Warehouse-Scale Computing and the BDAS Stack

Ion Stoica
UC Berkeley
Overview

Workloads

Hardware trends and implications in modern datacenters

BDAS stack
What is Big Data used For?

Reports, e.g.,
» Track business processes, transactions

Diagnosis, e.g.,
» Why is user engagement dropping?
» Why is the system slow?
» Detect spam, worms, viruses, DDoS attacks

Decisions, e.g.,
» Decide what feature to add
» Decide what ad to show
» Block worms, viruses, ...

Data is as useful as the decisions it enables
Data Processing Goals

Low latency queries on historical data: enable faster decisions
  » E.g., identify why a site is slow and fix it

Low latency queries on live data (streaming): enable decisions on real-time data
  » E.g., detect & block worms in real-time (a worm may infect 1mil hosts in 1.3sec)

Sophisticated data processing: enable “better” decisions
  » E.g., anomaly detection, trend analysis
**Typical Datacenter Node**

- **CPU:**
  - » 8-16 cores
  - » Transfer rate: quad channel DDR3: 4×10GB/s

- **RAM:** 64-256 GB

- **Storage:** 8-12 disks, each
  - » Capacity: 1-3TB
  - » Transfer rate: 50-100MB/s
Typical Datacenter Node

- RAM: (64-256GB)
- 8-16 cores
- Disk x10: (10-30TB)

Network:
- 40 GB/s
- 0.5-1GB/s
- 0.1-1GB/s

Nodes in same rack:
- 0.1-1GB/s

Nodes in other rack:
- 0.01-1GB/s

Top-of-rack switch
Typical Datacenter Node

- **RAM**: (64-256GB)
- **Disk**: x10 (10-30TB)
- **8-16 cores**
- **SSDs x4**: (1-4TB)

**Network**:
- 40 GB/s
- 0.8-2GB/s
- 0.5-1GB/s

**Top-of-rack switch**
- 0.1-1GB/s
- 0.01-1GB/s

**Nodes in same rack**
- 0.1-1GB/s

**Nodes in other rack**
- 0.01-1GB/s

**SSD drive**:• 256GB-1TB
• 200-500 MB/s (SATA 3)
How Much Does $1,000 Buy You Today?

15-20TB disk storage (consumer grade disks)
1 TB of SSD storage (SATA-3)
0.2TB of RAM
Memory Capacity Trends

RAM (64-256GB)

Disk x10 (10-30TB)

Node

8-16 cores

0.2-1GB/s

0.8-2GB/s

40 GB/s

SSDs x4 (1-4TB)

0.8-2GB/s

0.2-1GB/s

Top-of-rack switch

Nodes in same rack

Nodes in other rack

0.1-1GB/s

0.01-1GB/s

Capacity doubles every 18 months (Moore’s law)
Disk Capacity Trends

- **RAM**: (64-256GB)
- **8-16 cores**
- **Disk x10**: (10-30TB) with a capacity doubling every 18 months
- **Network**: 0.1-1GB/s
- **Top-of-rack switch**: 0.01-1GB/s, 0.1-1GB/s
- **Nodes in same rack**: 0.8-2GB/s, 0.2-1GB/s
- **Nodes in other rack**: 0.8-2GB/s, 0.2-1GB/s

**Node** (1-4TB)
SSD Capacity Trends

Node

- RAM (64-256GB)
- 8-16 cores
- Disk x10 (10-30TB)
- SSDs x5 (1-4TB)

Network
- Capacity doubles every 18 months (Moore’s law)
- 0.1-1GB/s
- 0.2-1GB/s
- 0.8-2GB/s
- 40 GB/s

Top-of-rack switch

Nodes in same rack

Nodes in other rack
CPU Trends

- RAM (64-256GB)
- Disk x10 (10-30TB)
- 8-16 cores
- • Frequency no longer increasing
  • # of cores doubles every 24 months

Source: http://www.drdobbs.com/parallel/design-for-manycore-systems/219200099
Network I/O Trends

Node

RAM (64-256GB)

8-16 cores

Disk x10 (10-30TB)

Node

SSDs x5 (1-4TB)

Network 0.1-1GB/s

Top-of-rack switch

Network 0.01-1GB/s

Nodes in same rack 0.1-1GB/s

Nodes in other rack 0.01-1GB/s

0.2-1GB/s

40 GB/s

0.8-2GB/s

Server Network I/O doubling every 24 months

Network I/O Trends

- RAM (64-256GB)
- 8-16 cores
- Disk x10 (10-30TB)

Memory (i.e., DDR) bandwidth doubling every ~36 months

source: http://www.theregister.co.uk/2009/05/26/rambus_pitches_xdr2/
SSD Throughput Trends

- RAM (64-256GB)
- Disk x10 (10-30TB)
- 8-16 cores

SSD (e.g., SATA) transfer rate doubles every 36 months

Bandwidth (Gbps)

