

Siraj Bagwan

Project on Big Data Platform Engineering

Challenges

- Improve click-through rates

Improving click-through rate (the number of clicks that your content receives divided by the number of times your content is shown.) by recommending content to relevant users.

- Personalize online ads and content

Showing more relevant ads and content basis on their past interaction to online users to improve user experience.

- Lead acquisition

By improving user experience and interaction can be able to raise the ratio of subscription.

Solution

- Collect

Collect and gather data of user behavior and interaction and store it to data lake from different sources in order to process.

- Process

Transform the click stream data on data lake and ingest it to data warehouse. Ingest the data from OLTP system to Data Warehouse.

- Analyze

Analyze the data present on the cluster using different tools available in order to generate insights.

- Data retention period: 1 year
- Total data collected per year: 250 TB
- Data to be stored on each data node: 50 TB (10 TB * 6)
- No. of data nodes required: $250/50 = 5$
- 10% overhead (node failure): 1
- Total data nodes required: 6
- Kafka nodes: 3

Cluster Planning

Hosts	No. of Hosts Required	Specification
Master Hosts	3	Instance Type: r6a.4xlarge Ram: 128 GB Core: 16
Utility Hosts	2	Instance Type: r6a.4xlarge Ram: 128 GB Core: 16
Edge hosts	1	Instance Type: c6a.8xlarge Ram: 64 GB Core: 32
Worker Hosts	6	Instance Type: c5.12xlarge Ram: 96 GB Core: 48 HDD: 10 TB * 6

Cluster Planning

Nodes	Services
Master Node 1:	NN, JN, Failover Controller, Zookeeper Resource Manager,
Master Node 2:	Standby NN, JN, Failover Controller, Standby Resource Manager, Zookeeper
Master Node 3:	JN, Zookeeper, JHS,SHS
Utility Node 1:	Cloudera Manager
Utility Node 2:	HMS,HS2,ICS,SS,
Edge Node:	Gateway of HDFS, YARN, HIVE. HUE, OOZIE
Data Nodes:	DN, NM, ID
3 Kafka Nodes:	Kafka Brokers

Cluster Planning

- Block size of HDFS: 128 MB
- 1 MB fsimage size for per 1000 blocks (Suggested by Cloudera)
- $250 \text{ TB} = 256000 \text{ GB} = 262,144,000 \text{ MB}$
- Block Size = 128 MB
- Total No. of Blocks = $262144000 / 128 = 2,048,000$
- Fsimage = $2,048,000 / 1000 = 2 \text{ GB}$
- Heap Size of name node = 2 GB = 4GB (Cloudera Suggest At least 4 GB)
- HDD: size 10 TB * 06 disks = 60TB

Worker Host Configuration

STEP 1: Worker Host Configuration

Enter your likely machine configuration in the input boxes below. If you are uncertain what machines you plan on buying, put in some minimum values that will suit what you expect to buy.

Host Components	Quantity	Size	Total	Description
RAM	96G		96G	Node memory
CPU	24	1	48	Number of CP
HyperThreading CPU	yes			Does the CPU
HDD (Hard Disk Drive)	6	10T	60G	Number of Ha
Ethernet	1	10G	10G	Number of Et

Worker Host Planning

STEP 2: Worker Host Planning

Now that you have your base Host configuration from Step 1, use the table below to allocate resources, mainly CPU and memory, to the various software components that run on the host.

		CPU (cores)	Memory (MB)	Notes
Service	Category			
Operating System	Overhead	1	8192	Most operating system overhead
Other services	Overhead	0	0	Enter the required resources for other services
Cloudera Manager agent	Overhead	1	1024	Allocate 1GB for Cloudera Manager agent
HDFS DataNode	CDH	1	2048	Allocation for HDFS DataNode
YARN NodeManager	CDH	1	2048	Allocation for YARN NodeManager
Impala daemon	CDH	1	16384	(Optional Ser
Hbase RegionServer	CDH	0	0	(Optional Ser
Solr Server	CDH	0	0	(Optional Ser
Kudu Server	CDH	0	0	(Optional Ser
Available Container Resources		43	68608	
Container resources				
Physical Cores to Vcores Multiplier		1		Set this ratio
YARN Available Vcores		43		This value wil
YARN Available Memory			68608	This value wil

