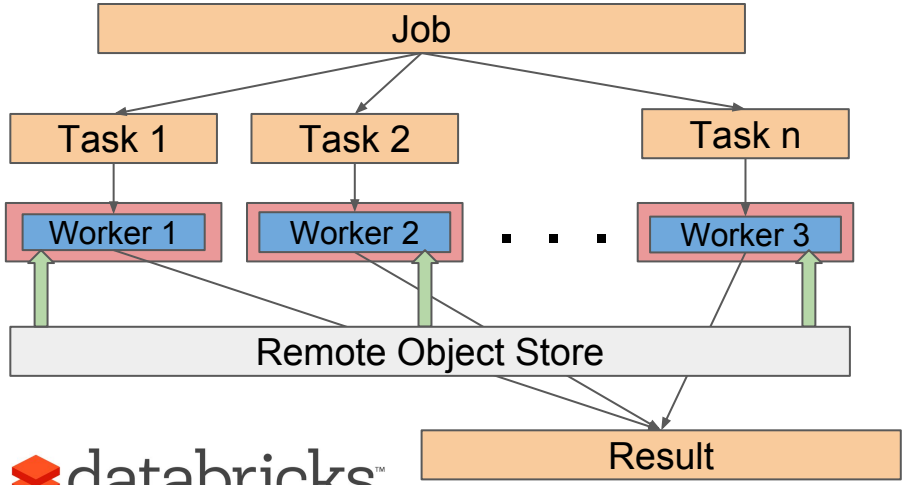
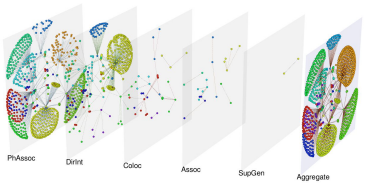
# Characterization of a Big DataStorage Workload in the Cloud

**Sacheendra Talluri**
Alicja Łuszczak
Cristina L. Abad
Alexandru Iosup

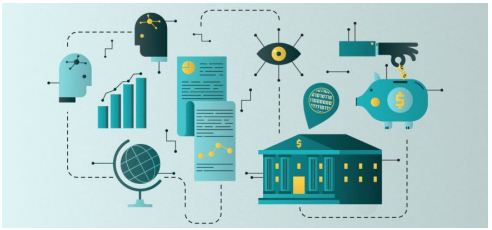



TU Delft VU Vrije Universiteit Amsterdam ESPOL databricks

# What is Big Data?




Google



LIGO

**Medicine** **Search** **Finance** **Science**

- Volume
- Velocity
- Variety
- ...

Processed by multiple machines concurrently.

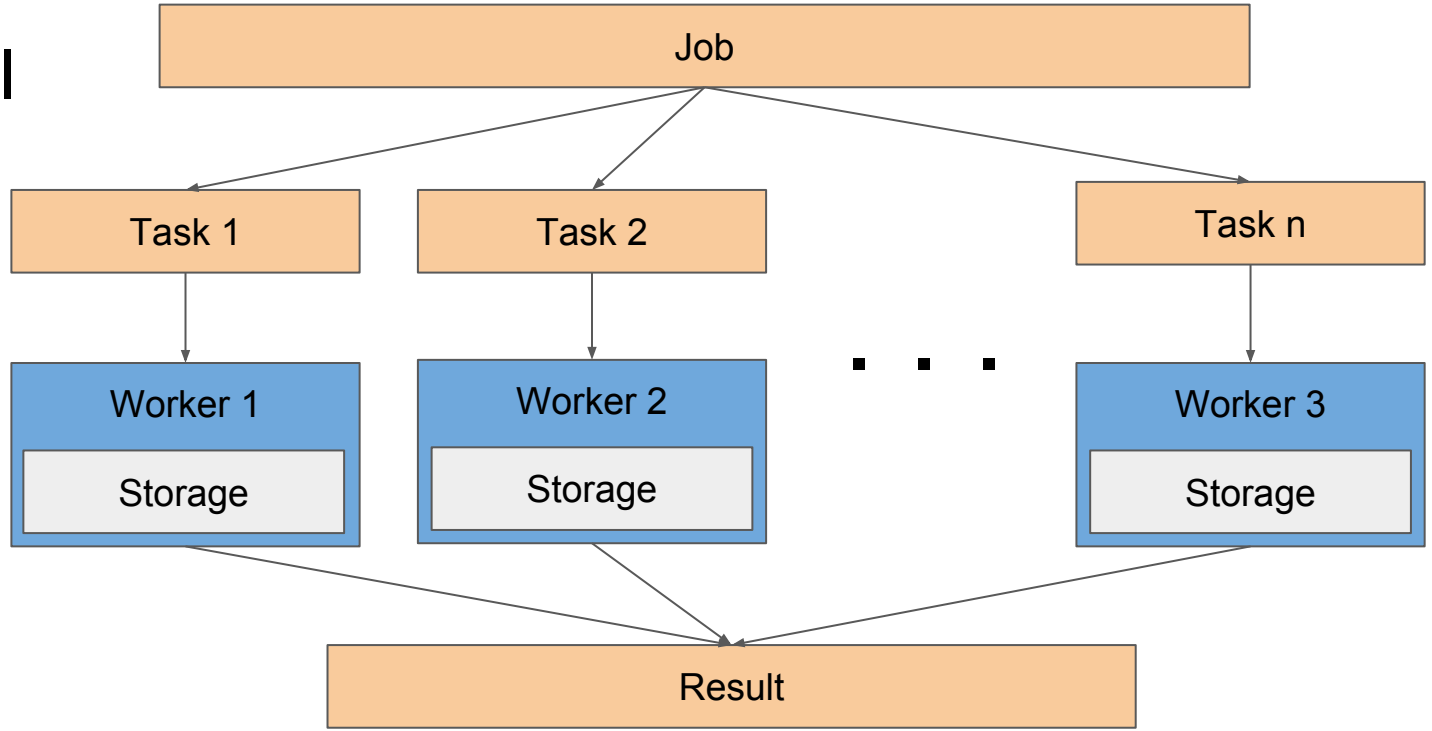
[1] Human HIV-1 protein interaction network, Wikimedia Commons
[2] Google Inc.
[3] Samsung Insights
[4] LIGO Scientific Collaboration

