AWS re:Invent

BDT305

Lessons Learned and Best Practices for Running Hadoop on AWS

Amandeep Khurana

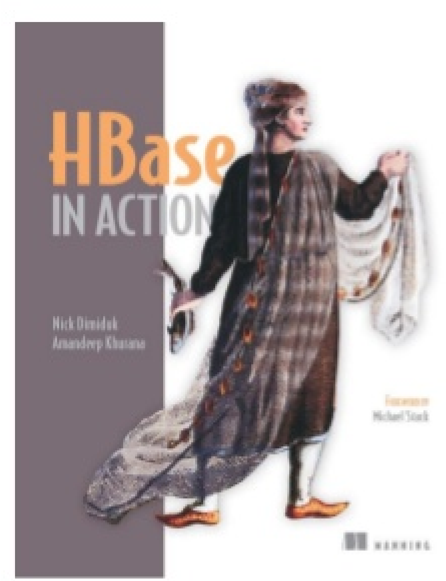
November 13, 2014 | Las Vegas, NV

amazon web services

cloudera

About me

- Principal Solutions Architect @ Cloudera
- Engineer @ AWS
- Co-author, HBase in Action



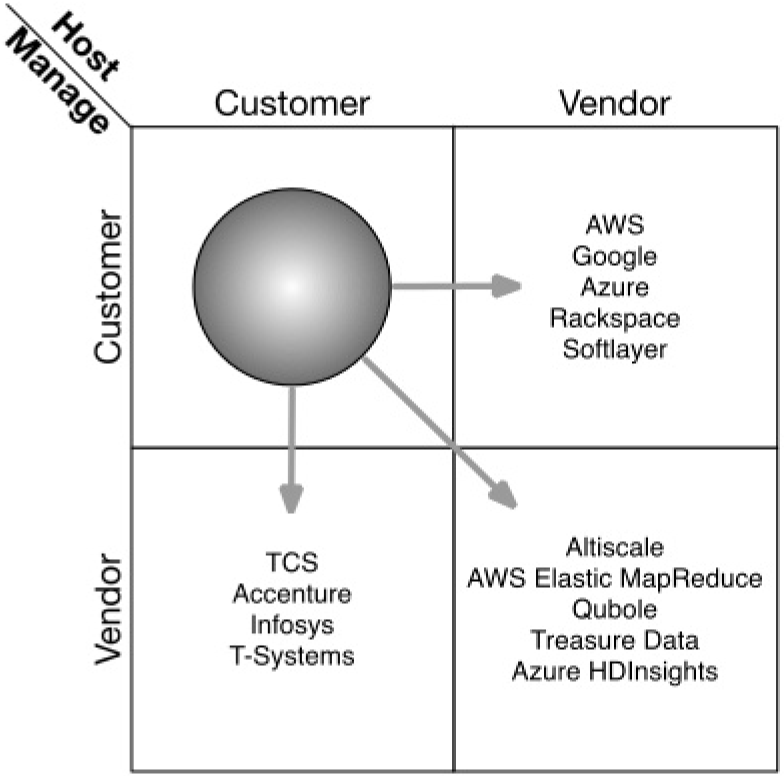
Agenda

- Motivation
- Deployment paradigms
- Storage
- Networking
- Instances
- Security
- High availability, backups, disaster recovery
- Planning your cluster
- Available resources

