Hadoop Infrastructure @Uber Past , Present and Future

Mayank Bansal

Uber’s Mission

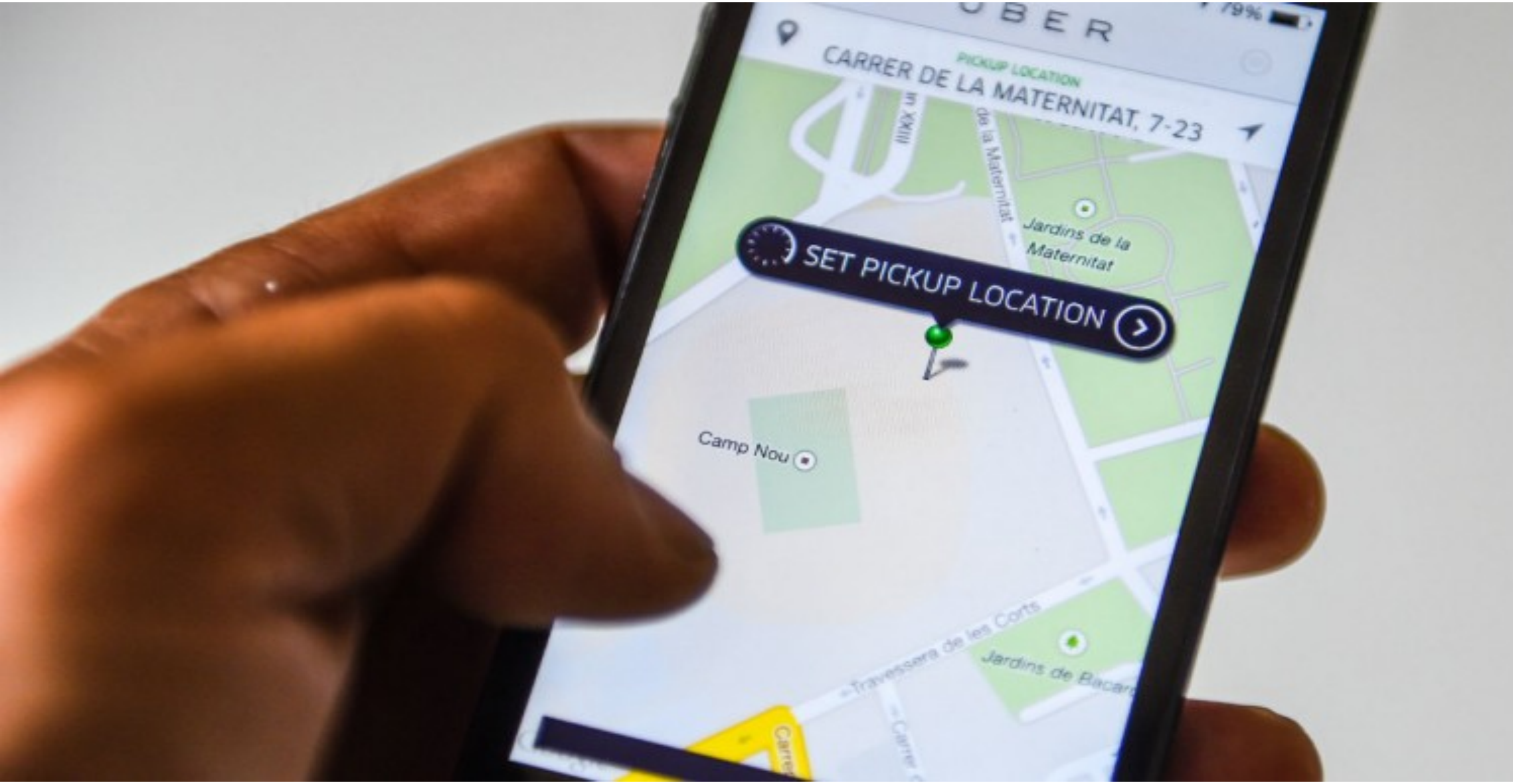
“ Transportation as reliable as running water , everywhere, for everyone ”