# Traditional Big Data Processing System Model

Uses local storage

1,000s of servers

Few can afford this

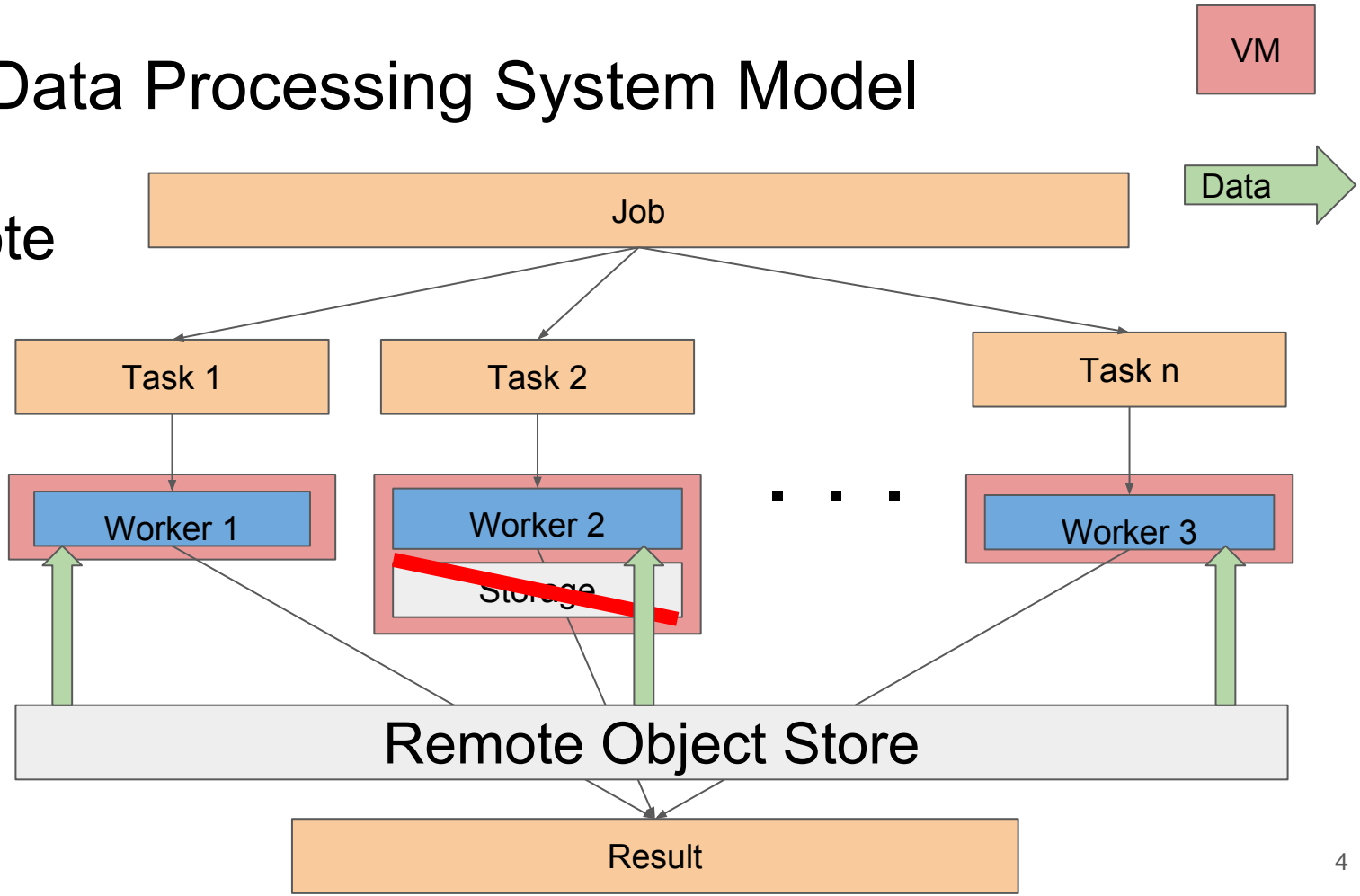


# Our Big Data Processing System Model

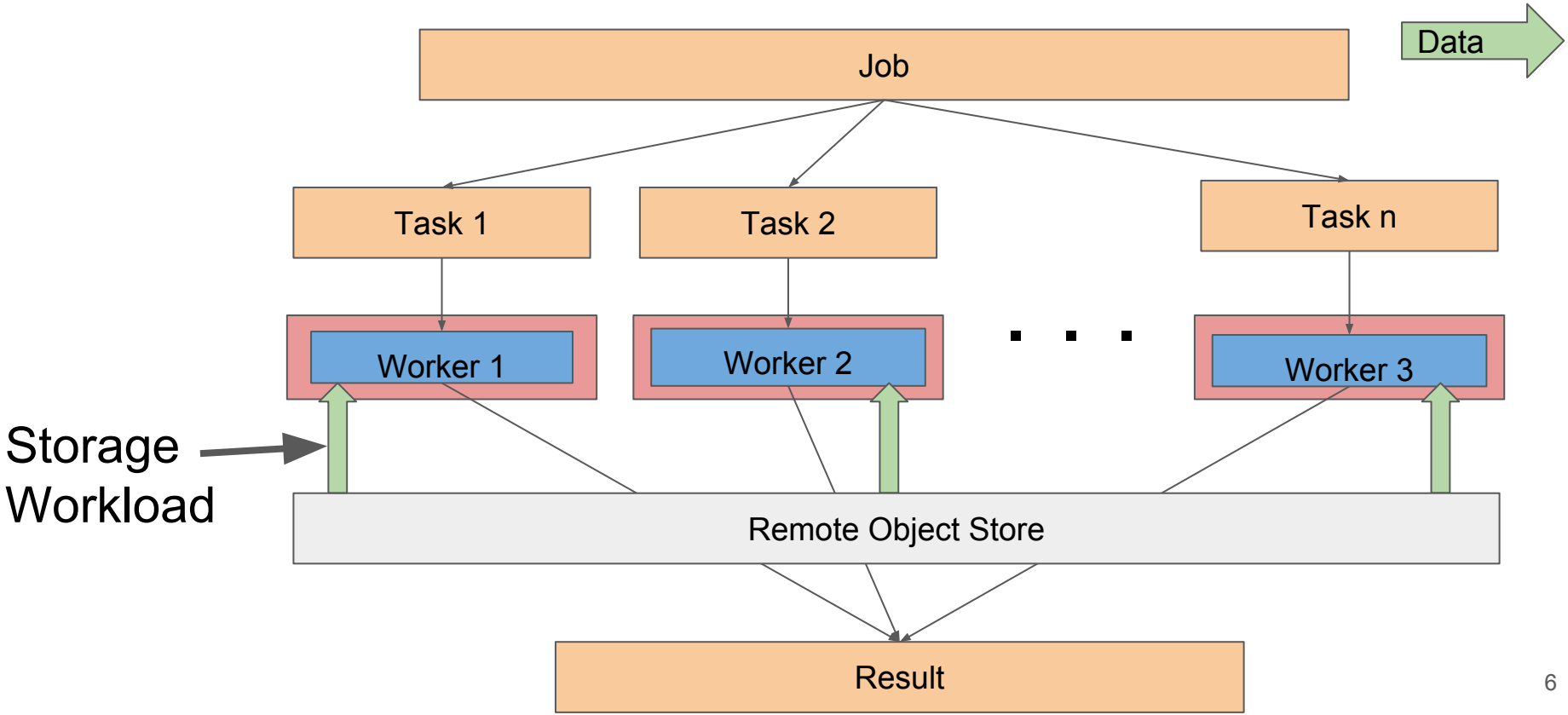
Uses remote storage

10s of ephemeral servers

Many can afford this



# Key Terms

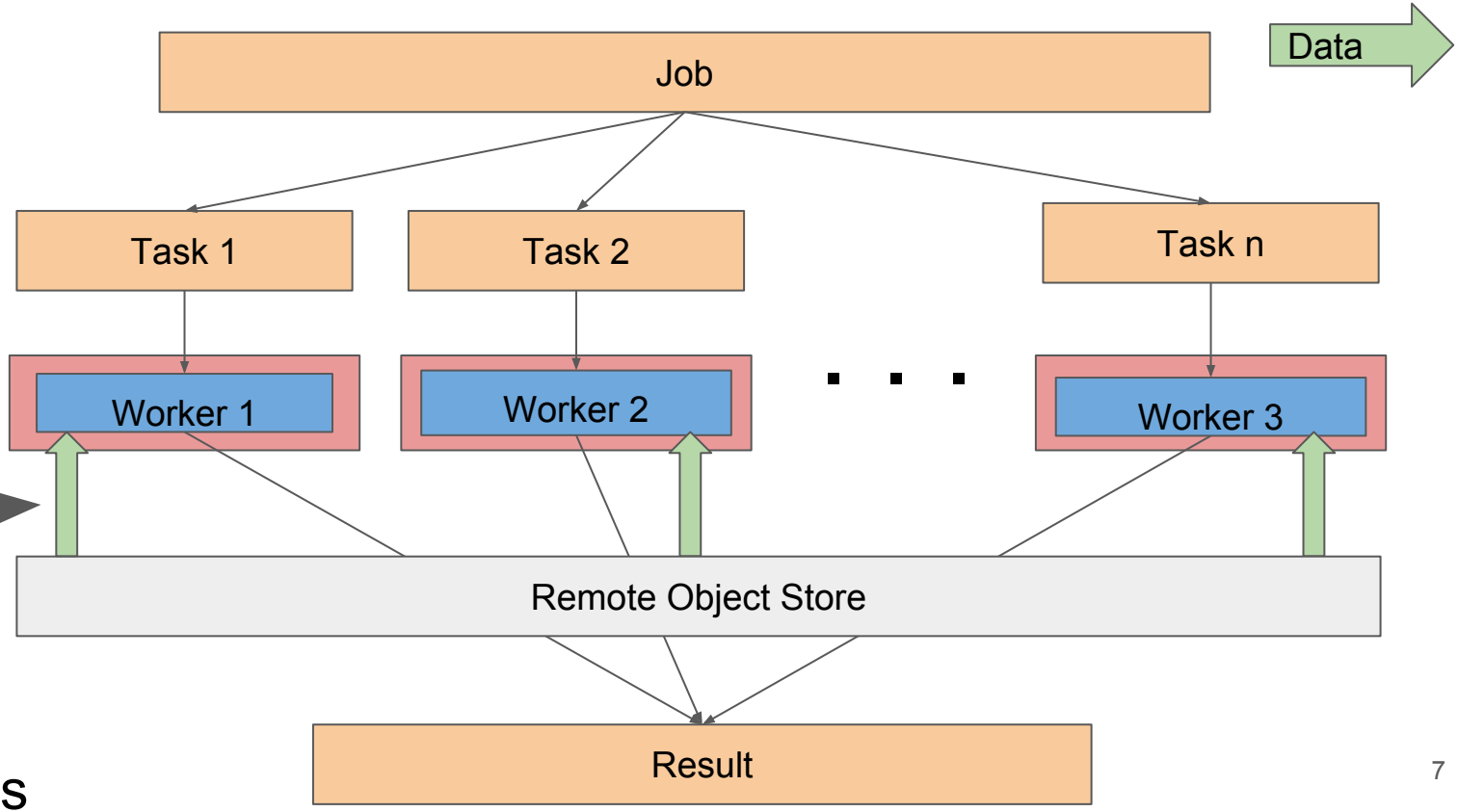

Data
Job
Task 1
Task 2
Task n
Worker 1
Worker 2
Worker 3
Storage Workload
Remote Object Store
Result

# Key Terms

Read: a small subset of data transferred together

Storage Workload

Trace: record of some characteristics



# Motivation

Scarcity of publicly available information about Spark based Workloads

| | Type |
|---|---|
| Chen 2011 | Enterprise |
| Carns 2011 | HPC |
| Abad 2012 | MapReduce + HDFS |
| Chen 2012 | MapReduce + HDFS |
| Atikoglu 2012 | Web Cache |
| Liu 2013 | Consumer Cloud |
| Harter 2014 | Messaging + HDFS |
| Gunasekaran 2015 | HPC |
| Summers 2016 | Video Delivery |
| **This Work** | **Spark + S3** |

# Motivation

Scarcity of publicly available information about Spark based Workloads

| | Type |
|---|---|
| Chen 2011 | Enterprise |
| Carns 2011 | HPC |
| Abad 2012 | MapReduce + HDFS |
