CISC 886- Cloud and Big Data

Anwar Hossain, Ph.D.
Queen's University
Email:
ahossain@queensu.ca
anwar.Hossain@gmail.com

Agenda

- What is big data?
- Properties of big data (the 5 V’s)
- How much data do we generate?
- Why big data?

What is big data?

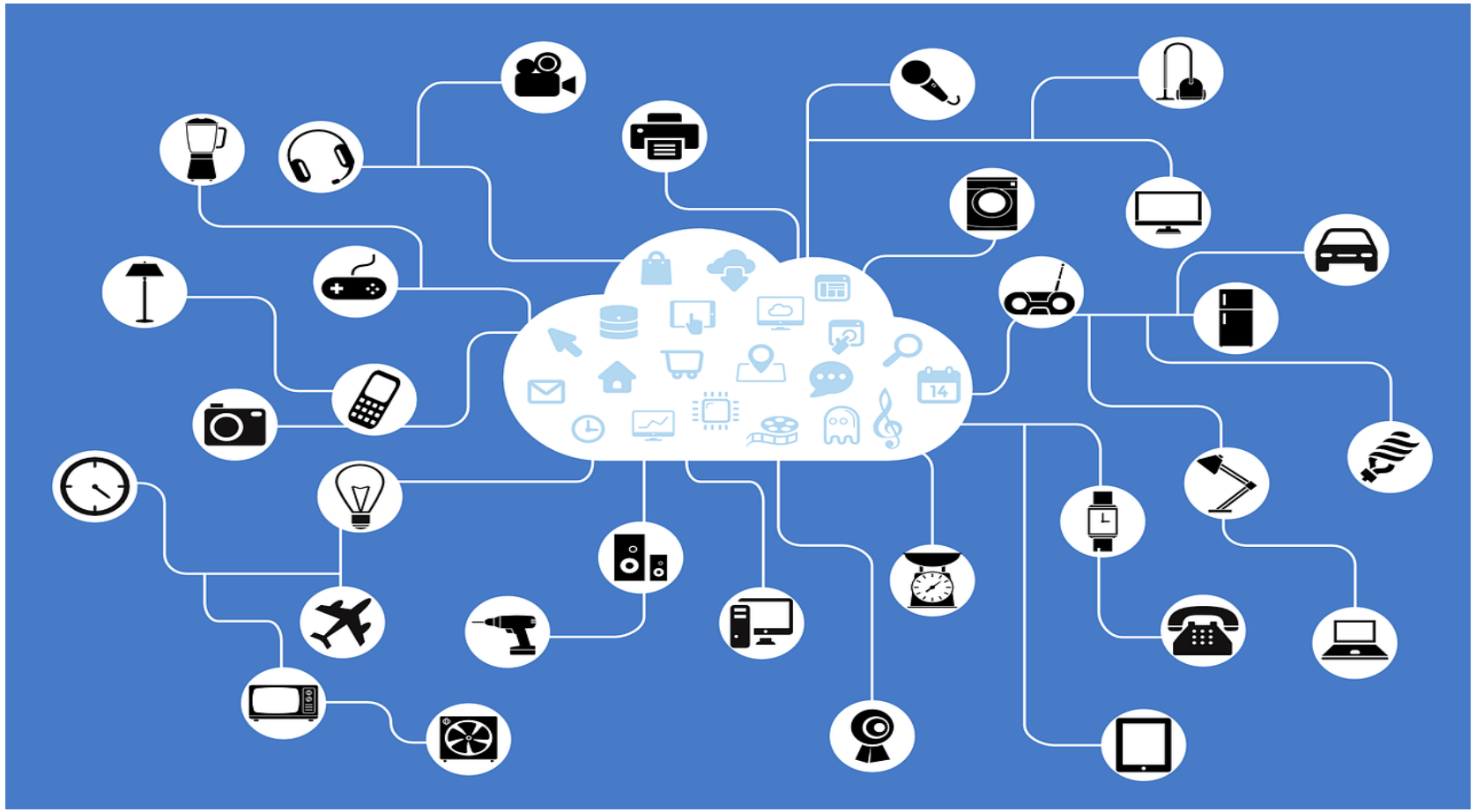
- Big data refers to data sets that are too large or complex to be dealt with by traditional data-processing application software –wiki
- Other definition-- Extremely large data sets that may be analysed computationally to reveal patterns and facts
- Big data for one may not be big enough for others- why?