- Top-of-rack switch
- Network 0.1-1GB/s
- 40 GB/s
- 0.8-2GB/s
- 0.2-1GB/s

Nodes in same rack 0.1-1GB/s

Nodes in other rack 0.01-1GB/s
Disk Throughput Trends

- **RAM**: (64-256GB)
- **Disk x10**: (10-30TB)
- **Network**: 0.1-1GB/s
- **Top-of-rack switch**:
  - Nodes in same rack: 0.1-1GB/s
  - Nodes in other rack: 0.01-1GB/s
- **Node SSDs x5**: (1.5-5TB)
- **Disk bandwidth increases very slowly**

Source: Freitas et al., 2011
Trend Summary

Rate doubles every 24 months

Capacity doubles every 18 months

Rate doubles every 36 months

Freq. no longer increasing; # of cores doubles every 24 months

Rate increases very slowly

Capacity doubles every 18 months

Rate doubles every 36 months
Trend Summary (cont’d)

Storage is cheap and capacity increases exponentially.

Most transfer rates increase exponentially but slower.

→ gap between capacity and transfer rate increases.

Multiple channels/disks alleviate problem but don’t come for free.

» E.g., disk striping increases block size.

Datacenter apps must carefully select where to place computations & data.
Challenge and Opportunity

Accessing disk very slow: 1,000s to read/write 100GB from/to disk
» Transfer rate not increasing
» Will get worse: 512 GB per node in one-two years
» Faster to access remote memory!
» SSDs not widely deployed in datacenters
Challenge and Opportunity (cont’d)

A few node cluster = 1TB RAM: enough to handle many large datasets
  » E.g., 1 billion users, 1KB metadata (on average)

# of cores doubles only every 24 months → memory/core increases exponentially

Judiciously using RAM is key
Existing Open Stack...

..mostly focused on large on-disk datasets
  
  » Massive data and sophisticated data processing, but slow

We add RAM to the mix
  
  » Enable interactive queries and data streaming: speedup queries and iterative algorithms by up to 30x
  
  » Dramatically increase ability to explore / mine the data

(SSDs in the future)
BDAS Software Stack

Leverage open source ecosystem (e.g., HDFS, Hadoop)

Abstractions to take advantage of storage hierarchy
  » Many real-world working sets fit in memory → RDDs
  » Controllable data placement to minimize communication

Virtualize cluster resources (Mesos)
  » Allow multiple frameworks to share cluster and data → Resource offers

Simplify parallel programming
  » Scala interface to Spark
  » Shark distributed SQL engine
Project History

Mesos started in early 2009, open sourced 2010
Spark started in late 2009, open sourced 2010
Shark started summer 2011, alpha April 2012
Used at Twitter, Foursquare, Klout, Quantifind, Conviva, Yahoo! Research, Airbnb & others
Today’s Open Analytics Stack

- **Hadoop Stack** (e.g., Hive, Pig, …)
- **Other Frameworks** (Storm, MPI, …)
- **HDFS**

Layer Diagram:
- **Storage Layer**
- **Frmwk. Layer**
BDAS Software Stack

Hadoop Stack (e.g., Hive, Pig, ...)

Other Frameworks (Storm, MPI)

Mesos

Management platform that allows multiple framework to share cluster
BDAS Software Stack

- **Spark**
  - In-memory processing framework:
    - Iterative computations
    - Interactive queries
  - (e.g., Hive, Pig, ...)

- **Mesos**
  - Resource Mgmt. Layer

- **HDFS**
  - Storage Layer

- **Frmwk. Layer**

Other Frameworks (Storm, MPI)
BDAS Software Stack

HIVE on top of Spark
- Interactive queries for data fitting in memory

HQL (Hive) Interface

Shark
Spark

Stack (e.g., Hive, Pig, ...)
Frameworks (Storm, MPI)

Mesos

HDFS
BDAS Software Stack

- Shark
- Spark Streaming
- Hadoop Stack (e.g., Hive, Pig, ...)
- Other Frameworks (Storm, MPI)

Low-latency streaming processing for very large data

HDFS

Frmwk. Layer
Resource Mgmt. Layer
Storage Layer
Provide **bounded error** and **bounded response time** on very large data.
BDAS Software Stack

- **BDAS HDAS Stack (e.g., Hive, Pig, …)**
- **Other Frameworks (Storm, MPI)**
- **Share in-memory storage (RDDs) across multiple frameworks**
Summary

Today: first iteration of BDAS Software Stack
» Mesos: enable multiple frameworks to share cluster resources and data
» Spark: enable interactive and iterative computations through the use of RDDs
» Shark: enable interactive Hive queries

Next: full stack to allow users to trade between answer (1) quality, (2) response time, and (3) cost

Fully compatible with open standards