| Chen 2012 | MapReduce + HDFS |
| Atikoglu 2012 | Web Cache |
| Liu 2013 | Consumer Cloud |
| Harter 2014 | Messaging + HDFS |
| Gunasekaran 2015 | HPC |
| Summers 2016 | Video Delivery |
| **This Work** | **Spark + S3** |

# Motivation

Scarcity of publicly available information about Spark based Workloads

# Research Question

What are the characteristics of Spark-based big data storage workloads in the cloud?

# Motivation

Scarcity of publicly available information about Spark based Workloads

# Research Question

What are the characteristics of Spark-based big data storage workloads in the cloud?

# Impact

**Speculation**

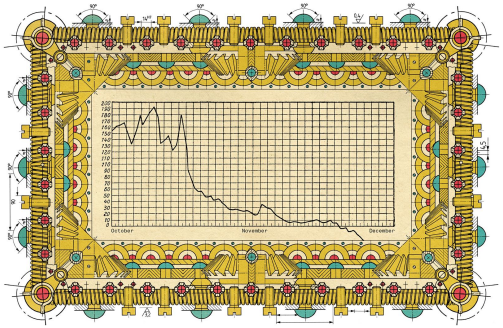
Enables new system designs, better tuning, and operations decisions

# Generalizable

- Databricks is large
- Customers from different sectors
- Applicable to spark-like stage based big data processing systems which read data from cloud object stores

# Process

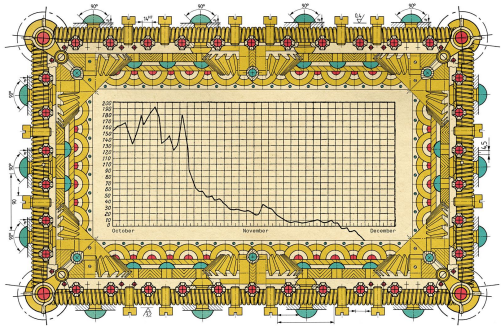
1. Collected real traces from Databricks



[1] Christian Gralingen

# Process

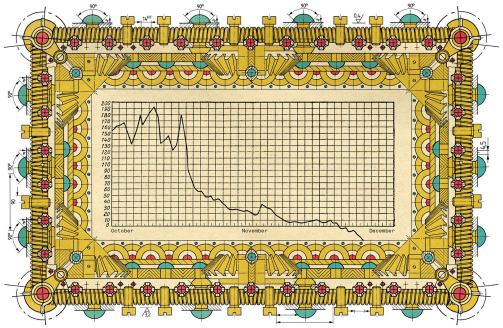
1. Collected real traces from Databricks
2. User data was not touched