The 5V's

- Volume
  - No starting limit
  - No end limit
  - Relative term
- Velocity
  - E.g. 1TB/month vs 1TB/day
- Variety
  - Different types and structure
  - Generated by man + machine

Different types of data

Data generation by sensors



Credits: pixabay

5V’s

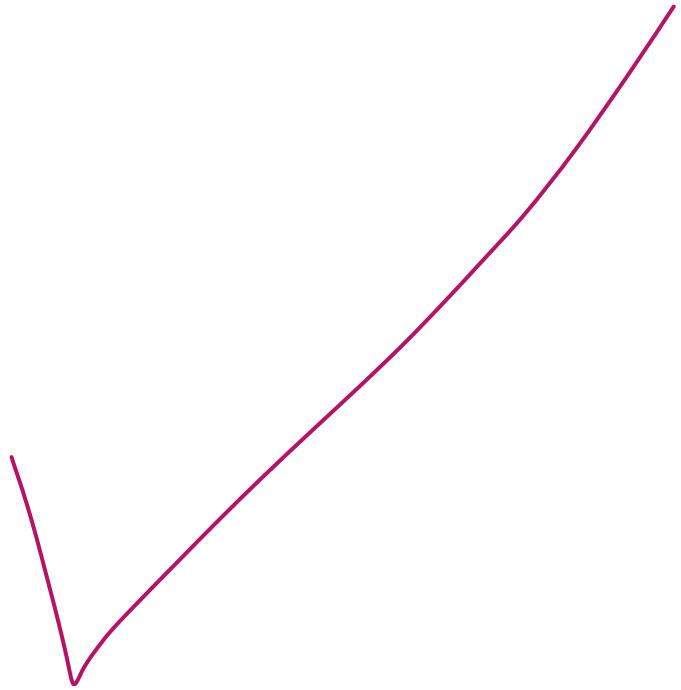
- Veracity
  - Authenticity
  - Trustworthiness
  - …
- Value
  - Intelligence
  - Prediction
  - Decision making
  - Insight
  - Etc.…

How much data do we generate?

Key Data Creation Statistics 2022

- https://earthweb.com/how-much-data-is-created-every-day/

- 1GB of data can create 350,000 emails.
- **2.5 quintillion bytes of data is created every day**.
- Skype has 3 billion minutes of calls per day.
- 5 billion Snapchat videos and photos are shared per day.
- 333.2 billion emails are sent per day.
- 20% of people online watch online games.
- Revenue from Bing is over $7 billion.
- People spend $1 million per minute online.