75+ Countries

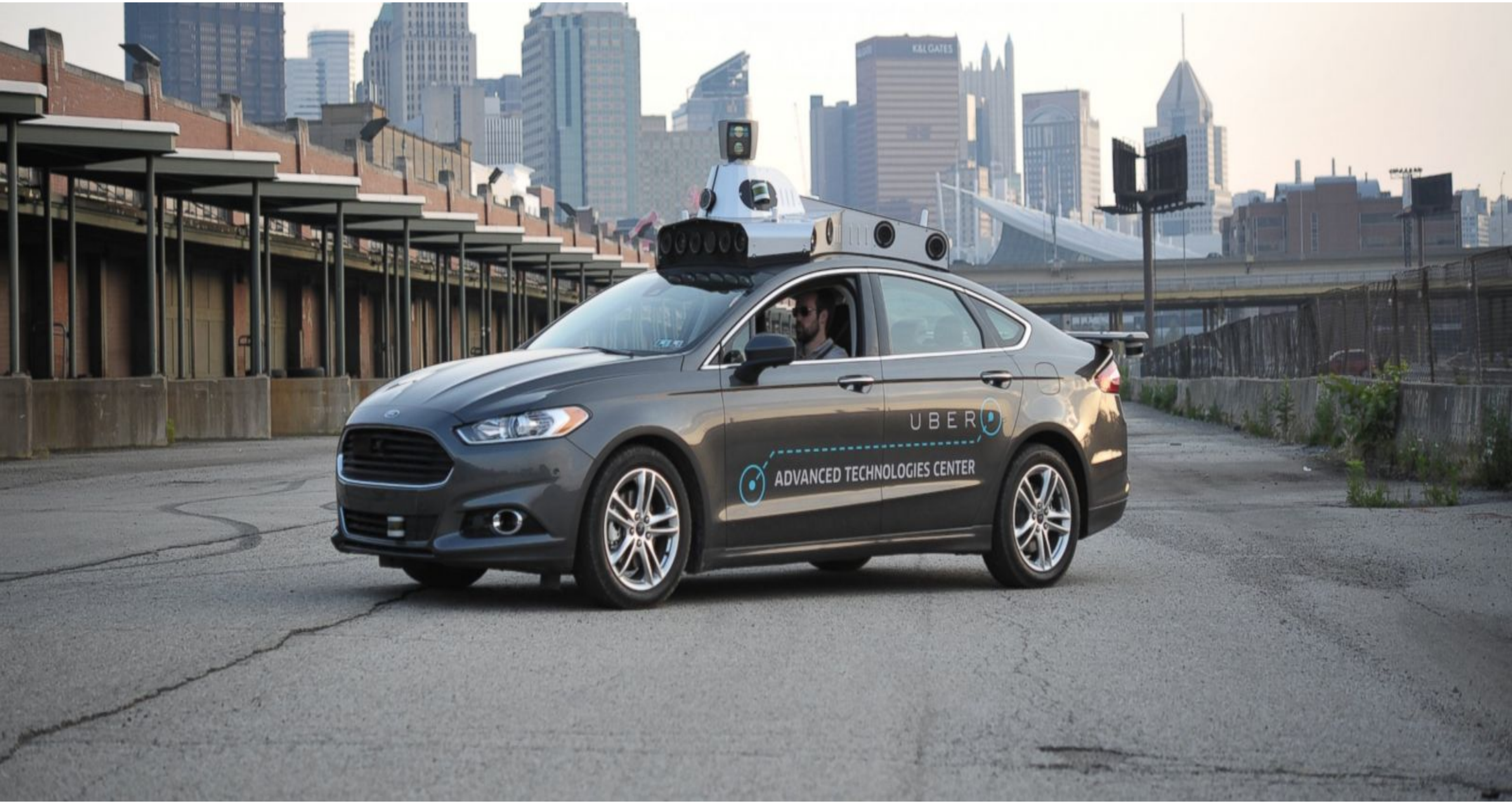
500+ Cities

And growing…

How Uber works



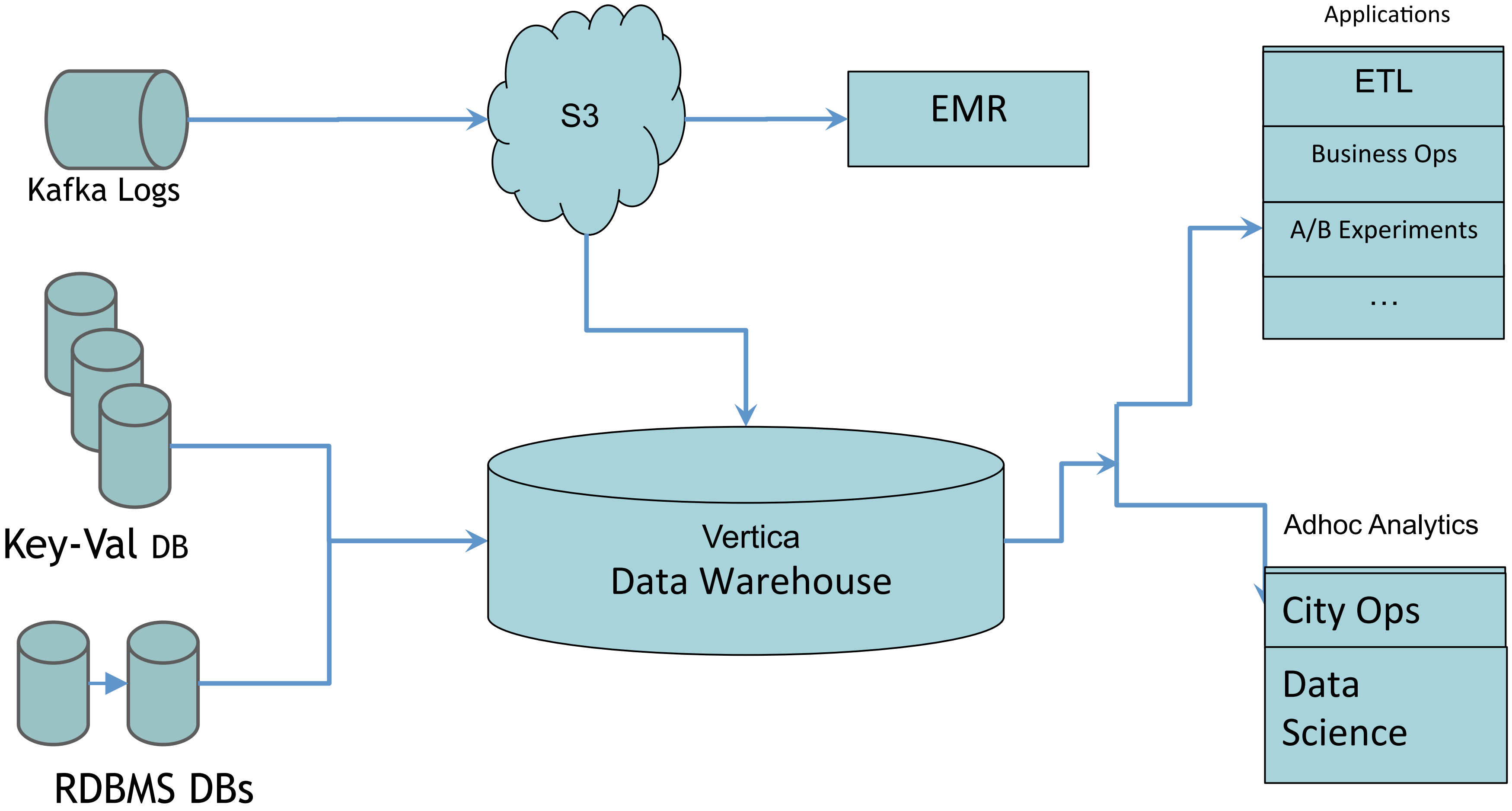
How Uber works



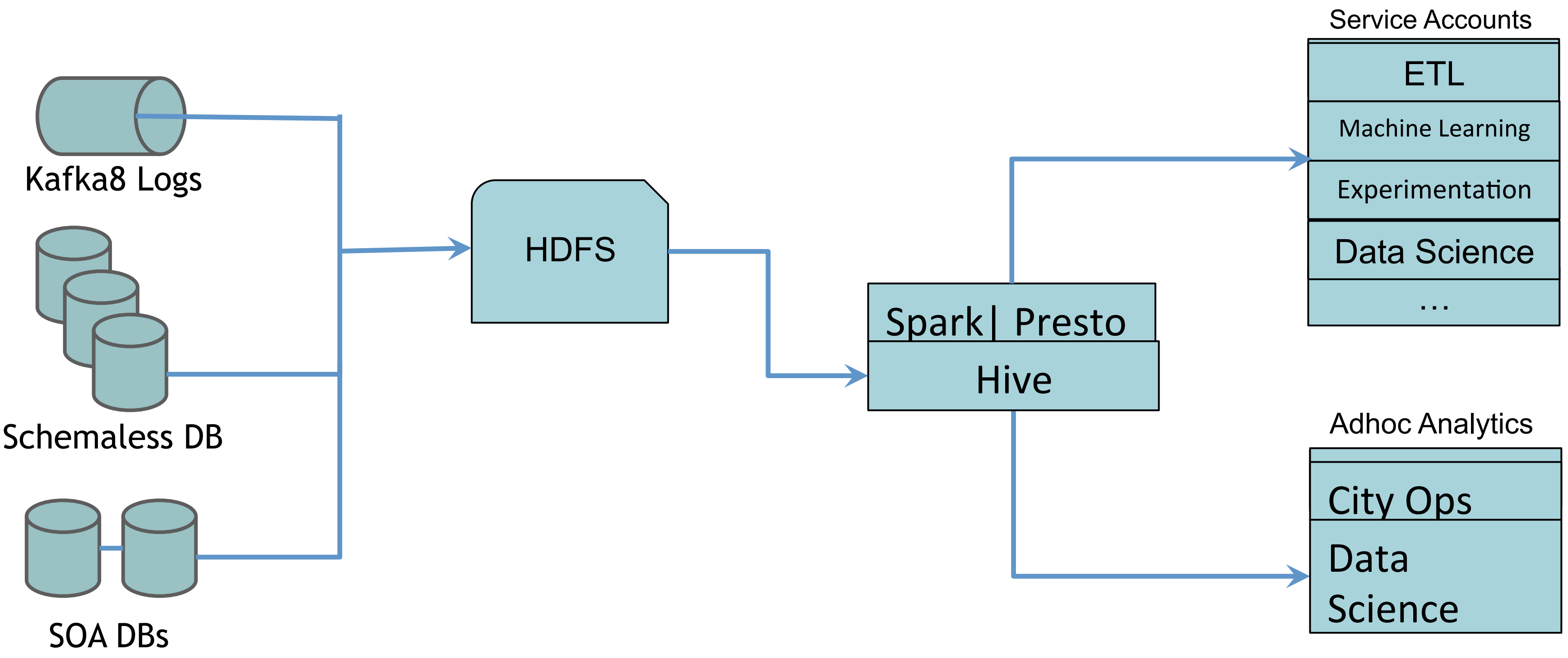
How Uber works

Data Driven Decisions

Data Infra Once Upon a time.. (2014)



Data Infrastructure Today

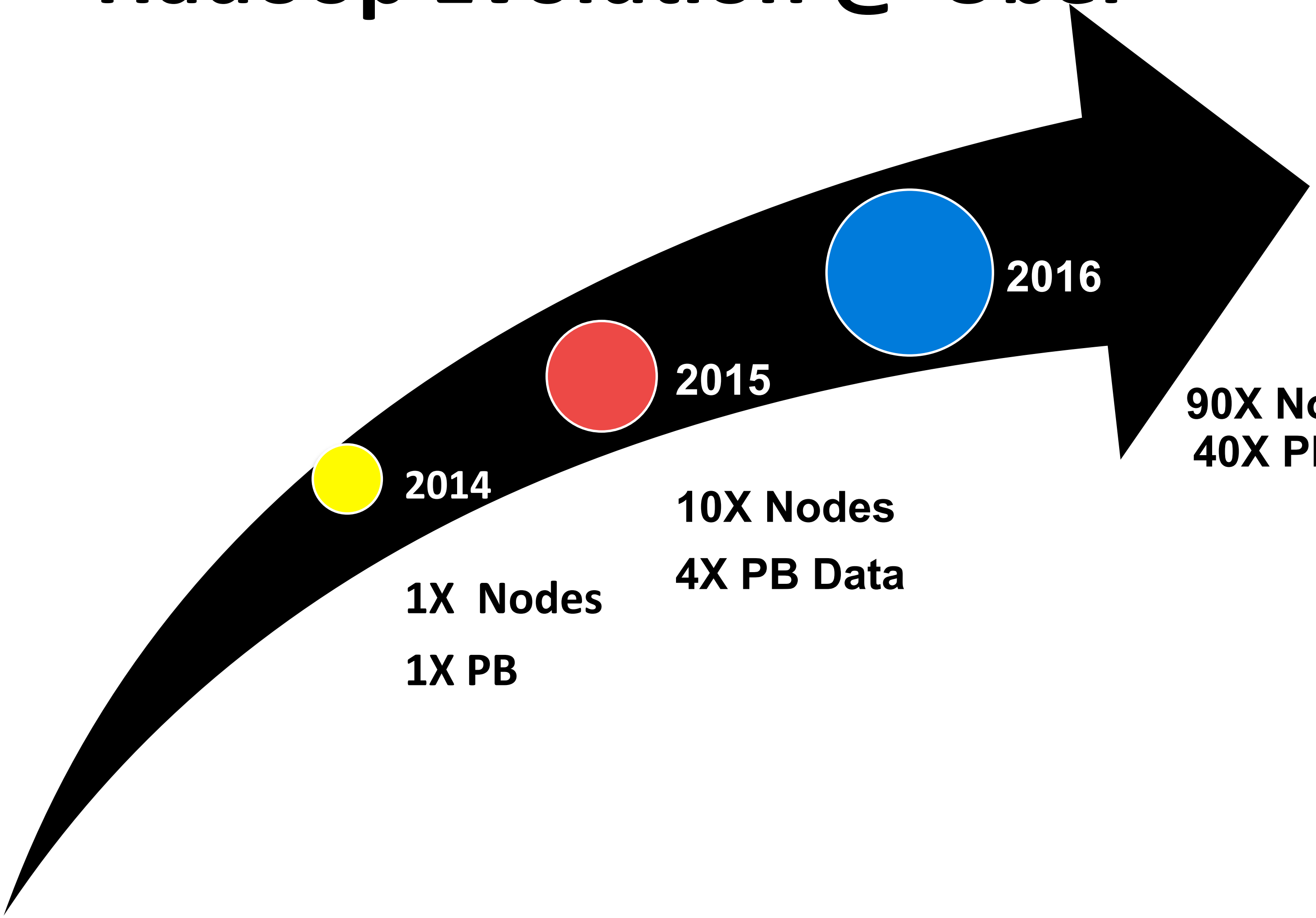


Few Takeaways …

- Strict Schema Management
  - Because our largest data audience are SQL Savvy! (1000s of Uber Ops!)
  - SQL = Strict Schema

- Big Data Processing Tools Unlocked - Hive, Presto and Spark
  - Migrate SQL savvy users from Vertica to Hive & Presto (1000s of Ops & 100s of data scientists & analysts)
  - Spark for more advanced users - 100s of data scientists

