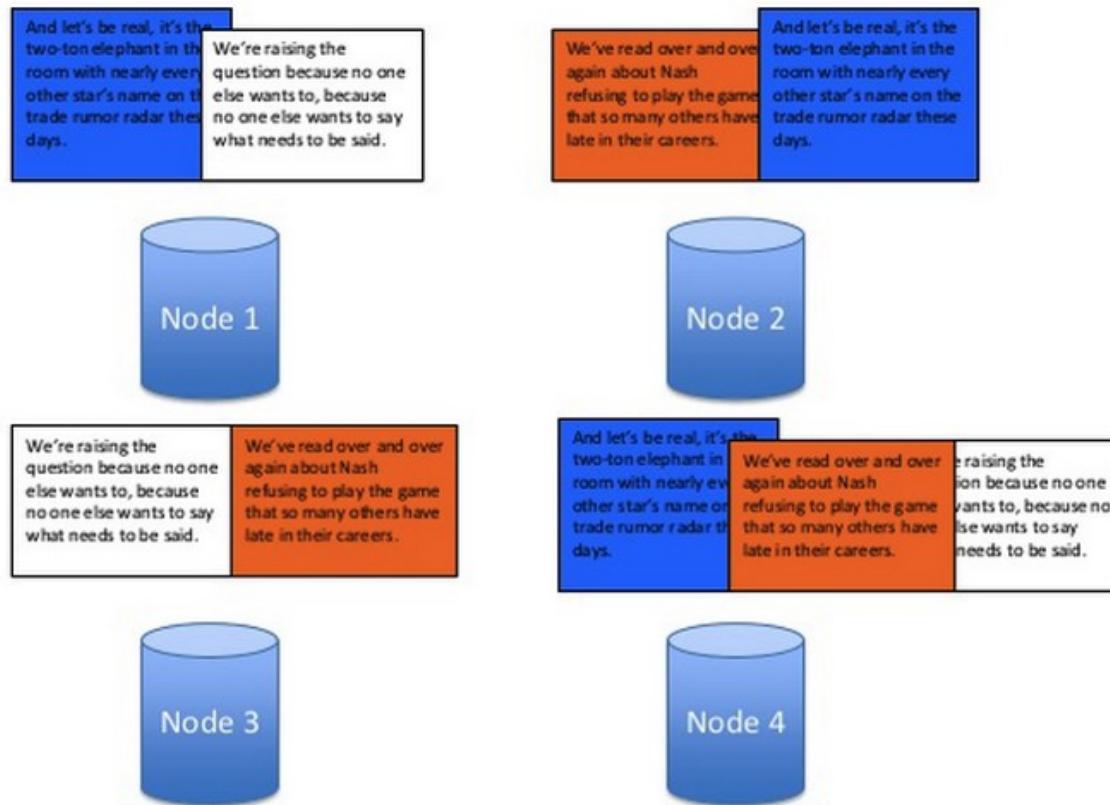
3 copies

We've read over and over again about Nash refusing to play the game that so many others have late in their careers.

Source Introduction to Apache Hadoop-Pig: PrashantKommireddi

How does HDFS work?

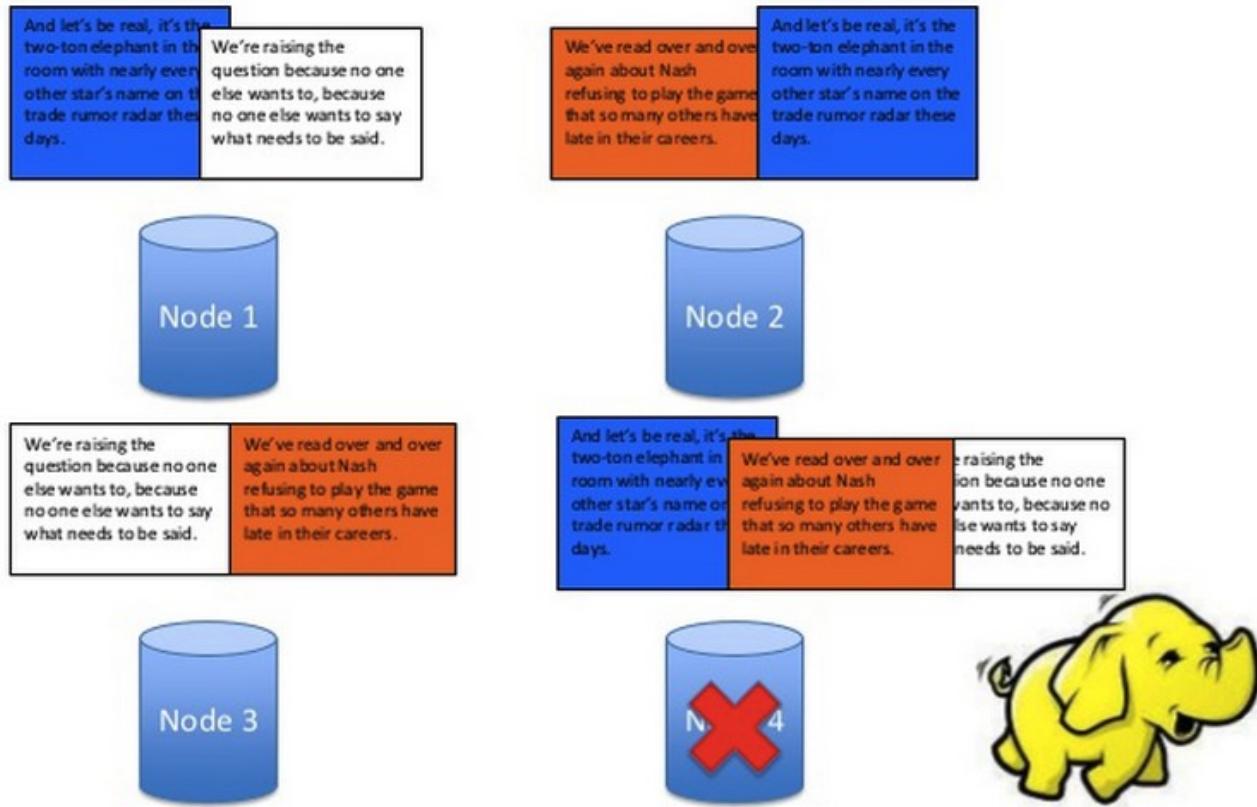
HDFS distributes these replicas across the cluster ...



Source Introduction to Apache Hadoop-Pig: PrashantKommireddi

How does HDFS work?

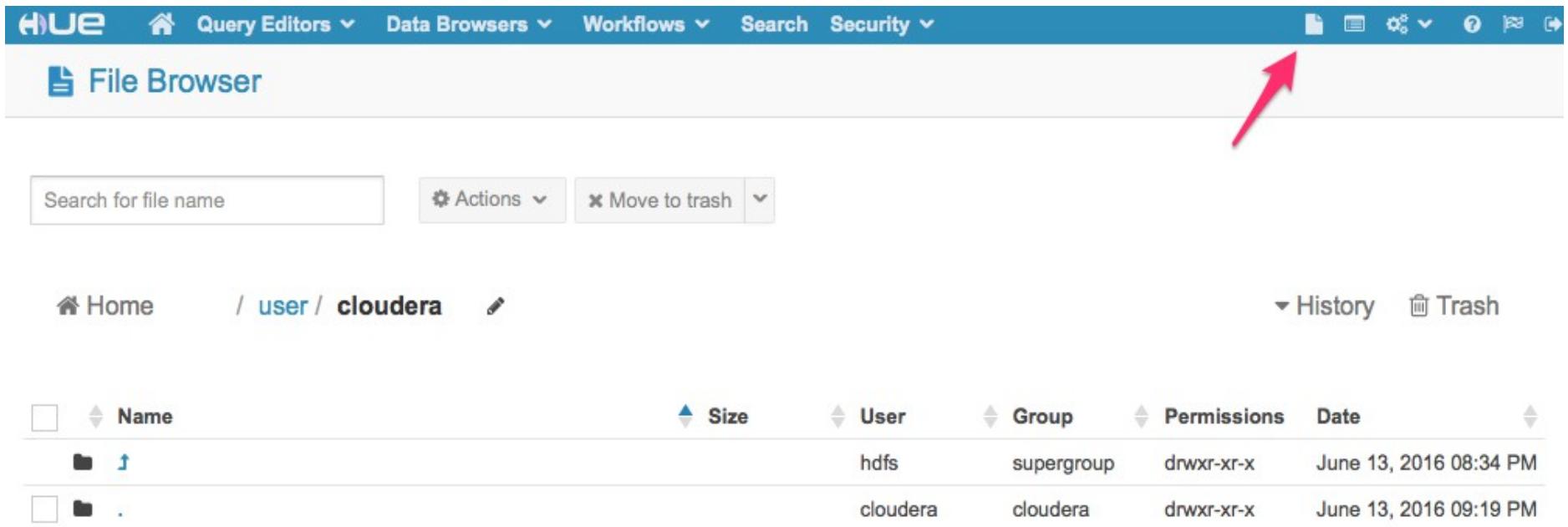
If a node goes down, we have copies elsewhere



Source Introduction to Apache Hadoop-Pig: PrashantKommireddi

Hands-On: Importing/Exporting Data to HDFS

Review file in Hadoop HDFS using File Browse



The screenshot shows the Hue File Browser interface. At the top, there is a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, Search, Security, and various system icons. Below the navigation bar, the title "File Browser" is displayed next to a folder icon. On the left, there is a search bar labeled "Search for file name" and a set of buttons for "Actions" and "Move to trash". The main area shows a file listing for the directory "/user/cloudera". The table has columns for Name, Size, User, Group, Permissions, and Date. Two entries are listed:

Name	Size	User	Group	Permissions	Date
..		hdfs	supergroup	drwxr-xr-x	June 13, 2016 08:34 PM
.		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:19 PM

Create a new directory name as: **input & output**

The screenshot shows the Hue File Browser interface. At the top, there is a navigation bar with links for Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the main title is "File Browser". On the left, there is a sidebar with actions like "Actions", "Move to trash", "Upload", and "New". The main area displays a list of files and directories under the path "/user/cloudera". The list includes two entries: "hdfs" (User: supergroup, Group: supergroup, Permissions: drwxr-xr-x, Date: June 13, 2016 08:34 PM) and "cloudera" (User: cloudera, Group: cloudera, Permissions: drwxr-xr-x, Date: June 13, 2016 09:19 PM). On the right side, there is a modal window for creating a new directory. The "Directory Name" field contains "input". Below the field are "Cancel" and "Create" buttons. A red arrow points to the "Directory" option in the "New" dropdown menu, which is highlighted with a gray box.

Size	User	Group	Permissions	Date
	hdfs	supergroup	drwxr-xr-x	June 13, 2016 08:34 PM
	cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:19 PM

Directory Name

Cancel Create

HUE  Query Editors ▾ Data Browsers ▾ Workflows ▾ Search Security ▾       

File Browser

Search for file name Actions ▾  Move to trash ▾

 Home / user / cloudera  History  Trash

<input type="checkbox"/>	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	 		hdfs	supergroup	drwxr-xr-x	June 13, 2016 08:34 PM
<input type="checkbox"/>	 .		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:21 PM
<input type="checkbox"/>	 input		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:20 PM
<input type="checkbox"/>	 output		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:21 PM

Upload a local file to HDFS

The screenshot shows the Hue File Browser interface. At the top, there is a navigation bar with links for Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the title "File Browser" is displayed. On the left, there is a search bar and a breadcrumb navigation path: "/ user / cloudera / input". To the right of the path are buttons for Actions, Move to trash, and a dropdown menu labeled "Upload" which has options for "Files" and "Zip/Tgz/Bz2 fil". A red arrow points to the "Files" option in the dropdown menu. Below the path, there is a toolbar with History and Trash buttons. The main area displays a table of files in the "/user/cloudera/input" directory. The table has columns for Name, Size, User, Group, Permissions, and Date. Two files are listed: "03_Suitability test.pdf" and "03_Suitability test.pdf".

Name	Size	User	Group	Permissions	Date
03_Suitability test.pdf		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:21 PM
03_Suitability test.pdf		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:20 PM

Upload to /user/cloudera/input

Select files

or drag and drop them here

03_Suitability test.pdf

99% from 0.3MB x

HUE Home Query Editors Data Browsers Workflows Search Security

File Browser

Search for file name Actions Move to trash

Home / user / cloudera / input

History Trash

	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	..		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:21 PM
<input type="checkbox"/>	.		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:22 PM
<input type="checkbox"/>	03_Suitability test.pdf	336.8 KB	cloudera	cloudera	-rw-r--r--	June 13, 2016 09:22 PM

Hands-On: Connect to a master node via SSH

SSH Login to a master node

```
THANACHARTs-MacBook-Air:elastic-mapreduce-cli THANACHART$ ssh -i "imchadoop.pem" ub  
untu@ec2-54-201-147-59.us-west-2.compute.amazonaws.com  
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-74-generic x86_64)
```

* Documentation: <https://help.ubuntu.com/>

System information as of Sun Mar 27 09:08:01 UTC 2016

System load: 0.0	Processes: 135
Usage of /: 27.6% of 29.40GB	Users logged in: 0
Memory usage: 24%	IP address for eth0: 172.31.10.53
Swap usage: 0%	

Graph this data and manage this system at:
<https://landscape.canonical.com/>

Get cloud support with Ubuntu Advantage Cloud Guest:
<http://www.ubuntu.com/business/services/cloud>

*** System restart required ***

```
Last login: Sun Mar 27 09:08:01 2016 from node-io5.pool-125-24.dynamic.totbb.net  
ubuntu@ip-172-31-10-53:~$ █
```

Hadoop syntax for HDFS

Command	Syntax
Listing of files in a directory	<code>hadoop fs -ls /user</code>
Create a new directory	<code>hadoop fs -mkdir /user/guest/newdirectory</code>
Copy a file from a local machine to Hadoop	<code>hadoop fs -put C:\Users\Administrator\Downloads\localfile.csv /user/rajn/newdirectory/hadoopfile.txt</code>
Copy a file from Hadoop to a local machine	<code>hadoop fs -get /user/rajn/newdirectory/hadoopfile.txt C:\Users\Administrator\Desktop\</code>
Tail last few lines of a large file in Hadoop	<code>hadoop fs -tail /user/rajn/newdirectory/hadoopfile.txt</code>
View the complete contents of a file in Hadoop	<code>hadoop fs -cat /user/rajn/newdirectory/hadoopfile.txt</code>
Remove a complete directory from Hadoop	<code>hadoop fs -rm -r /user/rajn/newdirectory</code>
Check the Hadoop filesystem space utilization	<code>hadoop fs -du /</code>

Install wget

- Command: yum install wget

```
[root@quickstart /]# yum install wget
Loaded plugins: fastestmirror
Setting up Install Process
Determining fastest mirrors
epel/metalink | 13 kB     00:00
 * base: mirrors.evowise.com
 * epel: mirror.cogentco.com
 * extras: mirror.us.leaseweb.net
 * updates: mirror.cs.pitt.edu
base | 3.7 kB     00:00
base/primary_db | 4.7 MB     00:06
```

Download an example text file

Make your own directory at a master node to avoid mixing with others

```
$mkdir guest1
$cd guest1
$wget https://s3.amazonaws.com/imcbucket/input/pg2600.txt
```

```
--2016-03-27 09:58:48-- https://s3.amazonaws.com/imcbucket/input/pg2600.txt
Resolving s3.amazonaws.com (s3.amazonaws.com)... 54.231.19.187
Connecting to s3.amazonaws.com (s3.amazonaws.com)|54.231.19.187|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3291648 (3.1M) [text/plain]
Saving to: 'pg2600.txt'

100%[=====>] 3,291,648 3.14MB/s in 1.0s

2016-03-27 09:58:50 (3.14 MB/s) - 'pg2600.txt' saved [3291648/3291648]
```

Upload Data to Hadoop

```
$hadoop fs -ls /user/cloudera/input  
$hadoop fs -rm /user/cloudera/input/*  
$hadoop fs -put pg2600.txt /user/cloudera/input/  
$hadoop fs -ls /user/cloudera/input
```

```
[root@quickstart guest1]# hadoop fs -ls /user/cloudera/input  
Found 1 items  
-rw-r--r-- 1 root cloudera 3291648 2016-06-14 04:29 /user/cloudera/input/pg2600.txt  
[root@quickstart guest1]#
```



Lecture

Understanding HBase

Introduction

An open source, non-relational, distributed database



HBase is an open source, non-relational, distributed database modeled after Google's BigTable and is written in Java. It is developed as part of Apache Software Foundation's Apache Hadoop project and runs on top of HDFS (, providing BigTable-like capabilities for Hadoop. That is, it provides a fault-tolerant way of storing large quantities of sparse data.

HBase Features

- Hadoop database modelled after Google's Bigtable
- Column oriented data store, known as Hadoop Database
- Support random realtime CRUD operations (unlike HDFS)
- No SQL Database
- Opensource, written in Java
- Run on a cluster of commodity hardware

When to use HBase?

- When you need high volume data to be stored
- Un-structured data
- Sparse data
- Column-oriented data
- Versioned data (same data template, captured at various time, time-elapse data)
- When you need high scalability

Which one to use?

- HDFS
 - Only append dataset (no random write)
 - Read the whole dataset (no random read)
- HBase
 - Need random write and/or read
 - Has thousands of operation per second on TB+ of data
- RDBMS
 - Data fits on one big node
 - Need full transaction support
 - Need real-time query capabilities

HBase vs. RDBMS

	HBase	RDBMS
Hardware architecture	Similar to Hadoop. Clustered commodity hardware. Very affordable.	Typically large scalable multiprocessor systems. Very expensive.
Fault Tolerance	Built into the architecture. Lots of nodes means each is relatively insignificant. No need to worry about individual node downtime.	Requires configuration of the HW and the RDBMS with the appropriate high availability options.
Typical Database Size	Terabytes to Petabytes - hundred of millions to billions of rows.	Gigabytes to Terabytes – hundred of thousands to millions of rows.
Data Layout	A sparse, distributed, persistent, multidimensional sorted map.	Rows or column oriented.
Data Types	Bytes only.	Rich data type support.
Transactions	ACID support on a single row only	Full ACID compliance across rows and tables
Query Language	API primitive commands only, unless combined with Hive or other technology	SQL
Indexes	Row-Key only unless combined with other technologies such as Hive or IBM's BigSQL	Yes
Throughput	Millions of queries per second	Thousands of queries per second

- Given this RDBMS:

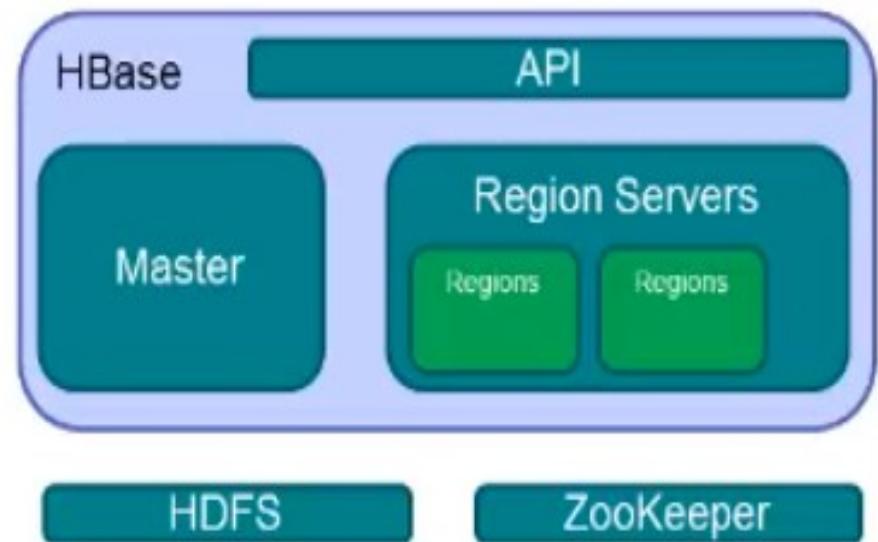
ID (Primary key)	Last name	First name	Password	Timestamp
1234	Smith	John	Hello, world!	20130710
5678	Cooper	Joyce	wysiwyg	20120825
5678	Cooper	Joyce	wisiwig	20130916

- Logical view in HBase:

Row-Key	Value (CF, Qualifier, Version)
1234	info {'lastName': 'Smith', 'firstName': 'John'} pwd {'password': 'Hello, world!'}
5678	info {'lastName': 'Cooper', 'firstName': 'Joyce'} pwd {'password': 'wysiwyg'@ts 20130916, 'password': 'wisiwig'@ts 20120825}

HBase Components

- Region
 - Row of table are stores
- Region Server
 - Hosts the tables
- Master
 - Coordinating the Region Servers
- ZooKeeper
- HDFS
- API
 - The Java Client API



HBase Shell Commands

- See the list of the tables

```
list
```

- Create a table:

```
create 'testTable', 'cf'
```

- Insert data into a table:

Insert at rowA, column "cf:columnName" with a value of "val1"

```
put 'testTable', 'rowA', 'cf:columnName', 'val1'
```

- Retrieve data from a table:

Retrive "rowA" from the table "testTable"

```
get 'testTable', 'rowA'
```

- Iterate through a table:

```
- scan 'testTable'
```

- Delete a table:

```
enable 'testTable'  
drop 'testTable'
```

Hands-On: Running HBase

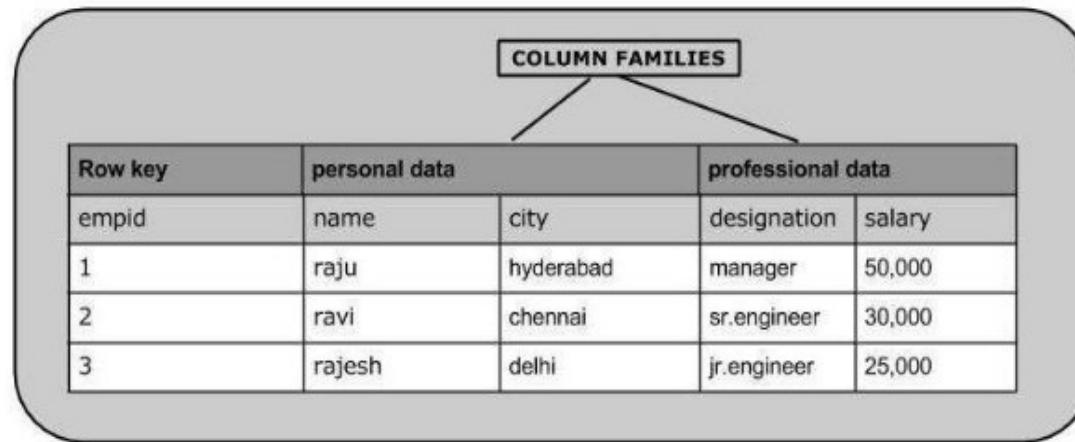
Hbase shell

Row key	personal data	professional data

```
$hbase shell  
hbase(main):001:0> create 'employee', 'personal data',  
'professional data'  
hbase(main):002:0> list
```

```
TABLE  
employee  
1 row(s) in 0.0310 seconds
```

Create Data



```
hbase(main):010:0> put 'employee','1','personal data:name','raju'  
0 row(s) in 0.1720 seconds
```

```
hbase(main):011:0> put 'employee','1','personal data:city','hyderabad'  
0 row(s) in 0.0140 seconds
```

```
hbase(main):018:0> put 'employee','1','professional data:designation','manager'  
0 row(s) in 0.0110 seconds
```

```
hbase(main):019:0> put 'employee','1','professional data:salary','50000'  
0 row(s) in 0.0070 seconds
```

Running HBase Browser

The screenshot shows the Hue Data Browser interface. The top navigation bar includes links for Home, Query Editors, Data Browsers (with Metastore Tables and HBase options), Workflows, Search, File Browser, Job Browser, and a user session (guest1). A red arrow points to the 'HBase' option in the 'Data Browsers' dropdown menu. The main content area is titled 'Home - HBase'. It features a search bar ('Search for Table Name'), checkboxes for 'Enable' (checked) and 'Disable', and a 'Drop' button. Below these controls is a table header with columns for 'Table Name' and 'Enabled'. The table body displays the message 'No data available in table'.

Viewing Employee Table

HUE Home Query Editors Data Browsers Workflows Search Security

H HBase Browser

Home - Cluster Switch Cluster

Search for Table Name Enable Disable Drop New Table

Table Name Enabled

employee

H HBase Browser

Home - Cluster / employee Switch Cluster

row_key, row_prefix* +scan_len [col1, family:col2, fam3:, col_prefix] personal data: professional data:

Filter Columns/Families All Sort By ASC

1			
personal data: city	personal data: name	professional data: designation	professional data: salary
hyderabad	raju	manager	50000

Create a table in HBase

H HBase Browser

Home - HBase Switch Cluster ▾

Search for Table Name Enable Disable Drop

Table Name Enabled

 New Table

Create New Table

Table Name:

Column Families:

Add a column property

Add an additional column family

Cancel Submit

Insert a new row in a table

HBase Browser

Home - HBase / Student

Switch Cluster ▾

row_key, row_prefix* +scan_len [col1, family:col2, fam3:, col_prefix* +3, fam:]

Filter Columns/Families

All Sort By DESC

No rows to display.

Fetched 123 in 0.374 seconds.

Drop Rows Bulk Upload New Row



Add field into a new row

Insert New Row

Row Key

[+ Add Field](#)

[Cancel](#)

[Submit](#)

Insert New Row

Row Key

[+ Add Field](#)

[Cancel](#)

[Submit](#)

Home - HBase / Student

[Switch Cluster ▾](#)

row_key, row_prefix* +scan_len [col1, family:col2, fam3:, col_prefix* +3, fam:

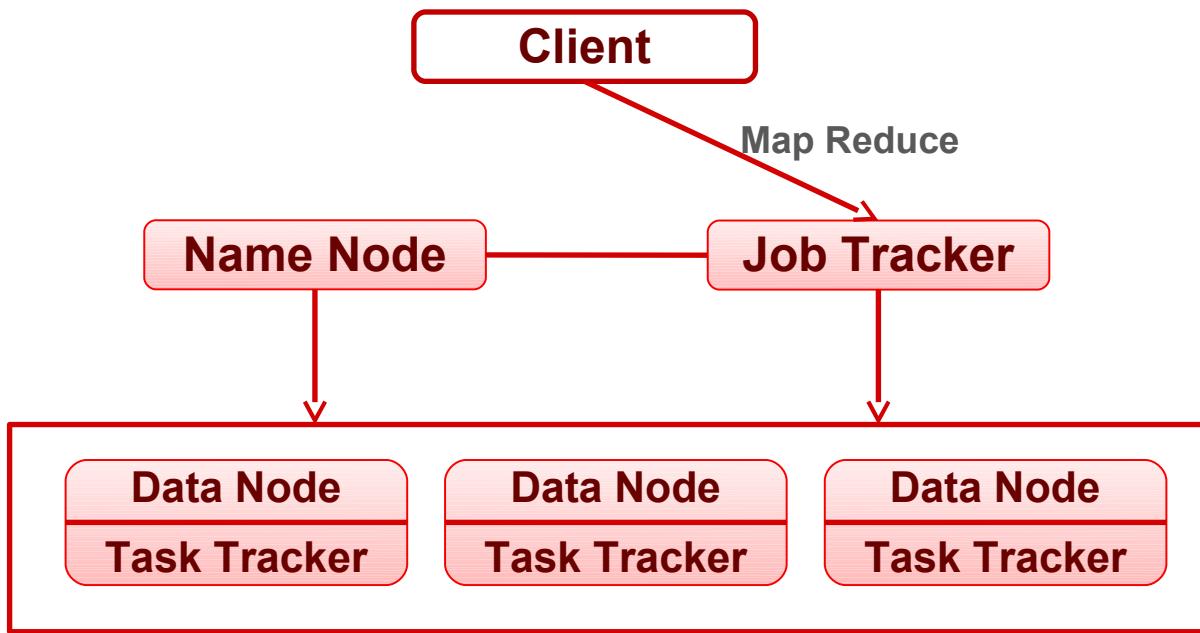


Filter Columns/Families

All

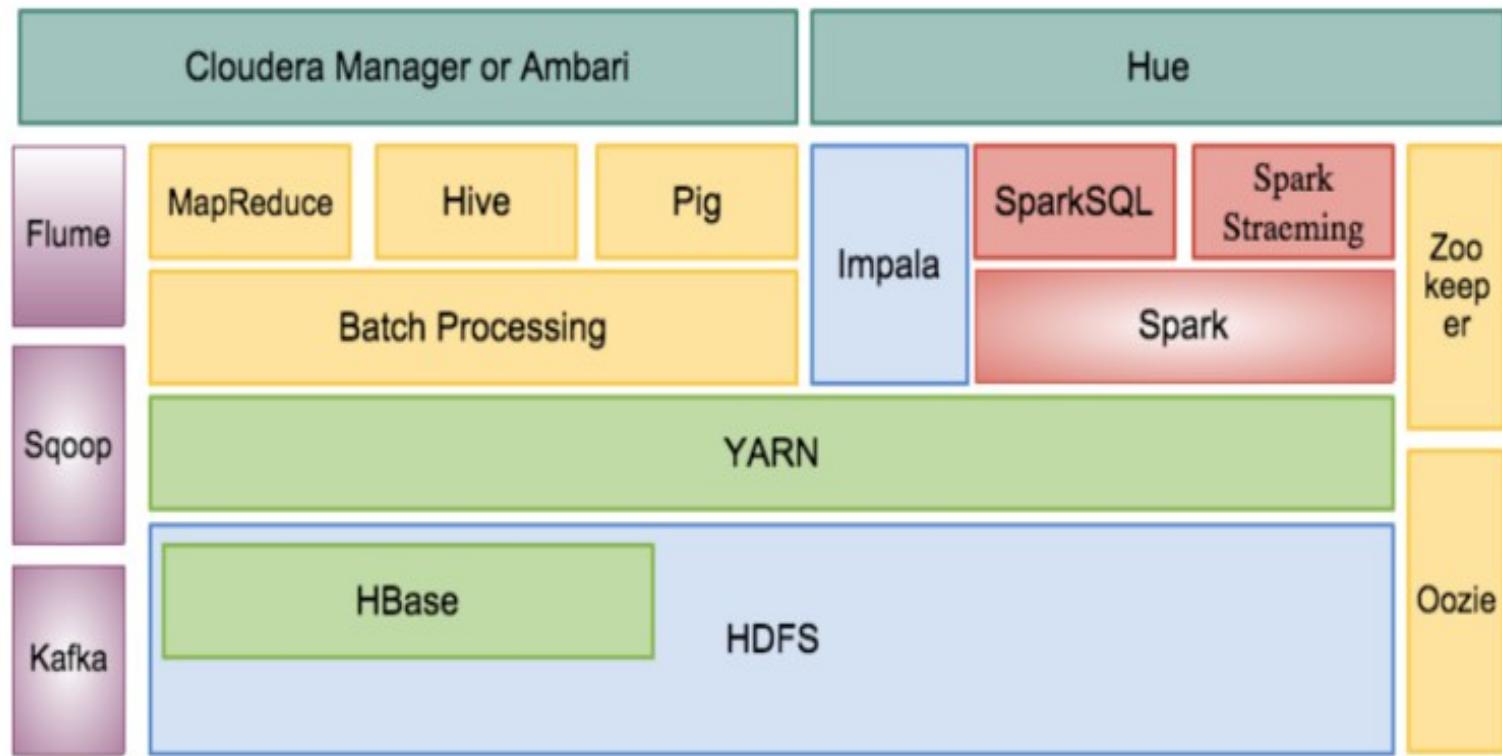
[Sort By DESC ▾](#)

124	cf:firstname	Somchai
123	cf:firstname	Thanachart
	cf:lastname	Numnonda



Lecture: Understanding Map Reduce Processing

Hadoop Ecosystem



Evolution of the Hadoop Platform

The stack is continually evolving and growing!