[1] Christian Gralingen

# Process

1. Collected real traces from Databricks
2. User data was not touched



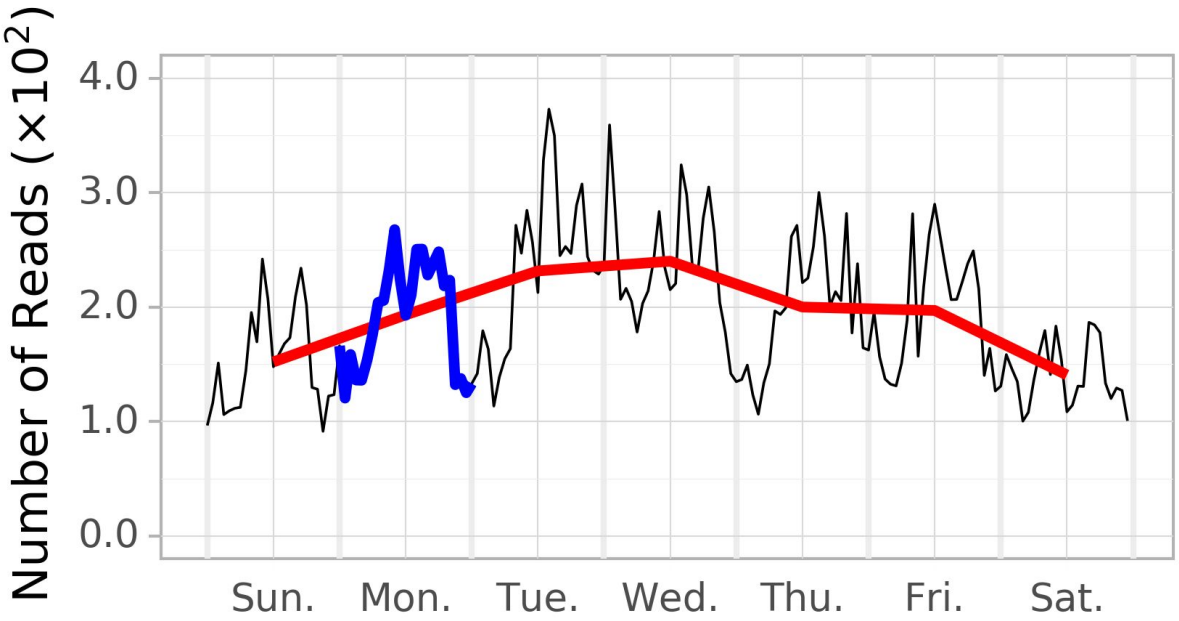
3. Anonymize and process large traces
4. Long-term trends via visualization
5. Bursts using Hurst exponent
6. Statistical characterization
7. File types

[1] Christian Gralingen

# Trace Subsets

- Whole Trace
  - 6 months long
  - 600TB data
- W1
  - 1 week long
  - First week of the six month trace
- W2
  - 1 week long
  - Last week of the six month trace
- First and last week to show that properties don’t change over time.