Hadoop Evolution @ Uber



2014

1X Nodes

1X PB

2015

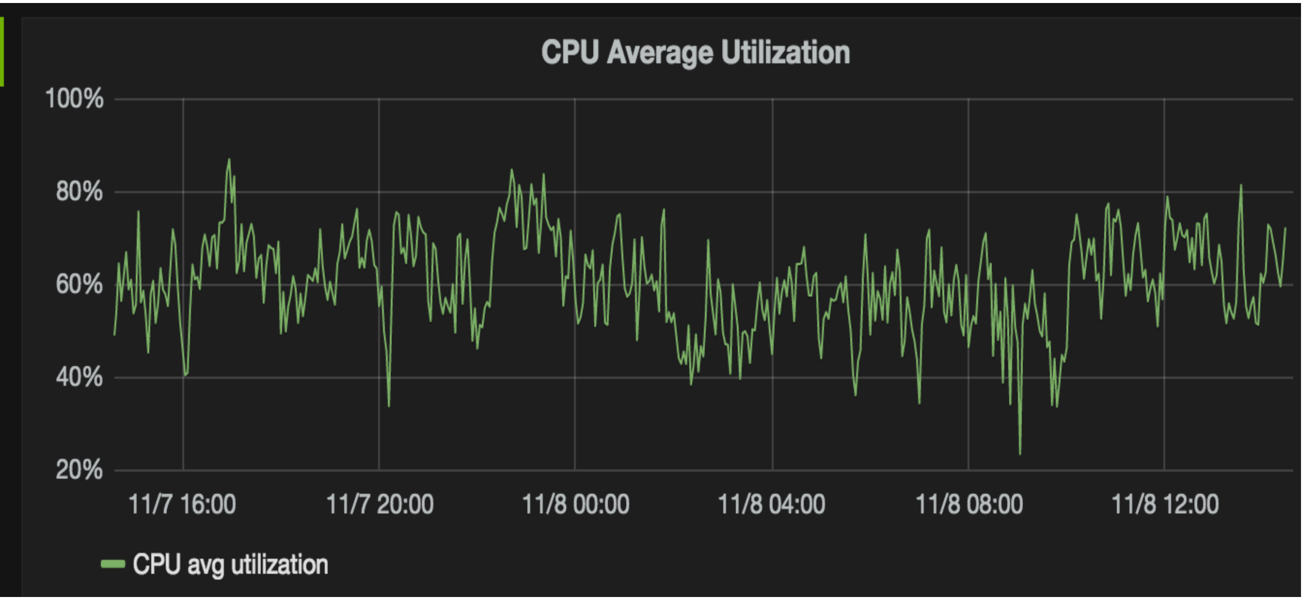
10X Nodes

4X PB Data

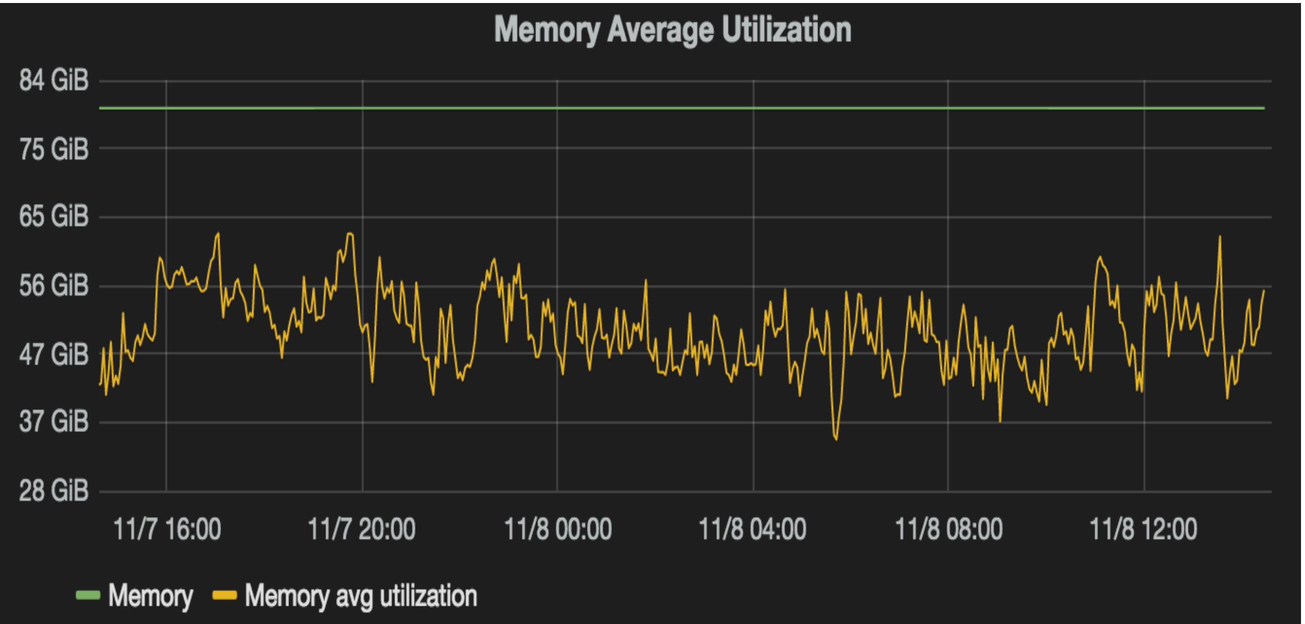
2016

90X Nodes
40X PB Data

Hadoop Cluster Utilization