2006	2007	2008	2009	2010	2011	2012	2013	2014-15
Core Hadoop (HDFS, MapReduce)	Solr Pig	Solr Pig	HBase ZooKeeper	Hive Mahout	Flume Bigtop Oozie MRUnit HCatalog Hue Sqoop Whirr Avro Hive Mahout	Parquet Sentry Spark Tez Impala Kafka Drill Flume Bigtop Oozie MRUnit HCatalog Hue Sqoop Whirr Avro Hive Mahout	Ibis Flink Parquet Sentry Spark Tez Impala Kafka Drill Flume Bigtop Oozie MRUnit HCatalog Hue Sqoop Whirr Avro Hive Mahout	Ibis Flink Parquet Sentry Spark Tez Impala Kafka Drill Flume Bigtop Oozie MRUnit HCatalog Hue Sqoop Whirr Avro Hive Mahout
Core Hadoop	Core Hadoop	Core Hadoop	ZooKeeper	HBase	Sqoop Whirr Avro Hive Mahout	ZooKeeper	ZooKeeper	ZooKeeper
					Solr Pig YARN	Solr Pig YARN	Solr Pig YARN	Solr Pig YARN
						Core Hadoop	Core Hadoop	Core Hadoop

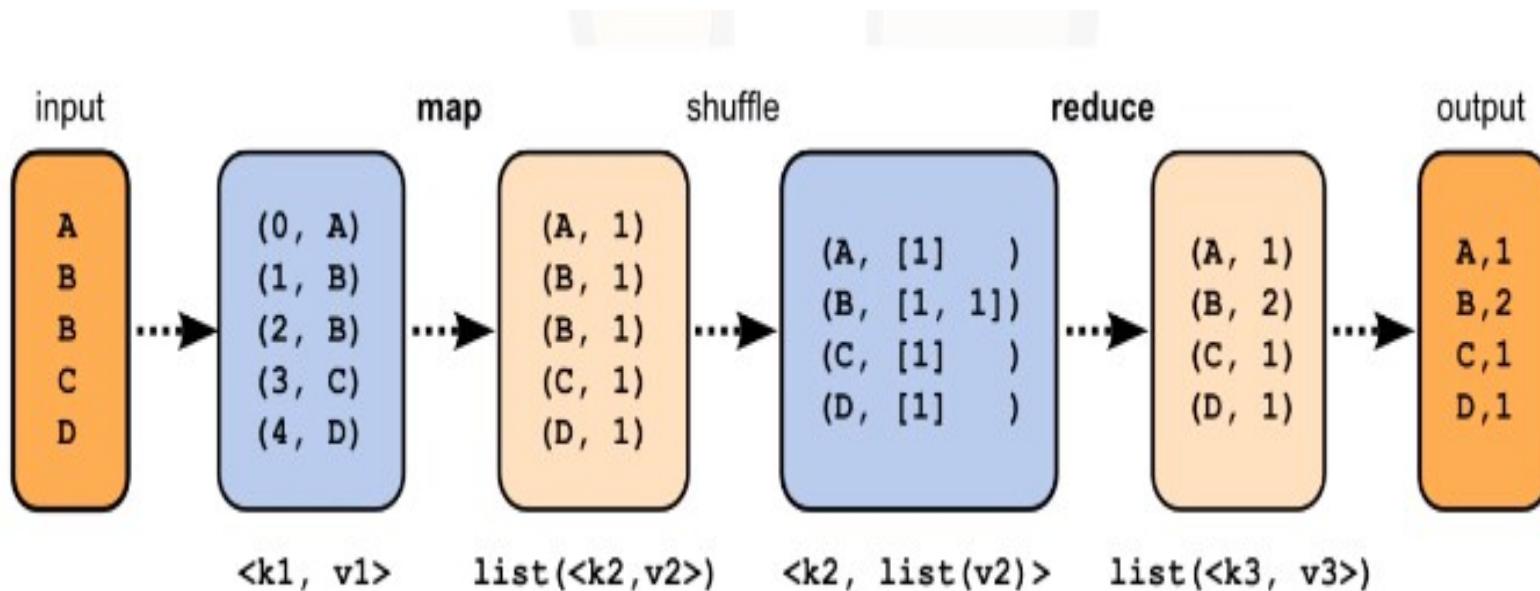
cloudera

Before MapReduce...

- Large scale data processing was difficult!
 - Managing hundreds or thousands of processors
 - Managing parallelization and distribution
 - I/O Scheduling
 - Status and monitoring
 - Fault/crash tolerance
- MapReduce provides all of these, easily!

Source: <http://labs.google.com/papers/mapreduce-osdi04-slides/index-auto-0002.html>

MapReduce Framework



Source: www.bigdatauniversity.com

How Map and Reduce Work Together

- Map returns information
- Reduces accepts information
- Reduce applies a user defined function to reduce the amount of data

Map Abstraction

- Inputs a key/value pair
 - Key is a reference to the input value
 - Value is the data set on which to operate
- Evaluation
 - Function defined by user
 - Applies to every value in value input
 - Might need to parse input
- Produces a new list of key/value pairs
 - Can be different type from input pair

Reduce Abstraction

- Starts with intermediate Key / Value pairs
 - Ends with finalized Key / Value pairs
-
- Starting pairs are sorted by key
 - Iterator supplies the values for a given key to the Reduce function.

Reduce Abstraction

- Typically a function that:
 - Starts with a large number of key/value pairs
 - One key/value for each word in all files being grep'd (including multiple entries for the same word)
 - Ends with very few key/value pairs
 - One key/value for each unique word across all the files with the number of instances summed into this entry
- Broken up so a given worker works with input of the same key.

Why is this approach better?

- Creates an abstraction for dealing with complex overhead
 - The computations are simple, the overhead is messy
- Removing the overhead makes programs much smaller and thus easier to use
 - Less testing is required as well. The MapReduce libraries can be assumed to work properly, so only user code needs to be tested
- Division of labor also handled by the MapReduce libraries, so programmers only need to focus on the actual computation

Hands-On: Writing your own Map Reduce Program

Example MapReduce: WordCount

```
package org.apache.hadoop.examples;

import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.util.GenericOptionsParser;

public class WordCount {

    public static class TokenizerMapper
        extends Mapper<Object, Text, Text, IntWritable>{

        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(Object key, Text value, Context context
                        ) throws IOException, InterruptedException {
            StringTokenizer itr = new StringTokenizer(value.toString());
            while (itr.hasMoreTokens()) {
                word.set(itr.nextToken());
                context.write(word, one);
            }
        }
    }
}
```

```
public static class IntSumReducer
    extends Reducer<Text,IntWritable,Text,IntWritable> {
private IntWritable result = new IntWritable();

    public void reduce(Text key, Iterable<IntWritable> values,
                      Context context
                      ) throws IOException, InterruptedException {
        int sum = 0;
        for (IntWritable val : values) {
            sum += val.get();
        }
        result.set(sum);
        context.write(key, result);
    }
}
```

```
public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    String[] otherArgs = new GenericOptionsParser(conf, args).getRemainingArgs();
    if (otherArgs.length != 2) {
        System.err.println("Usage: wordcount <in> <out>");
        System.exit(2);
    }
    Job job = new Job(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(otherArgs[0]));
    FileOutputFormat.setOutputPath(job, new Path(otherArgs[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
```

Running Map Reduce Program

```
[cloudera@quickstart ~]$ cd workspace/  
[cloudera@quickstart workspace]$ hadoop jar wordcount.jar org.myorg.WordCount input/* output/wordcount_output  
15/02/08 10:30:31 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032  
15/02/08 10:30:32 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032  
15/02/08 10:30:33 WARN mapreduce.JobSubmitter: Hadoop command-line option parsing not performed. Implement the Tool interface with ToolRunner to remedy this.  
15/02/08 10:30:33 INFO mapred.FileInputFormat: Total input paths to process : 1  
15/02/08 10:30:34 INFO mapreduce.JobSubmitter: number of splits:2  
15/02/08 10:30:34 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1423408479621_0009  
15/02/08 10:30:35 INFO impl.YarnClientImpl: Submitted application application_1423408479621_0009  
15/02/08 10:30:35 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_142  
15/02/08 10:30:35 INFO mapreduce.Job: Running job: job_1423408479621_0009  
15/02/08 10:30:52 INFO mapreduce.Job: Job job_1423408479621_0009 running in uber mode : false  
15/02/08 10:30:52 INFO mapreduce.Job: map 0% reduce 0%  
15/02/08 10:31:22 INFO mapreduce.Job: map 58% reduce 0%  
15/02/08 10:31:25 INFO mapreduce.Job: map 100% reduce 0%  
15/02/08 10:31:52 INFO mapreduce.Job: map 100% reduce 100%  
15/02/08 10:31:53 INFO mapreduce.Job: Job job_1423408479621_0009 completed successfully  
15/02/08 10:31:53 INFO mapreduce.Job: Counters: 49
```

```
$cd /root/guest1
```

```
$wget https://dl.dropboxusercontent.com/u/12655380/wordcount.jar
```

```
$hadoop jar wordcount.jar org.myorg.WordCount
```

```
/user/cloudera/input/* /user/cloudera/output/wordcount
```

Reviewing MapReduce Job in Hue

HUE Home Query Editors Data Browsers Workflows Search Security ?

Job Browser

Username Text Succeeded Running Failed Killed

Logs	ID	Name	Application Type	Status	User	Maps	Reduces	Queue	Priority	Duration	Submitted
	1465875170640_0001	wordcount	MAPREDUCE	SUCCEEDED	root	100%	100%	root.root	N/A	21s	06/13/16 21:32:37

Showing 1 to 1 of 1 entries ← Previous 1 Next →

Reviewing MapReduce Job in Hue

HUE Home Query Editors Data Browsers Workflows Search Security

Job Browser

JOB ID	wordcount
	146587517064...
TYPE	MR2
USER	root
STATUS	SUCCEEDED
LOGS	Logs

Attempts Tasks Metadata Counters

Recent Tasks

[View All Tasks »](#)

Logs	Tasks	Type
	task_1465875170640_0001_m_000000	MAP
	task_1465875170640_0001_m_000001	MAP
	task_1465875170640_0001_r_000000	REDUCE

Reviewing MapReduce Output Result

HUE Home Query Editors Data Browsers Workflows Search Security

File Browser

Search for file name Actions Move to trash

Home / user / cloudera / output / wordcount History Trash

Name	Size	User	Group	Permissions	Date
...		cloudera	cloudera	drwxr-xr-x	June 13, 2016 09:32 PM
.		root	cloudera	drwxr-xr-x	June 13, 2016 09:32 PM
_SUCCESS	0 bytes	root	cloudera	-rw-r--r--	June 13, 2016 09:32 PM
part-00000	44 bytes	root	cloudera	-rw-r--r--	June 13, 2016 09:32 PM

Reviewing MapReduce Output Result

HUE Home Query Editors Data Browsers Workflows Search Security

File Browser

ACTIONS

View as binary Edit file Download View file location Refresh

Home Page 1 of 1

/ user / cloudera / output / wordcount / part-00000

a	205807
e	315232
i	174282
o	192879
u	65433

INFO



Lecture

Understanding Oozie

Introduction

Workflow scheduler for Hadoop

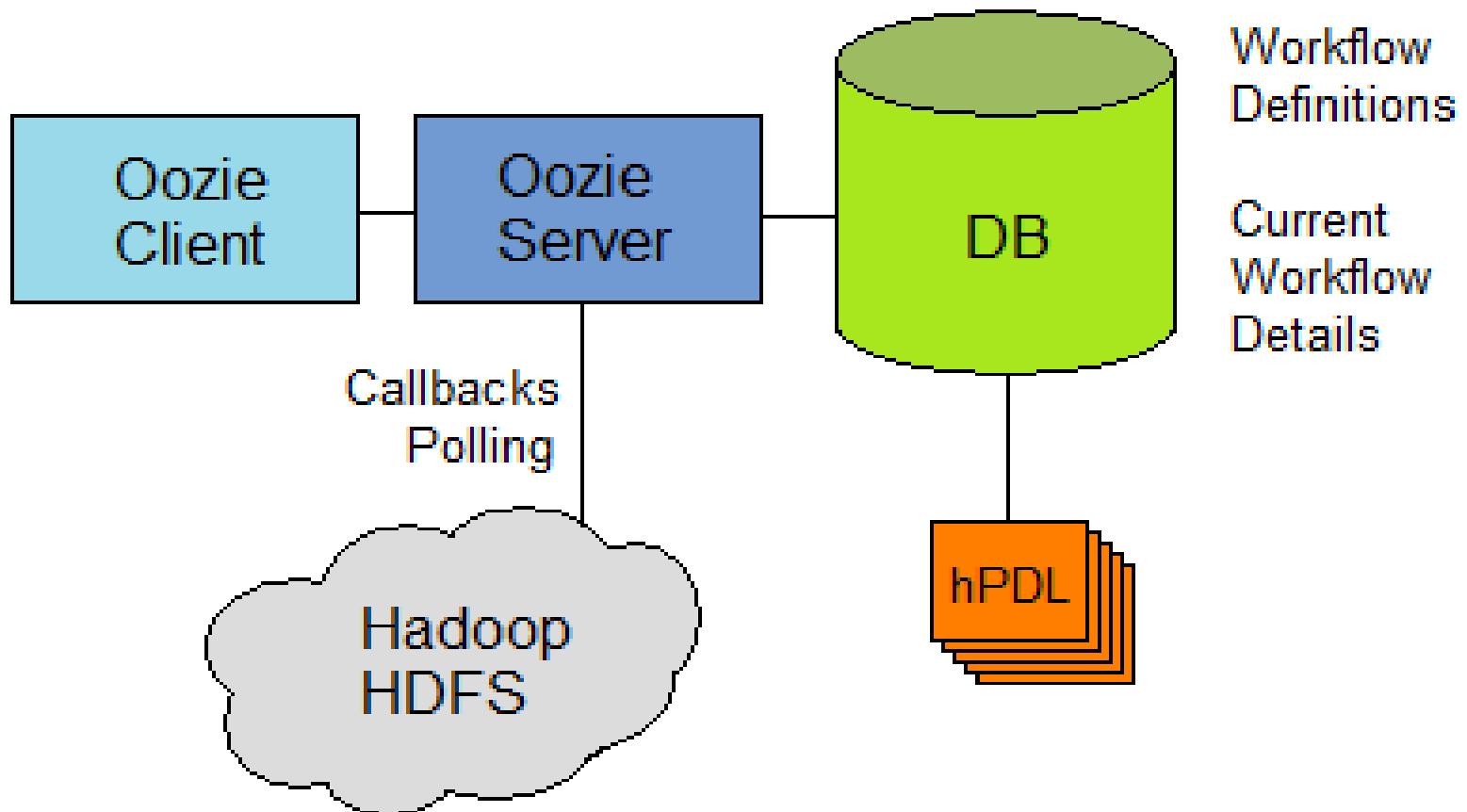


Oozie is a workflow scheduler system to manage Apache Hadoop jobs. Oozie is integrated with the rest of the Hadoop stack supporting several types of Hadoop jobs out of the box (such as Java map-reduce, Streaming map-reduce, Pig, Hive, Sqoop and Distcp) as well as system specific jobs (such as Java programs and shell scripts).

What is Oozie?

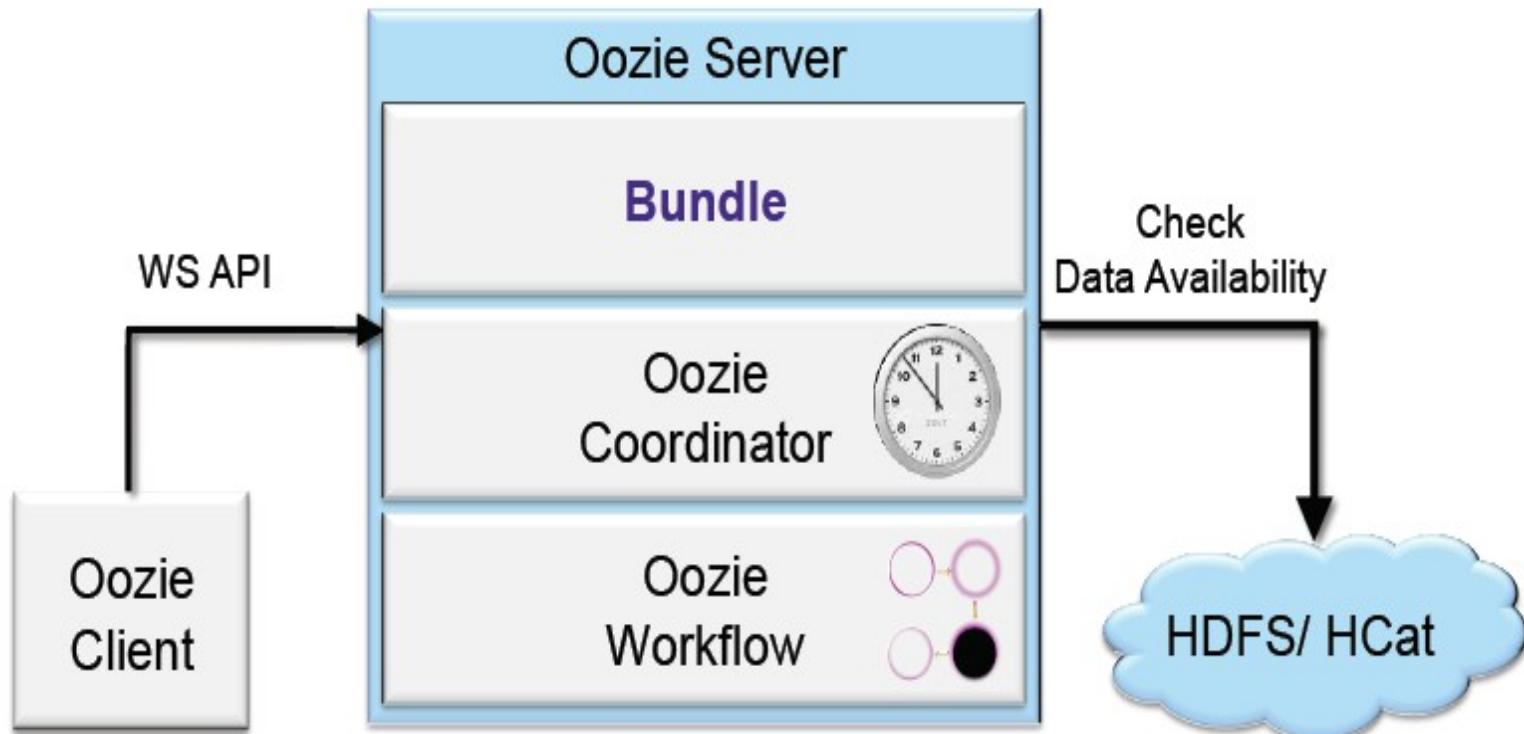
- Work flow scheduler for Hadoop
- Manages Hadoop Jobs
- Integrated with many Hadoop apps i.e. Pig, Hive
- Scaleable
- Schedule jobs
- A work flow is a collection of actions.
- A work flow is
 - Arranged as a DAG (direct acyclic graph)
 - Graph stored as hPDL (XML process definition)

Oozie Architecture



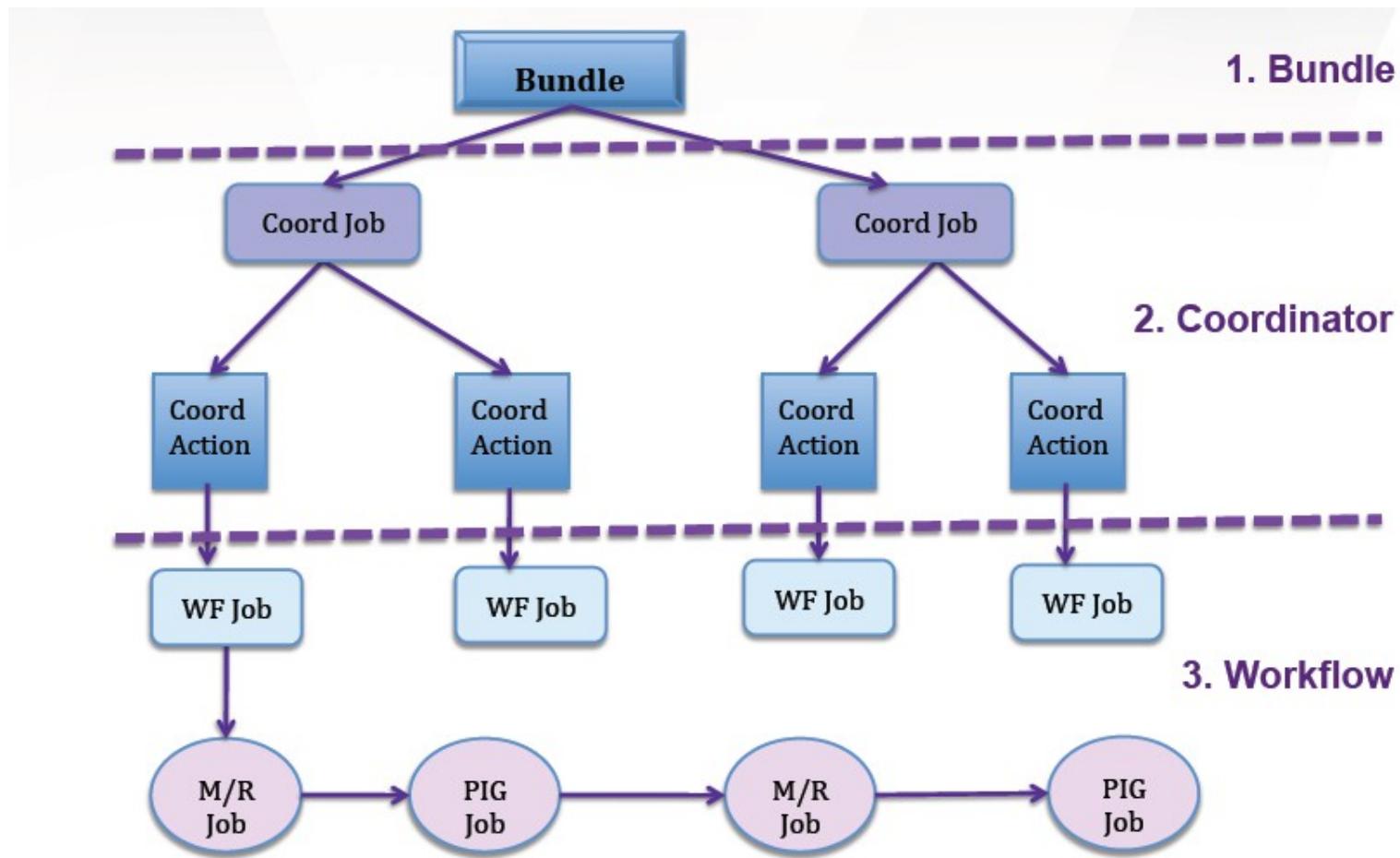
Source: info@semtech-solutions.co.nz

Oozie Server



Source: Oozie – Now and Beyond, Yahoo, 2013

Layer of Abstraction in Oozie



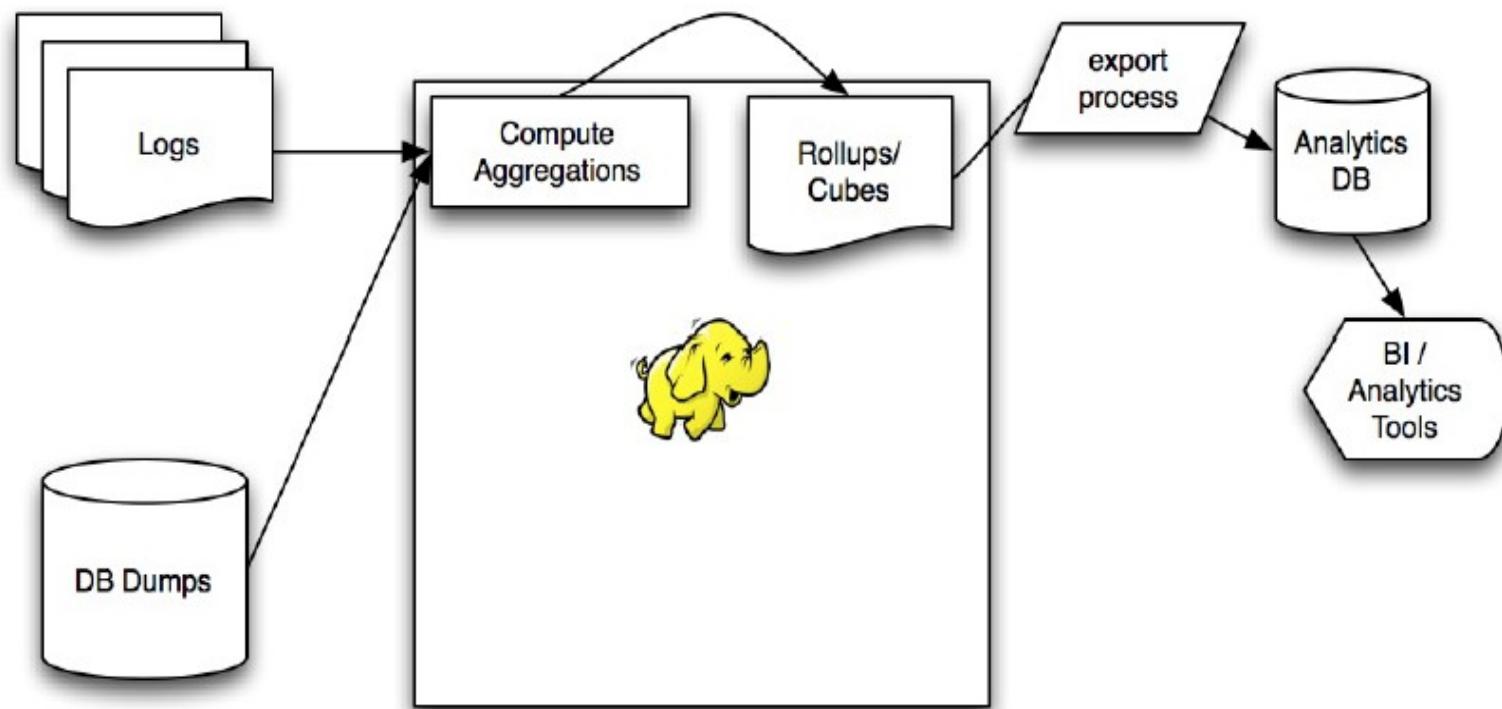
Source: Oozie – Now and Beyond, Yahoo, 2013