Why you should care

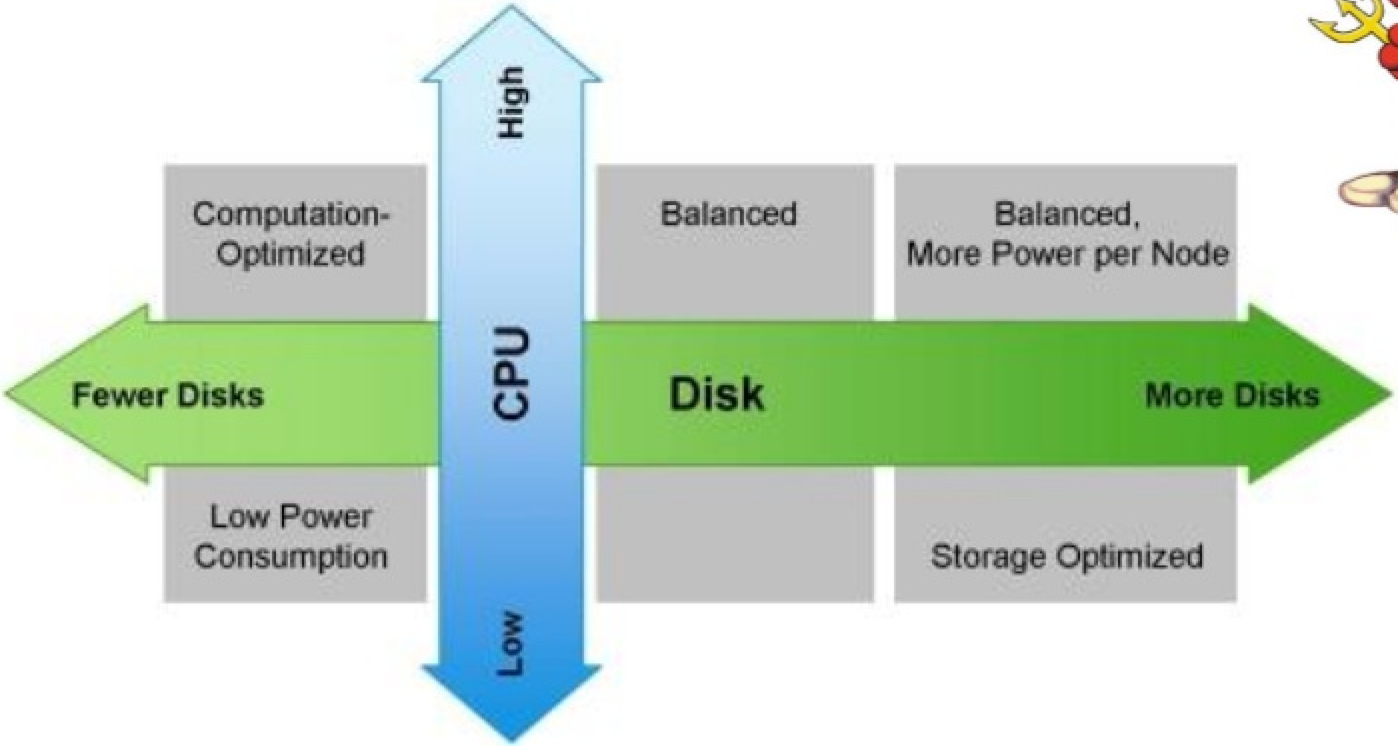
- Parallel trends
  - Commoditizing infrastructure
  - Commoditizing data
- Worlds converging… but with considerations
  - Cost
  - Flexibility
  - Ease of use
  - Operations
  - Location
  - Performance
  - Security

Intersection



Host Manage	Customer	Vendor
Customer		AWS Google Azure Rackspace Softlayer
Vendor	TCS Accenture Infosys T-Systems	Altiscale AWS Elastic MapReduce Qubole Treasure Data Azure HDInsights

The devil…



Computation-Optimized

Balanced

Balanced,
More Power per Node

Low Power Consumption

Storage Optimized

Primary consideration – Storage (source of truth)

Amazon S3

- Ad-hoc batch workloads
- SLA batch workloads

Predominantly transient clusters

HDFS

- Ad-hoc batch workloads
- SLA batch workloads
- Ad-hoc interactive workloads
- SLA interactive workloads

Long running clusters

Deployment models

	Transient clusters	Long-running clusters
Primary storage substrate	S3 or remote HDFS	HDFS
Backups	S3	S3 or second HDFS cluster
Workloads	• Batch (MapReduce, Spark) • Interactive is an anti-pattern	• Batch (MapReduce, Spark) • Interactive (HBase, Solr, Impala)
Role of cluster	Compute only	Compute and storage

