ENSF 612: Fall 2021
Lecture - Challenges in Working with Big Data

Dr. Gias Uddin, Assistant Professor
Electrical and Software Engineering
Schulich School of Engineering
University of Calgary
https://giasuddin.ca/

Topics

- Big data challenges
- Big data vs. traditional databases
- Cluster computing and big data

Big Data Challenges

- Wide variety of data science software available
  - MATLAB, Octave, R, Excel
- Most can only run on a single computer
- Can only analyze datasets that can fit computer's RAM
- Storage costs down ; size doubles every 18 months
- Provides incentive to keep data – not discard it
- But
  - CPU speeds are stalling
  - Storage bottlenecks prevent moving data in and out of CPUs
- So, single computer software can't exploit abundance of data

Big Data Challenges

- Let's look at concrete examples of data size
  - Facebook generates 60 TB of logs a day (1 TB = 1000 GB)
  - 1000 genome project has 200 TB of data
  - Google index is 10+ PB of data (1 PB = 1024 TB)
- Cost of a 1 TB disk is $35 – possible to keep data forever
- But it takes over 3 hours to read 1 TB from disk (@100 MB/s)
- Clear that single computer cannot hold/process all the data
- **Solution: distribute data across multiple computers**
- **Solution: cluster computing**

A Minimalistic Architectural Diagram

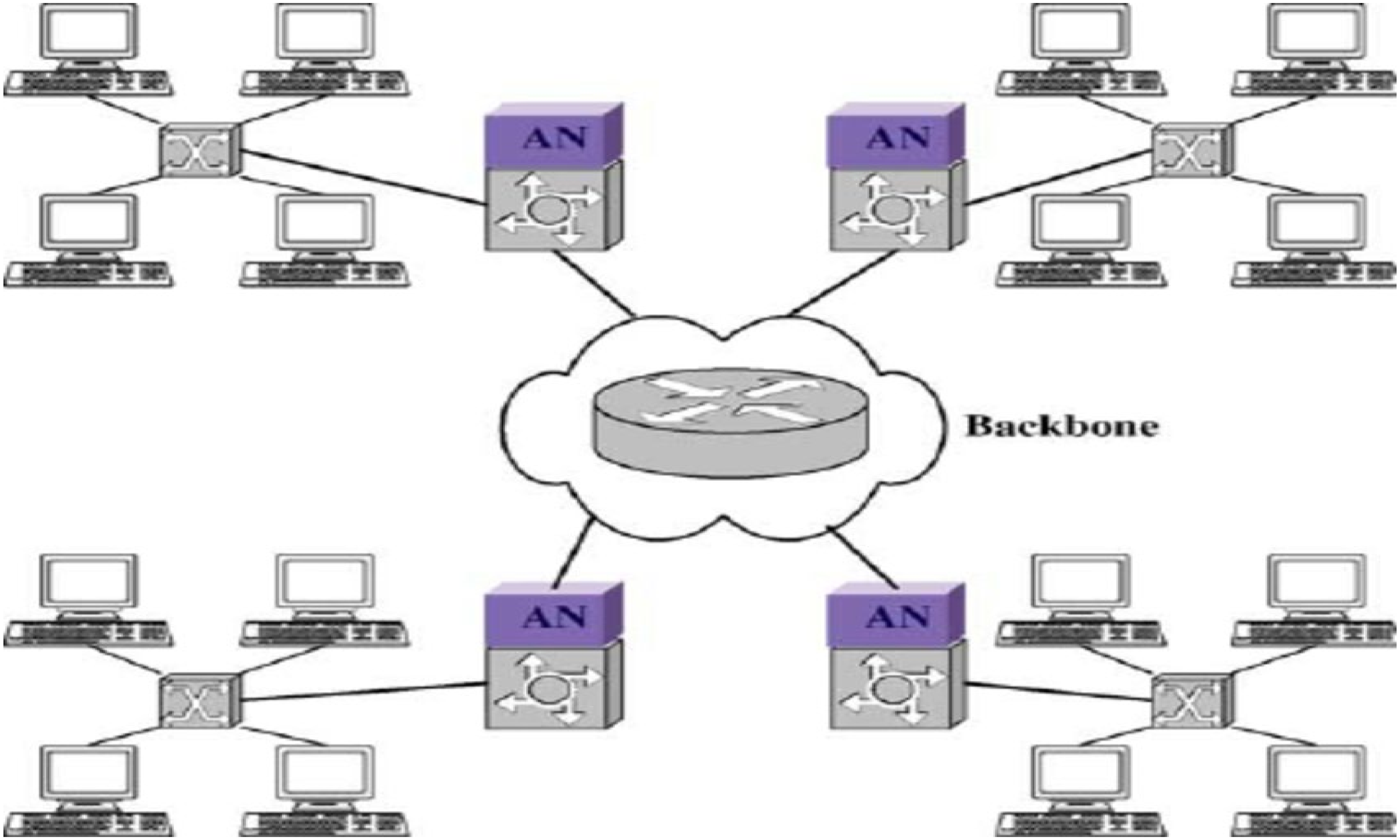


Image credit: https://www.researchgate.net/figure/Meta-cluster-computing_fig3_220285871

What it really looks like in real-world!

Image credit: https://en.wikipedia.org/wiki/Computer_cluster

Cluster Computing

NSA (US National Security Agency) Data Center

Image credit: https://www.herzing.ca/blog/montreal/a-look-at-the-nsas-new-data-centre-for-network-administration-students/

Big Data vs Traditional Database

- Databases, e.g., SQL server, have supported data science
- Work well when data can fit into one computer
- Big data analyses use “non-traditional” “NoSQL” databases
  - HBase – based on Google’s BigTable, used by Facebook
  - Cassandra – used by Facebook
  - Redis, MongoDB,………
- All of these hold data over several computers
- **How else are these different from traditional databases?**

