



# Dagens sesjon

- Hadoop



*“It does look similar—but this one  
is powered by Hadoop”*

# Utfordringer med Big Data

- Tregt å prosessere, dårlig skalering
- Mye jobbing mot disk
- Ikke lagringskapasitet, men hastighet på datatilgang som er flaskehalsen
- Utfordringer med å både lagre og analysere data
- Skalering er kostbart
- Datamaskiner går i stykker

# Utfordringer med Big Data

- Pålitelighet
  - Småfeil må ikke gi full stopp
  - Noen må ta over hvis enheter feiler
  - Skalering, legge til en ressurs må gi forbedret kapasitet
- Backup
  - Kostbart
- Sikkerhet og lett å bruke
- Prosessere data i parallell

# Idéen: Parallellprosessering av data

- Overføringshastighet forbedres hurtigere enn søkehastighet
- Prosessere read/write i parallell heller enn sekvensielt
  - 1 stk HD – 75MB/sec: 16 dager for 100TB
  - 1000 stk HD – 75GB/sec: 22 minutter for 100TB
- Dette er vanskelig!
  - Synkronisering
  - Deadlocks
  - Begrenset båndbredde
  - Timing og koordinering
  - Disk feil
  - Tilgjengelighet

# Håpet: Distributed Computing

- Dette er heller ikke lett!

Ressurs deling: tilgang til data og utnytte CPU ressurser

Concurrency: Tillate og håndtere samtidig tilgang og oppdatering

Skalering

Feiltoleranse: redundans og recovery

Heterogene systemer: ulike OS, ulik hardware

Gjennomsiktighet: skal fremstå som ett system



# Hadoops bakgrunn

- Apache Hadoop er et rammeverk som tillateter distribuert prosessering av store dataset over clusterne av ordinær hardware
- Skal ha en enkel programmeringsmodell
- Designet for skalere fra en til flere tusen maskiner
- Hver maskin har datalagring OG prosessering
- Bygger på Googles MapReduce og Google File System

# Hadoop History

- 2002-2004 Doug cutting and Mike Cafarella started working on Nutch
- 2003-2004: Google publishes GFS and MapReduce paper
- 2004 : Doug cutting adds DFS and Mapreduce support to Nutch
- Yahoo ! Hires Cutting , bulid team to develop Hadoop
- 2007: NY time converts 4TB of archive over 100 EC2 cluster of Hadoop.
- Web scale deployment at Yahoo!,Facebook,twitter.
- May 2009: Yahoo does fastest sort of a TB, 62secs over 1460nodes
- Yahoo sort a PB in 16.25hrs over 3658 nodes

# Hadoop design prinsipper

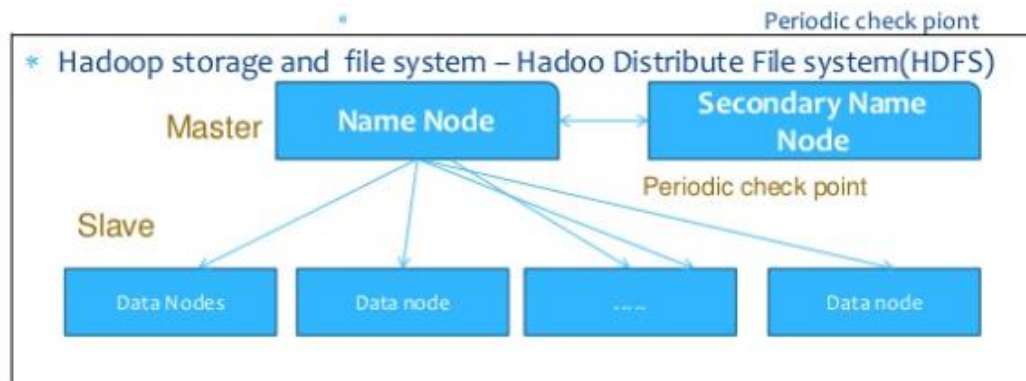
- Lagre og prosessere enorme datamengder
- Ytelse, datalagring og prosessering skal skalere lineært
- Prosessering skal skje der data er
- Enkel kjerne, moduler og utvidbar
- Man forventer at ting går feil
- Håndterbart og selvhelbredende
- Skal kjøre på ordinær hardware