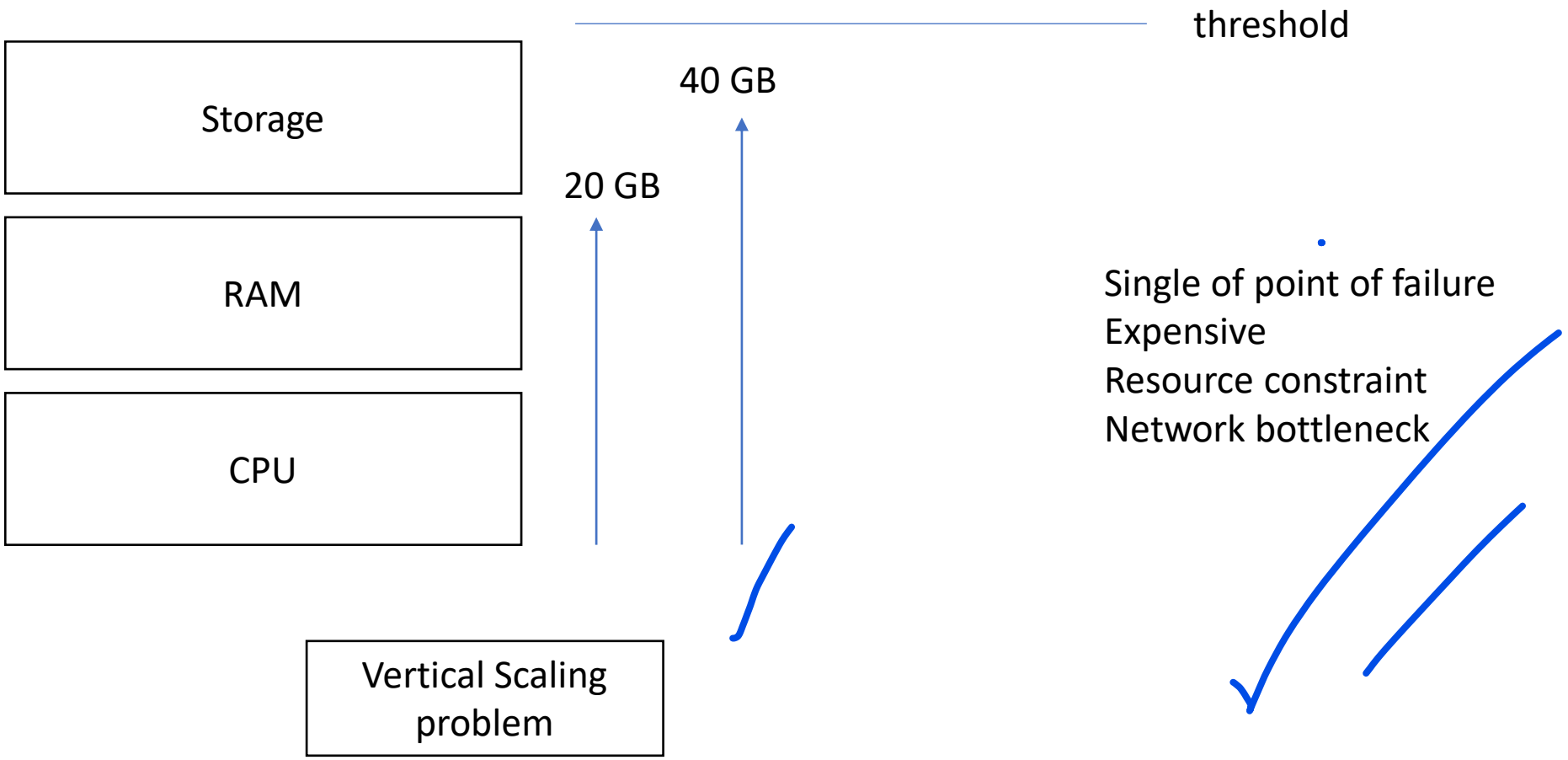


Big organizations generate huge data

Why big data?

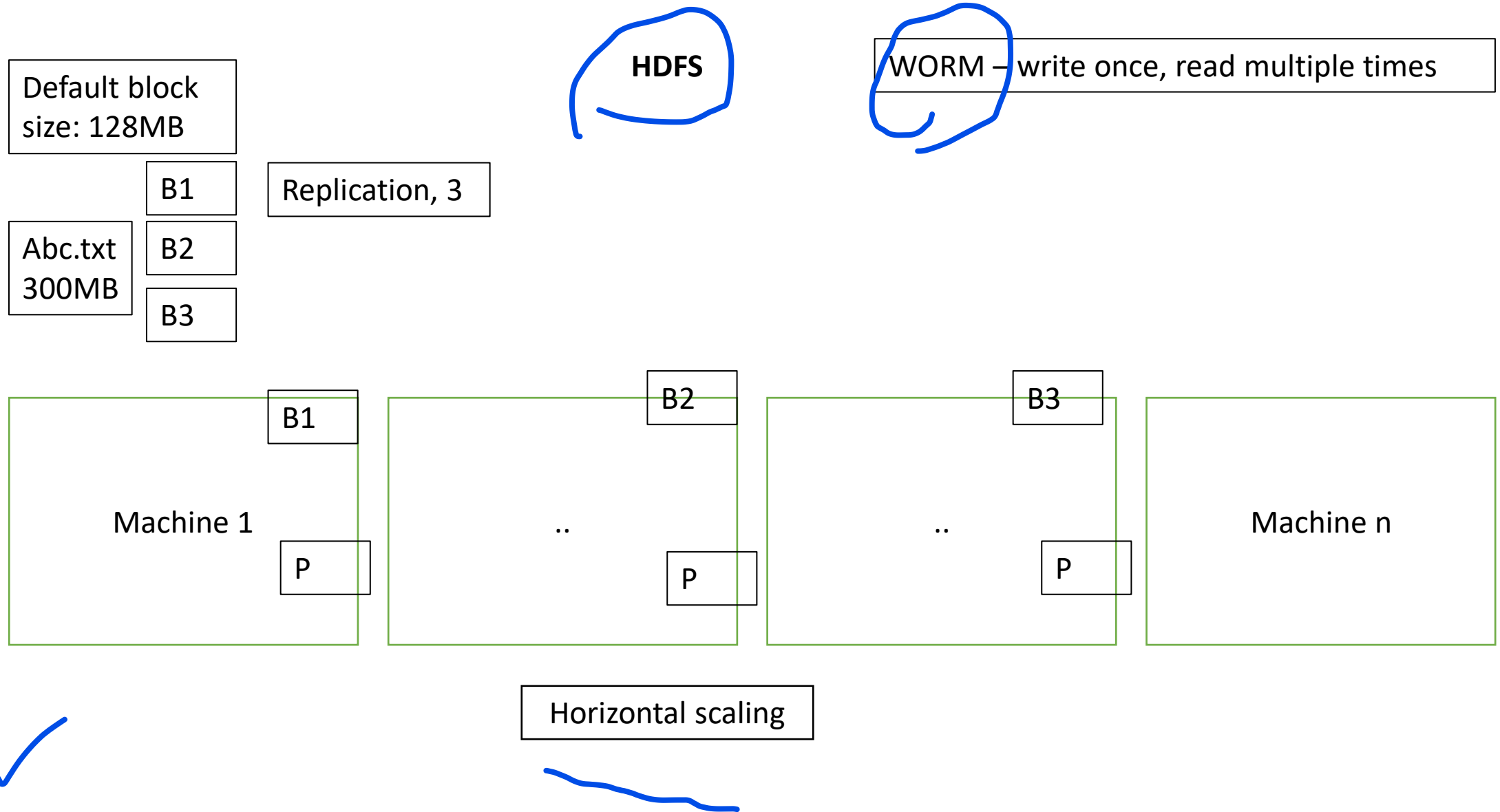
- Science – science experiments, satellite data and imageries
- Business – user behavior, shopping, recommendation
- Community – social network, blogging, twitting
- …