Big Data vs Traditional Database

Element	Databases	Data Science
Data Value	"Precious"	"Cheap"
Data Volume	Modest	Massive
Examples	Bank records, Personnel records, Census, Medical records	Online clicks, GPS logs, Tweets, tree sensor readings
Priorities	Consistency, Error recovery, Auditability	Speed, Availability, Query richness
Structured	Strongly (Schema)	Weakly or none (Text)
Properties	Transactions, ACID^{+}	CAP* theorem (2/3), eventual consistency
Realizations	Structured Query Language (SQL)	NoSQL: Riak, Memcached, Apache Hbase, Apache River, MongoDB, Apache Cassandra, Apache CouchDB,,...

*CAP = Consistency, Availability, Partition Tolerance

$^{+}$ACID = Atomicity, Consistency, Isolation and Durability

Cluster Computing and Big Data

- Clearly, data needs to be distributed
- Use a cluster of computers connected by a network
- E.g., scientific computing uses clusters of special computers
- Big data analysis exploits commodity computers and network
  - Cheap computers instead of specialized supercomputers
  - Ethernet instead of expensive networking technology
- System can be scaled by merely adding more computers
- **However, commodity computers create new challenges**

Cluster Computing and Big Data

- Cheap computers can fail a lot!
  - Stats for Google's data centres
    - 1%-5% of hard drives fail per year
    - 0.2% of DIMMs fail per year
- Network is slow
  - Slower to get data from network than from RAM or disk
- Uneven performance – some computers can be slow
- Distributed programming is hard!!
- **Software has to "hide" all these complexities**

Cluster Computing and Big Data

- Let's look at these challenges using an example
- "Hello World" of big data – the word count problem
  - Count number of occurrences of words in a document
  - Many applications – indexing, popularity of URLs

"Betty bought some butter and the

butter was so bitter. Betty bought

some better butter to make the

bitter butter better"

→

Betty – 2
Bought – 2
Some – 2
Butter – 4
And – 1
The -2
Was -1
So – 1
Bitter-2
Better – 2
To-1
Make -1

"Betty bought some butter and the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{}