Workflow Example: Data Analytics

- Logs => fact table(s)
- Database backup => Dimension tables
- Complete rollups/cubes
- Load data into a low-latency storage (e.g. Hbase, HDFS)
- Dashboard & BI tools

Source: Workflow Engines for Hadoop, Joe Crobak, 2013

Workflow Example: Data Analytics



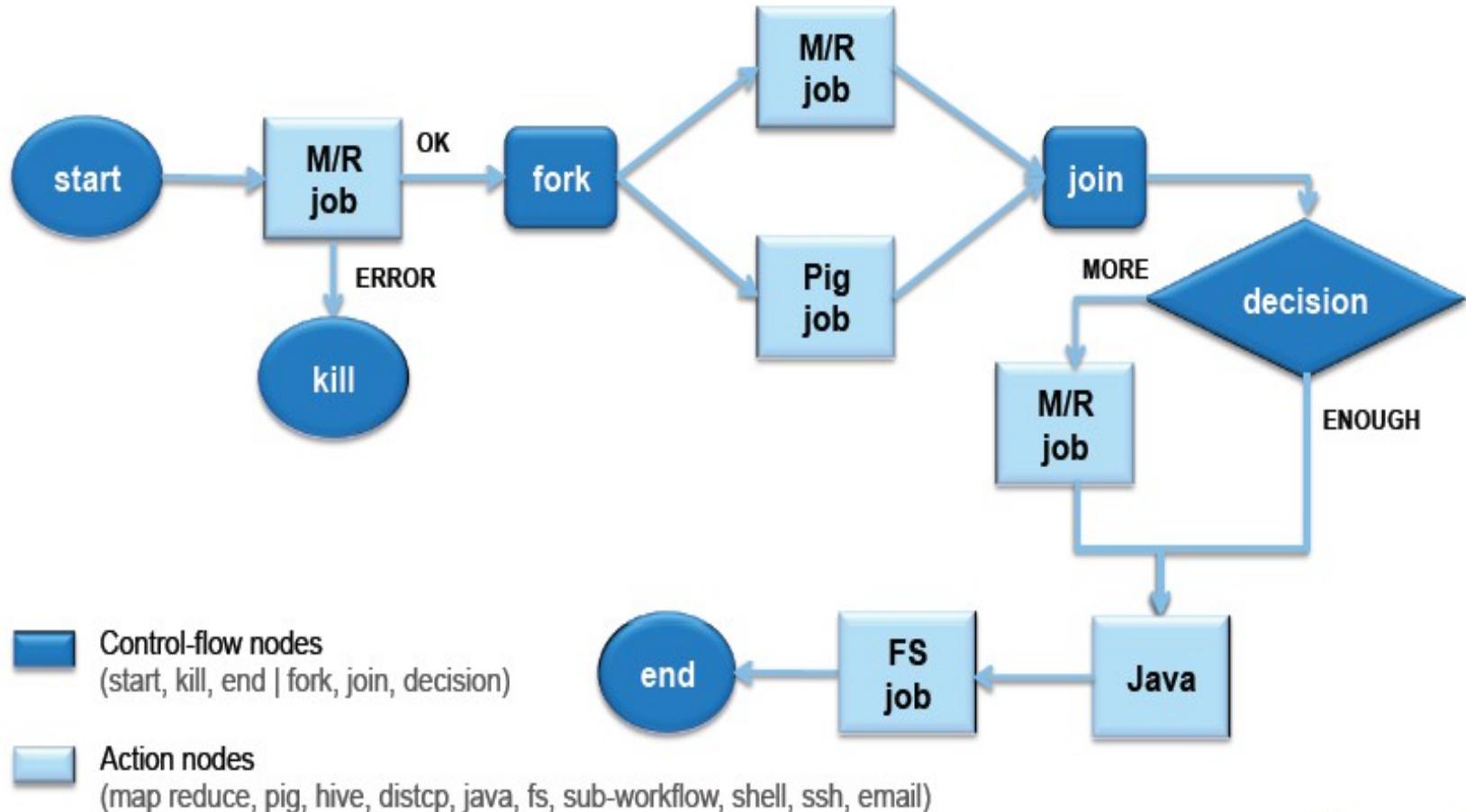
Source: Workflow Engines for Hadoop, Joe Crobak, 2013

Workflow Example: Data Analytics

- What happens if there is a failure?
 - Rebuild the failed day
 - .. and any downstream datasets
- With Hadoop Workflow
 - Possible OK to skip a day
 - Workflow tends to be self-contained, so you do not need to run downstream.
 - Sanity check your data before pushing to production.

Source: Workflow Engines for Hadoop, Joe Crobak, 2013

Oozie Workflow



Source: Oozie – Now and Beyond, Yahoo, 2013

Oozie Use Cases

- Time Triggers
 - Execute your workflow every 15 minutes
- Time and Data Triggers
 - Materialize your workflow every hour, but only run them when the input data is ready (that is loaded to the grid every hour)
- Rolling Window
 - Access 15 minute datasets and roll them up into hourly datasets

Source: Oozie – Now and Beyond, Yahoo, 2013

Hands-On: Running Map Reduce using Oozie workflow

Using Hue: select WorkFlows >> Editors >> Workflows

The screenshot shows the Hue interface for managing Oozie Workflows. The top navigation bar includes links for HUE, Home, Query Editors, Data Browsers, Workflows (selected), Search, File Browser, Job Browser, guest, and various help and settings icons. Below the navigation is a secondary header with tabs for Oozie Editor, Workflows (selected), Coordinators, and Bundles.

The main content area is titled "Workflow Editor". It features a search bar, a "Submit" button, and three action buttons: "Share", "Copy", and "Delete". A "Create" button is located in the top right corner of the search area.

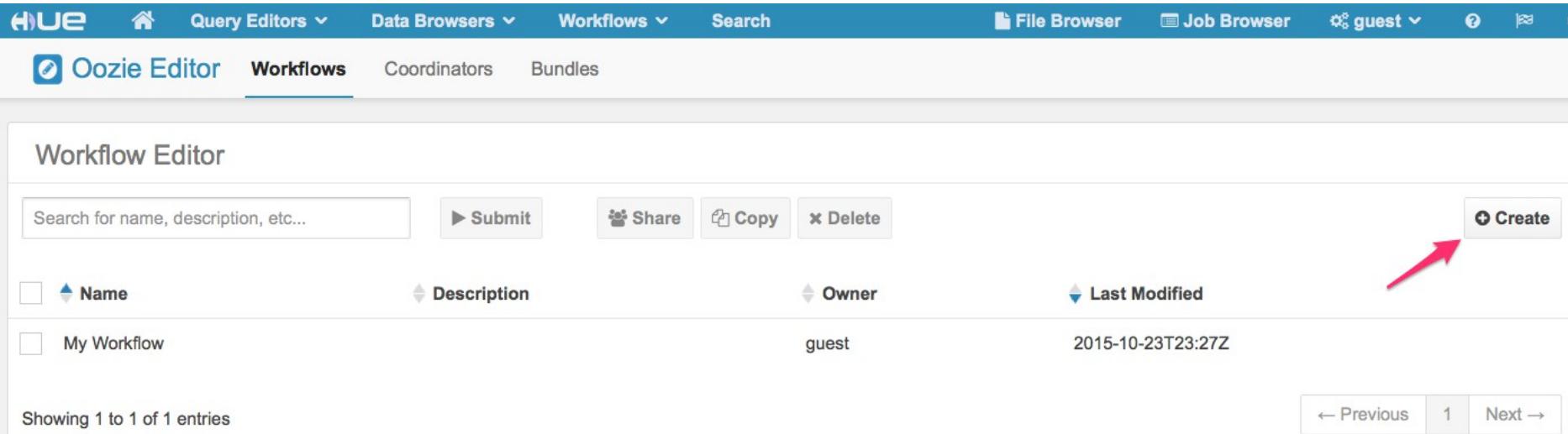
A table displays the workflow details:

Name	Description	Owner	Last Modified
My Workflow		guest	2015-10-23T23:27Z

At the bottom left, it says "Showing 1 to 1 of 1 entries". On the bottom right, there are navigation buttons for "← Previous", "1", and "Next →".

Create a new workflow

- Click Create button; the following screen will be displayed
- Name the workflow as WordCountWorkflow



The screenshot shows the Hue Oozie Editor interface. At the top, there is a navigation bar with links for HUE, Home, Query Editors, Data Browsers, Workflows (which is currently selected), Search, File Browser, Job Browser, guest, and other system icons.

The main area is titled "Workflow Editor". It features a search bar, a "Submit" button, and three action buttons: "Share", "Copy", and "Delete". On the far right of the header, there is a "Create" button with a plus sign, which is highlighted by a red arrow.

The main content area displays a table of workflow entries:

Name	Description	Owner	Last Modified
My Workflow		guest	2015-10-23T23:27Z

At the bottom left, it says "Showing 1 to 1 of 1 entries". At the bottom right, there are navigation links for "Previous", "1", and "Next".

HUE  Query Editors ▾ Data Browsers ▾ Workflows ▾ Search  File Browser  Job Browser  guest ▾ ?   

Oozie Editor **Workflows** Coordinators Bundles Unsaved 

ACTIONS              

WordCountWorkflow

Add a description...

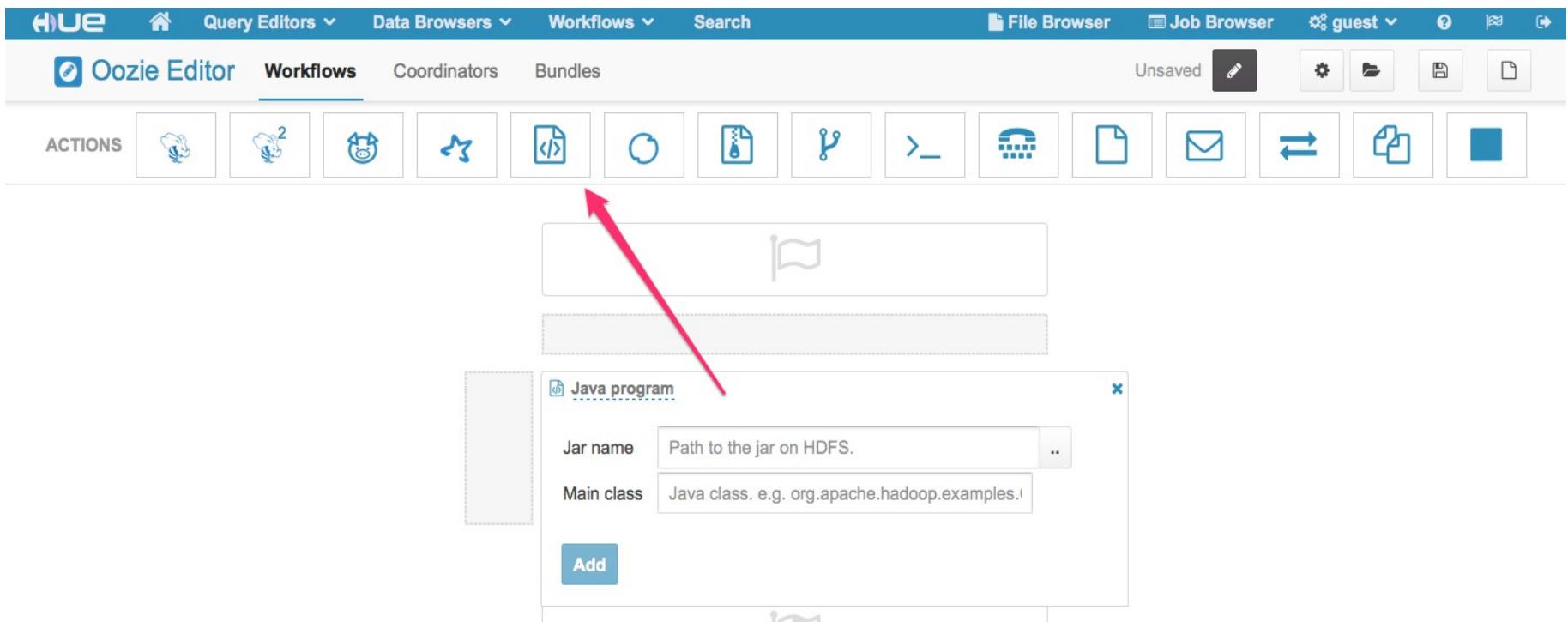


Drop your action here



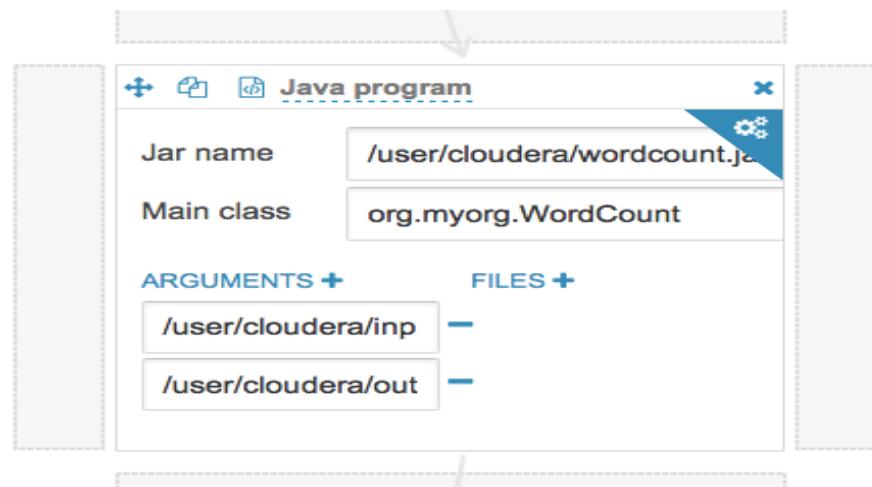
Select a Java job for the workflow

- From the Oozie editor, drag **Java Program** and drop between start and end



Edit the Java Job

- Assign the following value
 - Jar name: wordcount.jar (select ... choose upload from local machine)
 - Main Class: org.myorg.WordCount
 - Arguments: /user/cloudera/input/*
 - /user/cloudera/output/wordcount



Submit the workflow

- Click Done, follow by Save
- Then click submit

The screenshot shows the Hue Oozie Editor interface. At the top, there's a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, Search, File Browser, Job Browser, guest, and various system icons. Below the navigation bar, the main area has tabs for Oozie Editor, Workflows (which is selected), Coordinators, and Bundles. A toolbar below the tabs contains several icons for different actions like creating new components, deleting, and running jobs.

The central part of the screen is a workflow editor. It displays a sequence of steps: a start node (represented by a flag icon) followed by a Java program step. The Java program step is expanded, showing its configuration details:

- Java program**: The configuration fields are:
 - Jar name: /user/guest/wordcount.jar
 - Main class: org.myorg.WordCount
 - ARGUMENTS +:
 - input/*
 - FILES +:
 - output/wordcount

At the bottom right of the Java program configuration, there are two large red arrows pointing upwards towards the 'Submit' and 'Save' buttons in the top toolbar. The 'Submit' button is located next to the play icon, and the 'Save' button is located next to the document icon.

Introduction

A Petabyte Scale Data Warehouse Using Hadoop



Hive is developed by Facebook, designed to enable easy data summarization, ad-hoc querying and analysis of large volumes of data. It provides a simple query language called Hive QL, which is based on SQL

What Hive is NOT

Hive is not designed for online transaction processing and does not offer real-time queries and row level updates. It is best used for batch jobs over large sets of immutable data (like web logs, etc.).

Sample HiveQL

The Query compiler uses the information stored in the metastore to convert SQL queries into a sequence of map/reduce jobs, e.g. the following query

```
SELECT * FROM t where t.c = 'xyz'
```

```
SELECT t1.c2 FROM t1 JOIN t2 ON (t1.c1 = t2.c1)
```

```
SELECT t1.c1, count(1) from t1 group by t1.c1
```

Sample HiveQL

The Query compiler uses the information stored in the metastore to convert SQL queries into a sequence of map/reduce jobs, e.g. the following query

```
SELECT * FROM t where t.c = 'xyz'
```

```
SELECT t1.c2 FROM t1 JOIN t2 ON (t1.c1 = t2.c1)
```

```
SELECT t1.c1, count(1) from t1 group by t1.c1
```

System Architecture and Components

Metastore: To store the meta data.

Query compiler and execution engine: To convert SQL queries to a sequence of map/reduce jobs that are then executed on Hadoop.

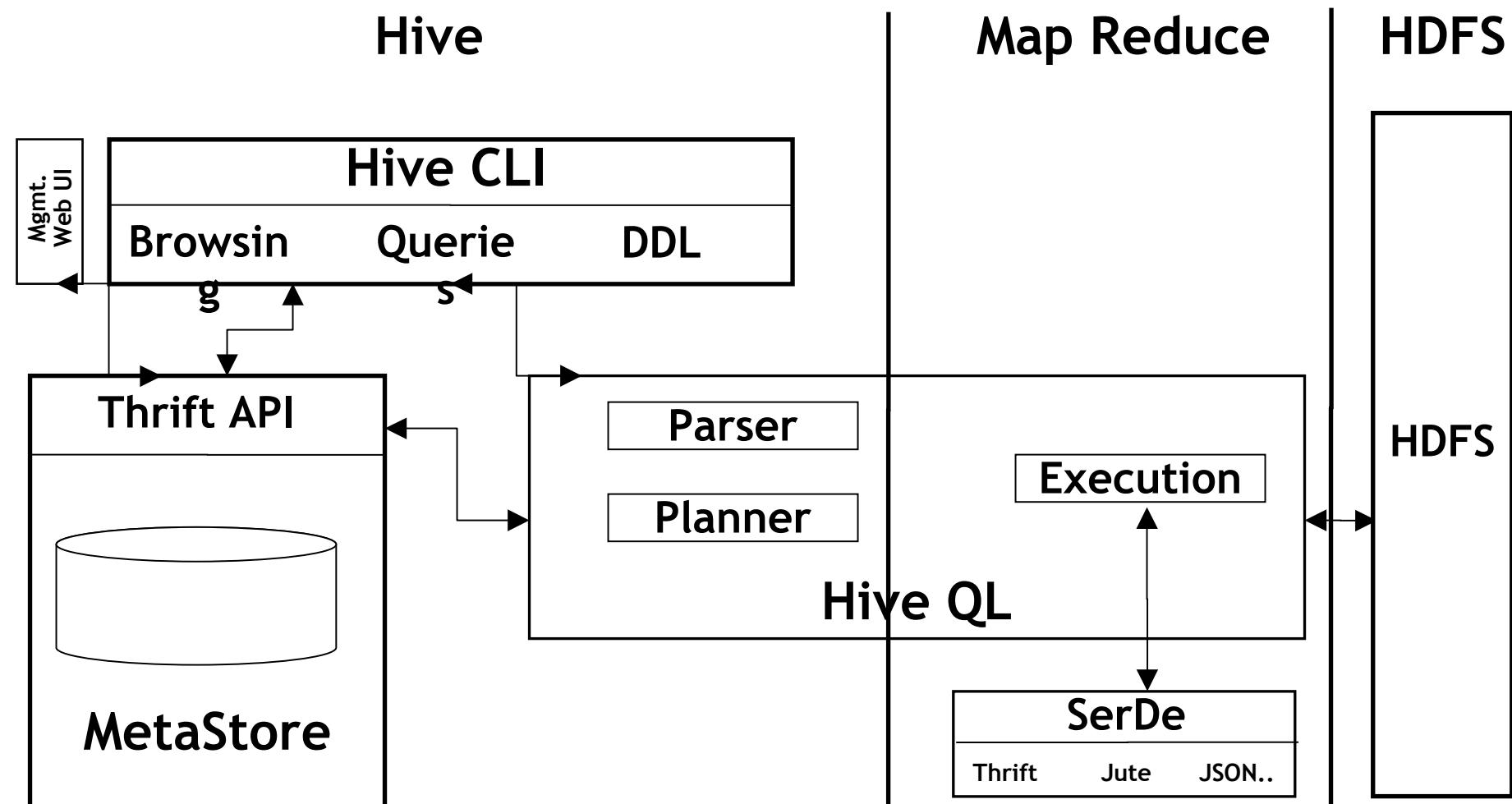
SerDe and ObjectInspectors: Programmable interfaces and implementations of common data formats and types.

A SerDe is a combination of a Serializer and a Deserializer (hence, Ser-De). The Deserializer interface takes a string or binary representation of a record, and translates it into a Java object that Hive can manipulate. The Serializer, however, will take a Java object that Hive has been working with, and turn it into something that Hive can write to HDFS or another supported system.

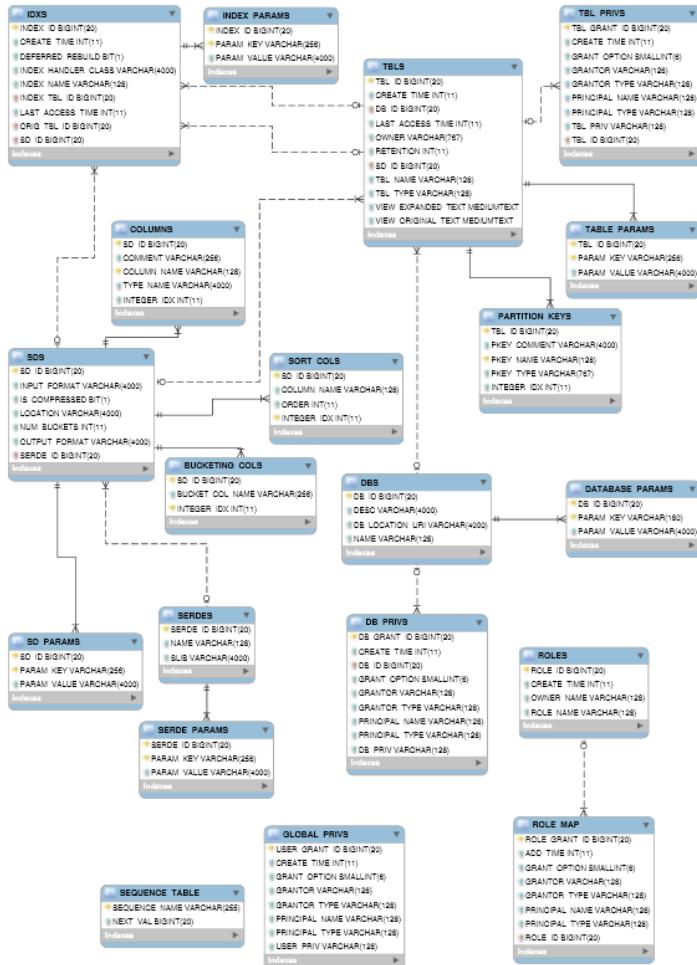
UDF and UDAF: Programmable interfaces and implementations for user defined functions (scalar and aggregate functions).

Clients: Command line client similar to Mysql command line.

Architecture Overview



Hive Metastore



Hive Metastore is a repository to keep all Hive metadata; Tables and Partitions definition.

By default, Hive will store its metadata in Derby DB

Hive Built in Functions

Return Type	Function Name (Signature)	Description
BIGINT	round(double a)	returns the rounded BIGINT value of the double
BIGINT	floor(double a)	returns the maximum BIGINT value that is equal or less than the double
BIGINT	ceil(double a)	returns the minimum BIGINT value that is equal or greater than the double
double	rand(), rand(int seed)	returns a random number (that changes from row to row). Specifying the seed will make sure the generated random number sequence is deterministic.
string	concat(string A, string B,...)	returns the string resulting from concatenating B after A. For example, concat('foo', 'bar') results in 'foobar'. This function accepts arbitrary number of arguments and return the concatenation of all of them.
string	substr(string A, int start)	returns the substring of A starting from start position till the end of string A. For example, substr('foobar', 4) results in 'bar'
string	substr(string A, int start, int length)	returns the substring of A starting from start position with the given length e.g. substr('foobar', 4, 2) results in 'ba'
string	upper(string A)	returns the string resulting from converting all characters of A to upper case e.g. upper('fOoBaR') results in 'FOOBAR'
string	ucase(string A)	Same as upper
string	lower(string A)	returns the string resulting from converting all characters of B to lower case e.g. lower('fOoBaR') results in 'foobar'
string	lcase(string A)	Same as lower
string	trim(string A)	returns the string resulting from trimming spaces from both ends of A e.g. trim(' foobar ') results in 'foobar'
string	ltrim(string A)	returns the string resulting from trimming spaces from the beginning(left hand side) of A. For example, ltrim(' foobar ') results in 'foobar '
string	rtrim(string A)	returns the string resulting from trimming spaces from the end(right hand side) of A. For example, rtrim(' foobar ') results in ' foobar'
string	regexp_replace(string A, string B, string C)	returns the string resulting from replacing all substrings in B that match the Java regular expression syntax(See Java regular expressions syntax) with C. For example, regexp_replace('foobar', 'oo ar',) returns 'fb'
string	from_unixtime(int unixtime)	convert the number of seconds from unix epoch (1970-01-01 00:00:00 UTC) to a string representing the timestamp of that moment in the current system time zone in the format of "1970-01-01 00:00:00"
string	to_date(string timestamp)	Return the date part of a timestamp string: to_date("1970-01-01 00:00:00") = "1970-01-01"
int	year(string date)	Return the year part of a date or a timestamp string: year("1970-01-01 00:00:00") = 1970, year("1970-01-01") = 1970
int	month(string date)	Return the month part of a date or a timestamp string: month("1970-11-01 00:00:00") = 11, month("1970-11-01") = 11
int	day(string date)	Return the day part of a date or a timestamp string: day("1970-11-01 00:00:00") = 1, day("1970-11-01") = 1
string	get_json_object(string json_string, string path)	Extract json object from a json string based on json path specified, and return json string of the extracted json object. It will return null if the input json string is invalid

Hive Aggregate Functions

Return Type	Aggregation Function Name (Signature)	Description
BIGINT	count(*), count(expr), count(DISTINCT expr[, expr_.])	count(*) - Returns the total number of retrieved rows, including rows containing NULL values; count(expr) - Returns the number of rows for which the supplied expression is non- NULL; count(DISTINCT expr[, expr]) - Returns the number of rows for which the supplied expression(s) are unique and non-NULL.
DOUBLE	sum(col), sum(DISTINCT col)	returns the sum of the elements in the group or the sum of the distinct values of the column in the group
DOUBLE	avg(col), avg(DISTINCT col)	returns the average of the elements in the group or the average of the distinct values of the column in the group
DOUBLE	min(col)	returns the minimum value of the column in the group
DOUBLE	max(col)	returns the maximum value of the column in the group

Running Hive

Hive Shell

Interactive

hive

Script

hive -f myscript

Inline

*hive -e 'SELECT * FROM mytable'*

Hive Commands

Command Line

Function	Hive
Run query	hive -e 'select a.col from tab1 a'
Run query silent mode	hive -S -e 'select a.col from tab1 a'
Set hive config variables	hive -e 'select a.col from tab1 a' -hiveconf hive.root.logger=DEBUG,console
Use initialization script	hive -i initialize.sql
Run non-interactive script	hive -f script.sql

Hive Shell

Function	Hive
Run script inside shell	source file_name
Run ls (dfs) commands	dfs -ls /user
Run ls (bash command) from shell	!ls
Set configuration variables	set mapred.reduce.tasks=32
TAB auto completion	set hive.<TAB>
Show all variables starting with hive	set
Revert all variables	reset
Add jar to distributed cache	add jar jar_path
Show all jars in distributed cache	list jars
Delete jar from distributed cache	delete jar jar_name

Hive Tables

- Managed- CREATE TABLE
 - LOAD- File moved into Hive's data warehouse directory
 - DROP- Both data and metadata are deleted.
- External- CREATE EXTERNAL TABLE
 - LOAD- No file moved
 - DROP- Only metadata deleted
 - Use when sharing data between Hive and Hadoop applications or you want to use multiple schema on the same data

Hive External Table

- `CREATE EXTERNAL TABLE external_Table (dummy STRING)`
- `LOCATION '/user/notroot/external_table';`