Big data perspective using current system



Single of point of failure
Expensive
Resource constraint
Network bottleneck

How Hadoop can address the issues?



Hadoop offers

- SW– free of cost
- Commodity hardware
- Increased reliability
- Data transfer over the network → Parallel processing/ distributed computation
- Vertical scaling → horizontal scaling

Hadoop works – rack awareness

HDFS: WORM – write once, read multiple times

Default block size: 128MB

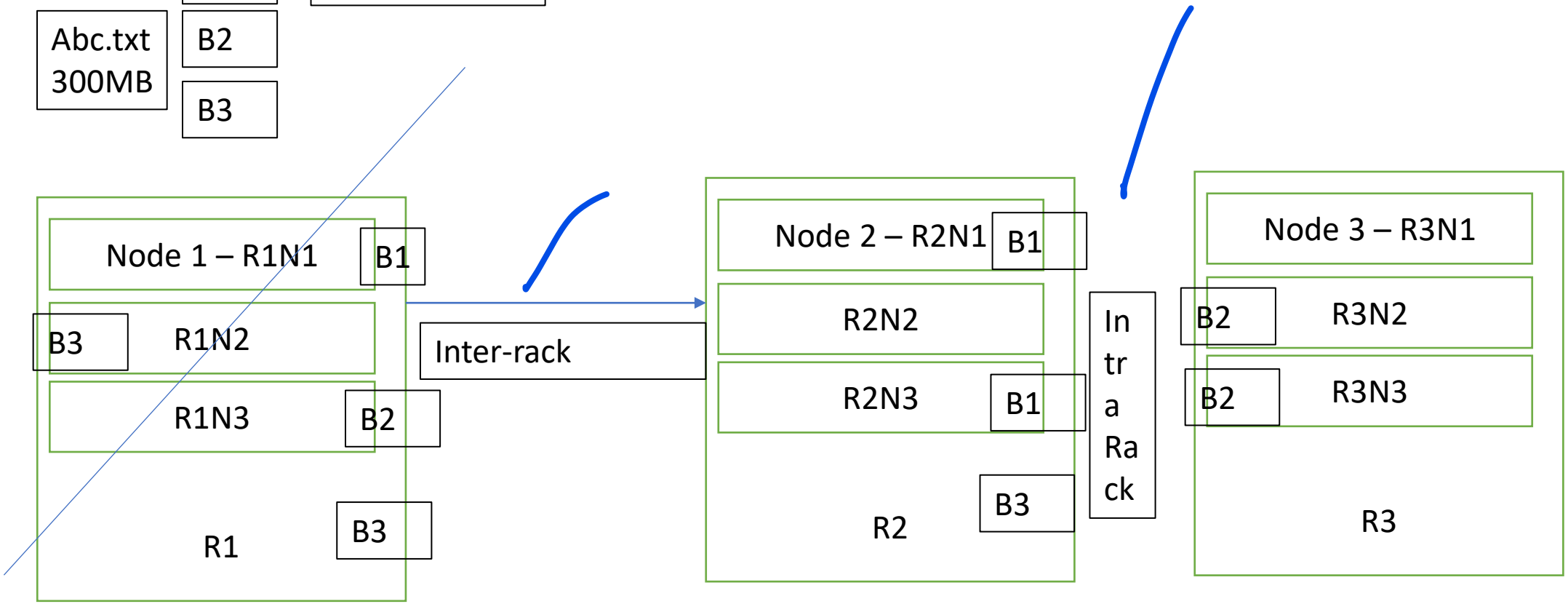
Abc.txt 300MB

B1

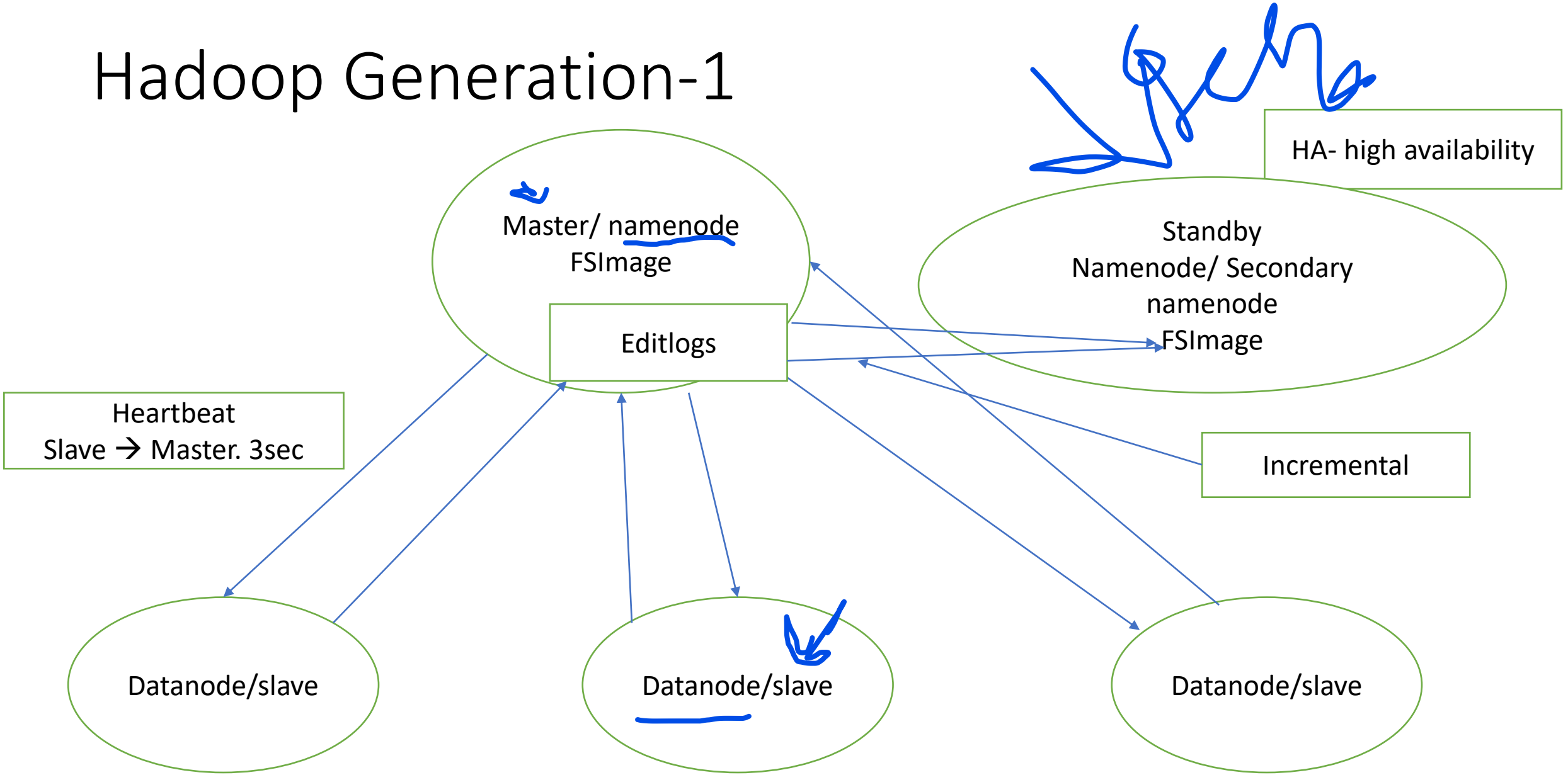
B2

B3

Replication, 3



Hadoop Generation-1



End-to-end write and read

- 1. Create a folder in local FS– mkdir
- 2. Create a file (import) in LFS – vi
- 3. Create a directory in HDFS
  - Hdfs dfs –mkdir
- 4. LFS→ HDFS (write anatomy)
  - Hdfs dfs –put lfspath hdfspath
- Read the file from HDFS (read anatomy)
  - Hdfs dfs –cat hdfspath