"$\boxed{\text{Betty}}$ bought some butter and the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{**Betty:1**}

"Betty $\boxed{\text{bought}}$ some butter and the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{Betty:1,
bought:1,}

"Betty bought $\boxed{\text{some}}$ butter and the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{Betty:1,
bought:1,
some:1,
}

"Betty bought some butter and the

butter was so bitter. Betty bought

some better butter to make the

bitter butter better"

{Betty:1,
bought:1,
some:1,
butter:1,
}

"Betty bought some butter $\boxed{\text{and}}$ the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{Betty:1,
 bought:1,
 some:1,
 butter:1,
 and:1,
}

"Betty bought some butter and the butter was so bitter. Betty bought some better butter to make the bitter butter better"

{Betty:1,
bought:1,
some:1,
butter:1,
and:1,
the:1,
}

“Betty bought some butter and the

$\boxed{\text{butter}}$ was so bitter. Betty bought

some better butter to make the

bitter butter better”

{Betty:1,
  bought:1,
  some:1,
  butter:2,
  and:1,
  the:1,
}

- What if the document is large?

"Betty bought some butter and the

butter was so bitter. Betty bought

some better butter to make the

bitter butter better. She sells sea shells

on the sea shore. I saw Susie sitting in a

shoe shine shop. Where she sits she shines,

and where she shines she sits."

Cluster Computing and Big Data

- Split document into partitions
- Assign each partition to a computer
- Calculate hash table at each of the computers

"Betty bought some butter and the butter was so bitter. Betty bought some better butter to make the

bitter butter better. She sells sea shells on the sea shore. I saw Susie sitting in a

shoe shine shop. Where she sits she shines, and where she shines she sits."

Computer 1

{Betty:2,bought:2,some:2,
butter:3,and:1,the:2,
was:1,so:1,bitter:1
better:1,to:1,make:1
}

Computer 2

{bitter:1,butter:1,better:1,
.....
}

Computer 3

{shoe:1,shine:1,shop:1,
.....
}

Cluster Computing and Big Data

- Combine results from partitions on another computer

Computer 1

{Betty:2,bought:2,some:2,
butter:3,and:1,the:2,
was:1,so:1,bitter:1
better:1,to:1,make:1
}

Computer 2

{bitter:1,butter:1,better:1,
.....
}

Computer 3

{shoe:1,shine:1,shop:1,
.....
}

Computer 4

{Betty:2,bought:2,
some:2,butter:4,
and:1,the:3,was:1
so:1,bitter:2,better:2,
to:1,make:1,she:5…
…
}

What's wrong with this approach?

- Combine results from partitions on another computer

Computer 1

{Betty:2,bought:2,some:2,
butter:3,and:1,the:2,
was:1,so:1,bitter:1
better:1,to:1,make:1
}