STEP 3: Cluster Size

Enter the number of nodes you have (or expect to have) in the cluster

		Quantity			
Number of Worker Hosts in the cluster		6			

Yarn Tuning

STEP 4: YARN Configuration on Cluster

These are the first set of configuration values for your cluster. You can set these values in YARN->Configuration

YARN NodeManager Configuration Properties	Value	Note
yarn.nodemanager.resource.cpu-vcores	43	Copied from S
yarn.nodemanager.resource.memory-mb	68608	Copied from S

STEP 5: Verify YARN Settings on Cluster

Go to the Resource Manager Web UI (usually <http://<ResourceManagerIP>:8088/> and verify the "Memory Total" and "Vcores Total" matches the values above. If your machine has no bad nodes, then the numbers should match exactly.

Resource Manager Property to Check	Value	Note
Expected Value for "Vcores Total"	258	Calculated from
Expected Value for "Memory Total" (in GB)	402	Calculated from

STEP 6: Verify Container Settings on Cluster

In order to have YARN jobs run cleanly, you need to configure the container properties.

YARN Container Configuration Properties (Vcores)	Value	Description
yarn.scheduler.minimum-allocation-vcores	1	Minimum vco
yarn.scheduler.maximum-allocation-vcores	43	Maximum vcc
yarn.scheduler.increment-allocation-vcores	1	Vcore allocati

YARN Container Configuration Properties (Memory)	Value	Description
yarn.scheduler.minimum-allocation-mb	1024	Minimum mei
yarn.scheduler.maximum-allocation-mb	68608	Maximum me
yarn.scheduler.increment-allocation-mb	512	Memory alloc

Cluster Container Capacity

Step 6A: Cluster Container Capacity

This section will tell you the capacity of your cluster (in terms of containers).

Cluster Container Estimates	Minimum	Maximum
Max possible number of containers, based on memory configuration		402
Max possible number of containers, based on vcore configuration		258
Container number based on 2 containers per disk spindles		72
Min possible number of containers, based on memory configuration	6	
Min possible number of containers, based on vcore configuration	6	