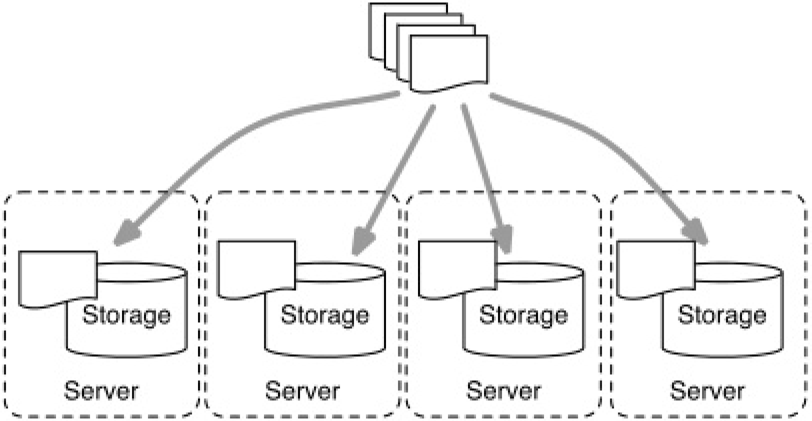
Storage

Access pattern, performance

Storage considerations

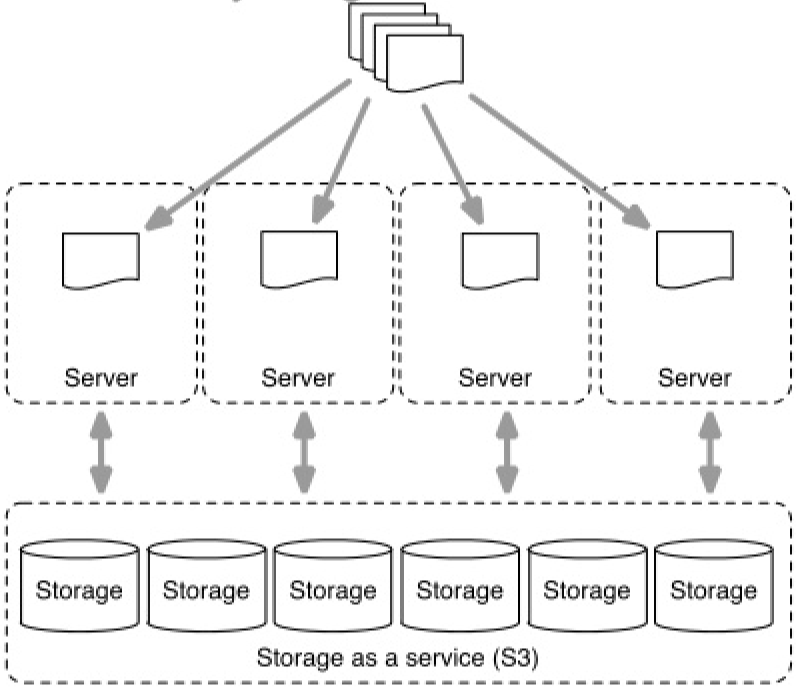
Hadoop paradigm:

Bring compute to storage



Cloud paradigm:

Everything as a service



Storage choices in AWS

- Instance store
  - Local storage attached to instance
  - Temporary
  - Instance dependent (not configurable)
- Amazon Elastic Block Store (EBS) - Block-level storage volume
  - External to instance
  - Lifecycle independent of instance
- Amazon Simple Storage Service (S3) – BLOB store
  - External data store
  - Simple API – Get, Put, Delete
  - Instance dependent bandwidth