Dropping External Table using Hive:-

Hive will delete metadata from metastore

Hive will NOT delete the HDFS file

You need to manually delete the HDFS file

Java JDBC for Hive

```
import java.sql.SQLException;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import java.sql.DriverManager;

public class HiveJdbcClient {
    private static String driverName = "org.apache.hadoop.hive.jdbc.HiveDriver";

    public static void main(String[] args) throws SQLException {
        try {
            Class.forName(driverName);
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
            System.exit(1);
        }
        Connection con = DriverManager.getConnection("jdbc:hive://localhost:10000/default", "", "");
        Statement stmt = con.createStatement();
        String tableName = "testHiveDriverTable";
        stmt.executeQuery("drop table " + tableName);
        ResultSet res = stmt.executeQuery("create table " + tableName + " (key int, value string)");
        // show tables
        String sql = "show tables '" + tableName + "'";
        System.out.println("Running: " + sql);
        res = stmt.executeQuery(sql);
        if (res.next()) {
            System.out.println(res.getString(1));
        }
        // describe table
        sql = "describe " + tableName;
        System.out.println("Running: " + sql);
        res = stmt.executeQuery(sql);
        while (res.next()) {
            System.out.println(res.getString(1) + "\t" + res.getString(2));
        }
    }
}
```

Java JDBC for Hive

```
import java.sql.SQLException;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import java.sql.DriverManager;

public class HiveJdbcClient {
    private static String driverName = "org.apache.hadoop.hive.jdbc.HiveDriver";

    public static void main(String[] args) throws SQLException {
        try {
            Class.forName(driverName);
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
            System.exit(1);
        }
        Connection con = DriverManager.getConnection("jdbc:hive://localhost:10000/default", "", "");
        Statement stmt = con.createStatement();
        String tableName = "testHiveDriverTable";
        stmt.executeQuery("drop table " + tableName);
        ResultSet res = stmt.executeQuery("create table " + tableName + " (key int, value string)");
        // show tables
        String sql = "show tables '" + tableName + "'";
        System.out.println("Running: " + sql);
        res = stmt.executeQuery(sql);
        if (res.next()) {
            System.out.println(res.getString(1));
        }
        // describe table
        sql = "describe " + tableName;
        System.out.println("Running: " + sql);
        res = stmt.executeQuery(sql);
        while (res.next()) {
            System.out.println(res.getString(1) + "\t" + res.getString(2));
        }
    }
}
```

HiveQL and MySQL Comparison

Metadata

Function	MySQL	HiveQL
Selecting a database	USE database;	USE database;
Listing databases	SHOW DATABASES;	SHOW DATABASES;
Listing tables in a database	SHOW TABLES;	SHOW TABLES;
Describing the format of a table	DESCRIBE table;	DESCRIBE (FORMATTED EXTENDED) table;
Creating a database	CREATE DATABASE db_name;	CREATE DATABASE db_name;
Dropping a database	DROP DATABASE db_name;	DROP DATABASE db_name (CASCADE);

HiveQL and MySQL Query Comparison

Query

Function	MySQL	HiveQL
Retrieving information	<code>SELECT from_columns FROM table WHERE conditions;</code>	<code>SELECT from_columns FROM table WHERE conditions;</code>
All values	<code>SELECT * FROM table;</code>	<code>SELECT * FROM table;</code>
Some values	<code>SELECT * FROM table WHERE rec_name = "value";</code>	<code>SELECT * FROM table WHERE rec_name = "value";</code>
Multiple criteria	<code>SELECT * FROM table WHERE rec1="value1" AND rec2="value2";</code>	<code>SELECT * FROM TABLE WHERE rec1 = "value1" AND rec2 = "value2";</code>
Selecting specific columns	<code>SELECT column_name FROM table;</code>	<code>SELECT column_name FROM table;</code>
Retrieving unique output records	<code>SELECT DISTINCT column_name FROM table;</code>	<code>SELECT DISTINCT column_name FROM table;</code>
Sorting	<code>SELECT col1, col2 FROM table ORDER BY col2;</code>	<code>SELECT col1, col2 FROM table ORDER BY col2;</code>
Sorting backward	<code>SELECT col1, col2 FROM table ORDER BY col2 DESC;</code>	<code>SELECT col1, col2 FROM table ORDER BY col2 DESC;</code>
Counting rows	<code>SELECT COUNT(*) FROM table;</code>	<code>SELECT COUNT(*) FROM table;</code>
Grouping with counting	<code>SELECT owner, COUNT(*) FROM table GROUP BY owner;</code>	<code>SELECT owner, COUNT(*) FROM table GROUP BY owner;</code>
Maximum value	<code>SELECT MAX(col_name) AS label FROM table;</code>	<code>SELECT MAX(col_name) AS label FROM table;</code>
Selecting from multiple tables (Join same table using alias w/"AS")	<code>SELECT pet.name, comment FROM pet, event WHERE pet.name = event.name;</code>	<code>SELECT pet.name, comment FROM pet JOIN event ON (pet.name = event.name);</code>

Hands-On: Loading Data using Hive

Start Hive

```
[root@quickstart guest1]# hive
2016-06-14 07:48:56,273 WARN  [main] mapreduce.TableMapReduceUtil: The
hbase-prefix-tree module jar containing PrefixTreeCodec is not present.
Continuing without it.

Logging initialized using configuration in file:/etc/hive/conf.dist/hiv
e-log4j.properties
WARNING: Hive CLI is deprecated and migration to Beeline is recommended
.
hive> █
```

Quit from Hive

```
hive> quit;
```

Create Hive Table

```
hive> CREATE TABLE test_tbl(id INT, country STRING) ROW FORMAT DELIMITED
      FIELDS TERMINATED BY ',' STORED AS TEXTFILE;
OK
Time taken: 0.886 seconds
hive> show tables;
OK
test_tbl
Time taken: 0.125 seconds, Fetched: 1 row(s)
hive> describe test_tbl;
OK
id          int
country      string
Time taken: 0.115 seconds, Fetched: 2 row(s)
hive> █
```

See also: <https://cwiki.apache.org/Hive/languagemanual-ddl.html>

Reviewing Hive Table in HDFS

The screenshot shows the Hue File Browser interface. The top navigation bar includes links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation is a search bar labeled "Search for file name" and a "Actions" dropdown menu. The main area displays a list of files and directories under the path "/user/hive/warehouse". The list includes three entries: a folder named ".", a folder named "test_tbl", and a file named "t". The columns in the table are Name, Size, User, Group, Permissions, and Date. The "Name" column contains icons for each item: a folder for ".", a folder for "test_tbl", and a file for "t". The "User" column shows "hive" for the first two and "root" for the third. The "Group" column shows "supergroup" for all. The "Permissions" column shows "drwxrwxrwx" for all. The "Date" column shows "April 05, 2016 07:27 PM" for the first two and "June 14, 2016 12:50 AM" for the third.

Name	Size	User	Group	Permissions	Date
..		hive	supergroup	drwxrwxrwx	April 05, 2016 07:27 PM
test_tbl		hive	supergroup	drwxrwxrwx	June 14, 2016 12:50 AM
t		root	supergroup	drwxrwxrwx	June 14, 2016 12:50 AM

Alter and Drop Hive Table

```
Hive > alter table test_tbl add columns (remarks STRING);
```

```
hive > describe test_tbl;
```

```
OK
```

```
id int
```

```
country string
```

```
remarks string
```

```
Time taken: 0.077 seconds
```

```
hive > drop table test_tbl;
```

```
OK
```

```
Time taken: 0.9 seconds
```

See also: <https://cwiki.apache.org/Hive/adminmanual-metastoreadmin.html>

Preparing Large Dataset

<http://grouplens.org/datasets/movielens/>



[about](#) [datasets](#) [publications](#) [blog](#)

MovieLens

GroupLens Research has collected and made available rating data sets from the MovieLens web site (<http://movielens.org>). The data sets were collected over various periods of time, depending on the size of the set. Before using these data sets, please review their README files for the usage licenses and other details.

Help our research lab: Please [take a short survey](#) about the MovieLens datasets

MovieLens 100k

100,000 ratings from 1000 users on 1700 movies.

- [README.txt](#)
- [ml-100k.zip](#)
- [Index of unzipped files](#)

MovieLens 1M

1 million ratings from 6000 users on 4000 movies.

- [README.txt](#)

Datasets

[MovieLens](#)

[HetRec 2011](#)

[WikiLens](#)

[Book-Crossing](#)

[Jester](#)

[EachMovie](#)

MovieLen Dataset

1) Type command > `wget`

`http://files.grouplens.org/datasets/movielens/ml-100k.zip`

2) Type command > `yum install unzip`

3) Type command > `unzip ml-100k.zip`

4) Type command > `more ml-100k/u.user`

```
[root@quickstart guest1]# more ml-100k/u.user
1|24|M|technician|85711
2|53|F|other|94043
3|23|M|writer|32067
4|24|M|technician|43537
5|33|F|other|15213
6|42|M|executive|98101
7|57|M|administrator|91344
8|36|M|administrator|05201
9|29|M|student|01002
10|53|M|lawyer|90703
11|39|F|other|30329
```

Moving dataset to HDFS

- 1) Type command > `cd ml-100k`
- 2) Type command > `hadoop fs -mkdir /user/cloudera/movielens`
- 3) Type command > `hadoop fs -put u.user /user/cloudera/movielens`
- 4) Type command > `hadoop fs -ls /user/cloudera/movielens`

```
[root@quickstart ml-100k]# hadoop fs -ls /user/cloudera/movielens
Found 1 items
-rw-r--r--  1 root cloudera      22628 2016-06-14 08:04 /user/cloudera/
movielens/u.user
[root@quickstart ml-100k]#
```

CREATE & SELECT Table

```
hive> CREATE EXTERNAL TABLE users (userid INT, age INT,  
> gender STRING, occupation STRING, zipcode STRING) ROW FORMAT  
> DELIMITED FIELDS TERMINATED BY '|' STORED AS TEXTFILE  
> LOCATION '/user/cloudera/movielens';
```

OK

Time taken: 0.646 seconds

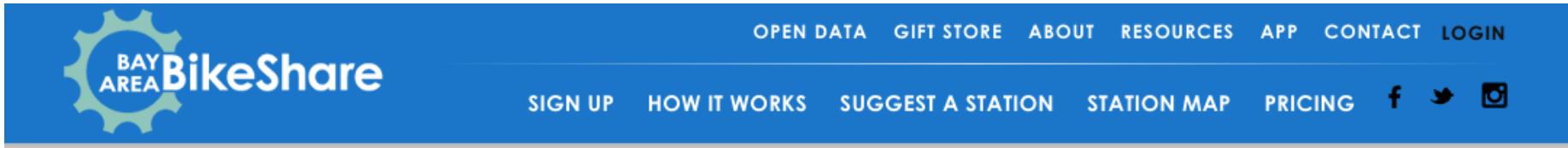
```
hive> SELECT * FROM users;
```

OK

1	24	M	technician	85711
2	53	F	other	94043
3	23	M	writer	32067
4	24	M	technician	43537
5	33	F	other	15213
6	42	M	executive	98101
7	57	M	administrator	91344
8	36	M	administrator	05201

Bay Area Bike Share (BABS)

<http://www.bayareabikeshare.com/open-data>



OPEN DATA

Here you'll find Bay Area Bike Share's trip data for public use. So whether you're a designer, developer, or just plain curious, feel free to download it and bring it to life!

THE DATA

Each trip is anonymized and includes:

- Bike number
- Trip start day and time
- Trip end day and time

YEAR 1 DATA

(August 2013 - August 2014)

YEAR 2 DATA

(September 2014 - August 2015)

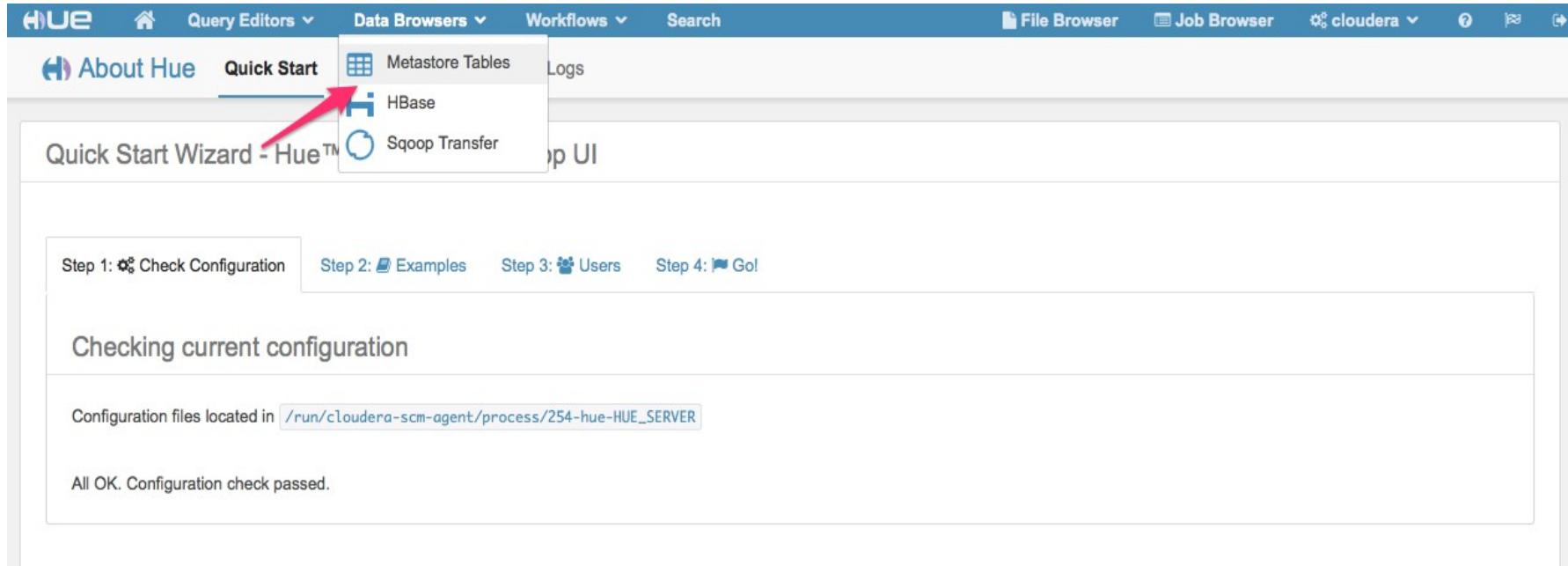
Preparing a bike data

```
$wget https://s3.amazonaws.com/babs-open-data/  
babs_open_data_year_1.zip  
$unzip babs_open_data_year_1.zip  
$cd 201402_babs_open_data/  
$hadoop fs -put 201402_trip_data.csv  
/user/cloudera  
$ hadoop fs -ls /user/cloudera
```

```
[root@quickstart 201402_babs_open_data]# hadoop fs -ls /user/cloudera  
Found 4 items  
-rw-r--r-- 1 root cloudera 17219022 2016-06-14 08:13 /user/cloudera/201402_t  
rip_data.csv  
drwxr-xr-x - cloudera cloudera 0 2016-06-14 04:29 /user/cloudera/input  
drwxr-xr-x - root cloudera 0 2016-06-14 08:04 /user/cloudera/movielen  
s  
drwxr-xr-x - cloudera cloudera 0 2016-06-14 04:32 /user/cloudera/output  
[root@quickstart 201402_babs_open_data]#
```

Importing CSV Data with the Metastore App

The BABS data set contains 4 CSVs that contain data for stations, trips, rebalancing (availability), and weather. We will import **trips** dataset using Metastore Tables



The screenshot shows the Hue web interface with the 'Quick Start' wizard open. The top navigation bar includes links for 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and 'cloudera'. The sidebar on the left has links for 'About Hue', 'Quick Start' (which is currently selected), 'HBase', and 'Scoop Transfer'. A red arrow points to the 'Metastore Tables' link in the 'Quick Start' dropdown menu. Below the sidebar, the 'Quick Start Wizard - Hue™' section displays four steps: 'Step 1: Check Configuration', 'Step 2: Examples', 'Step 3: Users', and 'Step 4: Go!'. The first step is active. The configuration check results are shown in a box: 'Checking current configuration', 'Configuration files located in /run/cloudera-scm-agent/process/254-hue-HUE_SERVER', and 'All OK. Configuration check passed.'

Select: Create a new table from a file

The screenshot shows the Hue Metastore Manager interface. The top navigation bar includes links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation is a toolbar with icons for refresh, search, and other functions. The main title is "Metastore Manager". On the left, there's a sidebar with icons for HDFS and SQL, and a list of databases including "default" and "sample_07". The main content area shows the "Databases > default" view. It displays "STATS" for the database, including its status as a "Default Hive database" and its ownership by the "public (ROLE)". There's also a "Location" link. Below this is a "TABLES" section with a search bar and buttons for "View", "Browse Data", and "Drop". A table lists existing tables, with "sample_07" being the only entry. The table has columns for "Table Name", "Comment", and "Type". A red arrow points to the "+" icon in the top right corner of the "TABLES" section, which is used to create new tables.

Table Name	Comment	Type
sample_07		

Name a table and select a file

Databases > default > Create a new table from a file

Step 1: Choose File

Step 2: Choose Delimiter

Step 3: Define Columns

Name Your Table and Choose A File

Table Name

trip

Name of the new table. Table names must be globally unique. Table names tend to correspond to the column names.

Description

Bay Area Bike Share

Use a table comment to describe the table. For example, note the data's provenance and any other relevant information.

Input File

/user/cloudera/201408_trip_data.csv

The HDFS path to the file on which to base this new table definition. It can be compressed (.zip) or not.

Import data from file



Check this box to import the data in this file after creating the table definition. Leave it unchecked to define an empty table.

Warning: The selected file is going to be moved during the import.

Choose a file

Home

/user / cloudera

..

201402_trip_data.csv

input

movielens

output

Upload a file



Choose Delimiter

Databases > default > Create a new table from a file

Step 1: Choose File

Step 2: Choose Delimiter

Step 3: Define Columns

Choose a Delimiter

Beeswax has determined that this file is delimited by **commas**.

Delimiter

Comma (,)

Preview

Enter the column delimiter which must be a single character. Use syntax like "\001" or "\t" for special characters.

Table preview

col_1	col_2	col_3	col_4	col_5	col_6	col_7	col_8	col_9	col_10	col_11
Trip ID	Duration	Start Date	Start Station	Start Terminal	End Date	End Station	End Terminal	Bike #	Subscriber Type	Zip Code
432946	406	8/31/2014 22:31	Mountain View Caltrain St...	28	8/31/2014 22:38	Castro Street and El Cami...	32	17	Subscriber	94040
432945	468	8/31/2014 22:07	Beale at Market	56	8/31/2014 22:15	Market at 4th	76	509	Customer	11231

Define Column Types

Databases > default > Create a new table from a file

Step 1: Choose File Step 2: Choose Delimiter Step 3: Define Columns

Define your columns

Use first row as column names Bulk edit column names

Column name	Column Type	Sample Row #1	Sample Row #2
TripID	int	432946	432945
Duration	int	406	468
StartDate	string	8/31/2014 22:31	8/31/2014 22:07
StartStation	string	Mountain View Caltrain Station	Beale at Market
StartTerminal	tinyint	28	56
EndDate	string	8/31/2014 22:38	8/31/2014 22:15



Create Table : Done

Databases > default > trip

Comment: Bay Area Bike Share			
	Columns	Sample	Properties
	Name	Type	Comment
0	tripid	int	
1	duration	int	
2	startdate	string	
3	startstation	string	
4	startterminal	tinyint	
5	enddate	string	
6	endstation	string	
7	endterminal	tinyint	
8	bike	smallint	
9	subscribertype	string	
10	zipcode	smallint	

HUE Home Query Editors Data Browsers Workflows Search Security

Metastore Manager

Databases > default

STATS

Default Hive database public (ROLE) Location

TABLES

Search for a table... View Browse Data Drop

<input type="checkbox"/>	Table Name	Comment	Type
<input type="checkbox"/>	test_tbl		
<input type="checkbox"/>	trip		
<input type="checkbox"/>	users		

Starting Hive Editor

The screenshot shows the Hue Query Editor interface. At the top, there's a navigation bar with links for 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and 'cloudera'. Below the navigation bar, the main area has tabs for 'Hive Editor' (which is selected), 'Query Editor', 'My Queries', 'Saved Queries', and 'History'. On the left, there's a sidebar titled 'DATABASE' with a dropdown set to 'default'. Below it, a table list shows 'airline_data' and 'trip' tables, each with a preview icon, a refresh icon, and a details icon. The 'trip' table has columns listed: tripid (int), duration (int), startdate (string), startstation (string), startterminal (tinyint), enddate (string), endstation (string), endterminal (tinyint), bike (smallint), subscriptype (string), and zipcode (smallint). The main workspace contains a query editor with a placeholder text 'Example: SELECT * FROM tablename, or press CTRL + space'. Below the editor are buttons for 'Execute', 'Save as...', 'Explain', and 'New query'. A 'Recent queries' section at the bottom lists three previous queries with their execution time, query text, and a 'See results...' link:

Time	Query	Result
11/04/2015 2:49:28 PM	DROP TABLE `default`.`babs`	See results...
11/04/2015 2:46:08 PM	SELECT startterminal, startstation, COUNT(1) AS count FROM babs GROUP BY startterminal, startstation ORDER BY count DESC LIMIT 10	See results...
11/04/2015 2:45:42 PM	SELECT startterminal, startstation, COUNT(1) AS count FROM bikeshare.trips GROUP BY startterminal, startstation ORDER BY count	

Find the top 10 most popular start stations based on the trip data

```
SELECT startterminal, startstation, COUNT(1) AS count FROM trip
GROUP BY startterminal, startstation ORDER BY count DESC LIMIT 10
```

The screenshot shows the Hue web interface. At the top, there's a navigation bar with links for Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the main area has tabs for Hive Editor (selected), Query Editor, My Queries, Saved Queries, and History. On the left, there's a sidebar for the 'default' database showing tables: test_tbl, trip, and users. The central area contains a code editor with the query:

```
1 SELECT startterminal, startstation, COUNT(1) AS count FROM trip GROUP BY startte
```

Below the code editor are buttons for Execute, Save as..., Explain, Format, or create a New query. The bottom section shows the results of the query in a table format. The table has three columns: startterminal, startstation, and count. The results are:

	startterminal	startstation	count
1	70	San Francisco Caltrain (Townsend at 4th)	9838
2	50	Harry Bridges Plaza (Ferry Building)	7343
3	60	Embarcadero at Sansome	6545
4	77	Market at Sansome	5922
5	55	Temporary Transbay Terminal (Howard at Beale)	5113
6	76	Market at 4th	5030
7	61	2nd at Townsend	4987
8	69	San Francisco Caltrain 2 (330 Townsend)	4976

```
1 SELECT startterminal, startstation, COUNT(1) AS count FROM trip GROUP BY startte
```

Execute

Save as...

Explain

Format

or create a

New query



Recent queries

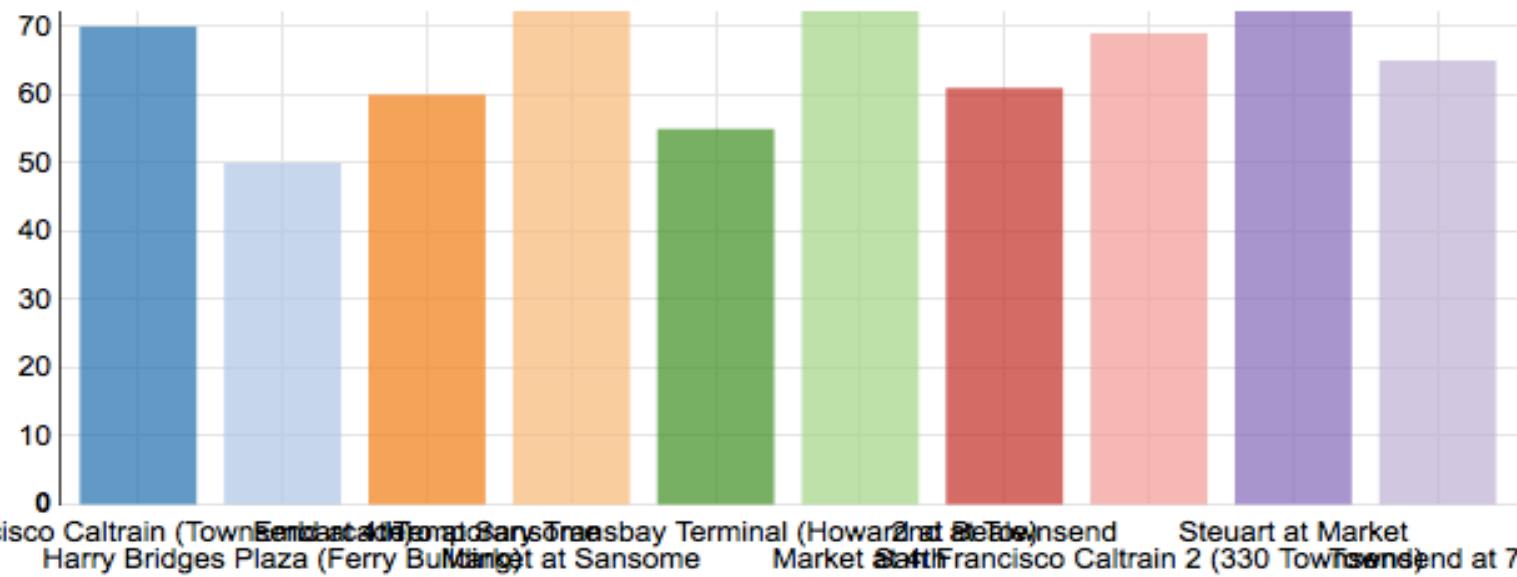
Query

Log

Columns

Results

Chart



Introduction

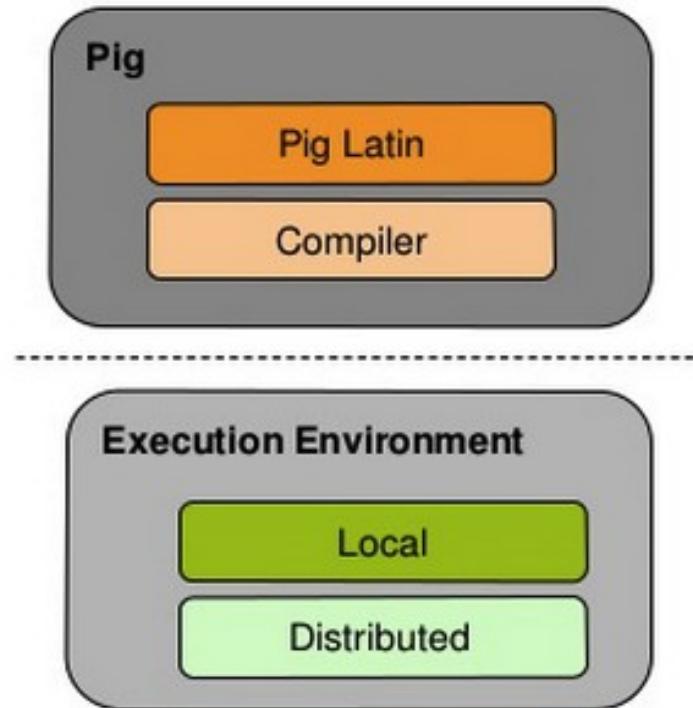
A high-level platform for creating MapReduce programs Using Hadoop



Pig is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs. The salient property of Pig programs is that their structure is amenable to substantial parallelization, which in turns enables them to handle very large data sets.