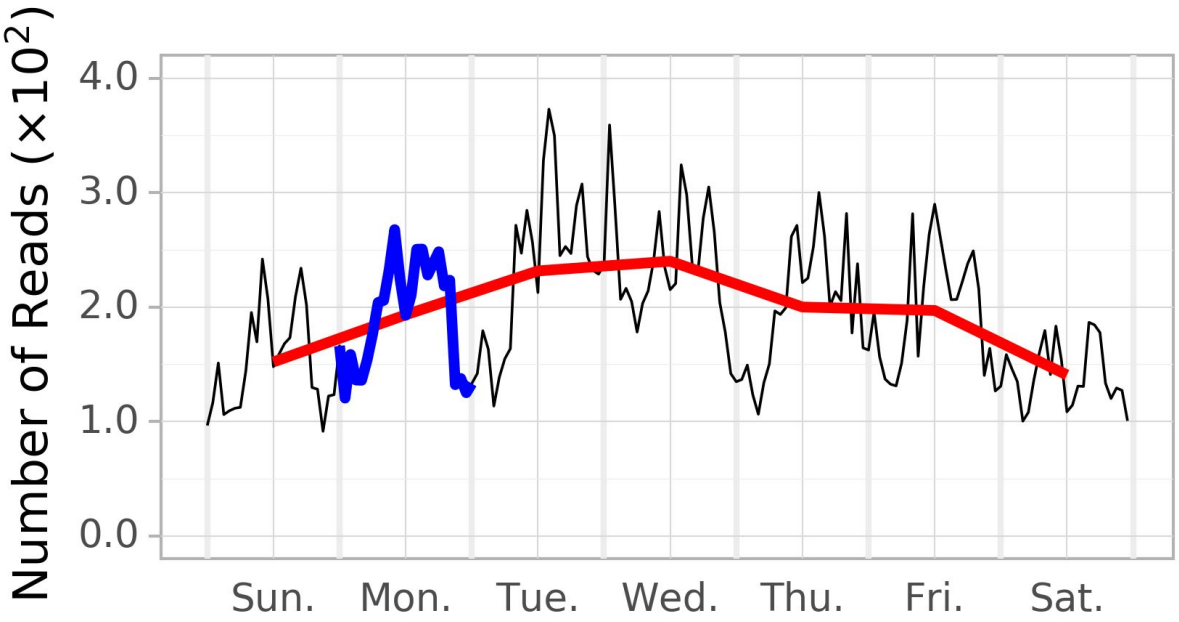
# Daily and Weekly Trend




**Weekly**

**Daily**

# Daily and Weekly Trend

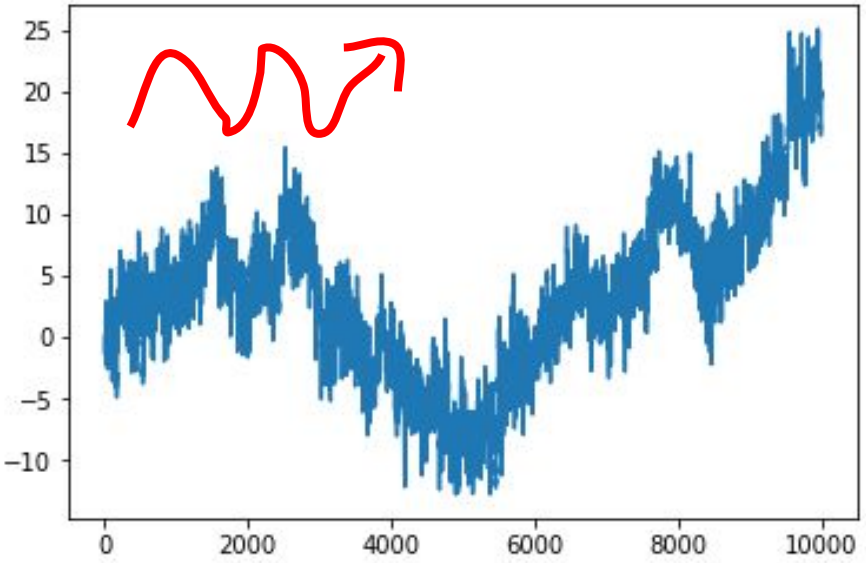



**Weekly**

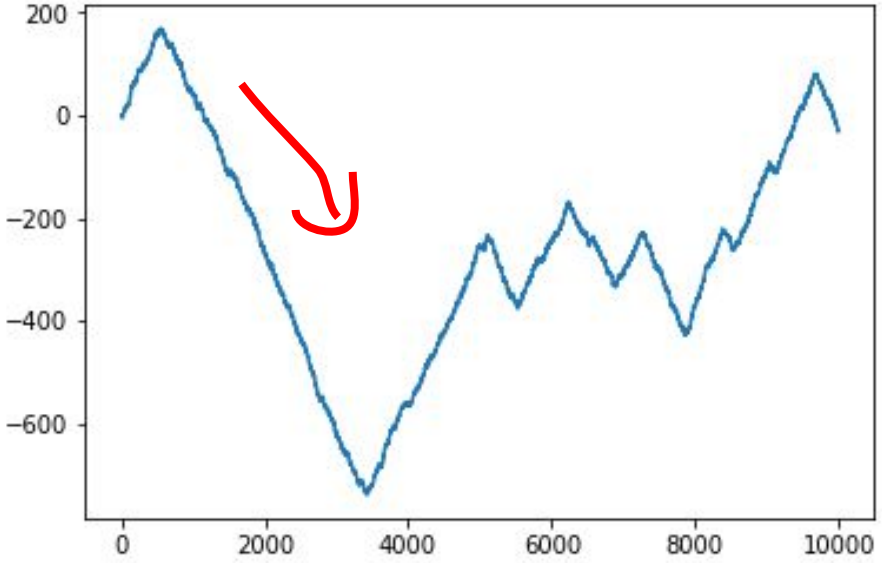
**Daily**

**Distribute demand evenly across days for lower EC2 spot instance prices**

# Bursts Quantified by Hurst Parameter (H)



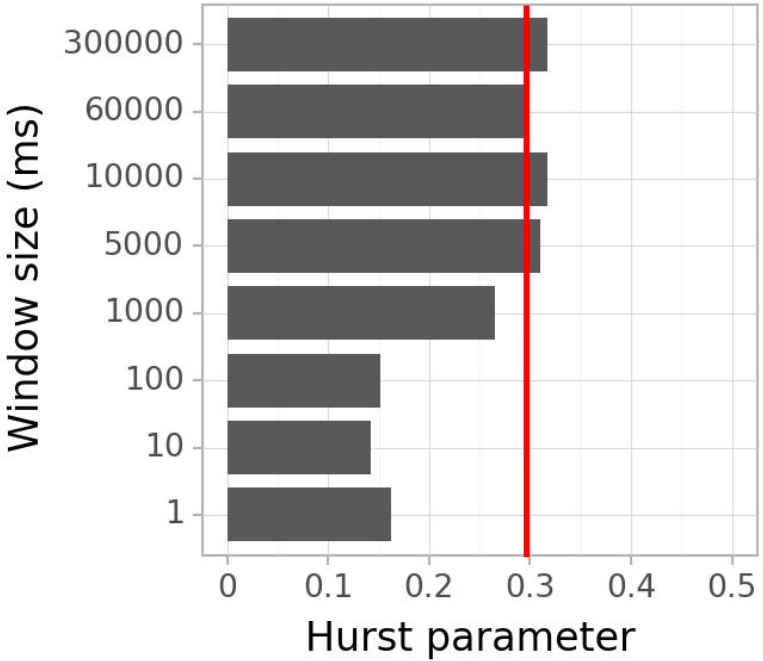

Tends to move in the opposite direction ($H < 0.5$)

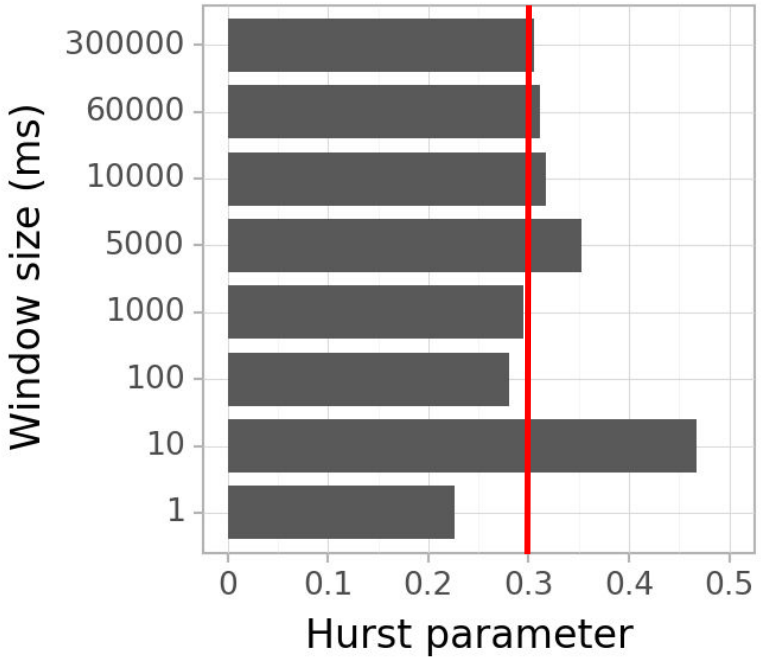



Tends to move in the same direction ($H > 0.5$)