Computer 2

{bitter:1,butter:1,better:1,
…..
}

Computer 3

{shoe:1,shine:1,shop:1,
…..
}

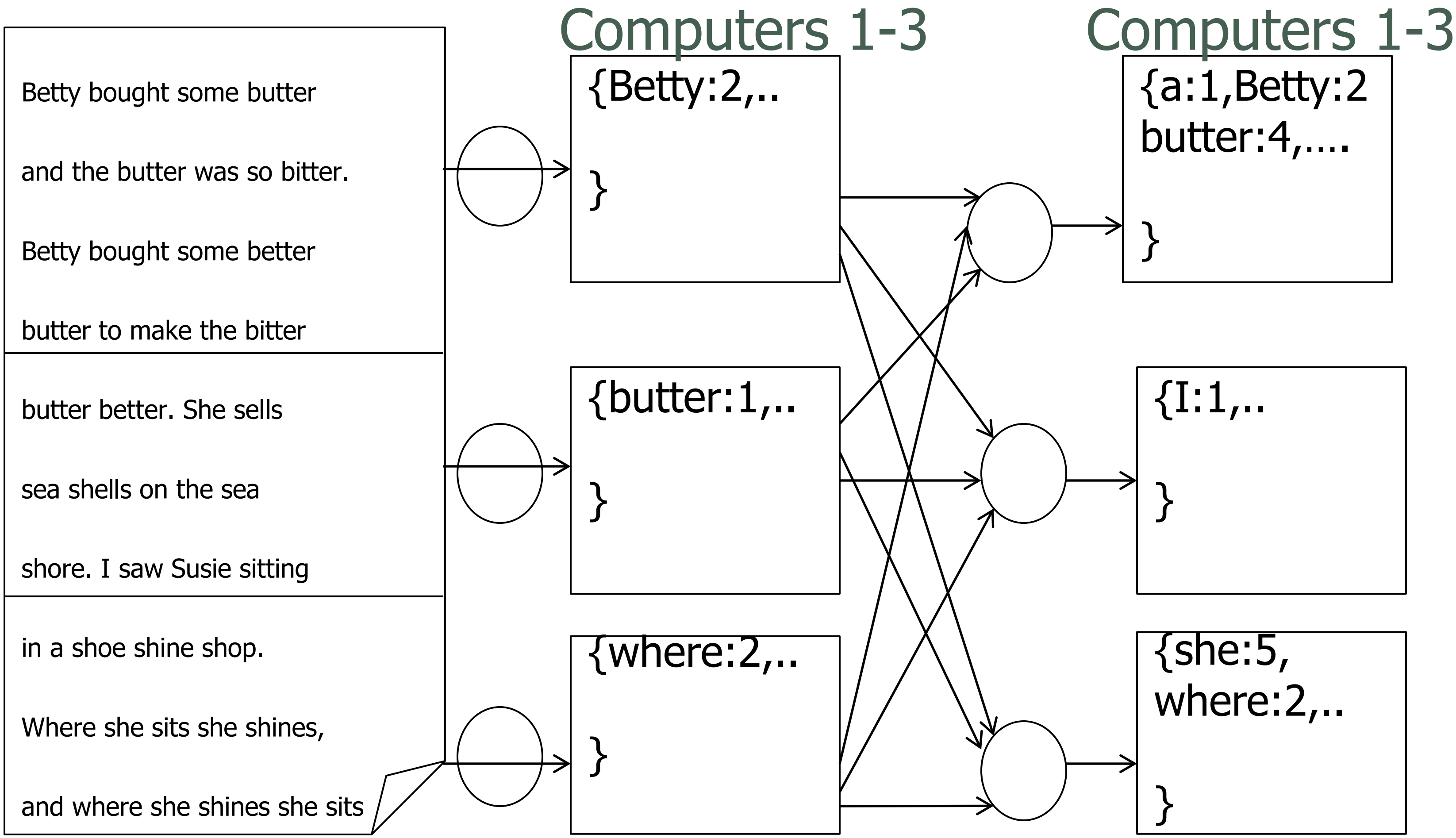
Computer 4

{Betty:2,bought:2,
some:2,butter:4,
and:1,the:3,was:1
so:1,bitter:2,better:2,
to:1,make:1,she:5…
…
}