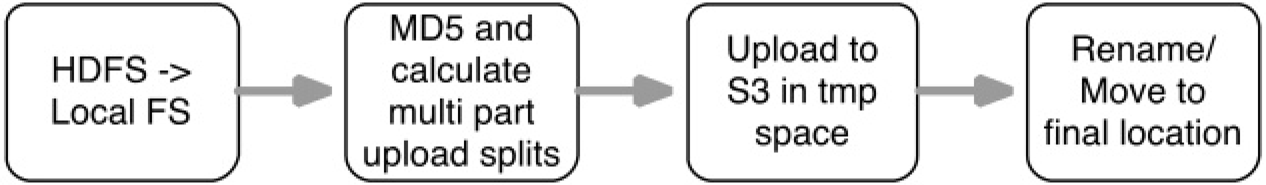
Interacting with S3

- In MapReduce jobs by using s3a URI
- Distcp
  - `hadoop distcp <options> hdfs:///foo/bar s3a:///mybucket/foo/`
- HBase snapshot export

  -
    ```
    hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot
        <options> -Dmapred.task.timeout=15000000
        -snapshot <name> -mappers <nmappers> -copy-to <dir>
    ```

Interacting with S3 – how it works

- Multiple implementations in the Hadoop project
  - S3 (block based)
  - S3N (file based, using jets3t)
  - S3A (file based, using AWS SDK) ←Latest stuff
- Bandwidth to S3 depends on instance type
  - <200 MB/s per instance on some of the larger ones
- Process


HDFS -> Local FS
MD5 and calculate multi part upload splits
Upload to S3 in tmp space
Rename/ Move to final location

Optimizing S3 interaction

- Tune
- Parallelize
- Writing to S3
  - Multi-part upload for > 5 GB files
  - Pick multiple drives for local staging (HADOOP-10610)
  - Up the task timeouts when writing large files
- Reading from S3
  - Range reads within map tasks via multiple threads
- Large objects are better (less load on metadata lookups)
- Randomize file names (metadata lookups are spread out)

HDFS in AWS

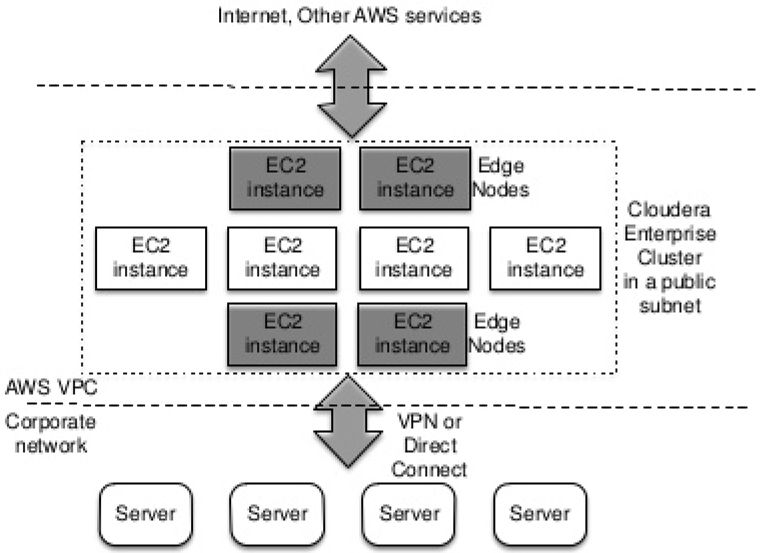
- Ephemeral drives on Amazon EC2 instances
- Persistent for as long as the instances are alive (no pausing)
- Use S3 for backups
- No EBS
  - Over the network
  - Designed for random I/O

Networking

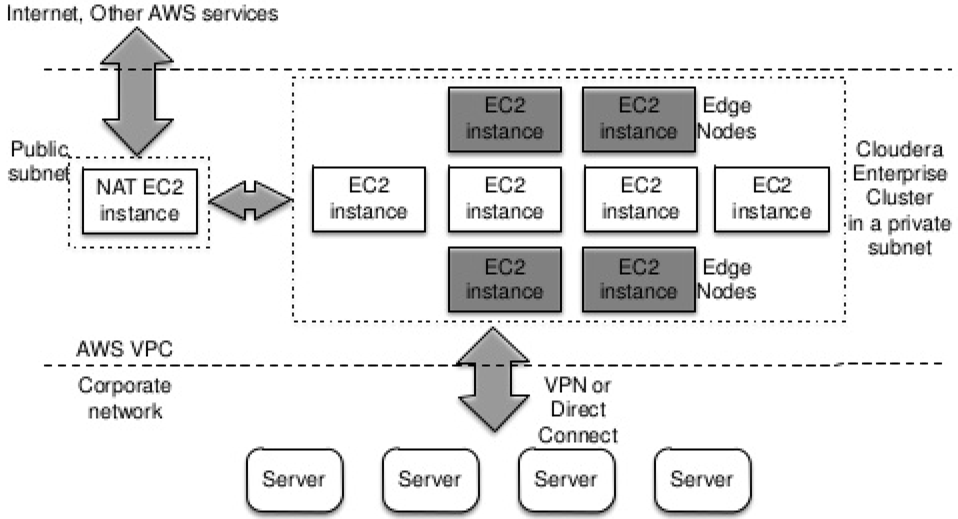
Performance, access, and security

Topologies – Deploy in Virtual Private Cloud (VPC)

Cluster in public subnet



Cluster in private subnet



Performance considerations

- Instance <-> Instance link
  - 10G
  - 10G + SR-IOV (HVM)
  - !10G
- Instance <-> S3 (equal to instance to public internet)
- Placement groups
  - Performance *may* dip outside of PGs
- Clusters within a single Availability Zone

EC2 instances

Storage, cost, performance, availability, and fault tolerance

Picking the right instance

Transient clusters

- Primary considerations:
  - Bandwidth
  - CPU
  - Memory
- Secondary considerations
  - Availability and fault tolerance
  - Local storage density
- Typical choices