Hadoop consists of

Storage
HDFS
Master: Namenode
Slave: Datanode

Outcome:
- Infinite scalability
- distributed storage
- Distributed processing
- reliability
- cheap

Processing
Hadoop Gen1
Master: Job Tracker
Slave: Task tracker

Further Version
YARN Processing
Master: Resource Manger
Slave: Node Manager

Hadoop ecosystem

- HDFS- Hadoop distributed file system
- YARN –Yet another resource negotiator
- MapReduce – programming based data processing
- Spark – In-memory data processing
- PIG, HIVE – query based processing
- Hbase – NoSQL
- Zookeeper- Manager cluster
- Oozie- Job scheduling
- …

MapReduce Framework

- No. of Mappers = No. of blocks (# of input split)
- 1 Mapper → 1 Block
- Line by line processing occurs
- At a time → 1 Line
- Mapper input:
- Key → Line number
- Value→ Complete line
- Output of the mapper→ Key, Value

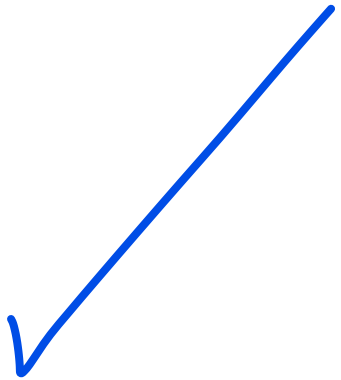
Abc.txt

Hello how are you

how is life

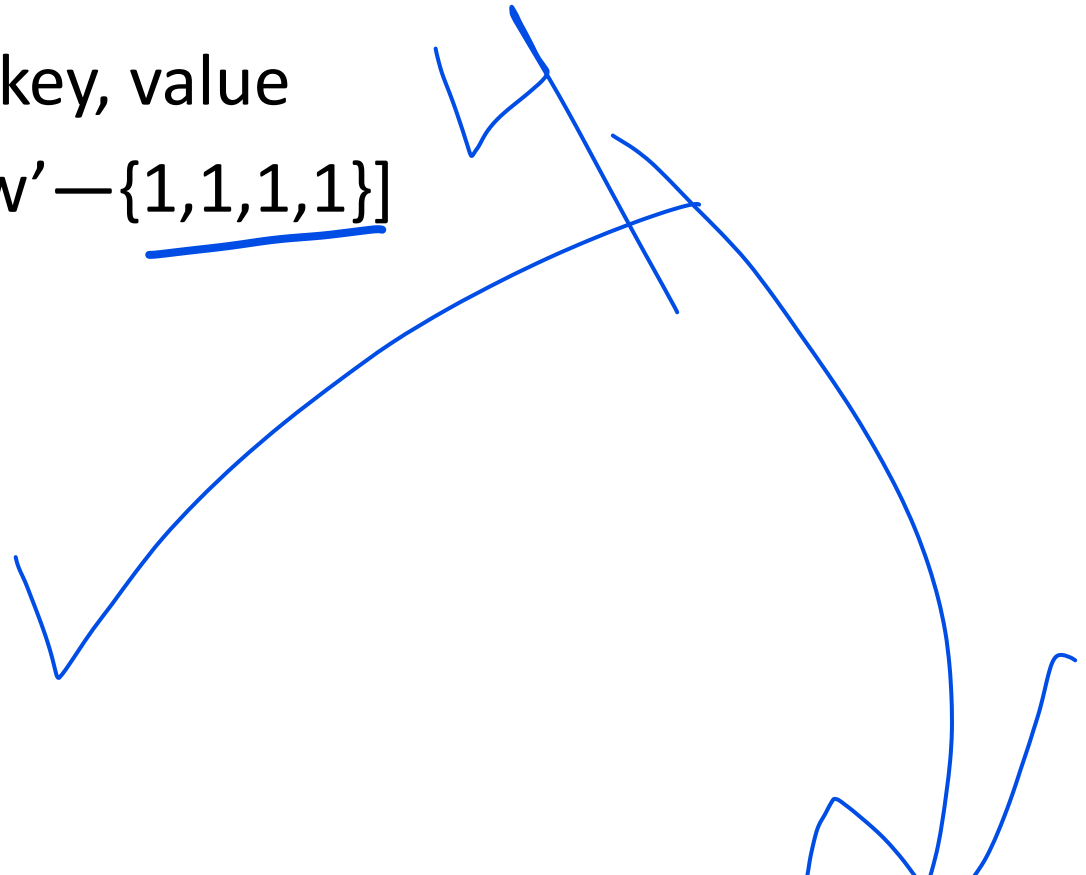
how is the weather

how is it going



Sort and Shuffle

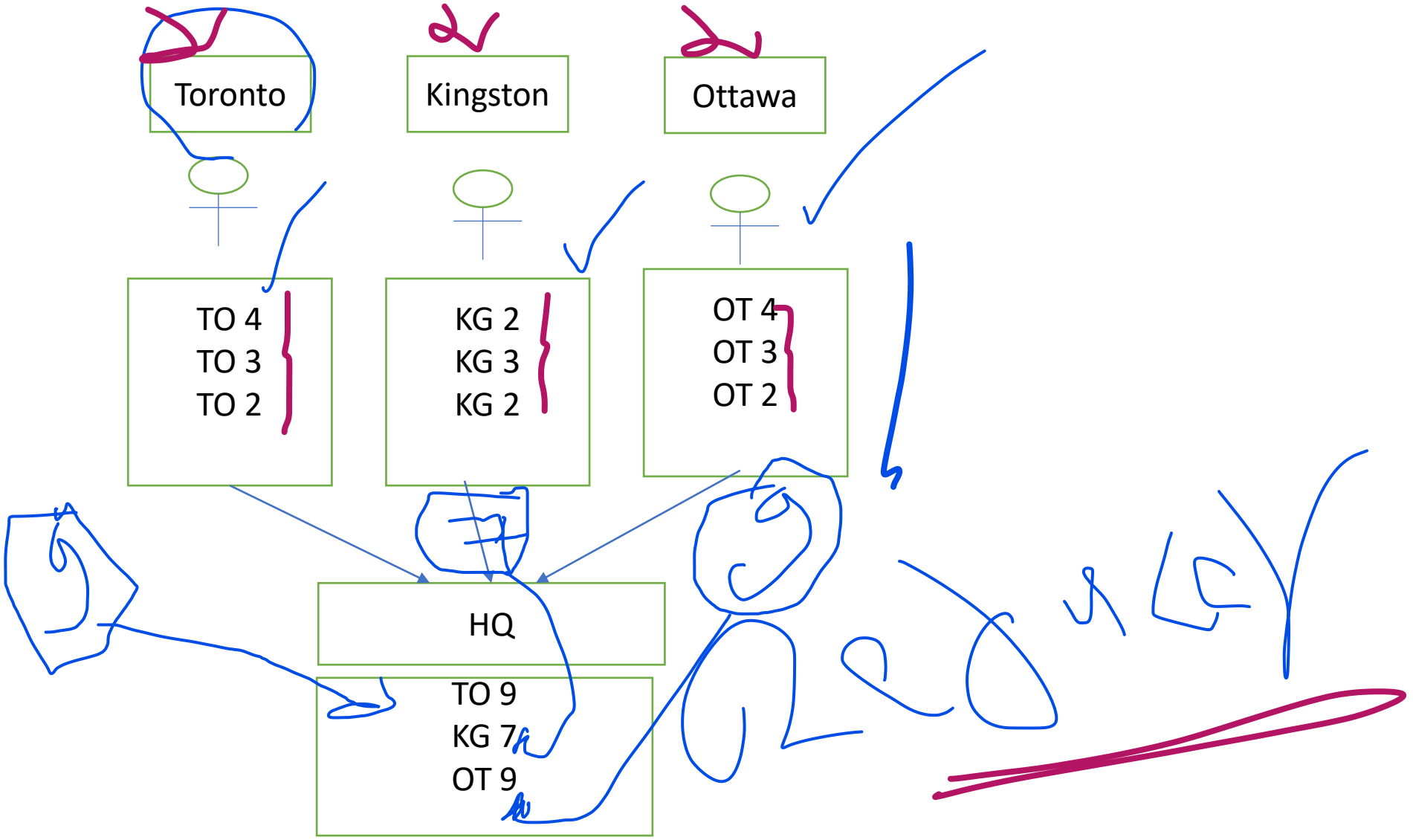
- Sort on the key
- Shuffle on the value
- Input (output of the mapper) → key, value
- Output → key, list of values [‘how’—{1,1,1,1}]