STEP 6B: Container Sanity Checking

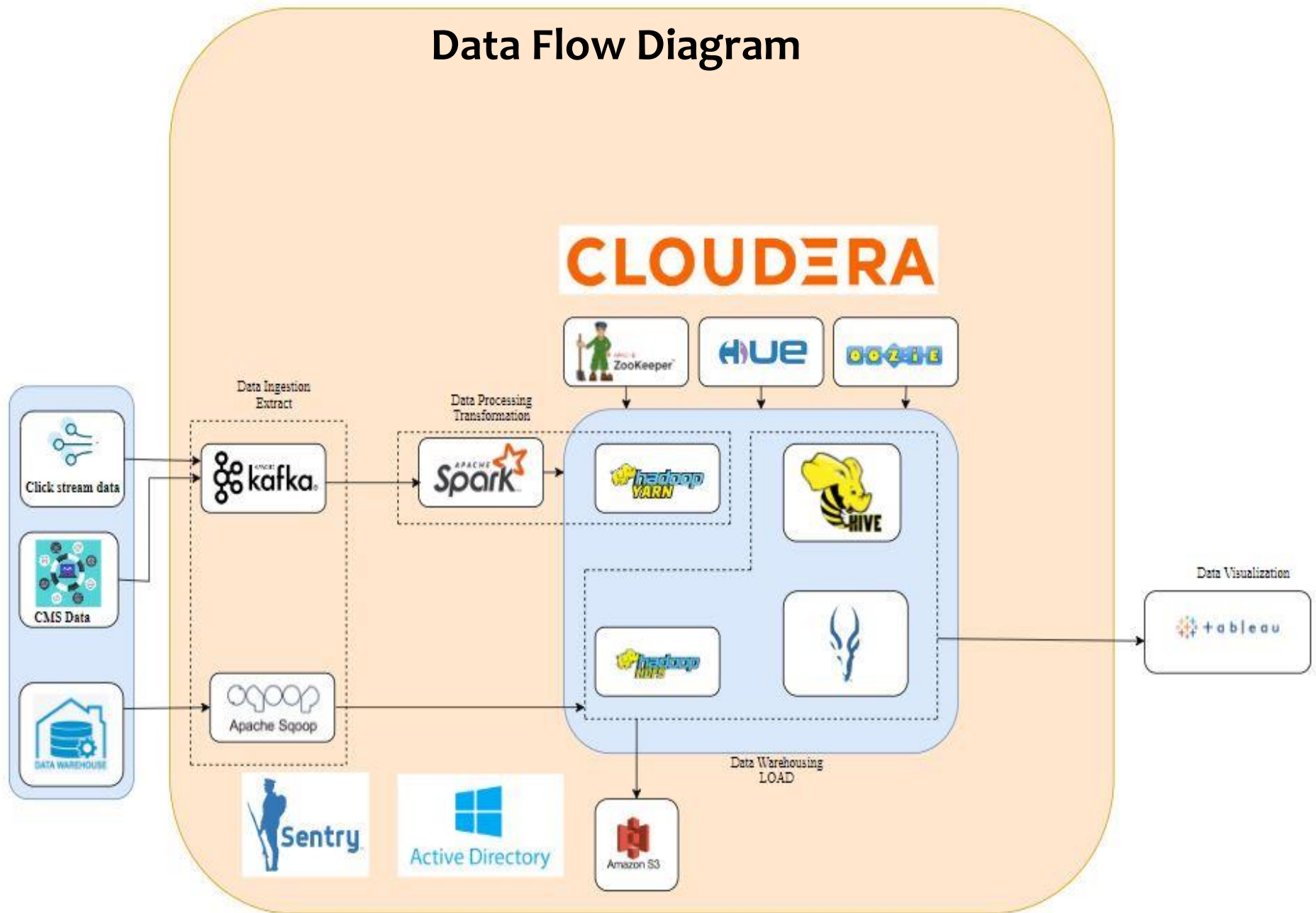
This section will do some basic checking of your container parameters in STEP 6 against the hosts.

	Check	
Sanity Check	Status	Description
Scheduler maximum vcores must be larger than minimum	GOOD	yarn.schedule
Scheduler maximum allocation MB must be larger than minimum	GOOD	yarn.schedule
Scheduler minimum vcores must be greater than or equal to 0	GOOD	yarn.schedule
Scheduler maximum vcores must be greater than or equal to 1	GOOD	yarn.schedule
Host vcores must be larger than scheduler minimum vcores	GOOD	yarn.nodema
Host vcores must be larger than scheduler maximum vcores	GOOD	yarn.nodemar
Host allocation MB must be larger than scheduler minimum	GOOD	yarn.nodemar
Host allocation MB must be larger than scheduler maximum	GOOD	yarn.nodemar
Small container limit	GOOD	If yarn.schedu

Service Stack

• Services	Versions
• Hadoop	3.0.0
• Kafka	2.1.0
• Sqoop	1.4.7
• Spark	2.4.0
• Hive	2.1.1
• Impala	3.2.0
• Hue	4.3.0
• Oozie	5.1.0
• Zookeeper	3.4.5
• Sentry	2.1.0

Data Flow Diagram





✓ Production ▾

CDH 6.2.0 (Parcels)

✓ 14 Hosts

✓ HDFS ▾

✓ Hive ▾

✓ Hue ▾

✓ Impala ▾

✓ Kafka ▾

✓ Oozie ▾

S3 Connector ▾

✓ Sentry ▴

✓ Spark ▴

Sqoop 1 Client ▾

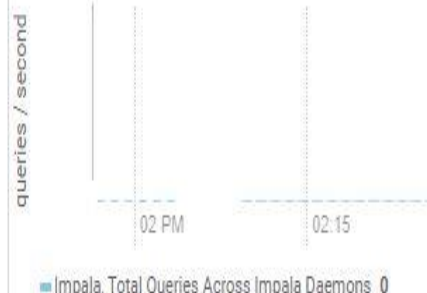
✓ YARN (MR2 In... ▴

✓ ZooKeeper ▴

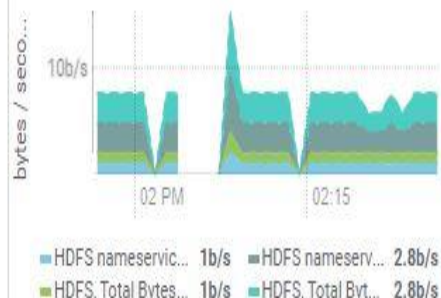
Charts

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

Completed Impala Queries



HDFS IO



Cluster Network IO



Cluster Disk IO



Cluster CPU



Cloudera Manager Web UI

Activate Windows
Go to Settings to activate Windows.