  - C3 family, M3 family, M1 family
  - Anti pattern to use storage dense

Long running clusters

- Primary considerations
  - Local storage is key
  - CPU
  - Memory
  - Availability and fault tolerance
  - Bandwidth
- Typical choices
  - hs1.8xlarge, cc2.8xlarge, i2.8xlarge

Amazon Machine Image (AMI)

- 2 kinds – PV and HVM.
- Pick a dependable base AMI
- Things to look out for
  - Kernel patches
  - Third-party software and library versions
- Increase root volume size

Security

Security considerations

- Amazon Virtual Private Cloud (VPC) options
  - Private subnet
    - All traffic outside of VPC via NAT
  - Public subnet
- Network ACLS at subnet level
- Security groups
- EDH guidelines for Kerberos, Active Directory, and Encryption
- S3 provides server-side encryption

High Availability, Backups, Disaster Recovery

HA, Backups, DR

- High Availability available in the Hadoop stack
  - Run Namenode HA with 5 Journal Nodes
  - Run 5 Zookeepers
  - Run multiple HBase masters
- Backups and disaster recovery (based on RPO/RTO requirements)
  - Hot backup: Active-Active clusters
  - Warm backup: S3
    - Hadoop level snapshots – HDFS, HBase
  - Cold backup: Amazon Glacier

Planning your cluster

Capacity, performance, access patterns

- Bad news – no simple answer. You have to think through it.
- Good news – mistakes are cheap. Learn from ours to make them even cheaper.
- Start with workload type (ad-hoc / SLA, batch / interactive)
- How much % of the day will you use your cluster?
- How much data do you want to store?
- What are the performance requirements?
- How are you ingesting data? What does the workflow look like?

To make life easier

- Just released – Cloudera Director!
- AWS Quickstart
- Available resources
  - Reference Architecture (just refreshed)
  - Best practices blog

cloudera

Thank you

We are hiring!

Opportunities

- Smarter with topology
- Amazon EBS as storage for HDFS
- Deeper S3 integration
- Amazon Kinesis integration
- Workflow management

AWS re:Invent

Please give us your feedback on this session.
Complete session evaluations and earn re:Invent swag.

BDT305

http://bit.ly/awsevals

amazon web services

Join the conversation on Twitter with #reinvent