Different kinds of bursts

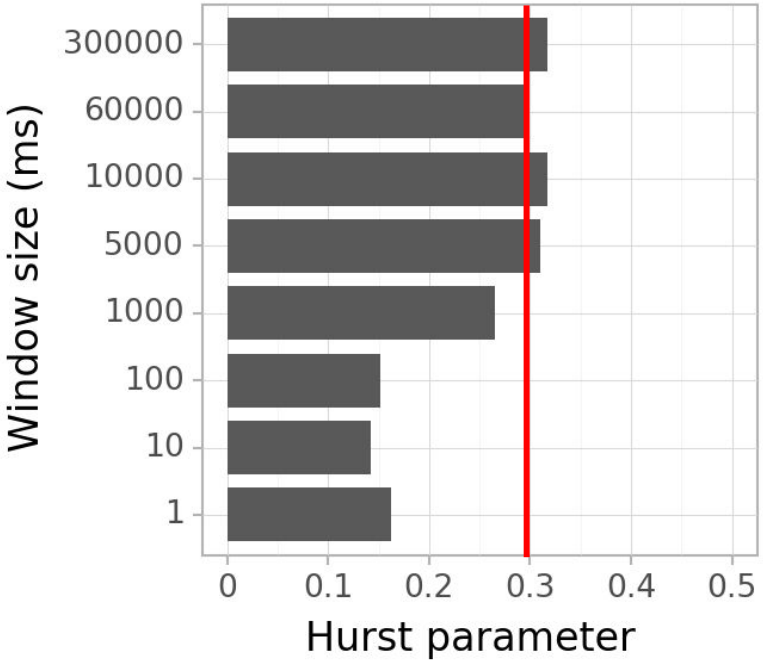
# Bursts Quantified by Hurst Parameter

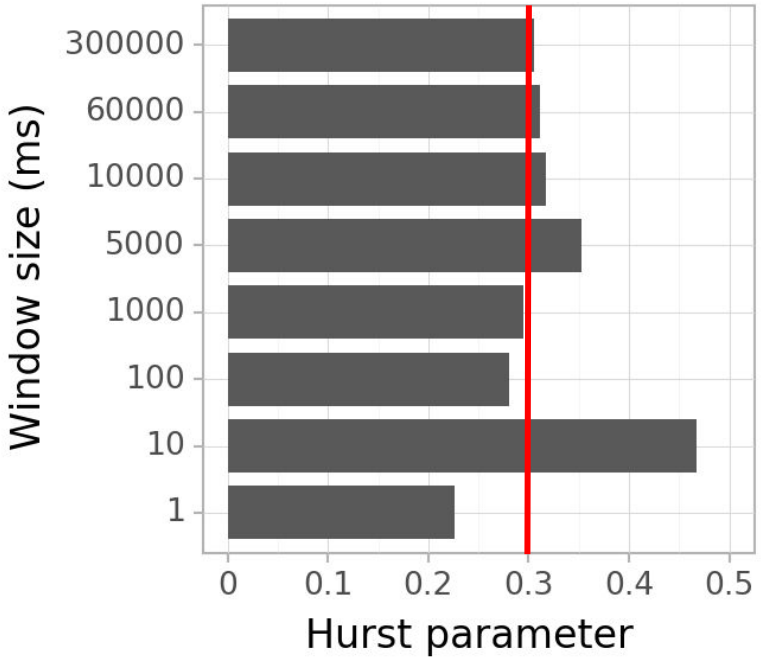



< 0.5 means sporadic bursts
Close to 0.3 most of the time

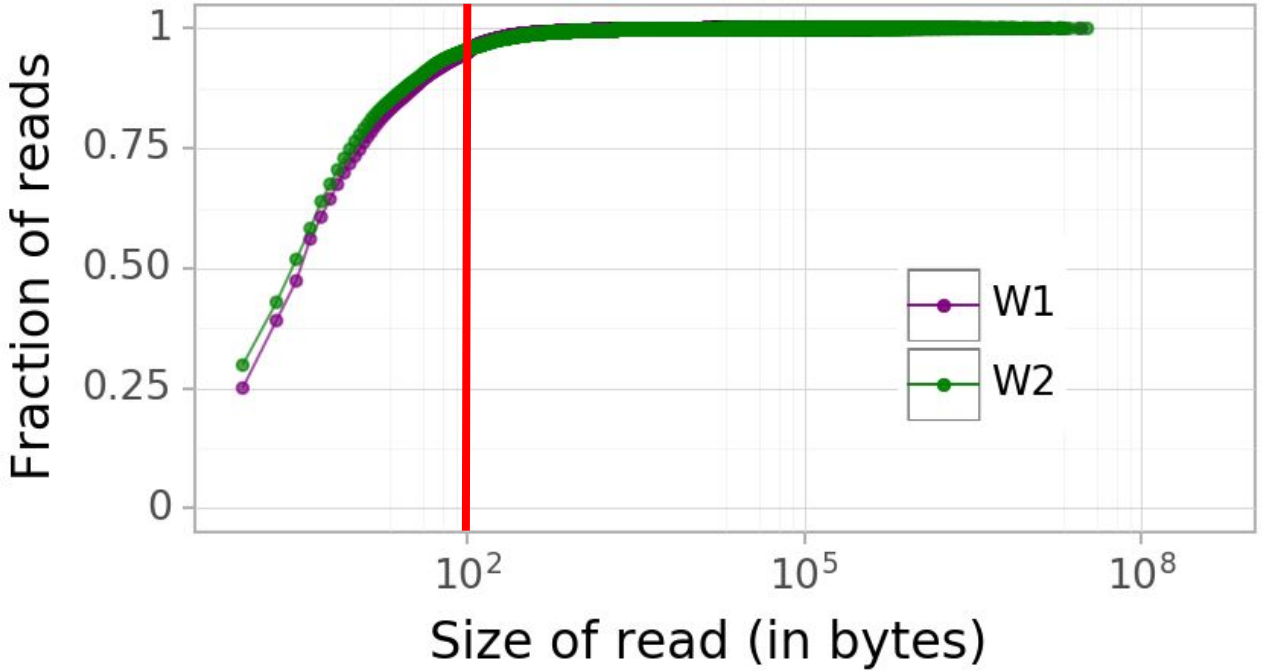
# Bursts Quantified by Hurst Parameter



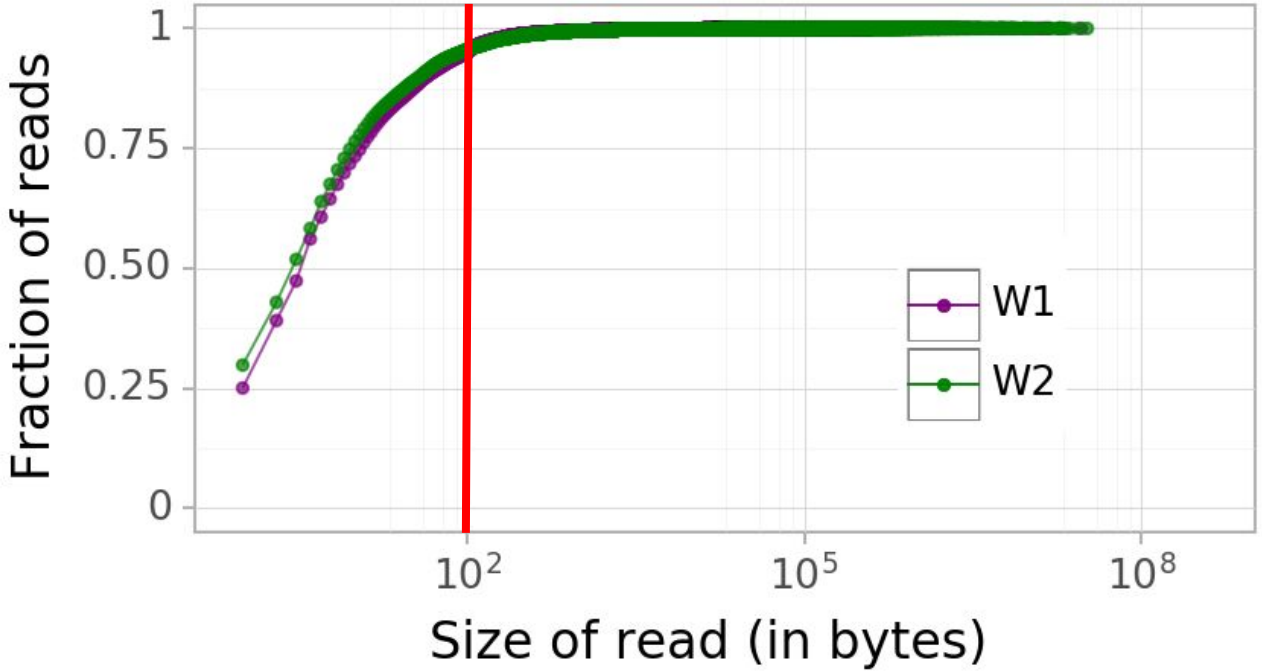