# Hadoop arkitektur

- Bygget på to uavhengige rammeverk
- HDFS  
Hadoop File System er et pålitelig distribuert feilsystem som gir rask tilgang til data
- MapReduce  
MapReduce er et rammeverk for å gjøre høy-ytelses, distribuert prosessering vha. splitt og aggreger programmering

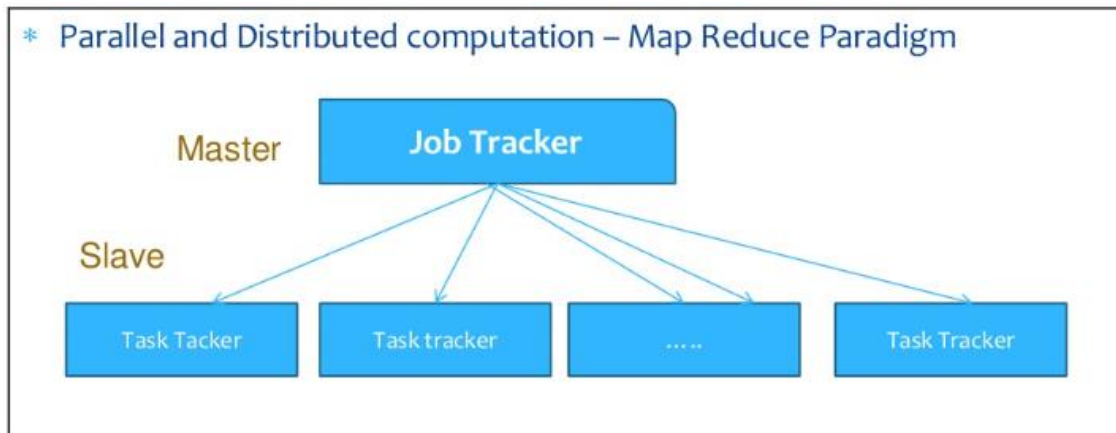
# Hadoop Master og Slave arkitektur

- Name Nodes er master. Den vedlikeholder kataloger og filer og holder rede på hvilke datablokker som ligger hvor.
- Datanoder er slavene som holder data og er distribuert i clusteret
- Secondary Name Node er en backup for Name Nodes og tar over hvis Name Nodes feiler