Pig Components

- **Two Components**
 - Language (Pig Latin)
 - Compiler
- **Two Execution Environments**
 - Local
 - pig -x local*
 - **Distributed**
 - pig -x mapreduce*



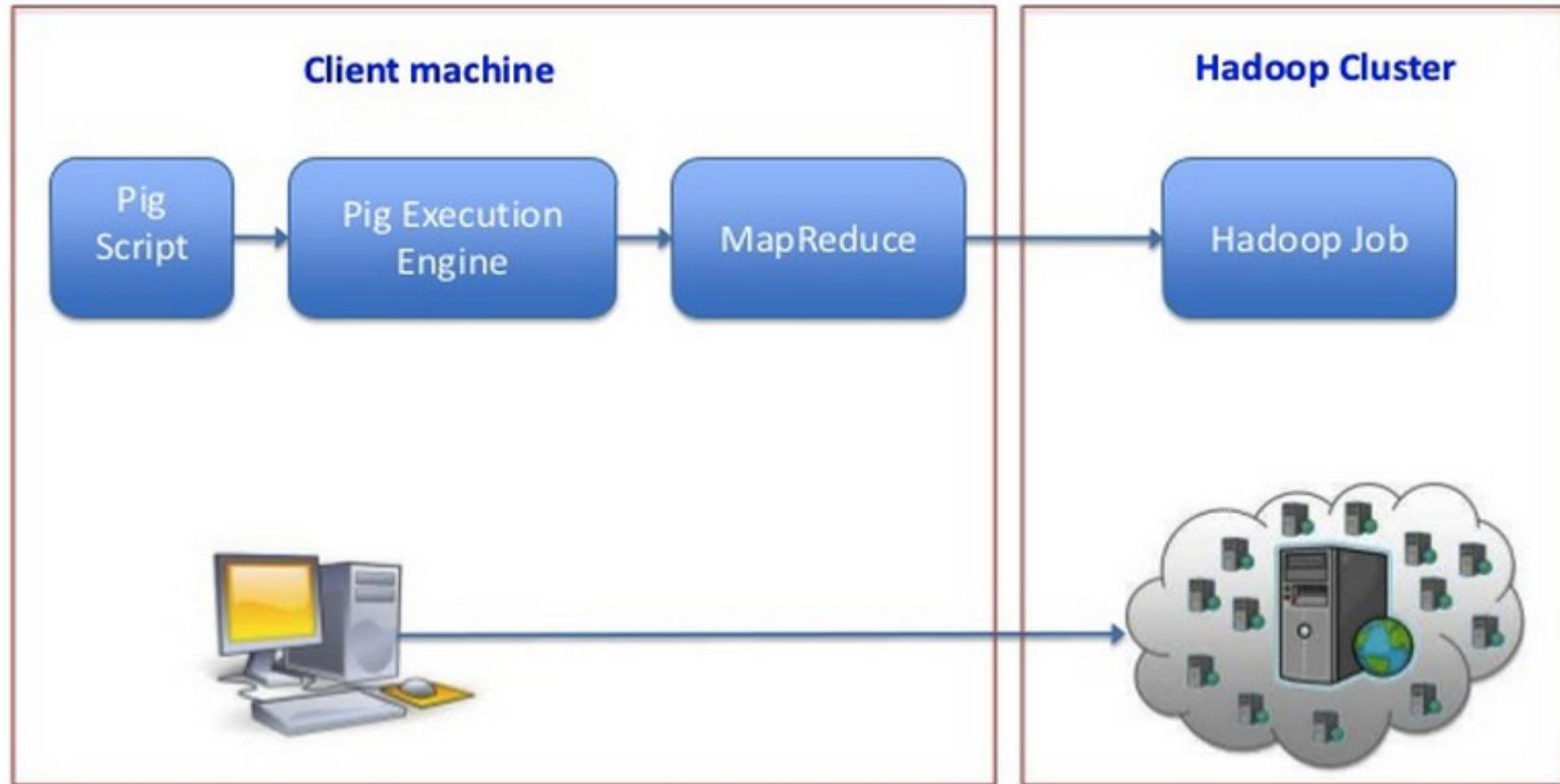
Running Pig

- **Script**
pig myscript
- **Command line (Grunt)**
pig
- **Embedded**
Writing a java program

Pig Latin

```
Users = load 'users' as (name, age);
Fltrd = filter Users by
    age >= 18 and age <= 25;
Pages = load 'pages' as (user, url);
Jnd = joinFltrdby name, Pages by user;
Grpd = group Jnd by url;
Smmd = foreach Grpd generate group,
COUNT(Jnd) as clicks;
Srtd = order Smmd by clicks desc;
Top5 = limit Srtd 5;
store Top5 into 'top5sites';
```

Pig Execution Stages



Why Pig?

- **Makes writing Hadoop jobs easier**
 - 5% of the code, 5% of the time
 - You don't need to be a programmer to write Pig scripts
- **Provide major functionality required for DatawareHouse and Analytics**
 - *Load, Filter, Join, Group By, Order, Transform*
 - **User can write custom UDFs (User Defined Function)**

Pig v.s. Hive



<i>Characteristic</i>	<i>Pig</i>	<i>Hive</i>
Developed by	Yahoo!	Facebook
Language name	Pig Latin	HiveQL
Type of language	Data flow	Declarative (SQL dialect)
Data structures it operates on	Complex, nested	
Schema optional?	Yes	No, but data can have many schemas
Relational complete?	Yes	Yes
Turing complete?	Yes when extended with Java UDFs	Yes when extended with Java UDFs

Hands-On: Running a Pig script

Starting Pig Command Line

```
$ pig -x mapreduce
2013-08-01 10:29:00,027 [main] INFO org.apache.pig.Main - Apache Pig
version 0.11.1 (r1459641) compiled Mar 22 2013, 02:13:53
2013-08-01 10:29:00,027 [main] INFO org.apache.pig.Main - Logging error
messages to: /home/hdadmin/pig_1375327740024.log
2013-08-01 10:29:00,066 [main] INFO org.apache.pig.impl.util.Utils -
Default bootup file /home/hdadmin/.pigbootup not found
2013-08-01 10:29:00,212 [main] INFO
org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting
to hadoop file system at: file:///
grunt>
```

Writing a Pig Script for wordcount

```
A = load '/user/cloudera/input/*';
B = foreach A generate flatten(TOKENIZE((chararray)$0)) as word;
C = group B by word;
D = foreach C generate COUNT(B), group;
store D into '/user/cloudera/output/wordcountPig';
```

```
2016-06-14 08:29:25,835 [main] INFO org.apache.pig.backend.hadoop.executionengine.m
apReduceLayer.MapReduceLauncher - More information at: http://localhost:50030/jobdet
ails.jsp?jobid=job_1465875170640_0004
2016-06-14 08:29:25,871 [main] INFO org.apache.pig.backend.hadoop.executionengine.m
apReduceLayer.MapReduceLauncher - 0% complete
2016-06-14 08:29:41,625 [main] INFO org.apache.pig.backend.hadoop.executionengine.m
apReduceLayer.MapReduceLauncher - 50% complete
2016-06-14 08:29:51,359 [main] INFO org.apache.pig.backend.hadoop.executionengine.m
apReduceLayer.MapReduceLauncher - 100% complete
2016-06-14 08:29:51,362 [main] INFO org.apache.pig.tools.pigstats.SimplePigStats -
Script Statistics:
```

HadoopVersion	PigVersion	UserId	StartedAt	FinishedAt	Features
2.6.0-cdh5.7.0	0.12.0-cdh5.7.0	root	2016-06-14 08:29:18	2016-06-14 08:29:51G	ROUP_BY

Success!

HUE Home Query Editors Data Browsers Workflows Search File Browser Job Browser hdfs ? Help

File Browser

ACTIONS	
	View as binary
	Download
	View file location
	Refresh
INFO	
Last modified	Nov. 7, 2015 8:40 a.m.
User	hdfs
Group	supergroup
Size	345.2 KB
Mode	100644

Home / user / hdfs / output / wordcountPig / part-r-00000

Page 35 of 87

```
wer.  
2    drawers  
199   drawing  
1     drawled  
7     dreaded  
14    dreamed  
2     dreamer  
1     dreams.  
65    dressed  
3     dresser  
23    dresses  
1     drifted  
3     driver.  
6     drivers  
34    driving  
1     drones.  
1     drooped  
29    dropped  
1     drought  
13    drowned  
14    drummer
```



Impala

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

Introduction

open source massively parallel processing (MPP) SQL query engine



Cloudera Impala is a query engine that runs on Apache Hadoop. Impala brings scalable parallel database technology to Hadoop, enabling users to issue low-latency SQL queries to data stored in HDFS and Apache HBase without requiring data movement or transformation.

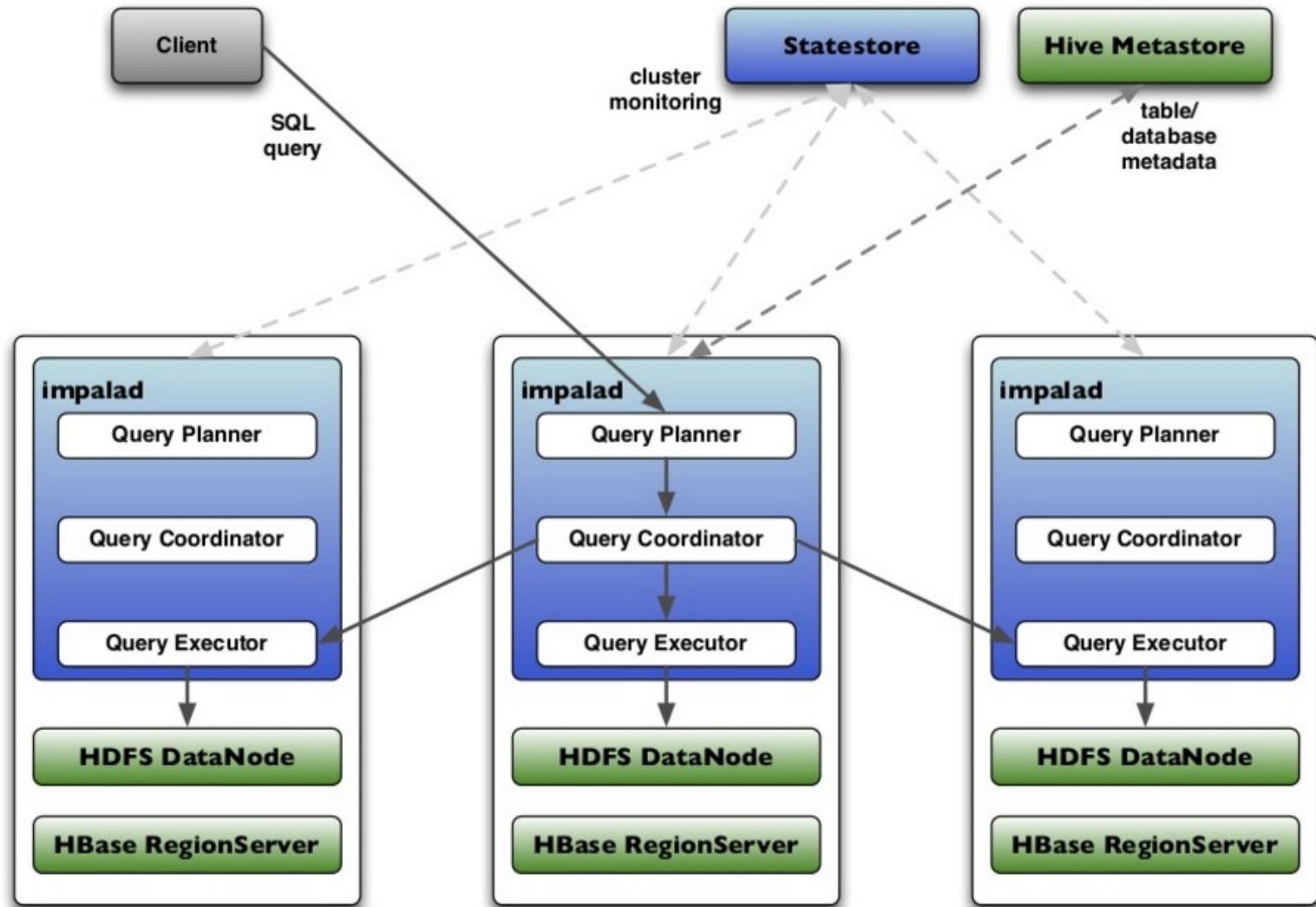
What is Impala?

- General--- purpose SQL engine
- Real--time queries in Apache Hadoop
- Opensource under Apache License
- Runs directly within Hadoop
- High performance
 - C++ instead of Java
 - Runtime code generator
 - Roughly 4-100 x Hive

Impala Overview

- Impala daemon run on HDFS nodes
- Statestore (for cluster metadata) v.s. Metastore (for database metastore)
- Queries run on “relevant” nodes
- Support common HDFS file formats
- Submit queries via Hue/Beeswax
- No fault tolerant

Impala Architecture



Start Impala Query Editor

The screenshot shows the Hue web interface for managing Hadoop data. At the top, there's a navigation bar with links for File Browser, Job Browser, guest1, and a help icon. Below the navigation bar is a header with tabs for Home, Query Editors (selected), Metastore Manager, Workflows, and Search. On the left, there's a sidebar with tabs for Assist (selected) and Settings, and sections for DATABASE (set to default) and TABLES (which is empty). The main area is titled 'Query Editors' and shows a dropdown menu with five options: Hive, Impala (which is highlighted with a red arrow), DB Query, Pig, and Job Designer. Below the dropdown is a text input field with placeholder text: 'Example: SELECT * FROM tablename, or press CTRL + space'. At the bottom of the input field are buttons for Execute, Save as..., Explain, and New query. To the right of the input field is a results panel with tabs for Recent queries, Query, Log, Columns, Results, and Chart. The 'Recent queries' tab is selected, showing a table with columns for Time, Query, and Result. The table displays the message 'No data available'.

Update the list of tables/metadata by execute the command **invalidate metadata**

The screenshot shows the Hue Query Editor interface. At the top, there's a navigation bar with links like 'Query Editors', 'Data Browsers', 'Workflows', 'Search', and 'Security'. Below the navigation bar, the 'Impala' tab is selected in the 'Query Editor' section, which also includes 'My Queries', 'Saved Queries', and 'History'.

In the main query editor area, the command `1 invalidate metadata` is typed into the text input field. Below the input field, there's a toolbar with several buttons: 'Execute' (which is highlighted with a red arrow pointing to it), 'Save as...', 'Explain', 'Format', 'or create a', 'New query', and a '...' button. There are also tabs for 'Recent queries', 'Query', 'Log', 'Columns', 'Results', and 'Chart'.

At the bottom of the interface, there are filters for 'Time' and 'Query' (set to 'No data available') and a 'Result' section.

Restart Impala Query Editor and refresh the table list

The screenshot shows the Hue Query Editor interface. At the top, there's a navigation bar with links for Home, Query Editors, Metastore Manager, Workflows, Search, File Browser, Job Browser, and a user icon. Below the navigation bar, the title bar says "HUE" and "Impala". The main area has tabs for "Query Editor" (which is selected), My Queries, Saved Queries, and History. On the left, there's a sidebar with tabs for Assist, Settings, and Session. Under the "DATABASE" section, a dropdown menu is open, showing "default" and a red arrow pointing to the refresh icon next to it. The main query editor area has a placeholder text "Example: SELECT * FROM tablename, or press CTRL + space". Below it are buttons for Execute, Save as..., Explain, and New query. A message says "or create a". At the bottom, there's a timeline view with tabs for Recent queries, Query, Log, Columns, Results, and Chart. The timeline shows a single entry: "03/27/2016 5:42:28 PM" followed by the query "invalidate metadata;".

Find the top 10 most popular start stations based on the trip data: Using Impala

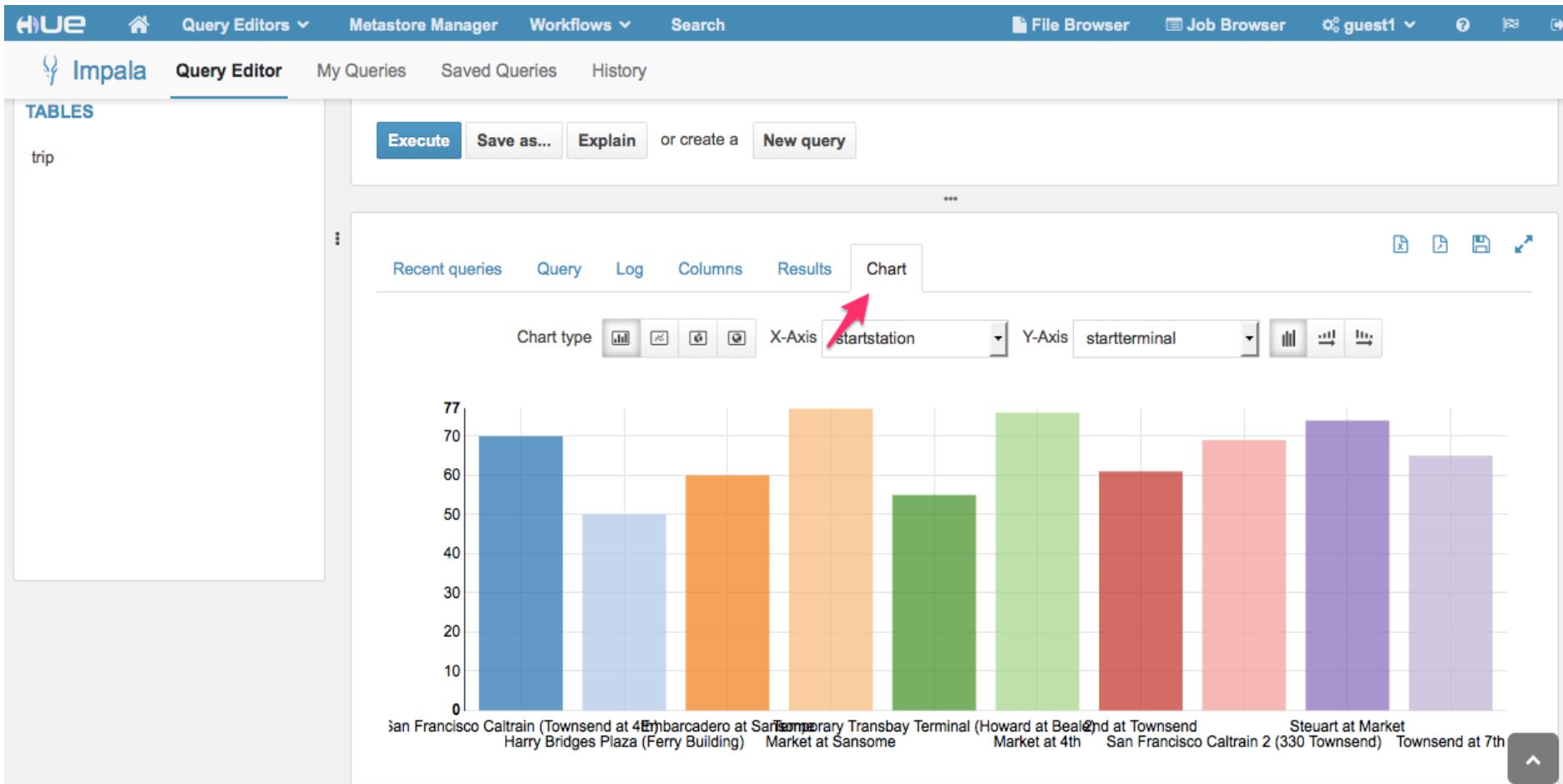
```
SELECT startterminal, startstation, COUNT(1) AS count FROM trip
GROUP BY startterminal, startstation ORDER BY count DESC LIMIT 10
```

The screenshot shows the Hue web interface for Apache Impala. At the top, there's a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the main area has tabs for Assist, Settings (which is selected), and Session. On the left, there's a sidebar with a 'default' connection icon, a 'Tables' section listing 'test_tbl' and 'trip', and a 'Recent queries' section showing '(3)' entries. The central part of the screen contains the query editor. The query itself is:

```
1 SELECT startterminal, startstation, COUNT(1) AS count FROM trip GROUP BY startte
2
```

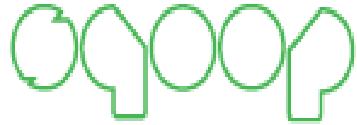
Below the query editor are buttons for Execute, Save as..., Explain, Format, or create a New query. The results section is titled 'Results' and shows the following table:

	startterminal	startstation	count
1	70	San Francisco Caltrain (Townsend at 4th)	9838
2	50	Harry Bridges Plaza (Ferry Building)	7343
3	60	Embarcadero at Sansome	6545
4	77	Market at Sansome	5922



Find the total number of trips and average duration (in minutes) of those trips, grouped by hour

```
SELECT
    hour,
    COUNT(1) AS trips,
    ROUND(AVG(duration) / 60) AS avg_duration
FROM (
    SELECT
        CAST(SPLIT(SPLIT(t.startdate, ' ') [1], ':') [0] AS INT) AS
hour,
        t.duration AS duration
    FROM `bikeshare`.`trips` t
    WHERE
        t.startterminal = 70
        AND
        t.duration IS NOT NULL
    ) r
GROUP BY hour
ORDER BY hour ASC;
```



Apache Sqoop

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

Introduction

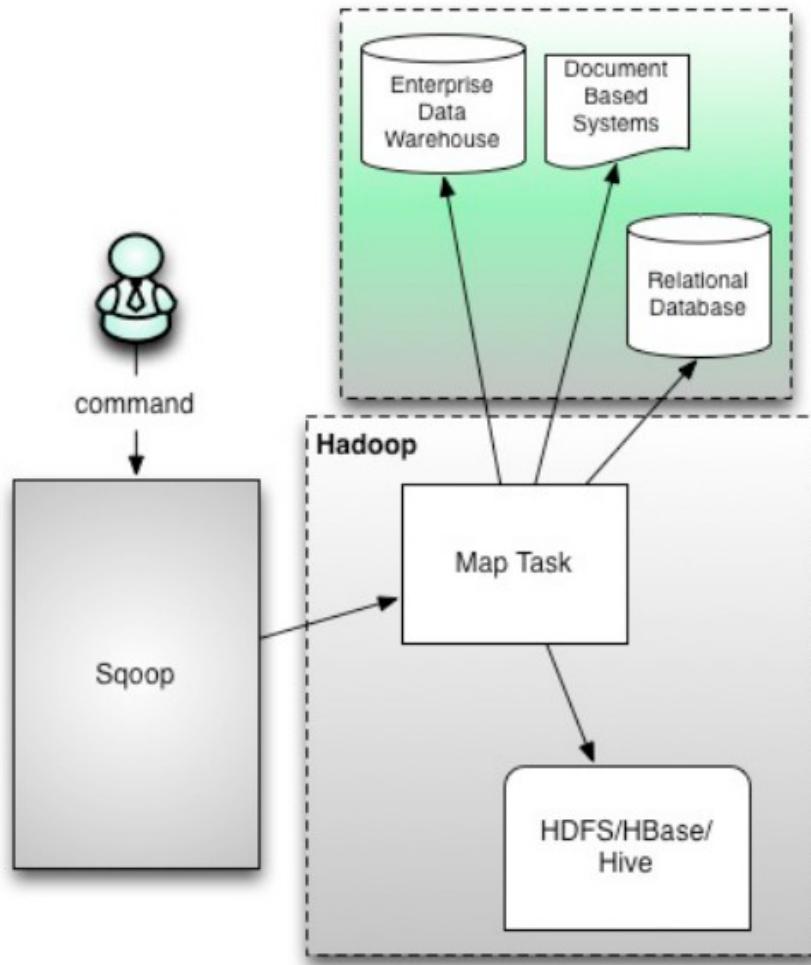


Sqoop (“SQL-to-Hadoop”) is a straightforward command-line tool with the following capabilities:

- Imports individual tables or entire databases to files in HDFS
- Generates Java classes to allow you to interact with your imported data
- Provides the ability to import from SQL databases straight into your Hive data warehouse

See also: <http://sqoop.apache.org/docs/1.4.2/SqoopUserGuide.html>

Architecture Overview



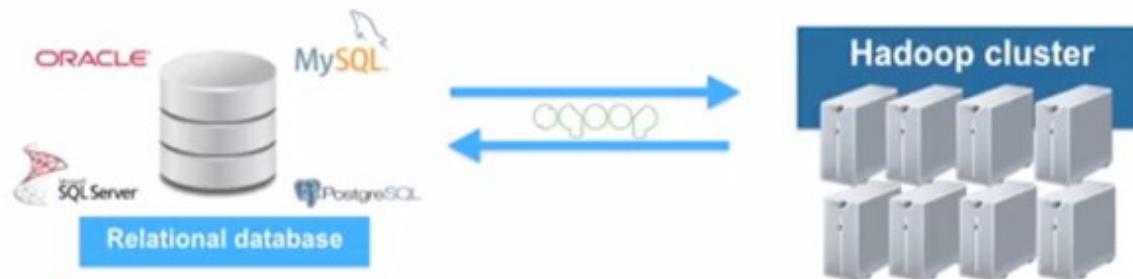
Hive.apache.org

Sqoop Benefit

- Leverages RDBMS metadata to get the column data types
- It is simple to script and uses SQL
- It can be used to handle change data capture by importing daily transactional data to Hadoop
- It uses MapReduce for export and import that enables parallel and efficient data movement

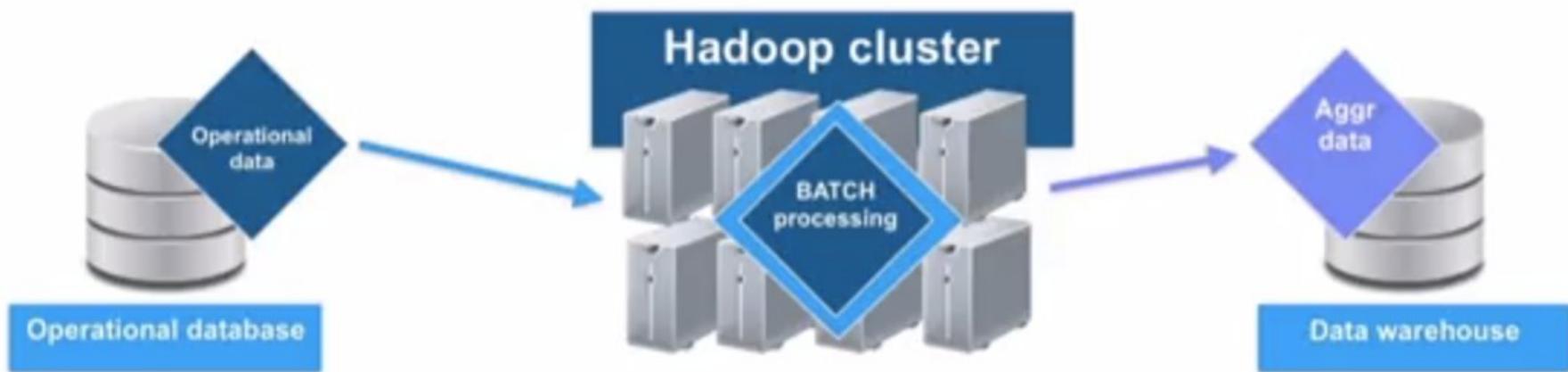
Sqoop Mode

- Sqoop import: Data moves from RDBMS to Hadoop
- Sqoop export: Data moves from Hadoop to RDBMS



Use Case #1: ETL for Data Warehouse

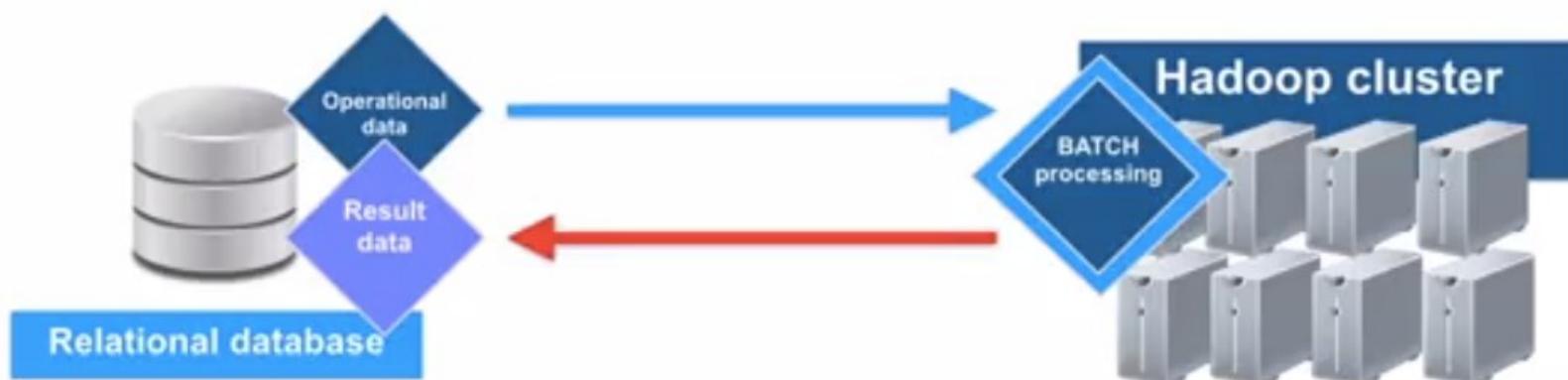
- Transform operational data for data warehouse reports in Hadoop as the batch transformation “engine”



Source: Mastering Apache Sqoop, David Yahalom, 2016

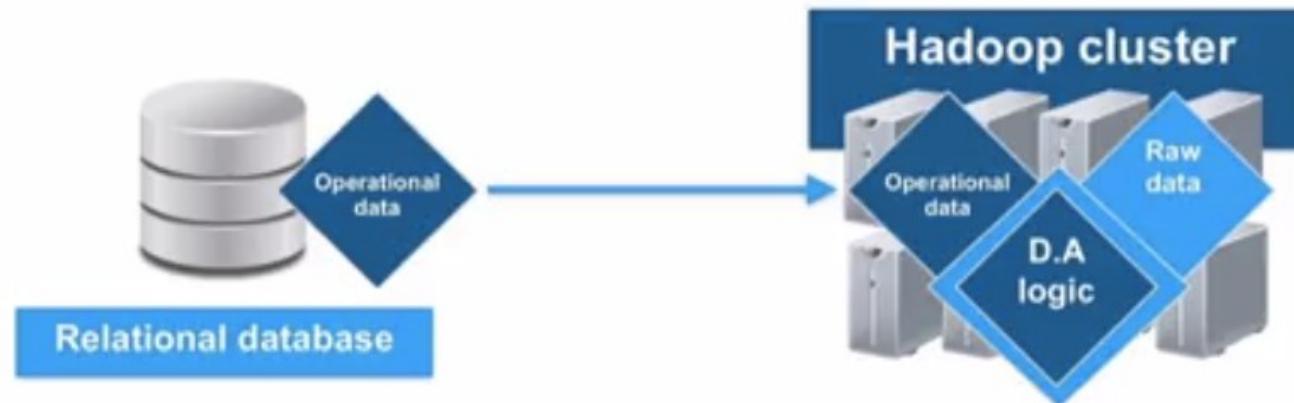
Use Case #2: ELT

- Extract operational data from RDBMS, process in Hadoop, return **result** to RDBMS



Use Case #3: Data Analysis

- Copy real-time data from RDBMS, combine with raw data on Hadoop using complex data analysis logic (not just SQL!)