**Prefetch or other activity in-between bursts**

# 90% reads are small in size



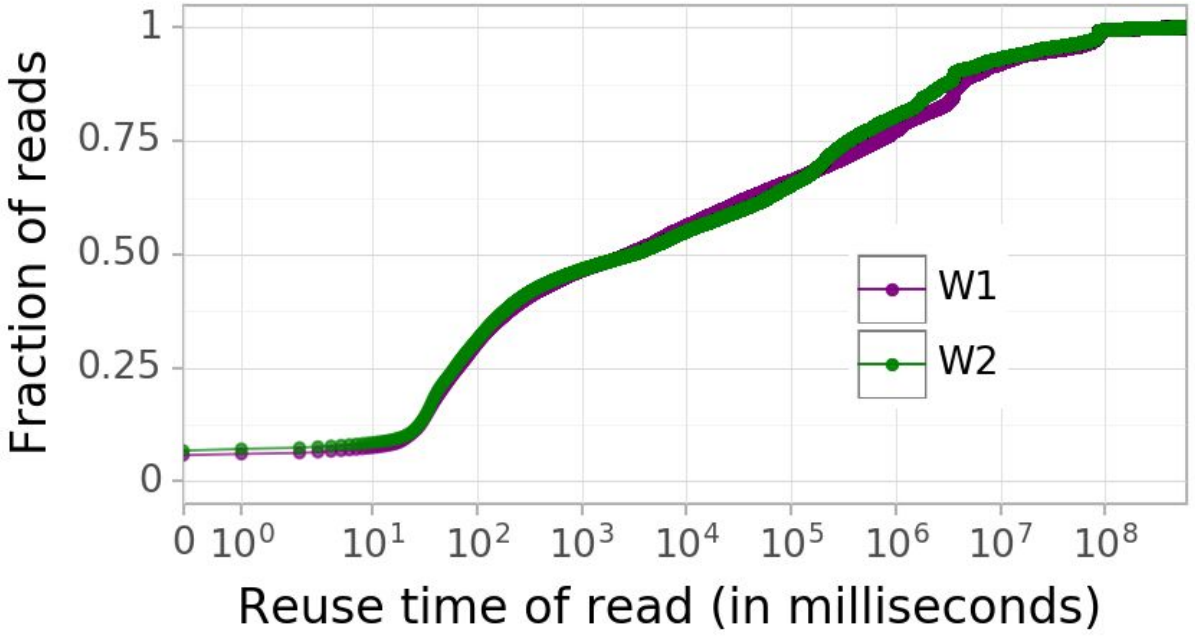
# 90% reads are small in size




**Likely metadata. Store metadata in a faster storage, separate from data.**

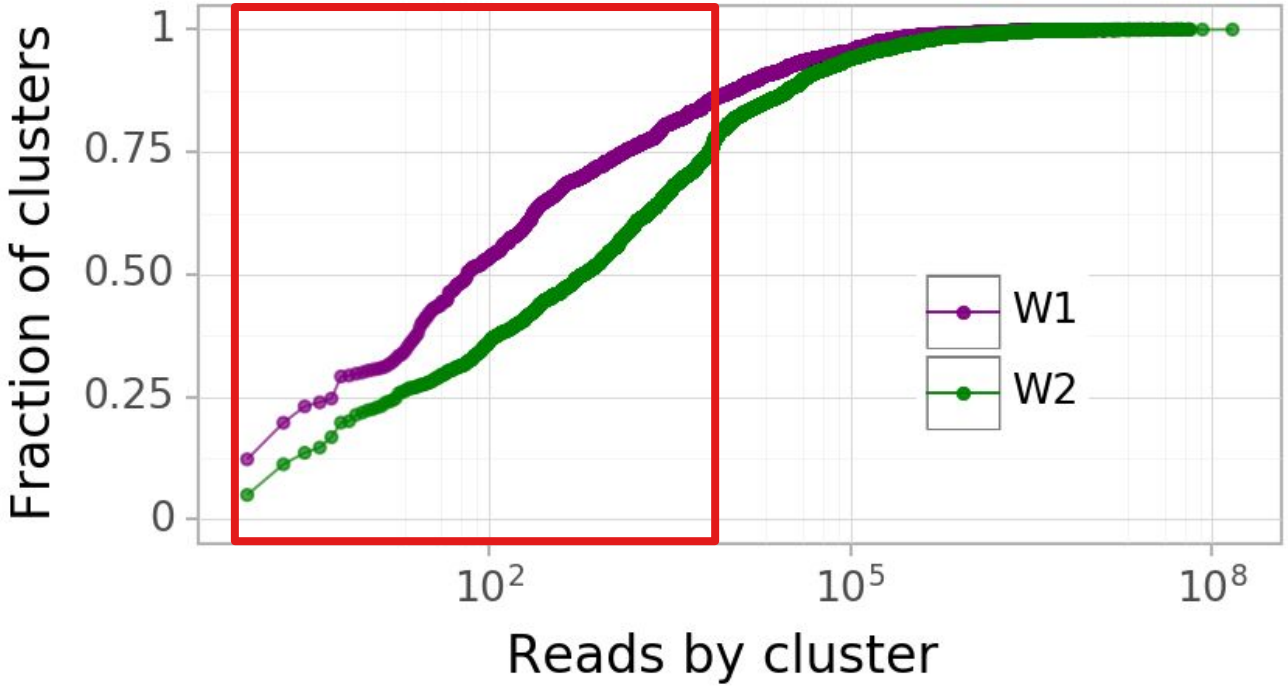
# Reuse time

# Reuse times are logarithmically distributed

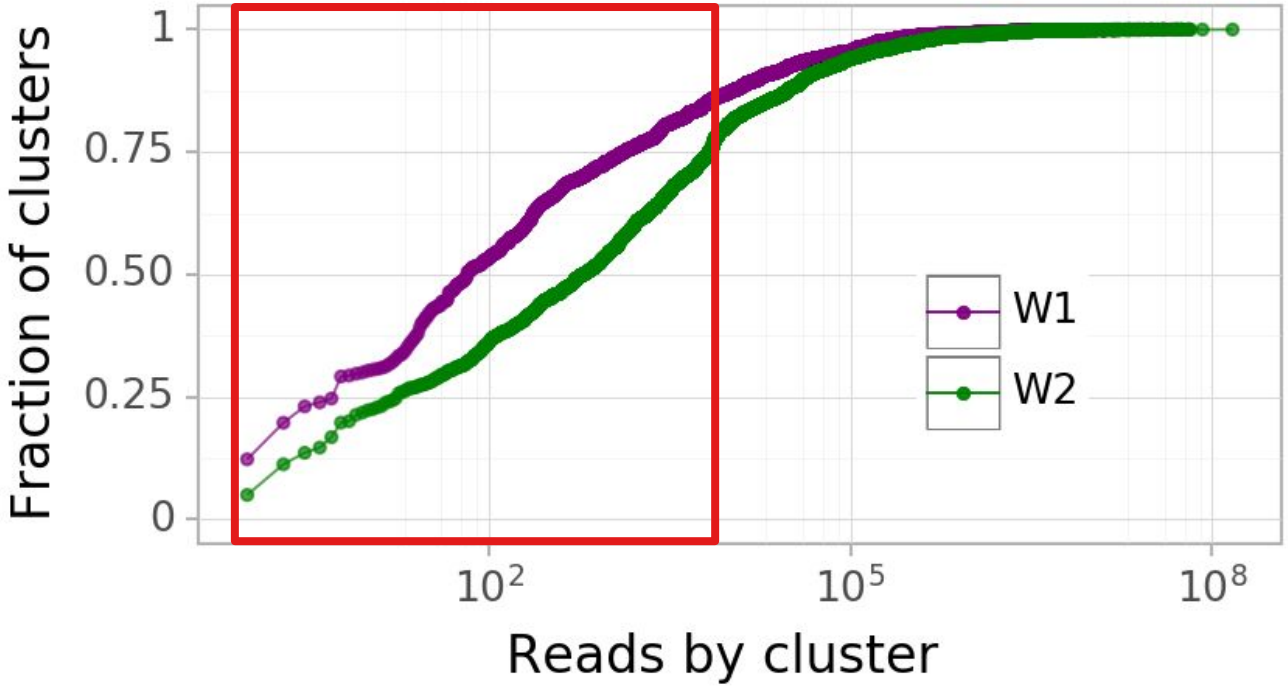



Each logarithmic interval has the same number of reads