Result has to fit one computer

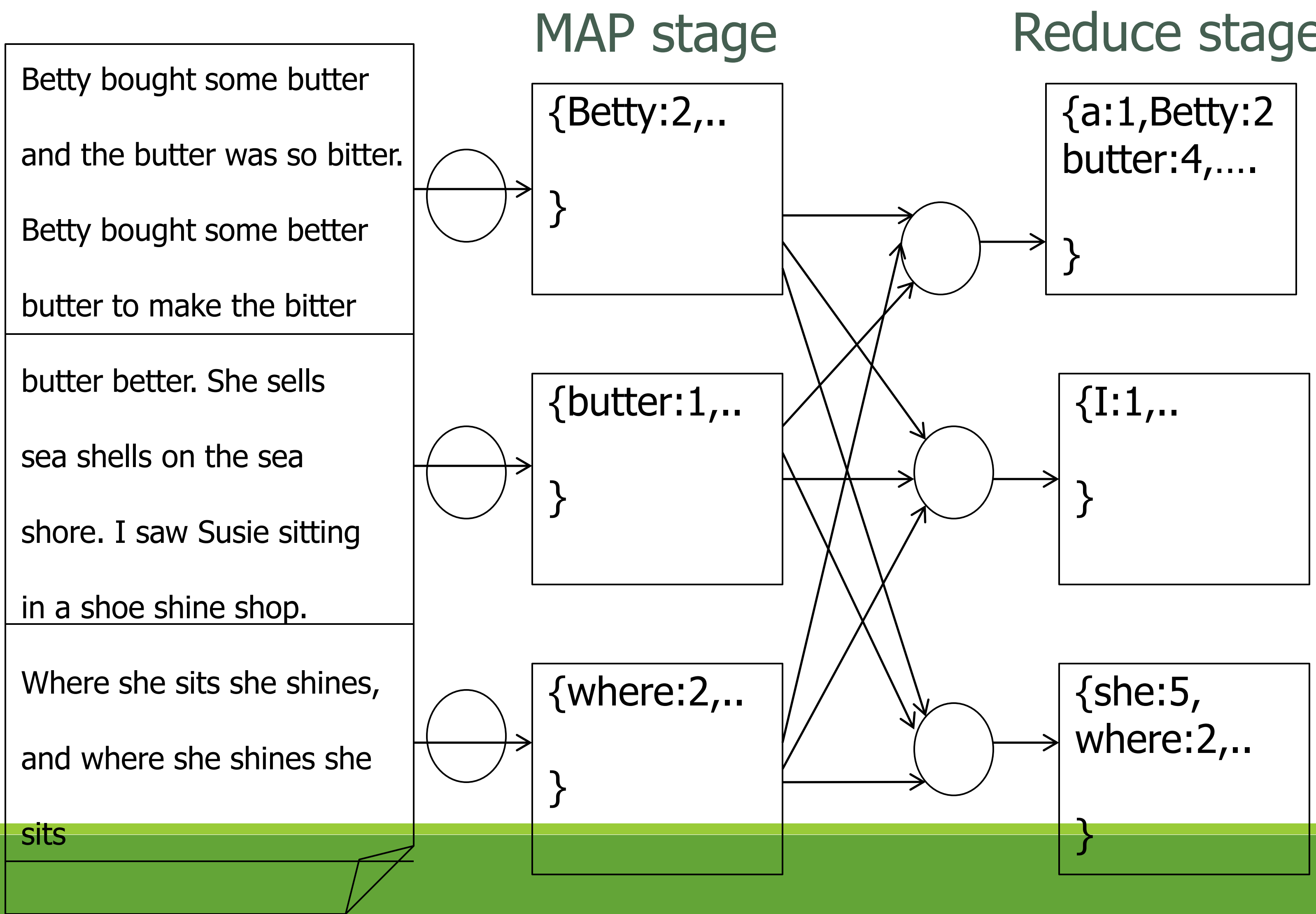
Cluster Computing and Big Data

- Use divide and conquer – partition results on many computers
- Each computer handles a set of words



Cluster Computing and Big Data

- MapReduce approach proposed by Google



Cluster Computing and Big Data

- Challenges in using cluster computing
  - How to divide work across machines?
    - Moving data over network is expensive!
    - Move computation to the data
  - How to deal with failures?
    - Machine fails or machine is too slow
    - Start map or reduce task after machine recovers
    - Works if map or reduce give same output for a given input
    - What if machine dies destroying its partition?

Topics

- Big data challenges
- Big data vs. traditional databases
- Cluster computing and big data

Acknowledgements

Portions of these slides were adapted from external material available under creative commons license CC-BY-NC-SA 4.0. This license grants the ability to share and adapt the material for non-commercial purposes.

Name of the creator: Dr. Anthony Joseph, University of Berkeley and team

Link to material: https://courses.edx.org/courses/BerkeleyX/CS100.1x/1T2015/info