Source: Mastering Apache Sqoop, David Yahalom, 2016

Use Case #4: Data Archival

- Move data from RDBMS after it expires to Hadoop, keeping the RDBMS “clean and lean”



Source: Mastering Apache Sqoop, David Yahalom, 2016

Use Case #5: Data Consolidation

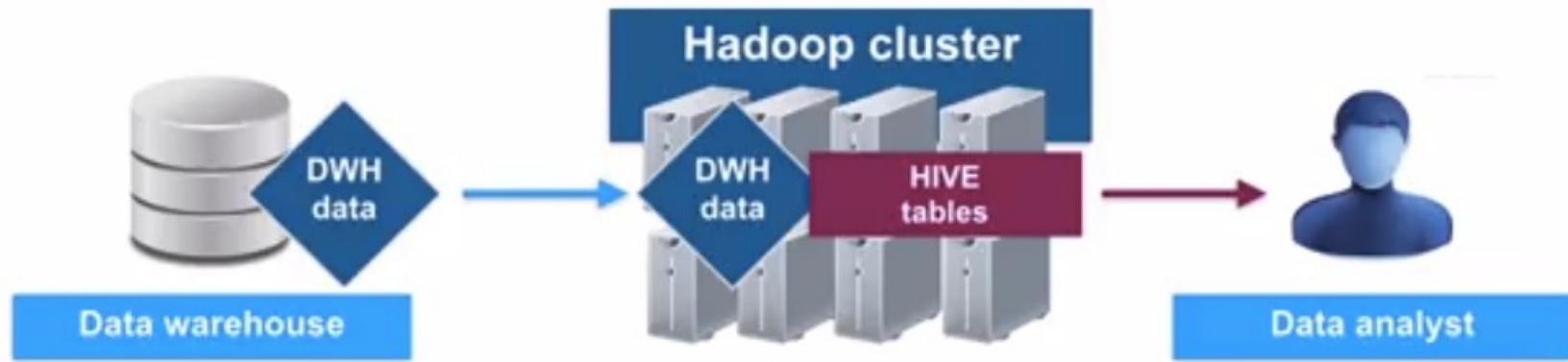
- Integrate data from various organizational “data stores” to Hadoop for various data processing requirements



Source: Mastering Apache Sqoop, David Yahalom, 2016

Use Case #6: Move reports to Hadoop

- Easily allow traditional data analysis and business intelligence using Hadoop's power

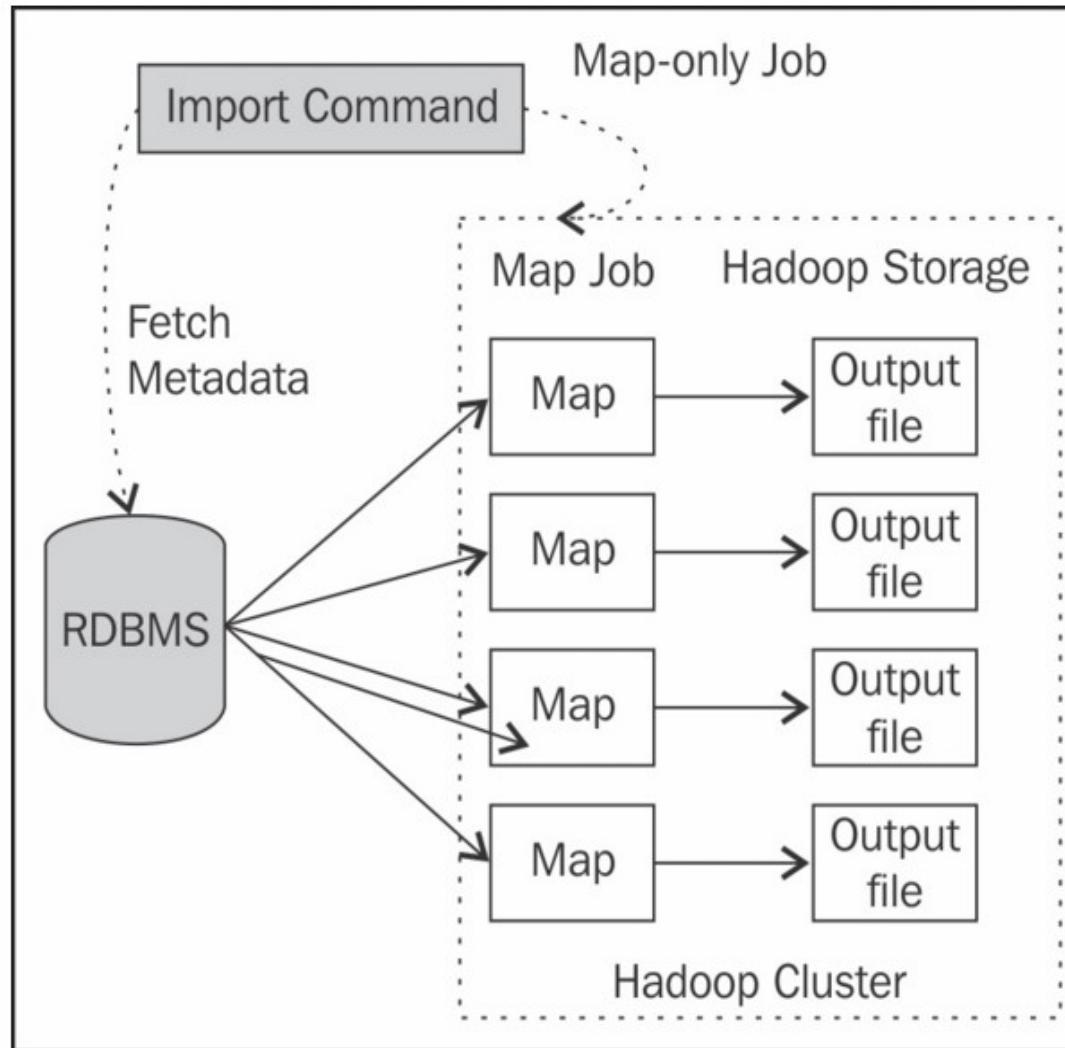


Source: Mastering Apache Sqoop, David Yahalom, 2016

Import Commands

Parameters	Description
<code>--connect <jdbc-uri></code>	Specifies the server or database to connect to. It also specifies the port. For example: <code>--connect jdbc:mysql://host:port/databaseName</code>
<code>--connection-manager <class-name></code>	Specifies the connection manager class name.
<code>--driver <class-name></code>	Specifies the fully qualified name of the JDBC driver class.
<code>--hadoop-home <dir></code>	This parameter is used to override the <code>\$HADOOP_HOME</code> environment variable.
<code>-P</code>	If a user doesn't want to specify the database password along with the command, we can use the <code>-P</code> option to read the password from the console.
<code>--password <password></code>	Sets the authentication password required to connect to the input source.
<code>--username <username></code>	Sets the authentication username.
<code>--connection-param-file <properties-file></code>	Specifies the connection parameter's file.
<code>--help</code>	This option will provide the usage instructions.
<code>--verbose</code>	Prints more information during a query execution.

Architecture of the import process



Incremental import

Parameter/argument	Description
--check-column <column-name>	The value of this column is used to determine the rows to be imported during the import process.
--incremental <incremental-type>	Specifies the type of incremental mode. Possible values are <code>append</code> and <code>lastmodified</code> .
--last-value <value>	Specifies the last value or the maximum value of the <code>check</code> column from the previous import. All the records whose <code>check</code> column value is greater than the value of the <code>--last-value</code> argument will be imported to HDFS.

```
bin/sqoop import -connect jdbc:mysql://localhost:3306/db1 -username root -password password -  
table student -target-dir /user/abc/student -columns "student_id,address,name" --incremental  
lastmodified -last-value "2012-11-06 19:01:35"--check-column col4
```

Export Commands

Parameters	Description
--direct	Use the direct mode to perform the export quickly. Note that it is only supported for MySQL.
--export-dir<dir>	The location of input files in HDFS.
--table <table-name>	Name of the output table (the RDBMS table).
-m, --num-mappers <n>	Refers to the number of map tasks.
--update-mode <mode>	Specifies how updates are performed when new rows are found with non-matching keys in the database. Legal values for the mode include updateonly (default) and allowinsert .
--update-key <col-name>	The value of this column is used to identify the records that a user wants to update during the update mode. Use a comma-separated list of columns if there is more than one column.
--staging-table <staging-table-name>	Specifies the name of the staging table. The staging table is used to stage the data before inserting it into the destination table.
--clear-staging-table	This argument is used to clean the data from the staging table.

Hands-On: Loading Data from RDBMS to Hadoop

Running MySQL Docker

- Command: sudo docker pull mysql
- Command: sudo docker run --name imcMysql -e MYSQL_ROOT_PASSWORD=imcinstiute -p 3306:3306 -d mysql
- Command: sudo docker exec -it imcMysql bash

```
root@f1922a70e09c:/# mysql -uroot -p"imcinstiute"
```

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> █
```

Prepare a test database table

```
mysql> CREATE DATABASE imc_db;  
mysql> USE imc_db;  
mysql> CREATE TABLE country_tbl(id INT NOT NULL, country  
VARCHAR(50), PRIMARY KEY (id));  
  
mysql> INSERT INTO country_tbl VALUES(1, 'USA');  
mysql> INSERT INTO country_tbl VALUES(2, 'CANADA');  
mysql> INSERT INTO country_tbl VALUES(3, 'Mexico');  
mysql> INSERT INTO country_tbl VALUES(4, 'Brazil');  
mysql> INSERT INTO country_tbl VALUES(61, 'Japan');  
mysql> INSERT INTO country_tbl VALUES(65, 'Singapore');  
mysql> INSERT INTO country_tbl VALUES(66, 'Thailand');
```

View data in the table

```
mysql> SELECT * FROM country_tbl;
```

+-----+	-----+
id	country
+-----+	-----+
1	USA
2	CANADA
3	Mexico
4	Brazil
61	Japan
65	Singapore
66	Thailand
+-----+	-----+

```
7 rows in set (0.00 sec)
```

```
mysql> exit;
```

Then **exit** from the container by press **Ctrl-P & Ctrl-Q**

Restart the Cloudera docker with linking to the MySQL Docker

Command: sudo run --hostname=quickstart.cloudera
--privileged=true --link imcMysql:mysql -t -i -p
8888:8888 cloudera/quickstart /usr/bin/docker-quickstart

*If both of these Dockers are up and running, you can find out the internal IP address of each of them by running this command. This gets the IP for **imcMysql**.*

- Command: sudo docker inspect imcMysql | grep IPAddress

"IPAddress": "172.17.0.7",

Restart the Cloudera docker with linking to the MySQL Docker

Command: `sudo docker run --hostname=quickstart.cloudera --privileged=true --link imcMysql:mysql -t -i -p 8888:8888 cloudera/quickstart /usr/bin/docker-quickstart`

If both of these Dockers are up and running, you can find out the internal IP address of each of them by running this command. This gets the IP for `imcMysql`.

- Command: `sudo docker inspect imcMysql | grep IPAddress`

`"IPAddress": "172.17.0.7",`

Check a MySQL driver for Sqoop

```
$ cd /var/lib/sqoop  
$ ls
```

```
[root@quickstart sqoop]# ls  
mysql-connector-java.jar
```

Note: If you do not see the driver file, you need to install one by using the following command

```
$ wget https://s3.amazonaws.com/imcbucket/apps/mysql-connector-java-  
5.1.23-bin.jar
```

Importing data from MySQL to HDFS

```
$sqoop import --connect jdbc:mysql://172.17.0.7/imc_db  
--username root --password imcinstiute --table country_tbl  
--target-dir /user/cloudera/testtable -m 1
```

The screenshot shows the Apache Hue interface, specifically the File Browser. The top navigation bar includes links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation is a toolbar with various icons. The main area is titled "File Browser". On the left, there's a sidebar with "ACTIONS" and options like "View as binary", "Edit file", "Download", "View file location", and "Refresh". The central content area displays a file tree path: Home / user / cloudera / testtable / part-m-00000. This path is highlighted with a red oval. To the right of the path is a page navigation section showing "Page 1 of 1" and arrows for navigating through the data. The main content area shows a list of data entries:

1,USA
2,CANADA
3,Mexico
4,Brazil
61,Japan
65,Singapore
66,Thailand

Importing data from MySQL to Hive Table

```
$sqoop import --connect jdbc:mysql://172.17.0.7/imc_db  
--username root --password imcinstiute --table country_tbl  
--hive-import --hive-table country -m 1
```

The screenshot shows the Apache Hive File Browser interface. At the top, there is a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the title "File Browser" is displayed. In the center, there is a breadcrumb navigation path: / user / hive / warehouse / country / part-m-00000. The word "part-m-00000" is highlighted with a red oval. On the left side, there is a sidebar with "ACTIONS" and several options: View as binary, Edit file, Download, View file location, and Refresh. The main content area displays a list of data rows:

1USA
2CANADA
3Mexico
4Brazil
61Japan
65Singapore
66Thailand

At the bottom right of the main content area, there are page navigation controls: Page 1 of 1, and arrows for navigating between pages.

Reviewing data from Hive Table

```
[root@quickstart /]# hive
```

```
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j.properties
```

```
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
```

```
hive> show tables;
```

```
hive> select * from country;
```

```
--  
1      USA  
2      CANADA  
3      Mexico  
4      Brazil  
61     Japan  
65     Singapore  
66     Thailand
```

```
Time taken: 0.587 seconds, Fetched: 7 row(s)
```

Running from Hue: Beewax

The screenshot shows the Hue Hive Editor interface. At the top, there's a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, Search, and Security. Below the navigation bar, the main area has tabs for Hive Editor (selected) and Query Editor, along with links for My Queries, Saved Queries, and History. On the left, there's a sidebar with Assist and Settings tabs, and sections for default database (Tables: country), and a search bar. The main workspace contains a query editor with the following code:

```
1 SELECT * FROM country;
```

Below the query editor are buttons for Execute, Save as..., Explain, Format, and a link to create a new query. The results section is currently active, showing a table with two columns: country.id and country.country. The data is as follows:

	country.id	country.country
1	1	USA
2	2	CANADA
3	3	Mexico
4	4	Brazil
5	61	Japan
6	65	Singapore
7	66	Thailand

Importing data from MySQL to HBase

```
$sqoop import --connect jdbc:mysql://172.17.0.7/imc_db  
--username root --password imcinstiute --table country_tbl  
--hbase-table country --column-family hbase_country_cf --hbase-row-key  
id --hbase-create-table -m 1
```

Start HBase

```
$hbase shell  
hbase(main):001:0> list
```

```
TABLE  
country  
1 row(s) in 0.2570 seconds  
=> ["country"]
```

Viewing Hbase data

```
hbase(main):003:0> scan 'country'
ROW                                COLUMN+CELL
 1        column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=USA
 2        column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=CANADA
 3        column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=Mexico
 4        column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=Brazil
 61       column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=Japan
 65       column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=Singapore
 66       column=hbase_country_cf:country, timestamp=1468081466623, val
                               ue=Thailand
7 row(s) in 0.1670 seconds
```

Viewing data from Hbase browser

The screenshot shows the HBase Browser interface within the Hue web application. The top navigation bar includes links for Query Editors, Data Browsers, Workflows, Search, and Security, along with various system icons.

The main title is "HBase Browser" under the "Data Browsers" section. The page title is "Home - Cluster / country". A search bar at the top contains the query: "row_key, row_prefix* +scan_len [col1, family:col2, fam3; col_prefix*]" and a search button. To the right are buttons for "Filter Columns/Families", "All" (checked), and "Sort By ASC".

The data is presented in three rows:

- Row 1:** hbase_country_cf: country
USA
- Row 2:** hbase_country_cf: country
CANADA
- Row 3:** hbase_country_cf: country
Mexico



Apache Flume

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

Introduction

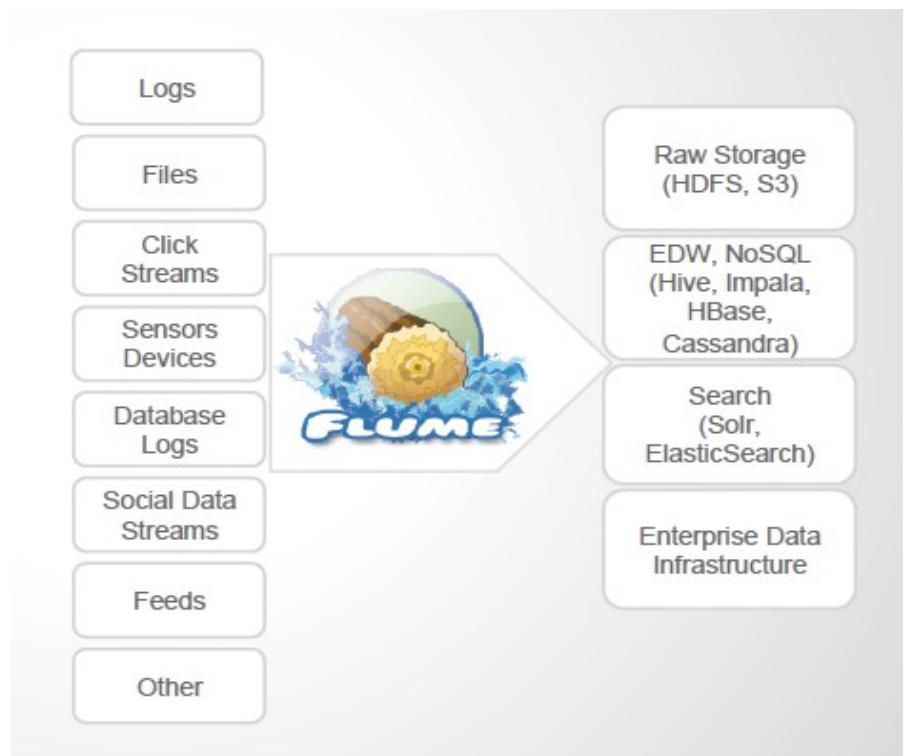


Apache Flume is:

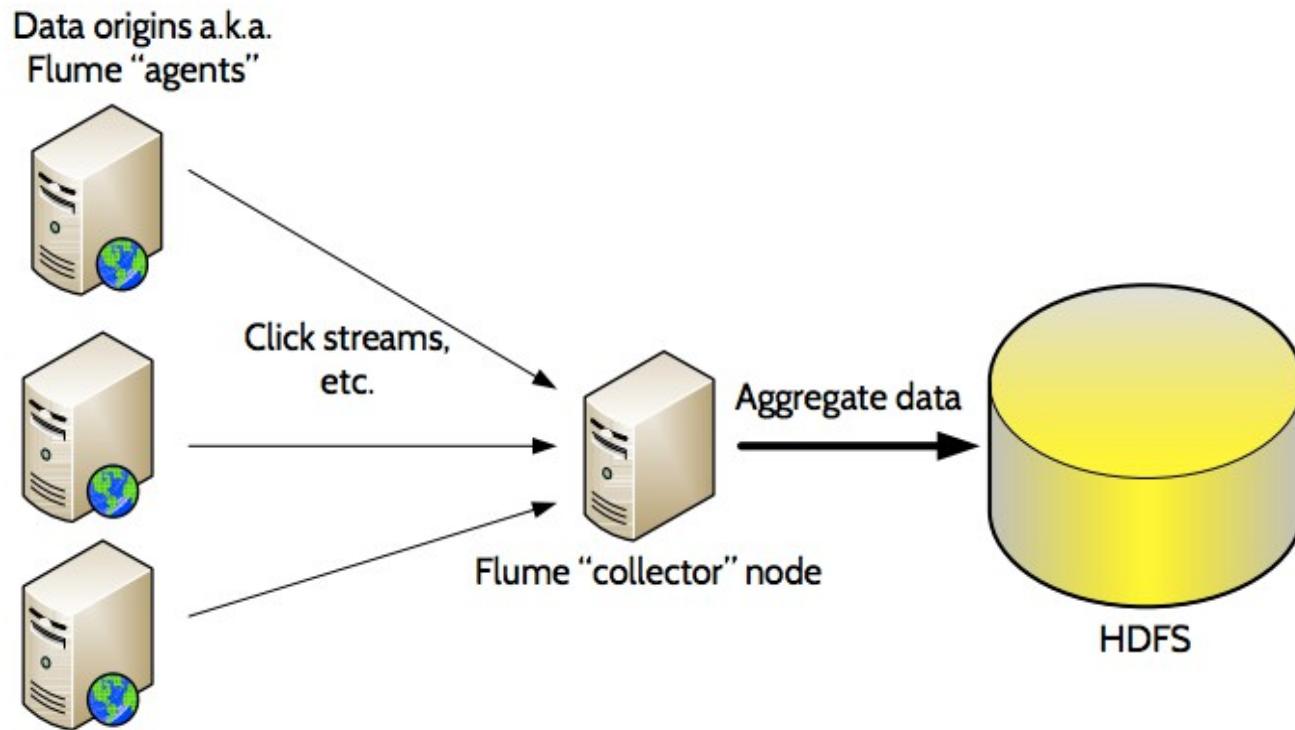
- **A distributed data transport and aggregation system for event- or log-structured data**
- **Principally designed for continuous data ingestion into Hadoop... But more flexible than that**

What is Flume?

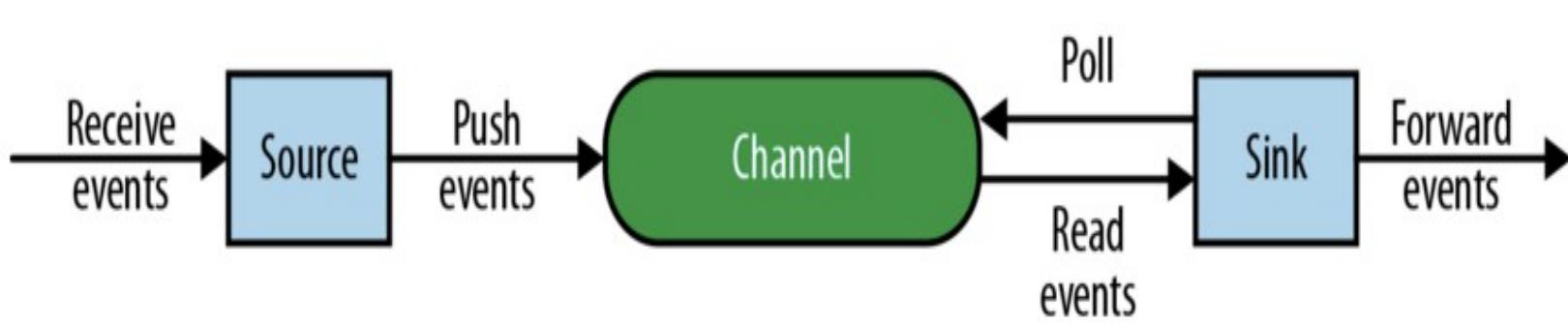
- Apache Flume is a continuous data ingestion system that is...
 - open-source,
 - reliable,
 - scalable,
 - manageable,
 - Customizable,
 - and designed for Big Data ecosystem



Architecture Overview



Flume Agent



- A source writes events to one or more channels.
- A channel is the holding area as events are passed from a source to a sink.
- A sink receives events from one channel only.
- An agent can have many channels.

Source: Using Flume, Hari Shreedharan, 2014

Sources

- Different Source types:
- Require at least one channel to function
- Specialized sources for integrating with well-known systems.
 - Example: Spooling Files, Syslog, Netcat, JMS
 - Auto-Generating Sources: Exec, SEQ
 - IPC sources for Agent-to-Agent communication: Avro, Thrift

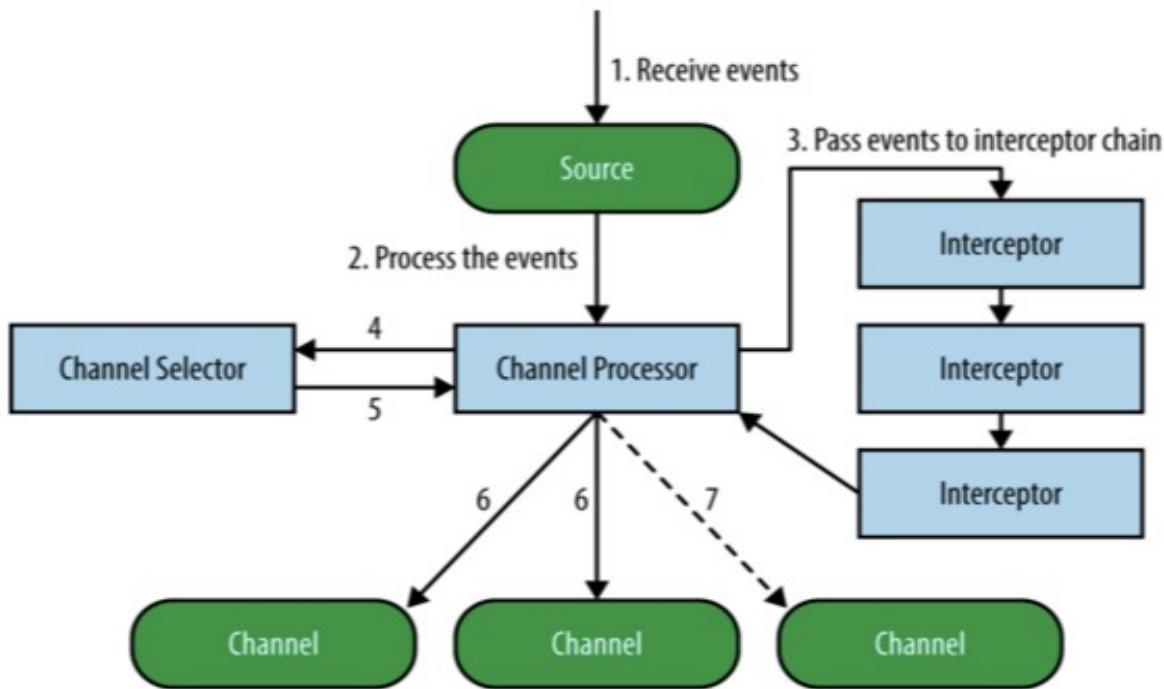
Channel

- Different Channels offer different levels of persistence:
 - Memory Channel
 - File Channel:
- Eventually, when the agent comes back data can be accessed.
- Channels are fully transactional
- Provide weak ordering guarantees
- Can work with any number of Sources and Sinks

Sink

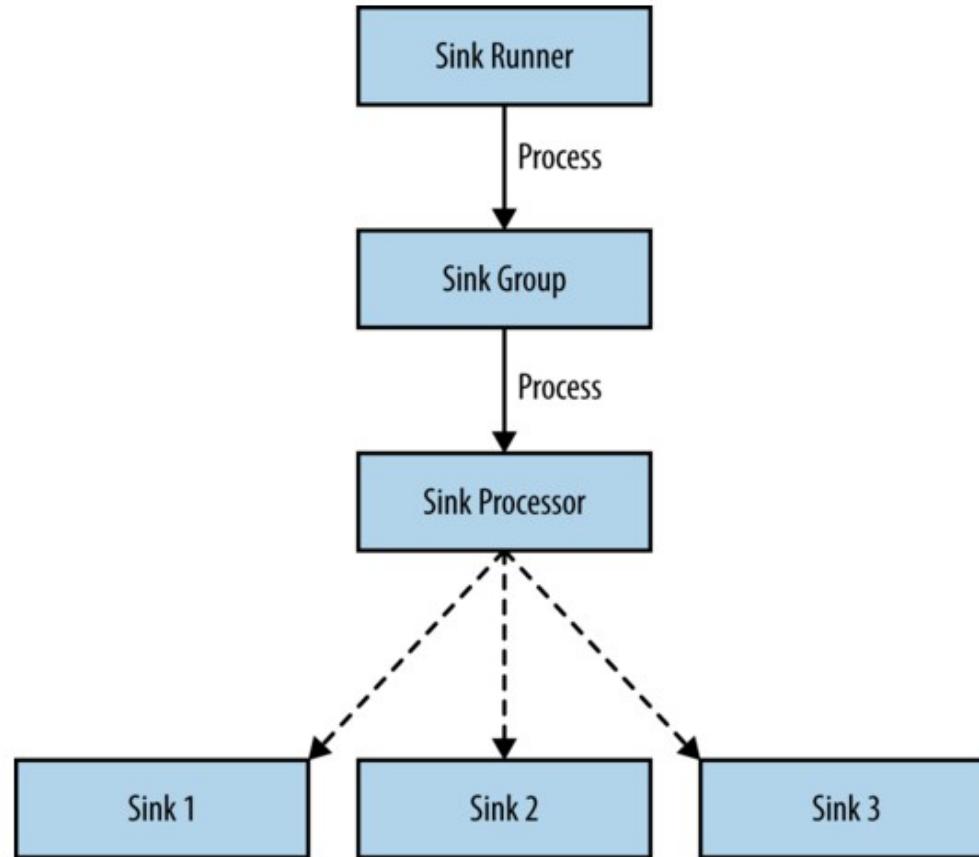
- Different types of Sinks:
 - Terminal sinks that deposit events to their final destination. For example: HDFS, HBase, Morphline-Solr, Elastic Search
 - Sinks support serialization to user's preferred formats.
 - HDFS sink supports time-based and arbitrary bucketing of data while writing to HDFS.
 - IPC sink for Agent-to-Agent communication: Avro, Thrift
- Require exactly one channel to function

Flume Process



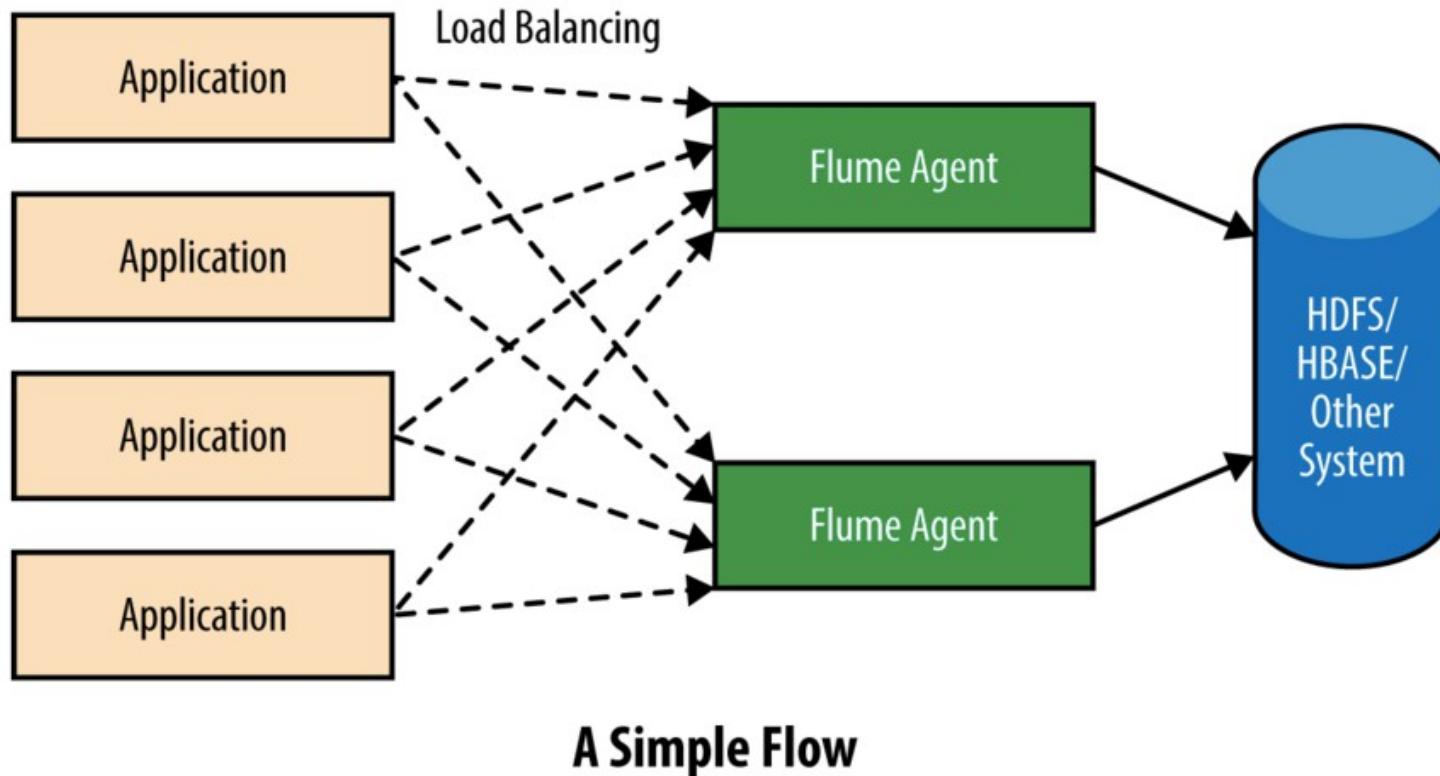
Source: Using Flume, Hari Shreedharan, 2014

Flume Process



Source: Using Flume, Hari Shreedharan, 2014

Flow



Source: Using Flume, Hari Shreedharan, 2014

Flume terminology

- A source writes events to one or more channels.
- A channel is the holding area as events are passed from a source to a sink.
- A sink receives events from one channel only.
- An agent can have many channels.

Flume Agent Configuration : Example