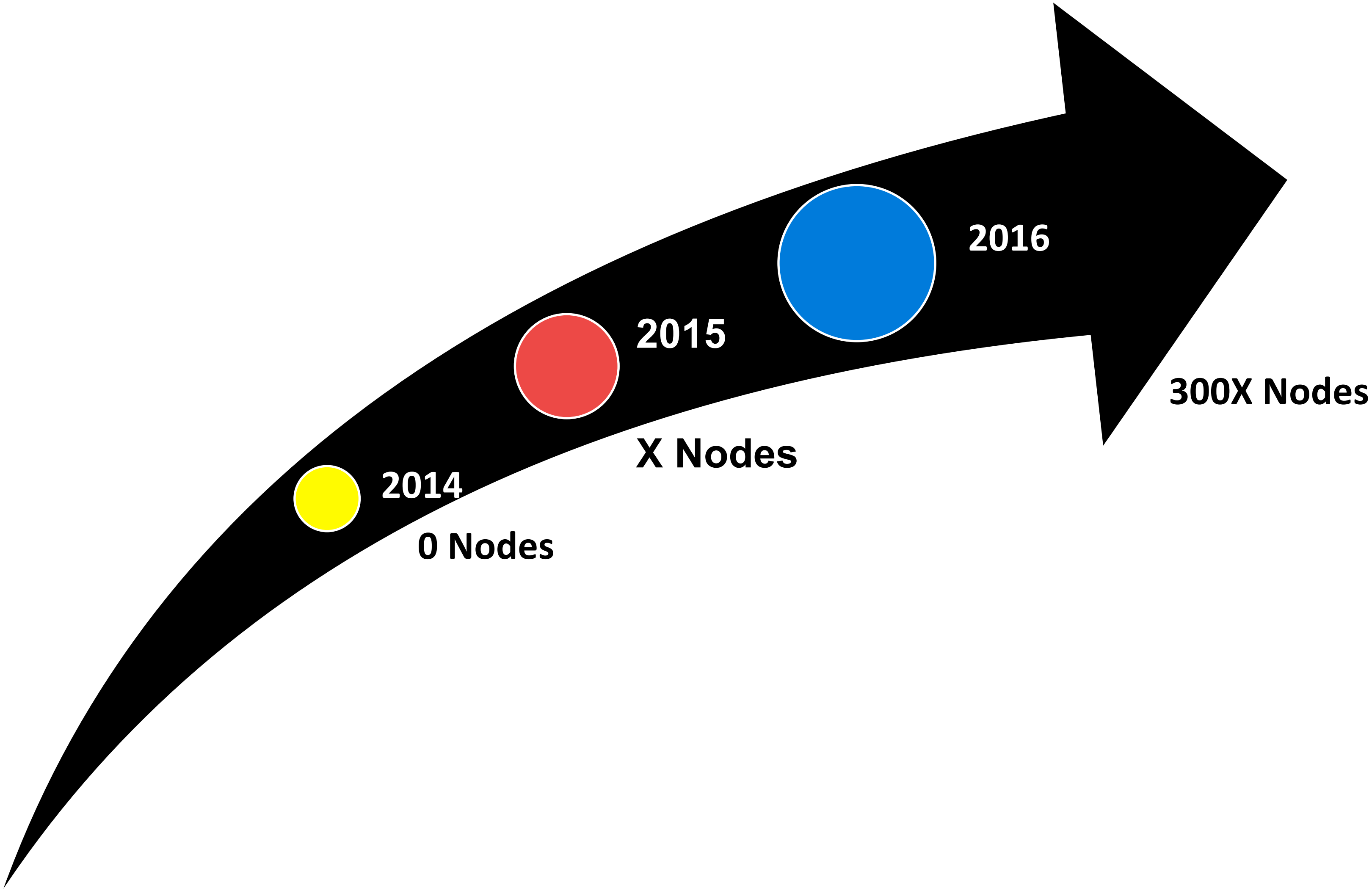


- Over provisioning for the peak loads.

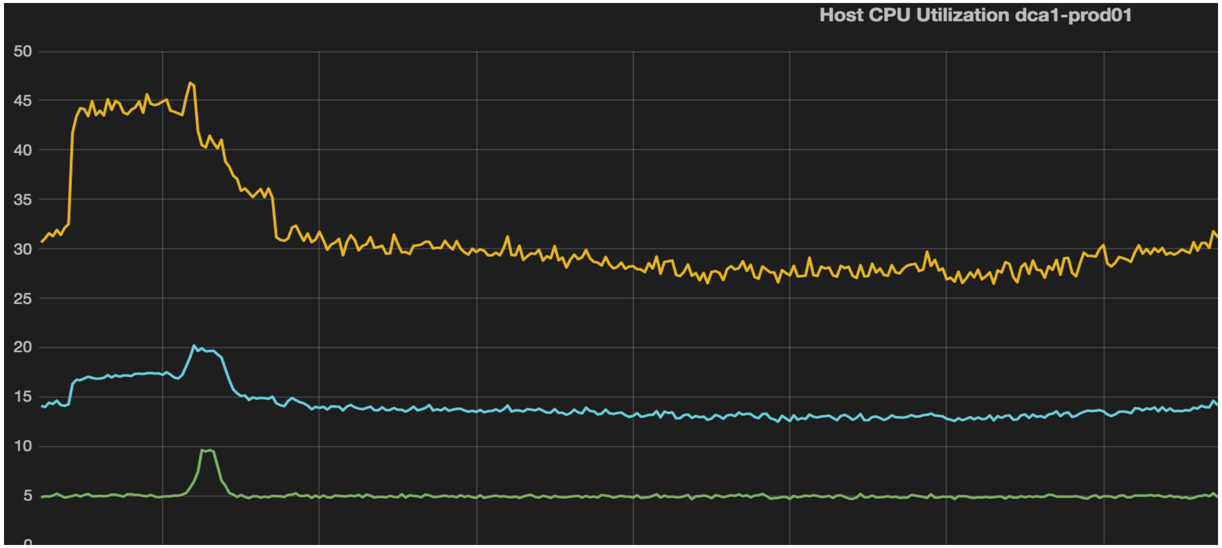


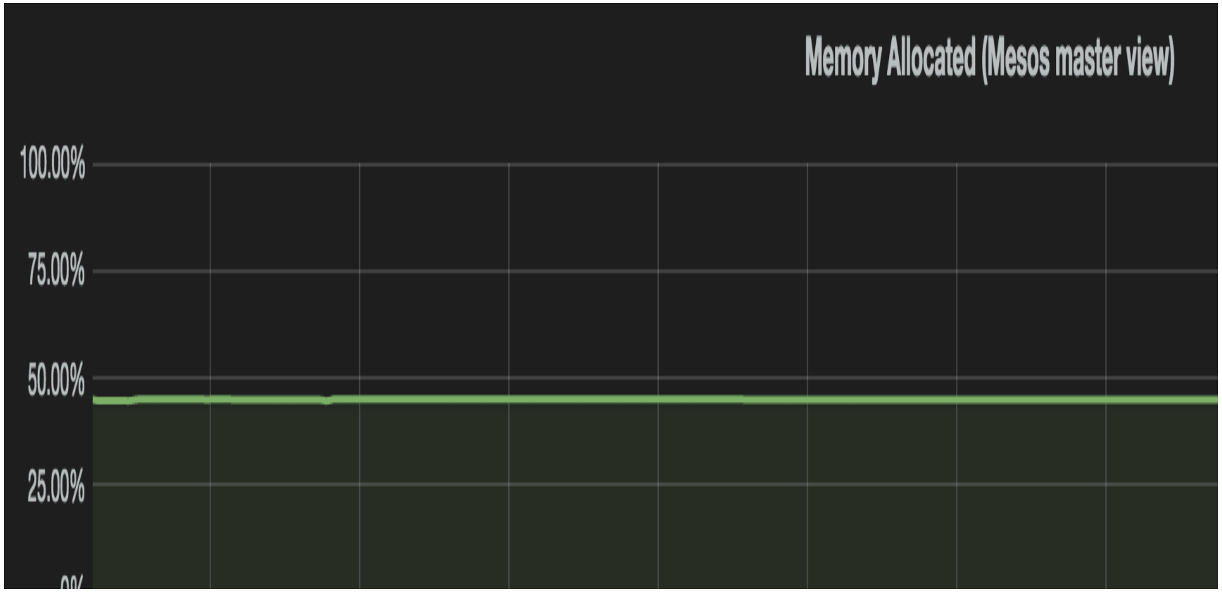
- Over capacity for anticipation of future growth