Roles

Roles Placement

Hosts	Count	Roles
ip-10-0-0-105.ap-south-1.compute.internal	1	FC JN NN RM S
ip-10-0-0-113.ap-south-1.compute.internal	1	G G LB HS KTR G OS G G G G
ip-10-0-0-120.ap-south-1.compute.internal	1	JN G SS HS JHS S
ip-10-0-0-121.ap-south-1.compute.internal	1	B FC JN NN RM S
ip-10-0-0-122.ap-south-1.compute.internal	1	AP ES HM RM SM
ip-10-0-0-89.ap-south-1.compute.internal	1	G HMS HS2 ICS ISS G
ip-10-0-0-[69, 82, 84, 104, 115].ap-south-1.compute.internal	5	DN ID NM
ip-10-0-0-[75, 77, 88].ap-south-1.compute.internal	3	KB

This table is grouped by hosts having the same roles assigned to them.

Improvements to be done: Could Be better by adding one more utility host and configuring high availability for Hive, oozie, Hue, Sentry

Security

StatusKerberos Credentials

- TLS Settings
- Security Inspector

Cluster			
Production	Successfully enabled Kerberos.	HDFS Data At Rest Encryption is disabled	Set up HDFS Data At Rest Encryption

Authentication Mechanism Enable

Active Directory Users and Computers

File Action View Help



- Active Directory Users and Com
- ▶ Saved Queries
- ▶ hadoopsecurity.local
 - ▶ Builtin
 - ▶ Computers
 - ▶ Domain Controllers
 - ▶ ForeignSecurityPrincipal
 - hadoop
 - ▶ Managed Service Account
 - ▶ Users

Name	Type	Description
advbXJwGCd	User	
AgLxLLJfdD	User	
cloudera ma...	User	
DzfQTACzBW	User	
ezDGvAGEWQ	User	
fqDDijLzkY	User	
hTMnOcpvnm	User	
hwZIDmKW...	User	
iFbvyjhbJr	User	
iHITOXzjzY	User	
JrcEJbuNRc	User	
KESgpkKVgf	User	
IsFoLYfYIC	User	
NANzPrfdBQ	User	
NlkgJulrvl	User	
PBgFidKrRU	User	
PCEASjumll	User	
piXRQskUfK	User	
pSDtZeaboZ	User	
qRNnYoMDGJ	User	
QvgPrtzbBz	User	

Integration with active directory:- two way trust

Instances (17) [Info](#)



Connect

















































Instance state

Actions

Launch instances

Q Search

< 1 > 

<input type="checkbox"/>	Name ▾	Instance ID	Instance state ▾	Status check	Alarm status	Availability Zone ▾	IPv6 IPs
<input type="checkbox"/>	DATABASE	i-06ec286a4f2c597d2	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	UN1-CM	i-0276dc5f2cc0a738d	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	UN2	i-0bb5c4d9d3a1408a6	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	MN1	i-04d246abdf9479ba	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	MN2	i-0d5c57448c2b6d03b	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	MN3	i-0cd8e75eb8ba9704d	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	EN	i-07a37ae78cf37e359	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	DN1	i-0020f878a0eda88da	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	DN2	i-0d432f97a65177bff	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	DN3	i-08b0ae299ca7cc955	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	DN4	i-0e6c488b527d7f9b1	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-
<input type="checkbox"/>	DN5	i-02bhf90f5508a154d	 Running 	 2/2 checks passed	No alarms 	ap-south-1b	-

What can be better: Can achieve High Fault tolerance if master nodes and utility nodes (mn,un) deployed in different AZ's, And HA for Gateway or edge node (EG).



THANK YOU