```
agent.sources = httpSrc
agent.channels = memory1 memory2
agent.sinks = hdfsSink hbaseSink

agent.sources.httpSrc.type = http
agent.sources.httpSrc.channels = memory1 memory2

# Bind to all interfaces
agent.sources.httpSrc.bind = 0.0.0.0
agent.sources.httpSrc.port = 4353

# Removing this line will disable SSL
agent.sources.httpSrc.ssl = true
agent.sources.httpSrc.keystore = /tmp/keystore
agent.sources.httpSrc.keystore-password = UsingFlume

agent.sources.httpSrc.handler = usingflume.ch03.HTTPSourceXMLHandler
agent.sources.httpSrc.handler.insertTimestamp = true

agent.sources.httpSrc.interceptors = hostInterceptor
agent.sources.httpSrc.interceptors.hostInterceptor.type = host
```

Flume Agent Configuration : Example

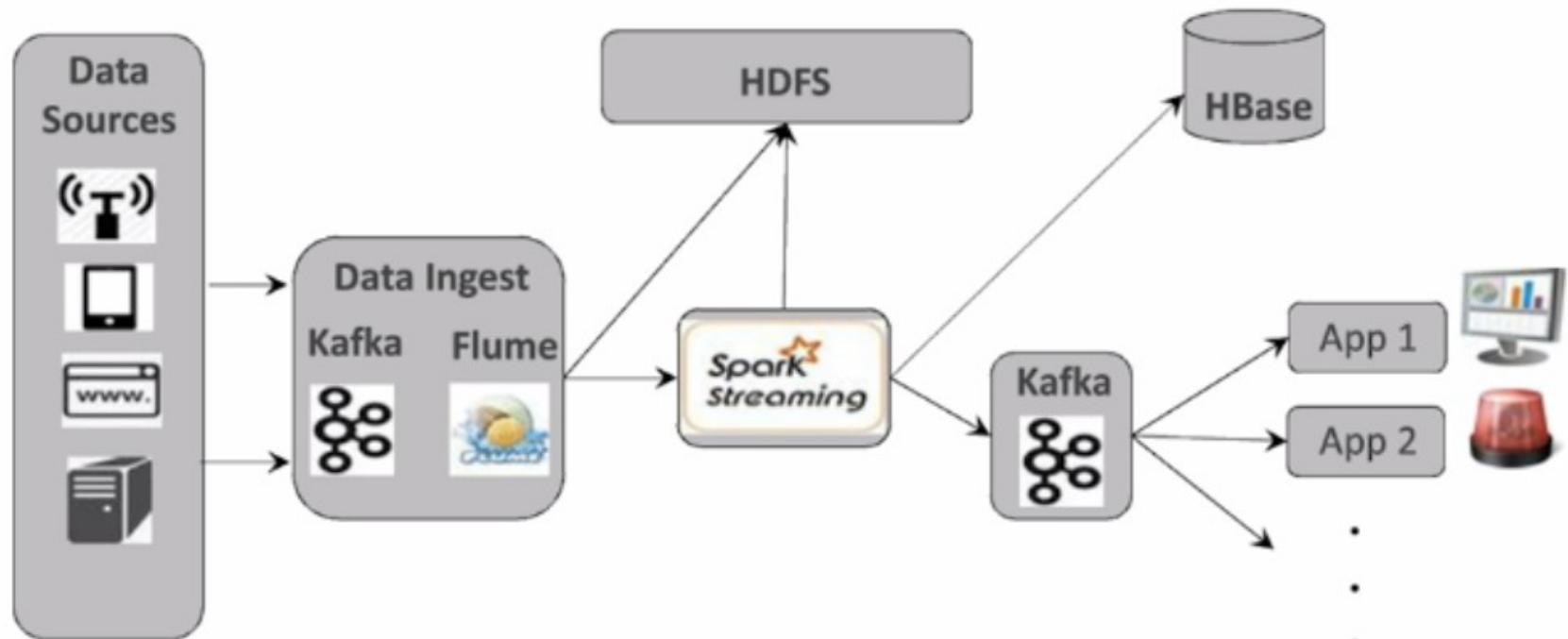
```
# Initializes a memory channel with default configuration
agent.channels.memory1.type = memory

# Initializes a memory channel with default configuration
agent.channels.memory2.type = memory

# HDFS Sink
agent.sinks.hdfsSink.type = hdfs
agent.sinks.hdfsSink.channel = memory1
agent.sinks.hdfsSink.hdfs.path = /Data/UsingFlume/{topic}/{Y}/{m}/{d}/{H}/{M}
agent.sinks.hdfsSink.hdfs.filePrefix = UsingFlumeData

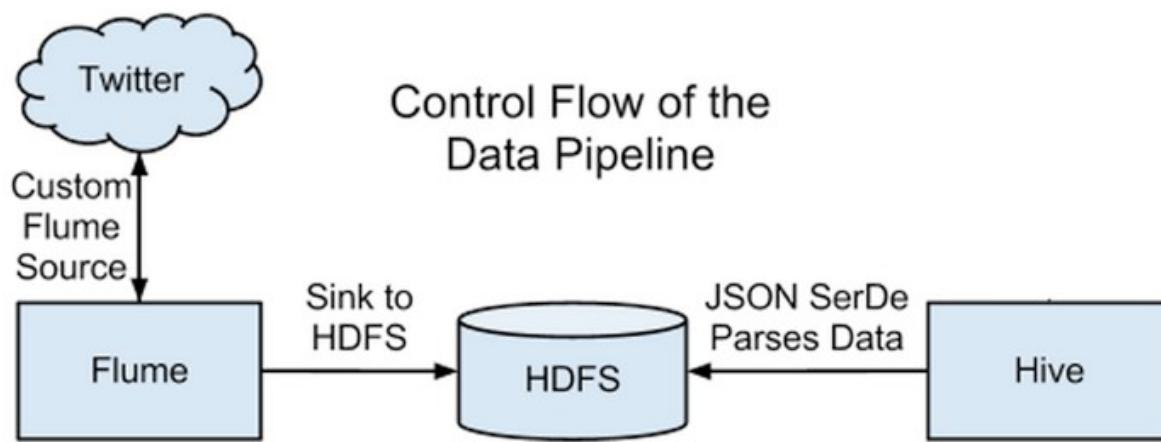
agent.sinks.hbaseSink.type = asynchbase
agent.sinks.hbaseSink.channel = memory2
agent.sinks.hbaseSink.serializer = usingflume.ch05.AsyncHBaseDirectSerializer
agent.sinks.hbaseSink.table = usingFlumeTable
```

Stream Processing Architecture



Hands-On: Loading Twitter Data to Hadoop HDFS

Exercise Overview



Installing Pre-built version of flume

```
$ wget  
http://files.cloudera.com/samples/flume-sources-1.0-SNAPSHOT.jar  
  
$ sudo cp flume-sources-1.0-SNAPSHOT.jar  
/opt/cloudera/parcels/CDH-5.5.1-1.cdh5.5.1.p0.11/lib/flume-  
ng/lib/  
  
$sudo cp /etc/flume-ng/conf/flume-env.sh.template  
/etc/flume-ng/conf/flume-env.sh
```

Create a new Twitter App

Login to your Twitter @ twitter.com

The screenshot shows the Twitter mobile application interface. At the top, there is a navigation bar with five items: Home, Notifications, Messages, Discover, and a search bar labeled "Search Twitter". To the right of the search bar are icons for profile picture, direct message, and refresh.

The main content area displays the "What's happening?" feed. The first tweet is from a user named "ninanews" (@nnanews) posted 1 hour ago. The tweet content is in Thai and mentions "สำนักข่าวเนชั่น" (@nnanews), "นายกฯ", "สั่งห้ามอุบลร่องน้ำเสีย", "ทำบ่อปลาในแม่น้ำป่าสักตามยกระดับ", "พร้อมให้ทบทวน", and "#nna". Below the tweet are interaction icons: reply, retweet, like, and more options.

The second tweet is from "HP OpenNFV" (@hpnfv). The tweet reads: "Where would we be without the carrier networks? Follow @hpnfv to learn more about what's next for telecom." It includes a blue HP logo and a "Follow" button.

The third tweet is from "Pongsuk Hiranprueck" (@nuishow) posted 2 hours ago. The tweet content is in Thai and mentions "Facebook เริ่มทดสอบการเชื่อมต่อระหว่าง WhatsApp กับ Facebook บน Android และ buff.ly/1xULvS9 #beartai". It includes a small profile picture of the user and a "View summary" link.

On the left side of the screen, there is a sidebar for the user's profile, "imcinstitute" (@imcinstitute). It shows the user's profile picture, name, handle, and statistics: 88 tweets, 9 accounts followed, and 23 followers.

Below the sidebar, there is a section titled "Get more from Twitter" with three items: "Sign up" (checkmark), "Follow 5 accounts" (checkmark), and "Complete your profile" (checkmark).

Create a new Twitter App (cont.)

Create a new Twitter App @ apps.twitter.com

The screenshot shows the Twitter Application Management interface. At the top, there's a navigation bar with a Twitter icon and the text "Application Management". On the right side of the bar is a user profile picture and a dropdown arrow. Below the bar, a large blue header bar spans the width of the page. The main content area has a title "Twitter Apps" in large, bold, dark gray font. Underneath the title is a light gray rectangular box containing the text "You don't currently have any Twitter Apps." A red arrow points from the bottom left towards this text. Below the text is a white rectangular button with a thin gray border and the text "Create New App" in a small, dark font.

Create a new Twitter App (cont.)

Enter all the details in the application:

 Application Management



Create an application

Application Details

Name *

IMC_Institute_App 

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

IMC Institute Demo App

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

<http://www.imcinstiute.com>

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.

(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Create a new Twitter App (cont.)

Your application will be created:

The screenshot shows a web browser window with the URL <https://apps.twitter.com/app/8158163> in the address bar. The page title is "Application Management". A green message box at the top states: "Your application has been created. Please take a moment to review and adjust your application's settings." Below this, the application details are listed: "IMC_Institute_App", "IMC Institute Demo App", and "http://www.imcinstiute.com". There are tabs for "Details", "Settings", "Keys and Access Tokens", and "Permissions". A "Test OAuth" button is visible on the right.

IMC_Institute_App

Test OAuth

Details

Settings

Keys and Access Tokens

Permissions



IMC Institute Demo App

<http://www.imcinstiute.com>

Organization

Information about the organization or company associated with your application. This information is optional.

Organization None

Organization website None

Application Settings

Create a new Twitter App (cont.)

Click on Keys and Access Tokens:

Application Management

IMC_Institute_App

Test OAuth

Details Settings **Keys and Access Tokens** Permissions

Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	MjpswndxVj27ylnpOoSBrnfLX
Consumer Secret (API Secret)	QYmuBO1smD5Yc3zE0ZF9ByCgeEQxnxUmhRVCisAvPFudYVjC4a
Access Level	Read and write (modify app permissions)
Owner	imcinstitute
Owner ID	921172807

Create a new Twitter App (cont.)

Your Access token got created:

Your Access Token

This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token **921172807-EfMXJj6as2dFECDH1vDe5goyTHcxPrF1RIJozqgx**

Access Token Secret **HppZEVip3D5j80GP21a37HxA4y10dH9BHcgEFXUNC9xy**

Access Level Read and write

Owner imcinstiute

Owner ID 921172807

Token Actions

[Regenerate My Access Token and Token Secret](#)

[Revoke Token Access](#)

Add classpath in Cloudera Manager

"Services" -> "flume1" -> "Configuration" -> -> "Advanced"
-> "Java Configuration Options for Flume Agent", add:

```
--classpath /opt/cloudera/parcels/CDH-5.5.1-  
1.cdh5.5.1.p0.11/lib/flume-ng/lib/flume-sources-1.0-  
SNAPSHOT.jar
```

The screenshot shows the Cloudera Manager interface for managing services. On the left, there's a sidebar with filters for Non-default (1), Has Overrides (0), SCOPE (Flume (Service-Wide) 1, Agent 8), and CATEGORY (Advanced 9, Flume-NG Solr Sink 3, Logs 4). The main area is titled "Java Configuration Options for Flume Agent" under "Agent Default Group". A configuration entry is visible: "--classpath /opt/cloudera/parcels/CDH-5.5.1-1.cdh5.5.1.p0.11/lib/flume-ng/lib/flume-sources-1.0-SNAPSHOT.jar". Below this, another entry for "HBase sink prefer hbase-site.xml over Zookeeper config" is shown under "Agent Default Group" with a checked checkbox.

Change the Flume Agent Name

cloudera manager

Clusters Hosts Diagnostics Audits Charts Backup Administration

Search (Hotkey: /) Support admin

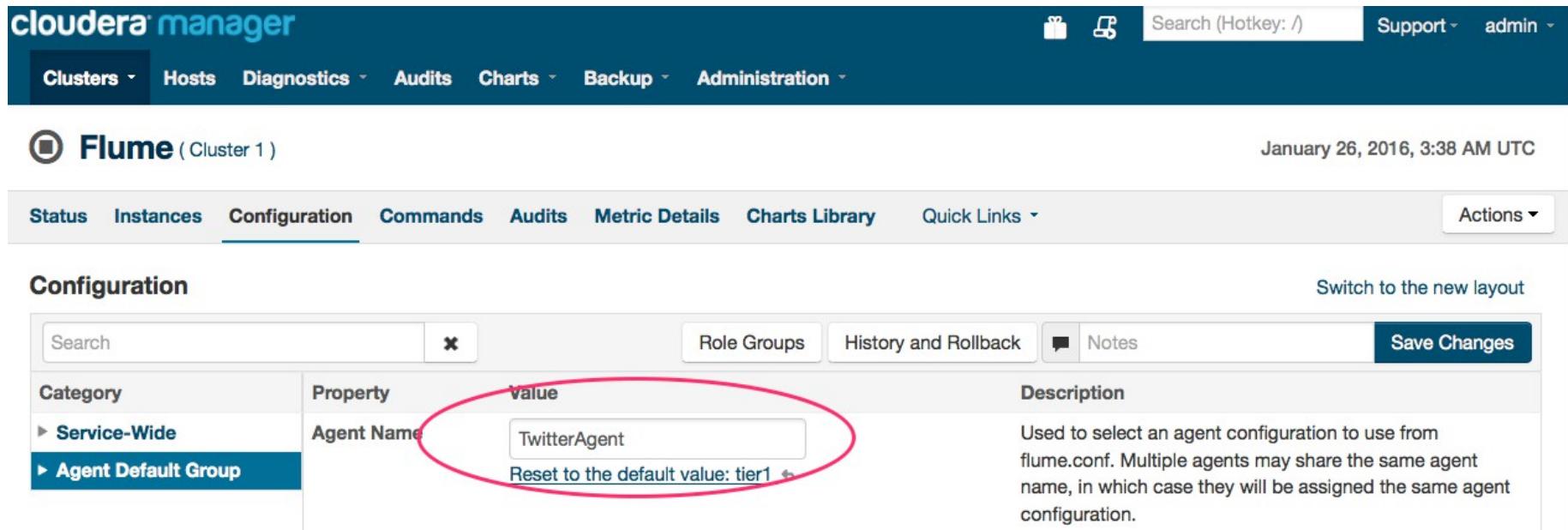
Flume (Cluster 1) January 26, 2016, 3:38 AM UTC

Status Instances Configuration Commands Audits Metric Details Charts Library Quick Links Actions

Switch to the new layout

Category	Property	Value	Description
► Service-Wide	Agent Name	TwitterAgent	Used to select an agent configuration to use from flume.conf. Multiple agents may share the same agent name, in which case they will be assigned the same agent configuration.
► Agent Default Group		Reset to the default value: tier1	

Save Changes



Configuring the Flume Agent

Flume (Cluster 1) January 25, 2016, 6:56 PM UTC

Status Instances Configuration Commands Audits Metric Details Charts Library Quick Links Actions

Configuration

Switch to the new layout

Category	Property	Value	Description
► Service-Wide	Agent Name	TwitterAgent Reset to the default value: tier1	Used to select an agent configuration to use from flume.conf. Multiple agents may share the same agent name, in which case they will be assigned the same agent configuration.
► Agent Default Group	Configuration File	<pre>TwitterAgent.sinks.HDFS.channel = MemChannel TwitterAgent.sinks.HDFS.type = hdfs TwitterAgent.sinks.HDFS.hdfs.path = hdfs://localhost:8020 /user/flume/tweets/</pre>	Verbatim contents of flume.conf. Multiple agents may be configured from the same configuration file; the Agent Name setting can be overridden to select which agent configuration to use for each agent. To integrate with a secured cluster, you can use the substitution strings "\$KERBEROS_PRINCIPAL" and "\$KERBEROS_KEYTAB", which will be replaced by the principal name and the keytab path respectively.

Agent Configuration

```
TwitterAgent.sources = Twitter
```

```
TwitterAgent.channels = MemChannel
```

```
TwitterAgent.sinks = HDFS
```

```
TwitterAgent.sources.Twitter.type =  
org.apache.flume.source.twitter.TwitterSource
```

```
TwitterAgent.sources.Twitter.channels = MemChannel
```

```
TwitterAgent.sources.Twitter.consumerKey =  
MjpswndxVj27y1npOoSBrnfLX
```

```
TwitterAgent.sources.Twitter.consumerSecret =  
QYmuBO1smD5Yc3zE0ZF9ByCgeEQnxUmhRVCisAvPFudYVjC4a
```

```
TwitterAgent.sources.Twitter.accessToken = 921172807-  
EfMXJj6as2dFECDH1vDe5goyTHcxPrF1RIJozqgx
```

```
TwitterAgent.sources.Twitter.accessTokenSecret =  
HbpZEVip3D5j80GP21a37HxA4y10dH9BHcgEFXUNcA9xy
```

Agent Configuration

```
TwitterAgent.sources.Twitter.keywords = hadoop, big data,
analytics, bigdata, cloudera, data science, data scientiest,
business intelligence, mapreduce, data warehouse, data
warehousing, mahout, hbase, nosql, newsql,
businessintelligence, cloudcomputing

TwitterAgent.sinks.HDFS.channel = MemChannel

TwitterAgent.sinks.HDFS.type = hdfs

TwitterAgent.sinks.HDFS.hdfs.path =
hdfs://xx.xx.xx.xx:8020/user/flume/tweets/

TwitterAgent.sinks.HDFS.hdfs.fileType = DataStream

TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text

TwitterAgent.sinks.HDFS.hdfs.batchSize = 1000

TwitterAgent.sinks.HDFS.hdfs.rollSize = 0

TwitterAgent.sinks.HDFS.hdfs.rollCount = 10000

TwitterAgent.channels.MemChannel.type = memory

TwitterAgent.channels.MemChannel.capacity = 10000

TwitterAgent.channels.MemChannel.transactionCapacity = 100
```

Restart Flume

cloudera manager

Clusters Hosts Diagnostics Audits Charts Backup Administration

Search (Hotkey: /) Support admin

Flume (Cluster 1) January 26, 2016, 3:40 AM UTC

Status Instances Configuration Commands Audits Metric Details Charts Library Quick Links Actions

Configuration

Search Agent Name TwitterAgent

Role Groups History and Rollback Notes

Category Property Value Description

Service-Wide Agent Name TwitterAgent Used to select an agent configuration. flume.conf. Multiple agents may have the same agent name, in which case they will be identified by their configuration.

Agent Default Group Configuration File TwitterAgent.sources = Twitter
TwitterAgent.channels = MemChannel Verbatim contents of flume.conf. Configuration settings can be overridden to set different configuration to use for each agent.

Actions: Start, Stop, **Restart** (circled), Rolling Restart, Add Role Instances, Rename, Enter Maintenance Mode, Update Config

View an agent log file

cloudera manager

Clusters Hosts Diagnostics Audits Charts Backup Administration

Agent (Cluster 1 , Flume , ip-10-0-0-96.ec2.internal)

◀ 30 minutes preceding January 26, 2016, 3:45 AM UTC ▶ ⏪ ⏫ ⏪ ⏫

Status Configuration Processes Commands Audits Charts Library Log File ⌂ Stacks Logs Quick Links Actions

Health Tests Create Trigger

Show 7 Good

Heap Dump Directory Free Space Suppress...
Test disabled because role is not configured to dump heap when out of memory. Test of whether this role's heap dump directory has enough free space.

Health History

3:42:12 AM 4 Became Good Show
3:41:25 AM 2 Became Good Show
2:20:25 AM 2 Became Disabled Show
2:19:49 AM 4 Became Disabled Show
2:19:02 AM 4 Became Good Show

Charts

Flume Channel Sizes

percent

03:30

flume-AGENT-88902087059ab421d11e10091e5... 2.4

Health ?

percent

03:30

30m 1h 2h 6h 12h 1d 7d 30d

View an agent log file

Log Details

[Download Full Log](#)

Host [ip-10-0-0-96.ec2.internal](#) [Change...](#) ▾

Role Agent - [Change...](#) ▾

File /var/log/flume-ng/flume-cmf-flume-AGENT-ip-10-0-0-96.ec2.internal.log



January 26, 2016 3:43 AM - January 26, 2016 3:46 AM

Jan 26, 3:44:35.128 AM INFO org.apache.flume.source.twitter.TwitterSource Processed 12,200 docs

Jan 26, 3:44:36.204 AM INFO org.apache.flume.source.twitter.TwitterSource Processed 12,300 docs

Jan 26, 3:44:38.303 AM INFO org.apache.flume.source.twitter.TwitterSource Processed 12,400 docs

View a result using Hue

Screenshot of the Hue File Browser interface showing a list of files in the directory `/user/flume/tweets`.

The URL bar shows the path: `Home / user / flume / tweets`. The "tweets" folder is circled in red.

Name	Size	User	Group	Permissions	Date
..		flume	supergroup	drwxrwxrwx	January 25, 2016 06:06 PM
.		flume	supergroup	drwxrwxrwx	January 25, 2016 06:09 PM
FlumeData.1453773971971	528.0 KB	flume	supergroup	-rw-r--r--	January 25, 2016 06:06 PM
FlumeData.1453774003928	504.7 KB	flume	supergroup	-rw-r--r--	January 25, 2016 06:07 PM
FlumeData.1453774034008	511.9 KB	flume	supergroup	-rw-r--r--	January 25, 2016 06:07 PM
FlumeData.1453774064983	6.8 MB	flume	supergroup	-rw-r--r--	January 25, 2016 06:08 PM
FlumeData.1453774098110	9.9 MB	flume	supergroup	-rw-r--r--	January 25, 2016 06:08 PM
FlumeData.1453774128268	9.9 MB	flume	supergroup	-rw-r--r--	January 25, 2016 06:09 PM
FlumeData.1453774158410.tmp	0 bytes	flume	supergroup	-rw-r--r--	January 25, 2016 06:09 PM

Stop the agent

cloudera manager

Clusters Hosts Diagnostics Audits Charts Backup Administration

Agent (Cluster 1 , Flume , ip-10-0-0-96.ec2.internal)

◀ 30 minutes preceding January 26, 2016, 5:04 AM UTC ▶ ⏪ ⏩ ⏴ ⏵

Status Configuration Processes Commands Audits Charts Library Log File ⌂ Stacks Logs Quick Links Actions

Health Tests Create Trigger

Show 7 Good

Heap Dump Directory Free Space Suppress...
Test disabled because role is not configured to dump heap when out of memory. Test of whether this role's heap dump directory has enough free space.

Health History

5:00 AM Unexpected Exits Good Show
4:55:38 AM Process Status Good Show 1 Still Bad

Charts

Flume Channel Sizes

percent

04:45 05 AM

flume-AGENT-88902087059ab421d11e10091e5... 5.4

Health ?

Start this Agent
Stop this Agent (circled)
Restart this Agent
Enter Maintenance Mode
Update Config
List Open Files (lsof)
Collect Stack Traces (jstack)
Heap Dump (jmap)
Heap Histogram (jmap -histo)

30m 1h 2

7. Analyse data using Hive

Get a Serde Jar File for parsing JSON file

```
$ wget  
http://files.cloudera.com/samples/hive-serdes-1.0-SNAPSHOT.jar  
  
$ mv hive-serdes-1.0-SNAPSHOT.jar /usr/local/apache-hive-  
1.1.0-bin/lib/  
  
$ hive
```

Register the Jar file.