Mesos Evolution @ Uber



Mesos Cluster Utilization

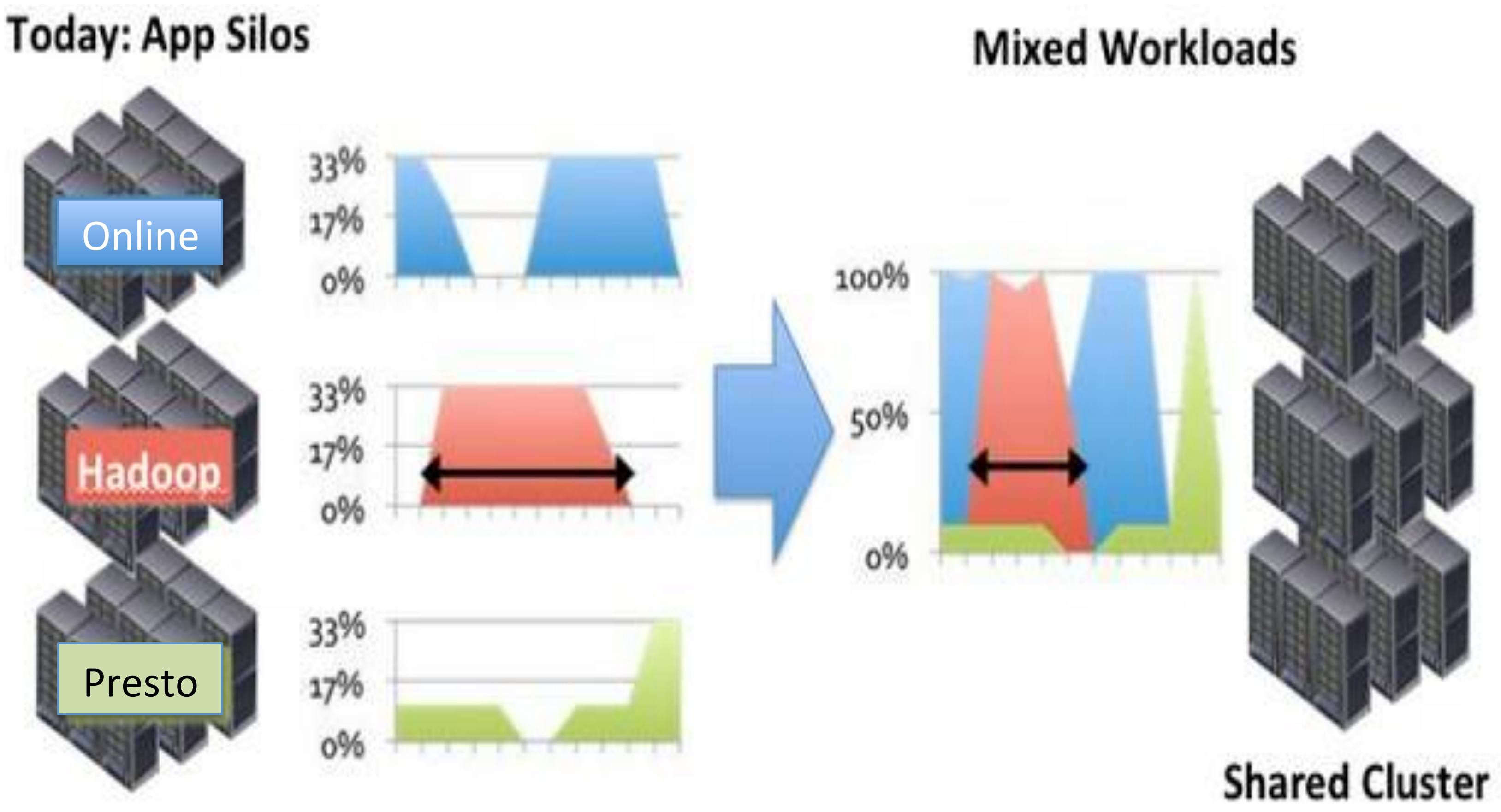




- Over provisioning for the peak loads

- Over capacity for anticipation of future growth

End Goal


Today: App Silos
Online
Hadoop
Presto
33%
17%
0%
33%
17%
0%
33%
17%
0%
Mixed Workloads
100%
50%
0%
Shared Cluster

What we need ?

GLOBAL VIEW OF RESOURCES

Available Resource Managers

Mesos vs YARN

Similar Isolation

Scales Better

YARN	MESOS
Single Level Scheduler	Two Level Scheduler
Use C groups for isolation	Use C groups for Isolation
CPU, Memory as a resource	CPU, Memory and Disk as a resource
Works well with Hadoop work loads	Works well with longer running services
YARN support time based reservations	Mesos does not have support of reservations
Dominant resource scheduling	Scheduling is done by frameworks and depends on case to case basis