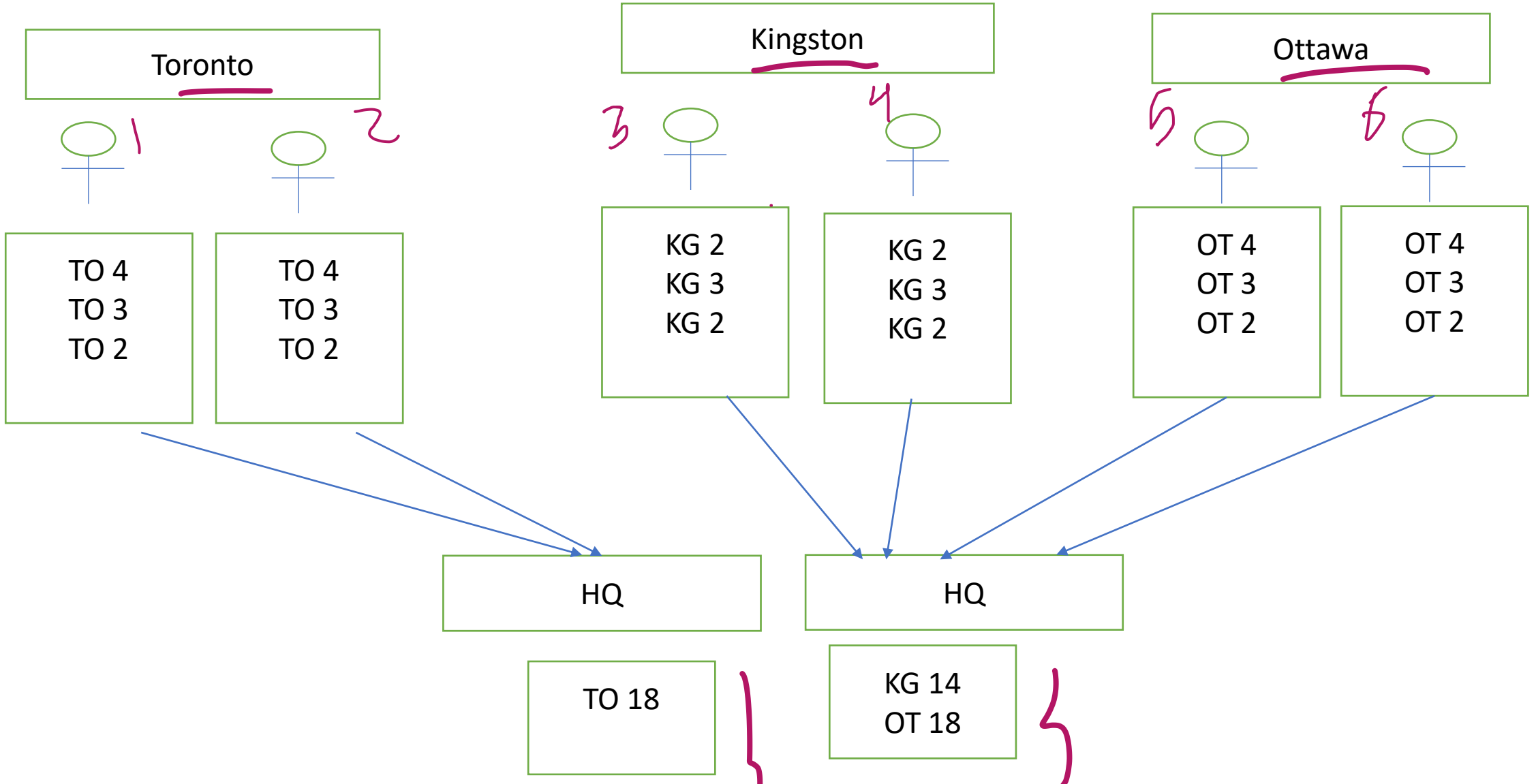
Reducer

- Input → Key, List of values [how—{1,1,1,1}] (output of sort and shuffle)
- Output → Key, value [‘how’, 4] (final output)

Census example (with limited resource)



Census example (with more resource)



With One Reducer