```
hive> ADD JAR /usr/local/apache-hive-1.1.0-bin/lib/hive-  
serdes-1.0-SNAPSHOT.jar;
```

Analyse data using Hive (cont.)

Running the following hive command

```
1 CREATE EXTERNAL TABLE tweets (
2     id BIGINT,
3     created_at STRING,
4     source STRING,
5     favorited BOOLEAN,
6     retweet_count INT,
7    retweeted_status STRUCT<
8         text:STRING,
9         user:STRUCT<screen_name:STRING,name:STRING>>,
10    entities STRUCT<
11        urls:ARRAY<STRUCT<expanded_url:STRING>>,
12        user_mentions:ARRAY<STRUCT<screen_name:STRING,name:STRING>>,
13        hashtags:ARRAY<STRUCT<text:STRING>>>,
14        text STRING,
15        user STRUCT<
16            screen_name:STRING,
17            name:STRING,
18            friends_count:INT,
19            followers_count:INT,
20            statuses_count:INT,
21            verified:BOOLEAN,
22            utc_offset:INT,
23            time_zone:STRING>,
24            in_reply_to_screen_name STRING
25        )
26    ROW FORMAT SERDE 'com.cloudera.hive.serde.JSONSerDe'
27    LOCATION '/user/flume/tweets';
```

<http://www.thecloudavenue.com/2013/03/analyse-tweets-using-flume-hadoop-and.html>

Analyse data using Hive (cont)

Finding user who has the most number of followers

```
hive> select user.screen_name, user.followers_count c from tweets order by c desc;
```

```
Starting Job = job_201504051617_0010, Tracking URL = http://localhost:50030/jobdetails.jsp?jobid=job_201504051617_0010
Kill Command = /usr/local/hadoop/libexec/../bin/hadoop job -kill job_201504051617_0010
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2015-04-06 15:37:27,782 Stage-1 map = 0%, reduce = 0%
2015-04-06 15:37:31,837 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.27 sec
2015-04-06 15:37:39,899 Stage-1 map = 100%, reduce = 33%, Cumulative CPU 1.27 sec
2015-04-06 15:37:40,908 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 2.42 sec
MapReduce Total cumulative CPU time: 2 seconds 420 msec
Ended Job = job_201504051617_0010
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 2.42 sec HDFS Read: 170686 HDFS Write: 687 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 420 msec
OK
vinnaum 11523
navchatterji 5485
HCITExpert 4751
NWDSCScoop 4097
7wdata 3005
MotivasiMariaP 2007
WesleyBackelant 1977
IFTTTMarketing 1307
jonathangibs 968
ephraimcohen 914
feshob 716
DKajouri 713
```

Apache Kafka

Dr.Thanachart Numnonda
IMC Institute
thanachart@imcinstitute.com

Introduction

Open-source message broker project



An open-source message broker project developed by the Apache Software Foundation written in Scala. The project aims to provide a unified, high-throughput, low-latency platform for handling real-time data feeds. It is, in its essence, a "massively scalable pub/sub message queue architected as a distributed transaction log", making it highly valuable for enterprise infrastructures.

What is Kafka?

- An apache project initially developed at LinkedIn
- Distributed publish-subscribe messaging system
- Designed for processing of real time activity stream data
e.g. logs, metrics collections
- Written in Scala
- Does not follow JMS Standards, neither uses JMS APIs

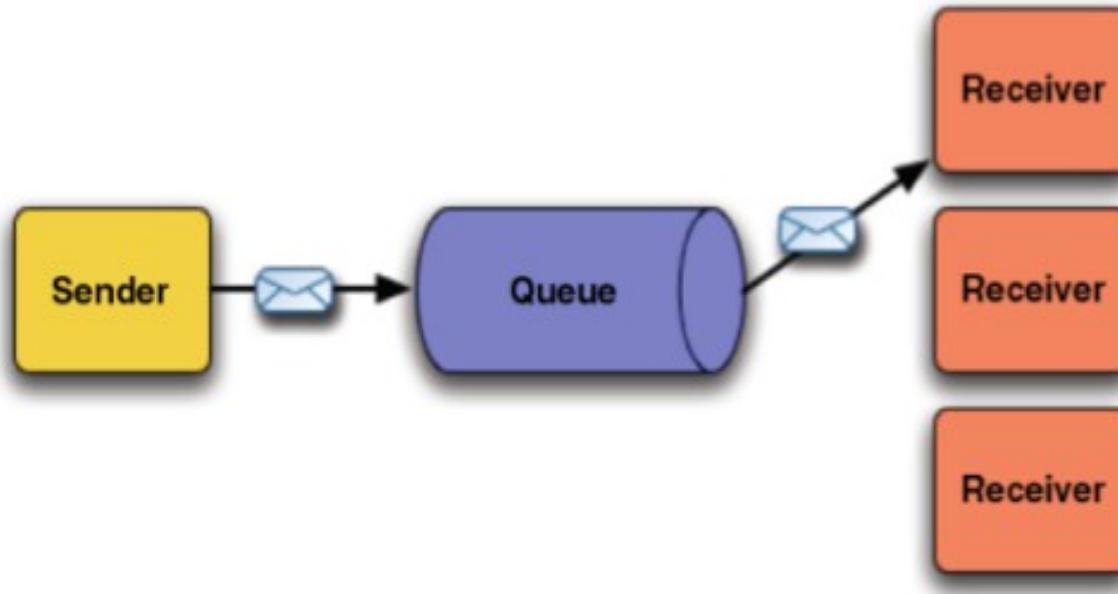
Kafka: Features

- Persistent messaging
- High-throughput
- Supports both queue and topic semantics
- Uses Zookeeper for forming a cluster of nodes (producer/consumer/broker)
- and many more...

Why Kafka?

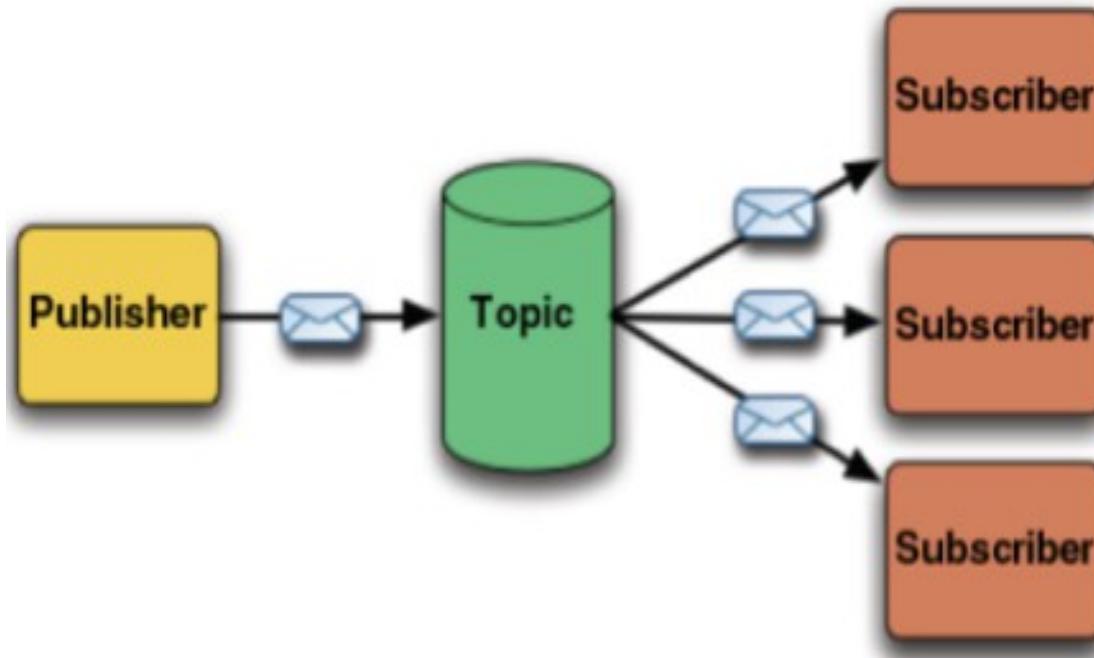
- Built with speed and scalability in mind.
- Enabled near real-time access to any data source
- Empowered hadoop jobs
- Allowed us to build real-time analytics
- Vastly improved our site monitoring and alerting capability
- Enabled us to visualize and track our call graphs.

Messaging System Concept: Queue



Source: Real time Analytics with Apache Kafka and Spark, Rahul Jain

Messaging System Concept: Topic

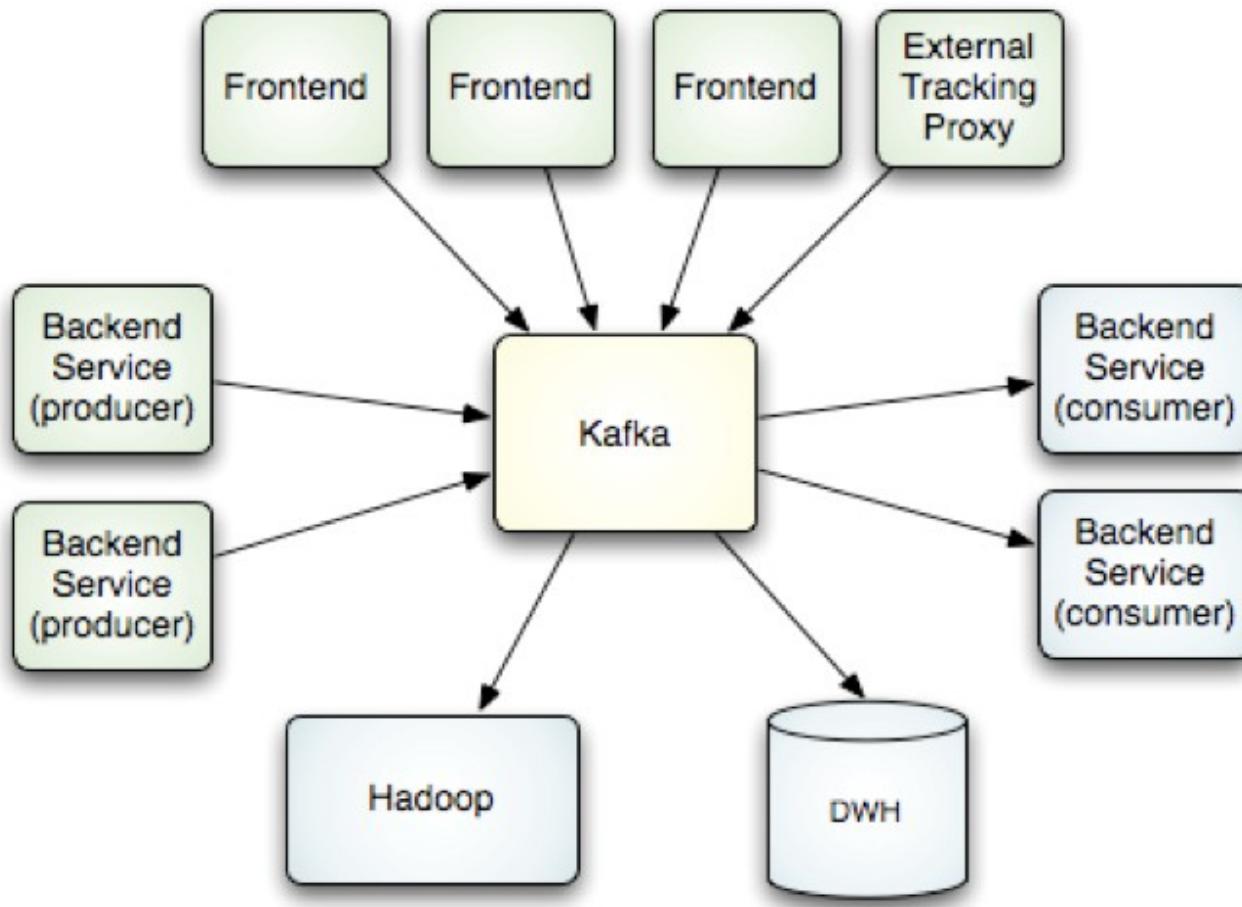


Source: Real time Analytics with Apache Kafka and Spark, Rahul Jain

Terminology

- Kafka maintains feeds of messages in categories called topics.
- Processes that publish messages to a Kafka topic are called producers.
- Processes that subscribe to topics and process the feed of published messages are called consumers.
- Kafka is run as a cluster comprised of one or more servers each of which is called a broker.

Kafka

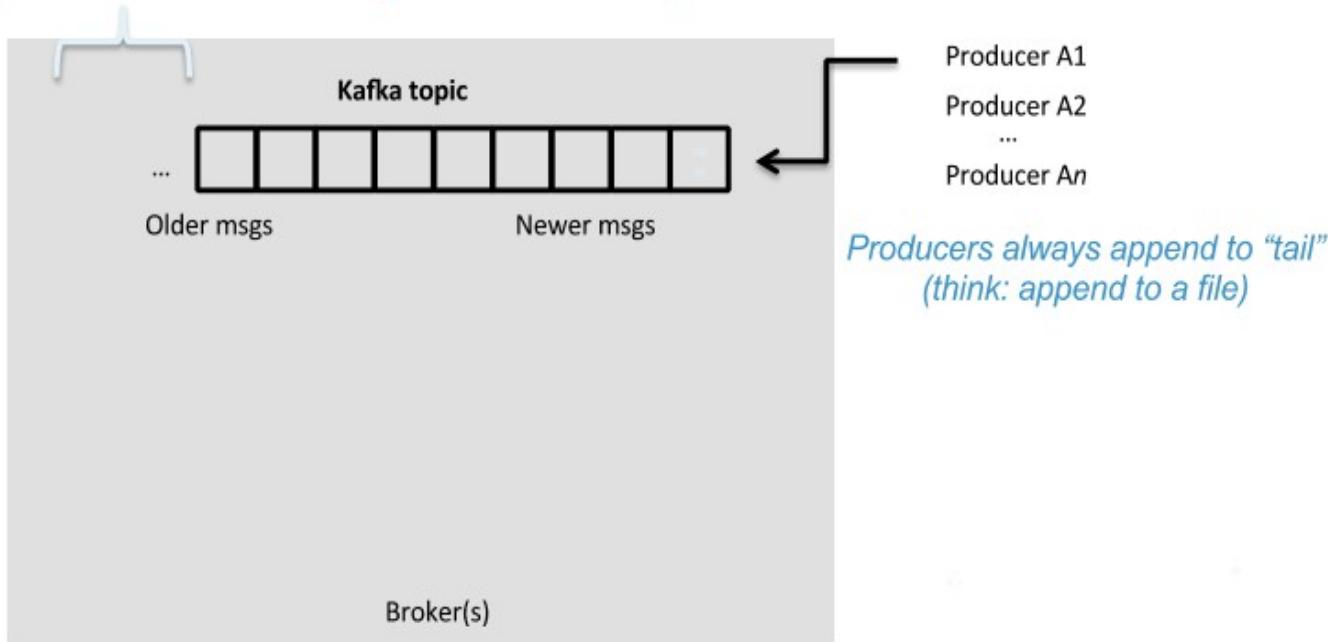


Source: Real time Analytics with Apache Kafka and Spark, Rahul Jain

Topics

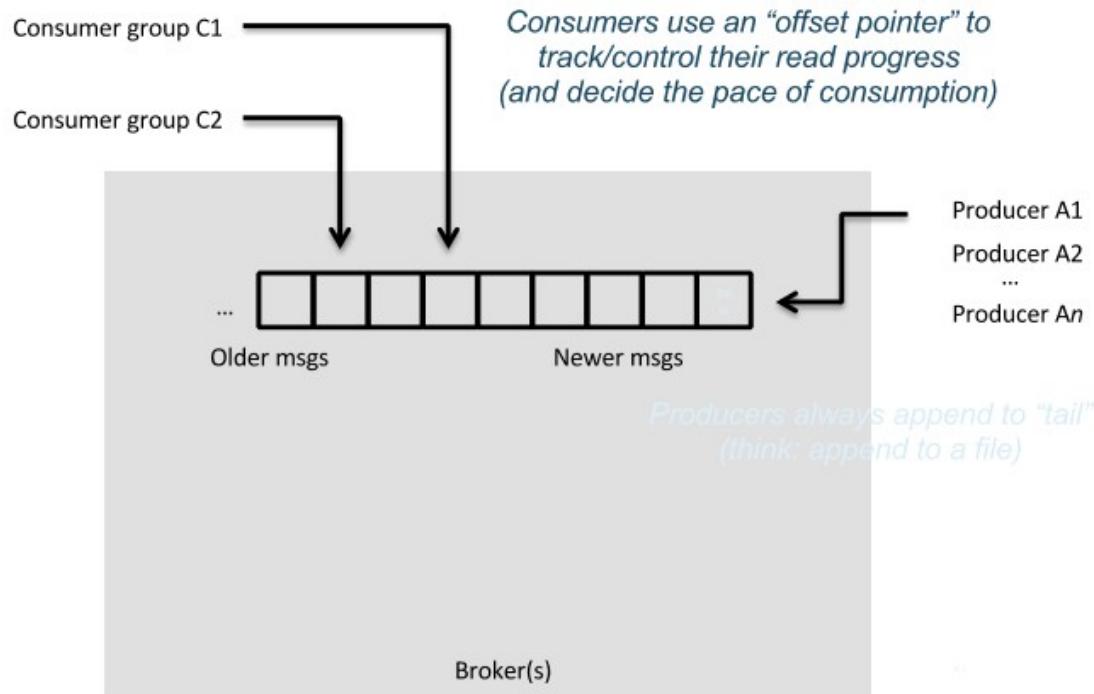
- Topic: feed name to which messages are published

Kafka prunes “head” based on age or max size or “key”



Source: Apache Kafka with Spark Streaming - Real Time Analytics Redefined

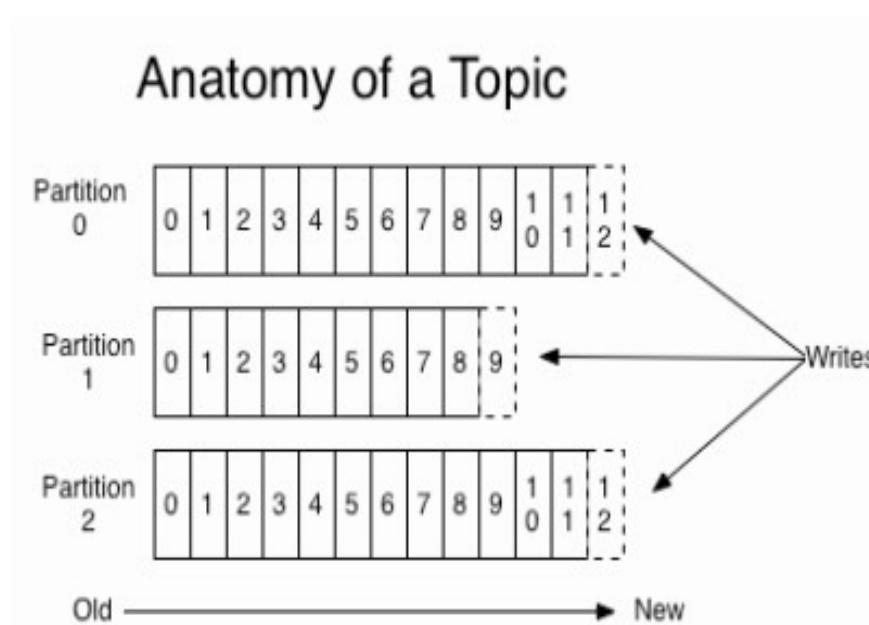
Topics



Source: Apache Kafka with Spark Streaming - Real Time Analytics Redefined

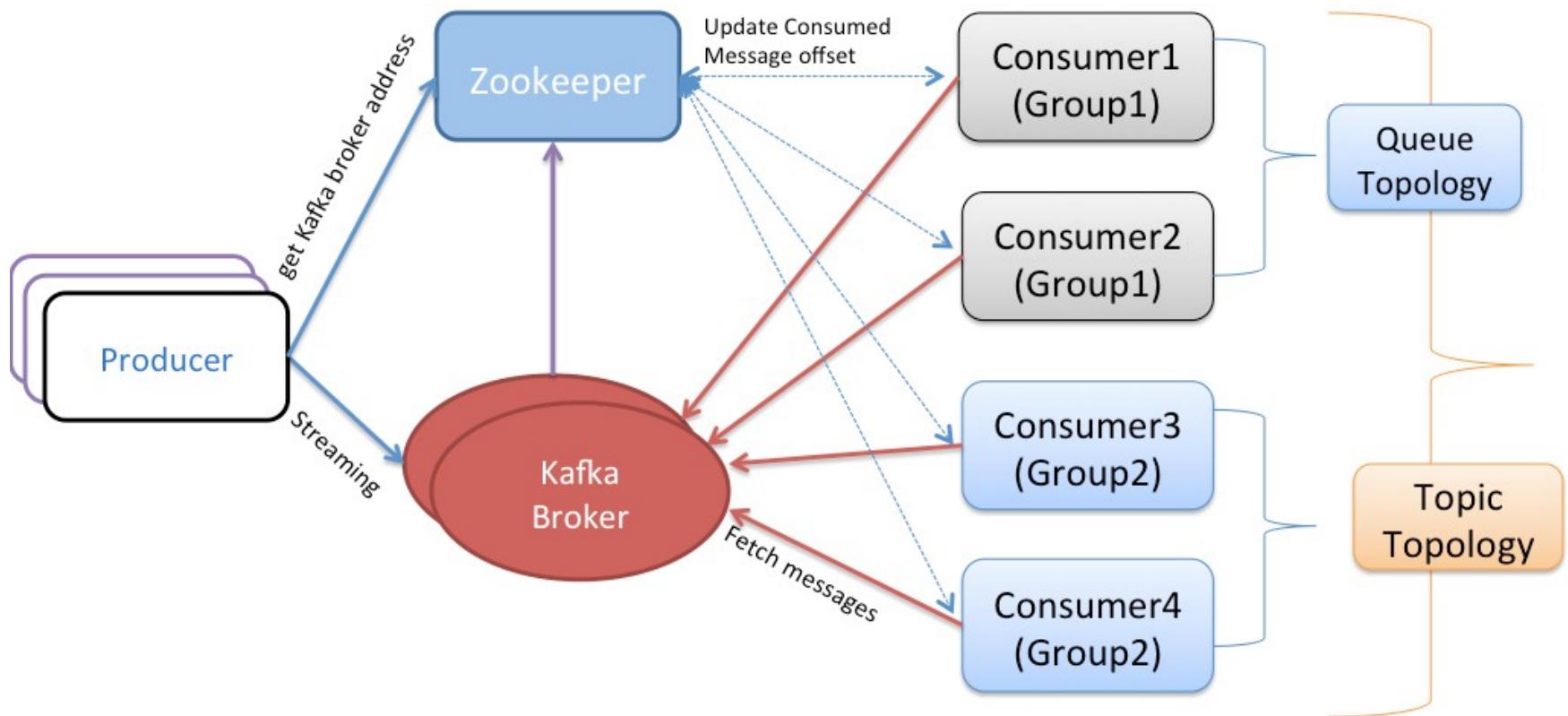
Topics

- A topic consists of partitions.
- Partition: ordered + immutable sequence of messages that is continually appended



Source: Apache Kafka with Spark Streaming - Real Time Analytics Redefined

Kafka Architecture



Source: Real time Analytics with Apache Kafka and Spark, Rahul Jain

Hands-on

SparkStreaming with Kafka

Install & Start Kafka Server

```
# wget http://www-us.apache.org/dist/kafka/0.9.0.1/kafka_2.10-  
0.9.0.1.tgz  
# tar xzf kafka_2.10-0.9.0.1.tgz  
# cd kafka_2.10-0.9.0.1  
# bin/kafka-server-start.sh config/server.properties&
```

```
[2016-06-23 04:37:21,426] INFO Kafka commitId : 23c69d62a0cabf06 (o  
rg.apache.kafka.common.utils.AppInfoParser)  
[2016-06-23 04:37:21,430] INFO [Kafka Server 0], started (kafka.ser  
ver.KafkaServer)  
[2016-06-23 04:37:21,446] INFO New leader is 0 (kafka.server.Zookeee  
perLeaderElector$LeaderChangeListener)
```

Running Kafka Producer

```
# bin/kafka-console-producer.sh --topic test --broker-list  
localhost:9092
```

type some random messages followed by Ctrl-D to finish

```
[root@quickstart kafka_2.10-0.9.0.1]# bin/kafka-console-producer.sh  
--topic test --broker-list localhost:9092  
This is a test message from IMC Institute
```

Big Data School

Test

```
[root@quickstart kafka_2.10-0.9.0.1]# █
```

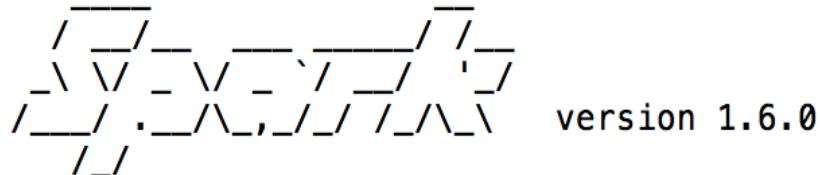
Running Kafka Consumer

```
# bin/kafka-console-consumer.sh --topic test --zookeeper localhost:2181 --from-beginning
```

```
[root@quickstart kafka_2.10-0.9.0.1]# bin/kafka-console-consumer.sh --topic test --zookeeper localhost:2181 --from-beginning
This is a test message from IMC Institute
Big Data School
Test
```

Start Spark-shell with extra memory

```
[root@quickstart ~]# spark-shell --driver-memory 1G
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/zookeeper/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/jars/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel).
Welcome to
```



Spark Streaming with Kafka

```
$ scala> :paste
import org.apache.spark.SparkConf
import org.apache.spark.streaming.{Seconds, StreamingContext}
import org.apache.spark.storage.StorageLevel
import StorageLevel._
import org.apache.spark._
import org.apache.spark.streaming._
import org.apache.spark.streaming.StreamingContext._
import org.apache.spark.streaming.kafka.KafkaUtils
val ssc = new StreamingContext(sc, Seconds(2))
val kafkaStream = KafkaUtils.createStream(ssc,
"localhost:2181","spark-streaming-consumer-group", Map("spark-
topic" -> 5))
kafkaStream.print()
ssc.start
```

Running Kafka Producer on another terminal

```
# docker ps
```

CONTAINER ID	IMAGE	COMMAND
CREATED	STATUS	PORTS
NAMES		
c77e4dc1ed9b	cloudera/quickstart:latest	"/usr/bin/docker-q
ui 22 minutes ago	Up 22 minutes	0.0.0.0:8888->8888/tcp
trusting_newton		

```
# docker exec -i -t c77e4dc1ed9b /bin/bash
```

```
[root@quickstart ~]# cd /root/kafka_2.10-0.9.0.1
[root@quickstart kafka_2.10-0.9.0.1]# bin/kafka-console-
producer.sh --broker-list localhost:9092 --topic spark-topic
```

Test & View the result

```
[root@quickstart kafka_2.10-0.9.0.1]# bin/kafka-console-producer.sh  
--broker-list localhost:9092 --topic spark-topic  
Hello from IMC Institute
```

Result from another terminal

```
85200 replicated to only 0 peer(s) instead of 1 peers  
-----  
Time: 1466658086000 ms  
-----  
(null,Hello from IMC Institute)
```

<http://www.imcinstitute.com/bigdatacert> Tel. 088-192-7975

Big Data Certification Course

120 Hrs

Start on 15 September 2016



Thank you

www.imcinstitute.com
www.facebook.com/imcinstitute