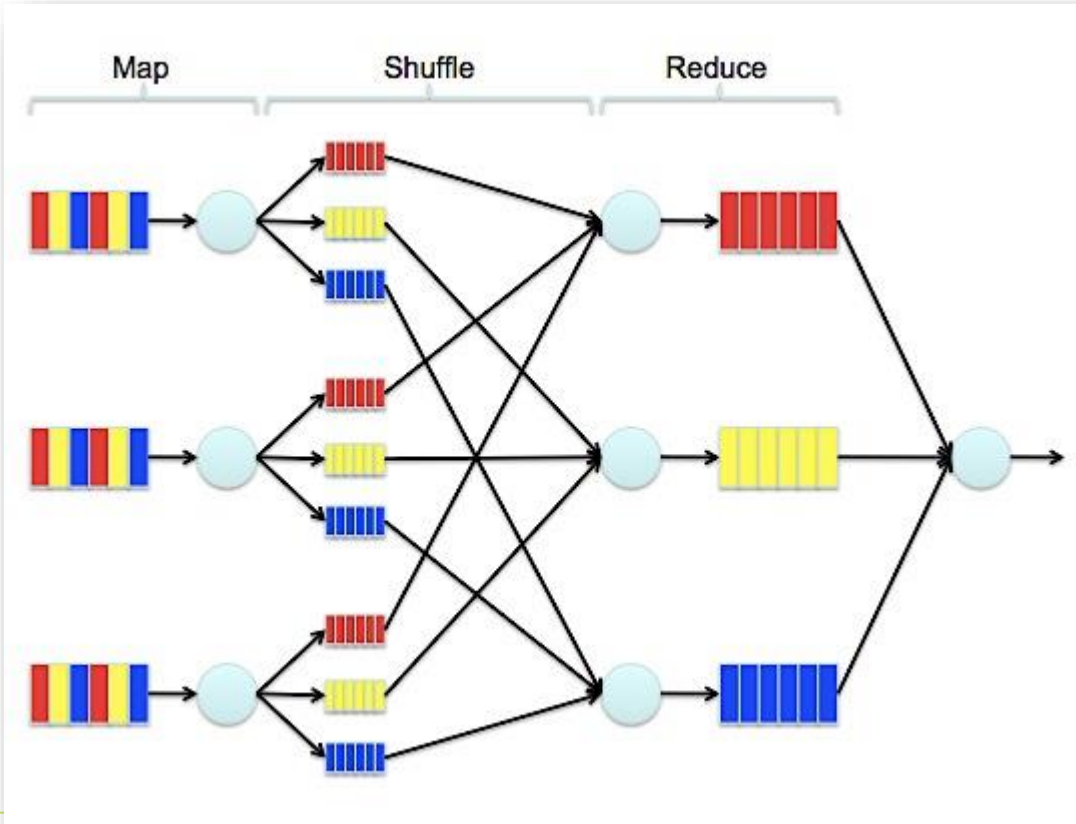


# Hadoop Master og Slave arkitektur

- JobTracker er master for prosessering. Den setter opp jobblister, men passer også på at prosessering skjer så nærme dataene som overhodet mulig
- TaskTrackers er slavene som er distribuert i clusteret og utfører map og reduce jobbene

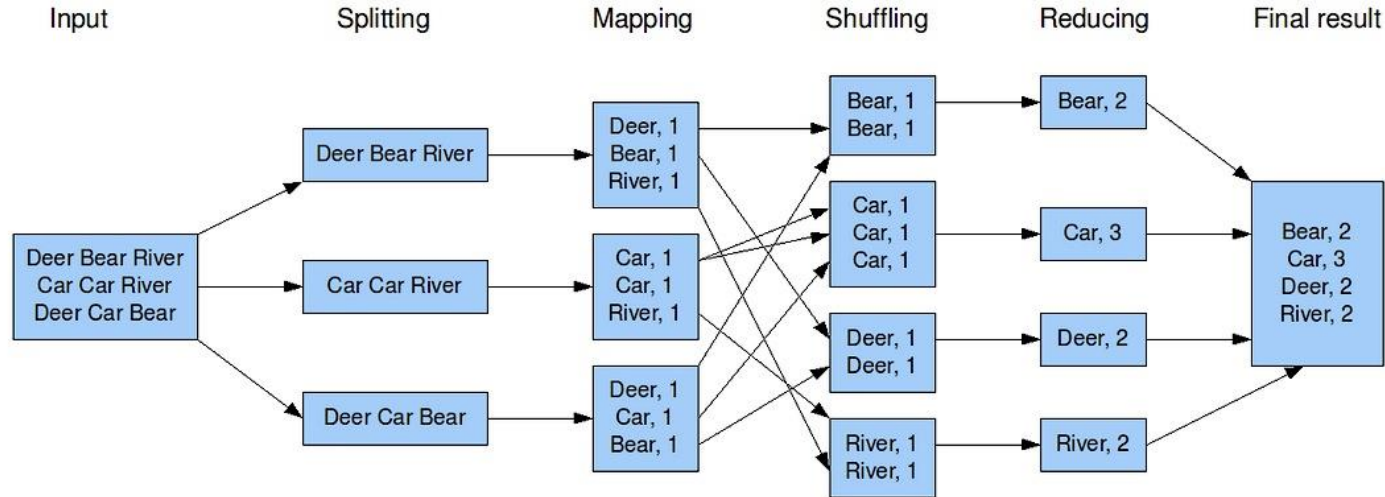


# MapReduce Job – Logical View



# MapReduce Example - WordCount

The overall MapReduce word count process





# Hadoop oppnår parallellisering

- Splitter data
- Prosesserer i parallell
- Sortere og kombinere for å finne svaret
- Schedule, prosessere og aggregere individuelt
- Robust feilhåndtering

# Spørringer mot Hadoop

- HiveQL
  - SQL lignende spørrespråk
  - Generer MapReduce i bakgrunnen
- Pig
  - Høynivå scriptspråk
  - Generer MapReduce i bakgrunnen

# Er Hadoop er alternativ for RDBMs

- Hadoop kan ikke erstatte relasjonsdatabaser, men er heller et supplement
- Den har ingen relasjoner, bruker key-value pairs
- Den håndtere ustrukturert (tekst) og semi-strukturert (xml)
- Går fint sammen med relasjonsdatabaser. Hadoop brukes ofte til å gjøre store mengder data om til noe håndterbart.