Disk is better

This is Important

Better for batch

Imp for batch SLA's

Let’s tied them together

In a Nutshell

YARN is good for Hadoop

Mesos is good for Longer Running Services

apache™
myriad

apache™ myriad

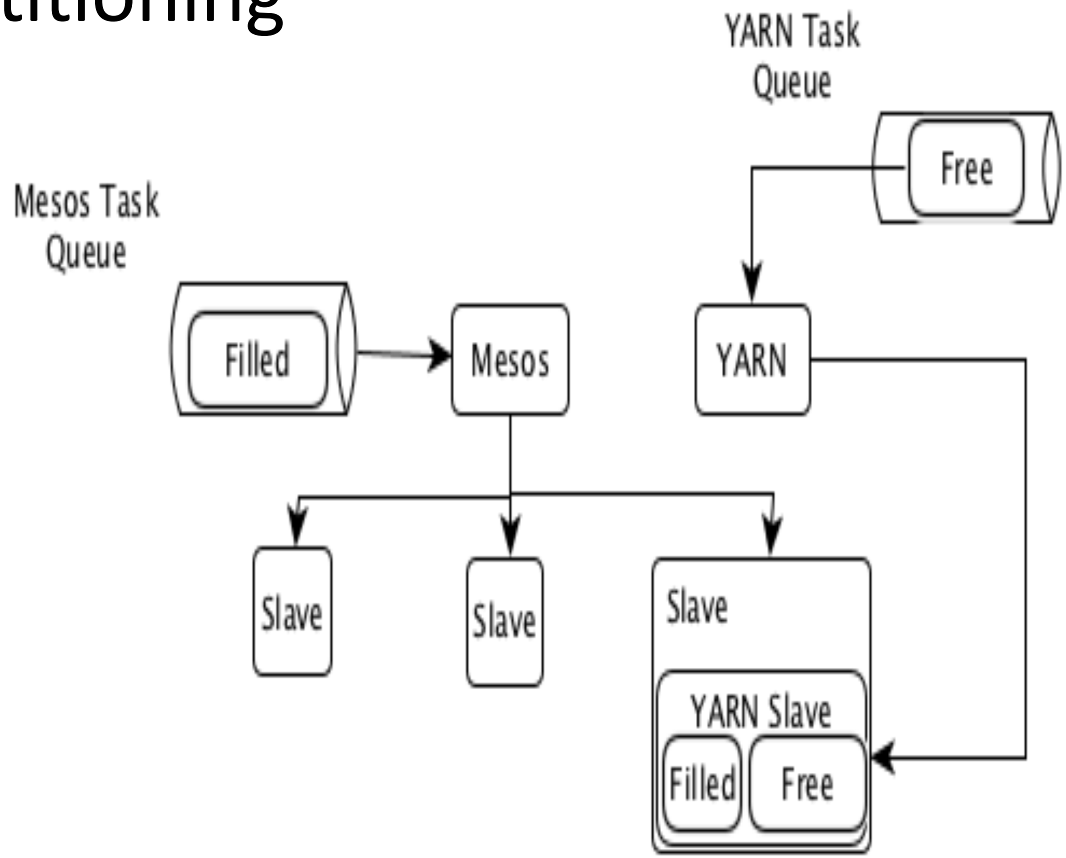
- Myriad is Mesos Framework for Apache YARN
- Mesos manages Data Center resources
- YARN manages Hadoop workloads
- Myriad
  - Gets resources from Mesos
  - Launches Node Managers

Myriad's Limitations

Static Resource Partitioning

- YARN will handle resources handed over to it.

- Mesos will work on rest of the resources

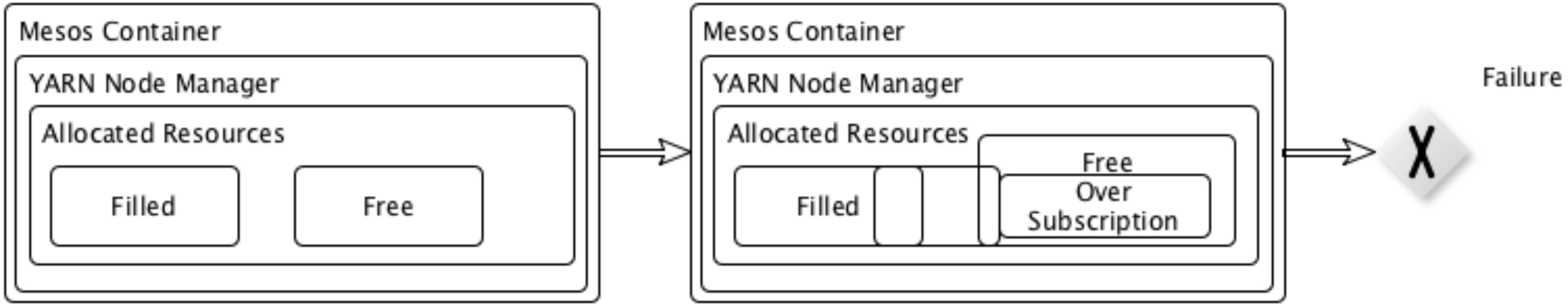


Myriad's Limitations

Resource Over Subscription

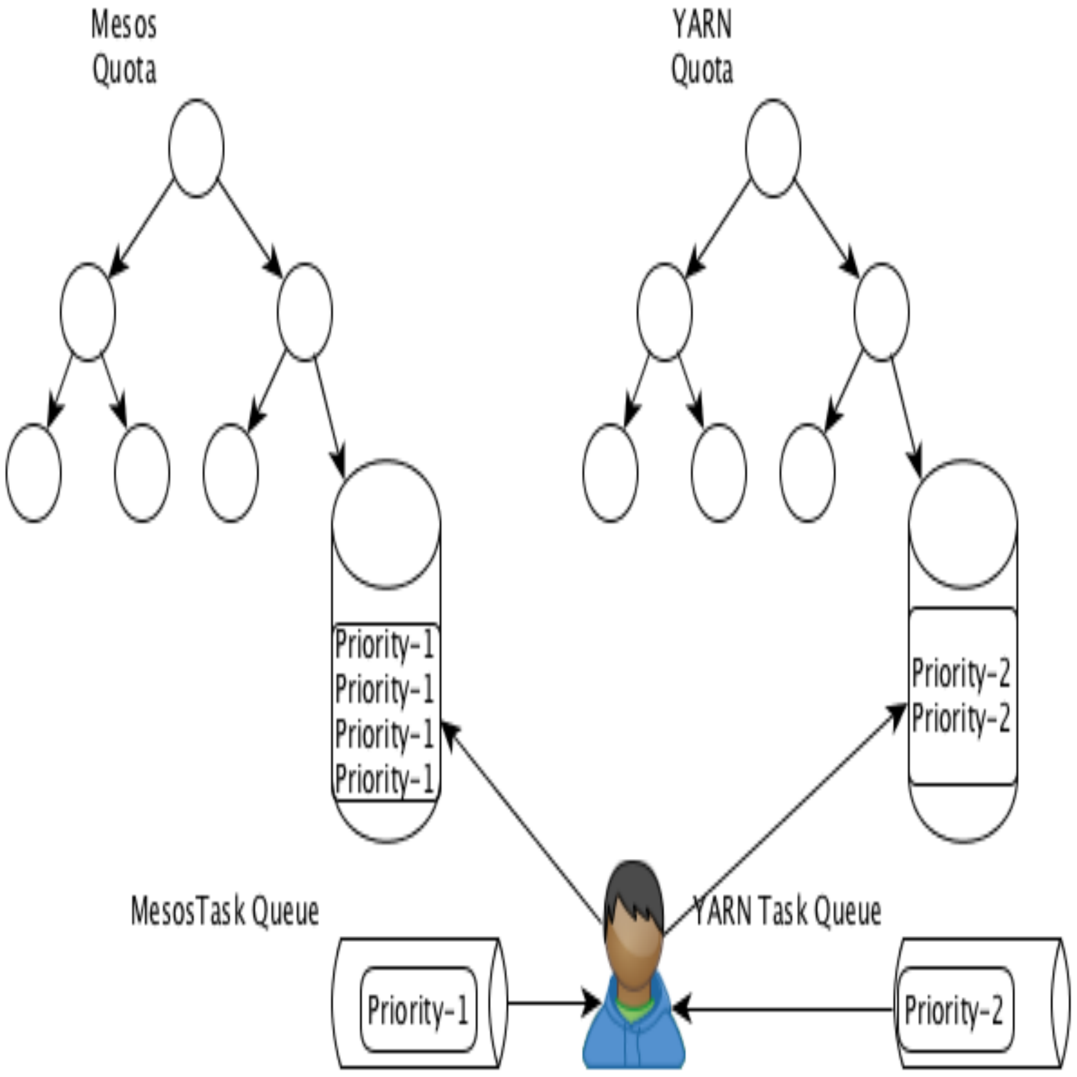
- YARN will never be able to do over subscription.
  - Node Manager will go away
  - Fragmentation of resources

- Mesos over subscription can kill YARN too



Myriad's Limitations

- No Global Quota Enforcement
- No Global Priorities



Myriad's Limitations

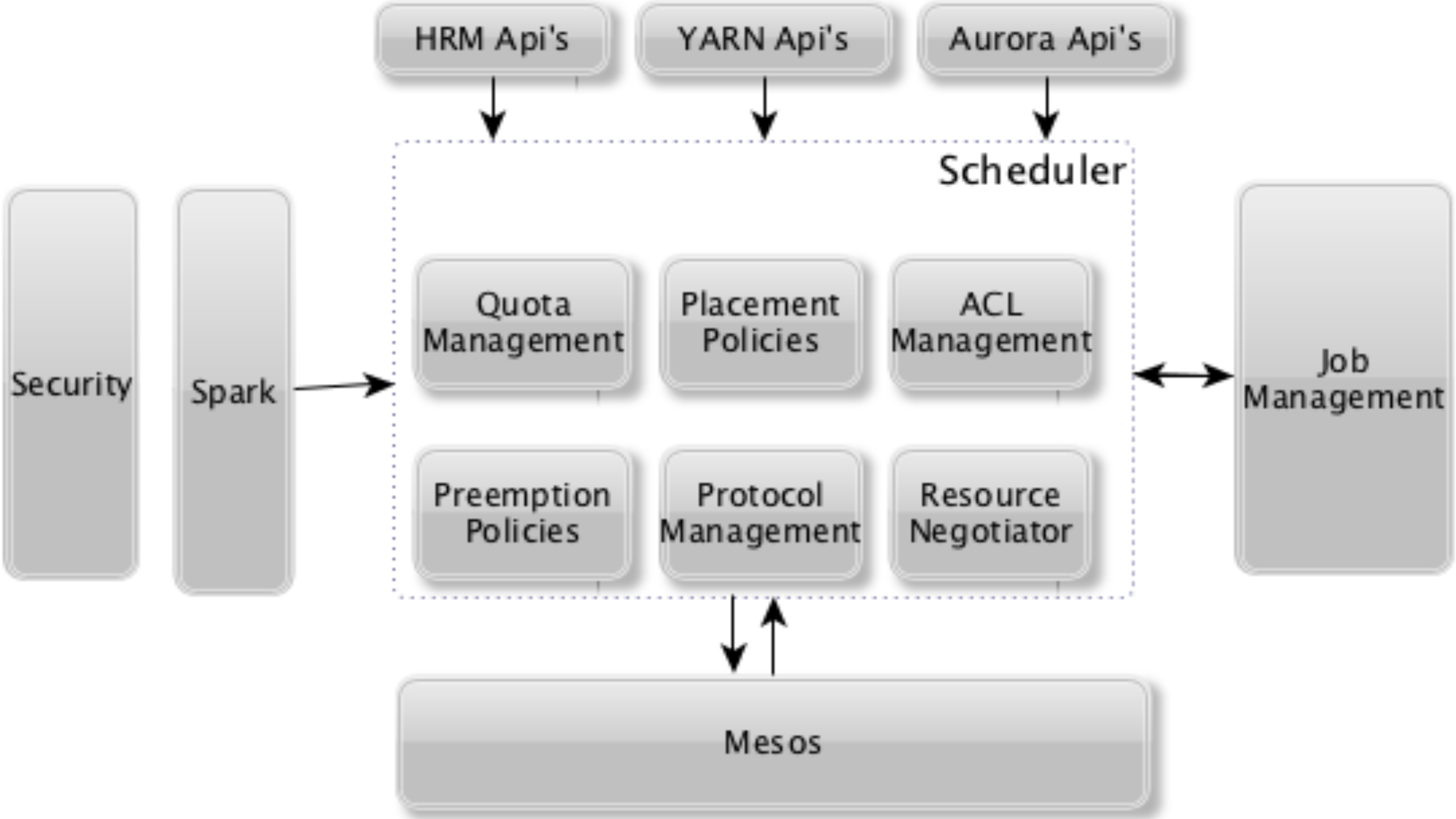
- Elastic Resource Management
- Bin Packing
- Stability
- Long List …

Unified Scheduler

High Level Characteristics

- Global Quota Management
- Central Scheduling policies
- Over subscription for both Online and Batch
- Isolation and bin packing
- SLA guarantees at Global Level

Unified Scheduler



Few Takeaways …

- We need one scheduling layer across all workloads
- Partitioning resources are not good
    - At least can save 30% resources
- Stability and simplicity wins in Production
    - Multi Level of resource Management and scheduling will not be scalable

Yes We're
HIRING

Questions?

mabansal@uber.com
mayank@apache.org

Thank You !!!