# Hvem skriver Hadoop

- Open Source fra Apache
- Ingen strategisk agenda
- Community driver software fremover
- Tillater konkurrerende prosjekter
- Ingen kan kjøpe Hadoop
- Organisasjoner som bidrar:  
Cloudera, Hortonworks, Yahoo!, Facebook, eBay, LinkedIn, Apple NetFlix, Twitter, IBM, Google, Adobe, Cisco, EMC, Berkely m.fl.

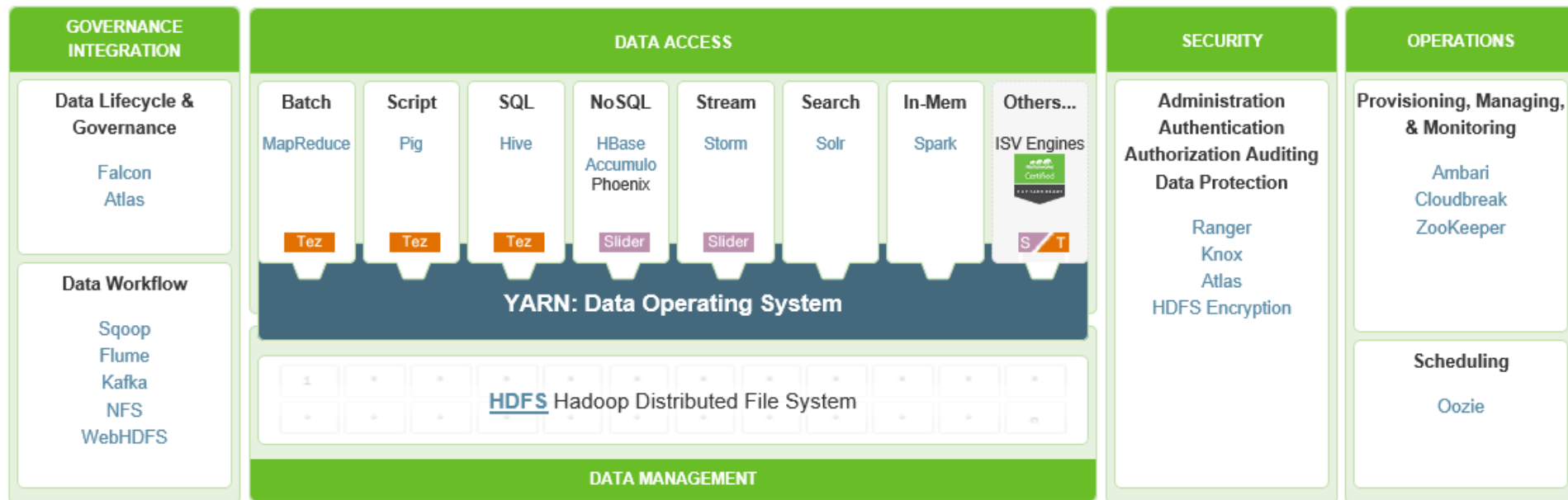
# Hva brukes Hadoop til

- Søk  
Yahoo, Amazon
- Loggprosessering  
Facebook, Yahoo, last.fm
- Anbefalingssystemer  
Facebook
- Datavarehus  
Facebook, AOL
- Video- og bildeanalyse  
New York Times

# Hadoop retning

- Opprinnelig designet for batch prosessering
- Drar mer og mer i retning av sanntid og transaksjonshåndtering  
Apache Storm
- Mer in-memory  
Apache Spark

# Hadoop og tilhørende teknologier



# Hadoop oppsummering

- Distribuert rammeverk for lagring og prosessering av data på ordinær hardware
- Open Source og skrevet i Java
- Lagrer all slags data
- Skalerer lineært
- Eksekverer kode der data er
- Håndterer feil på egenhåndt
- Replikerer data, ingen behov for back-up
- Veldig kosteffektivt





**“So you want to hire me as a Data Scientist for Intelligent Virtualized Deep Machine Learning Real-time Big Data in the Cloud for Social Networks? Ok, but if you also want Hadoop, increase my salary by 50%.”**