# Most clusters (75%) read few files

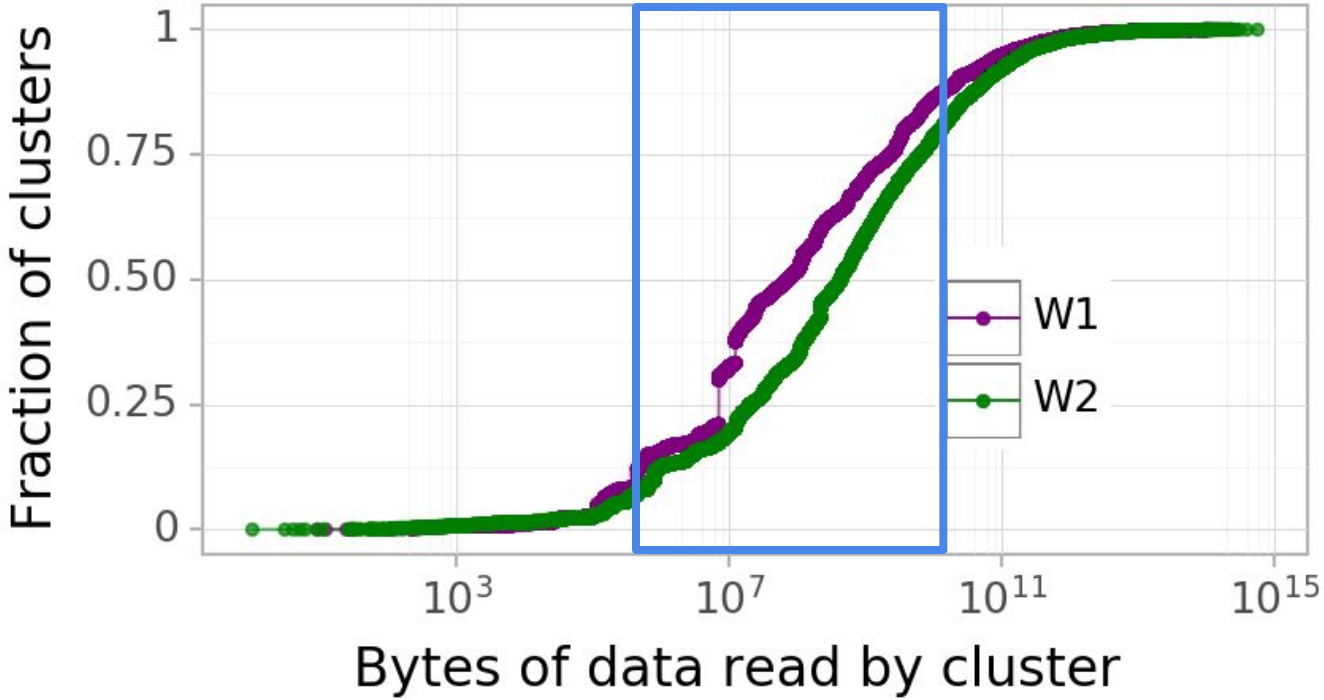


# Most clusters (75%) read few files

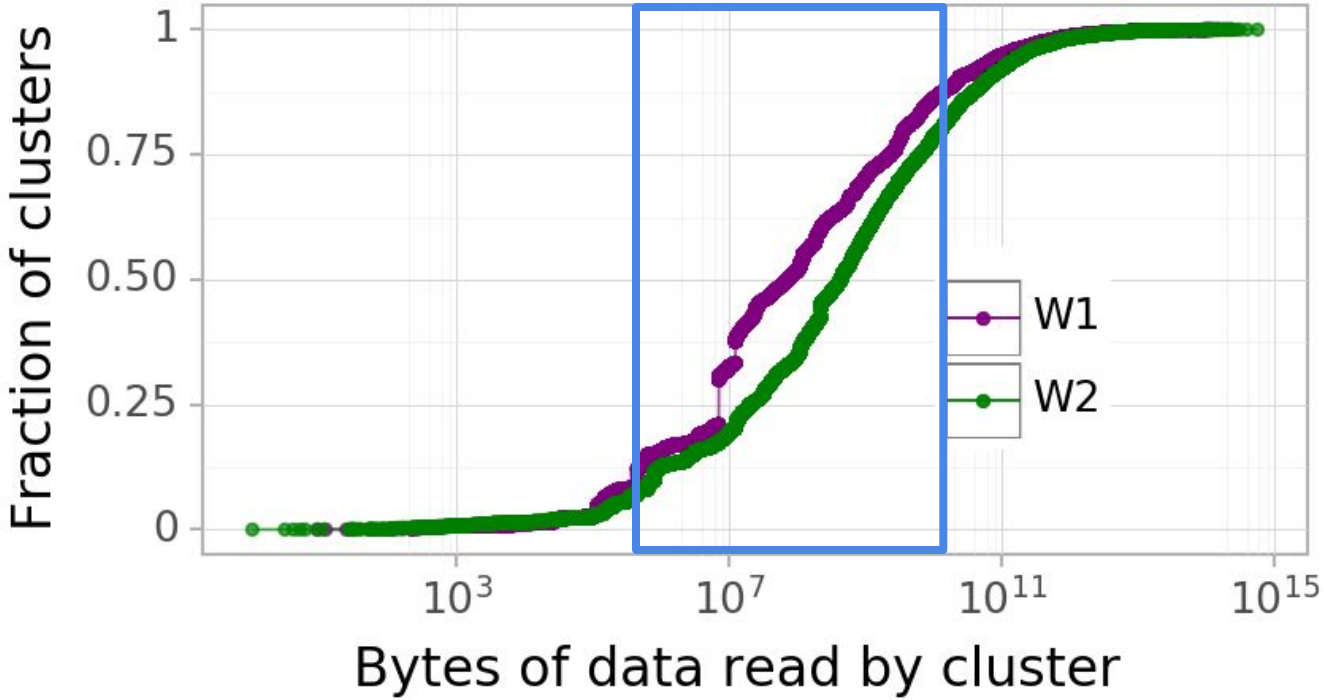



**Optimize clusters for few reads and short lifetimes**

# Most clusters (75%) read a median amount of data

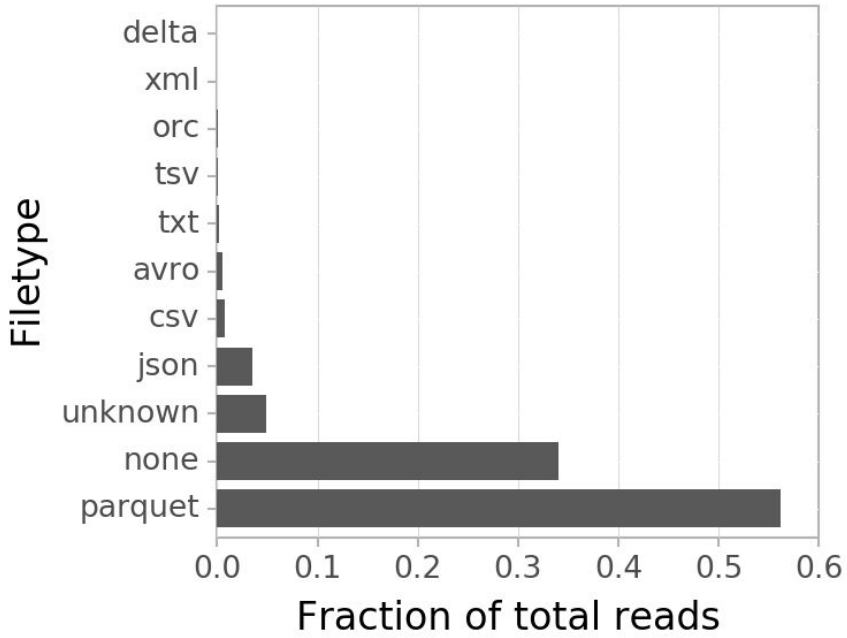


# Most clusters (75%) read a median amount of data

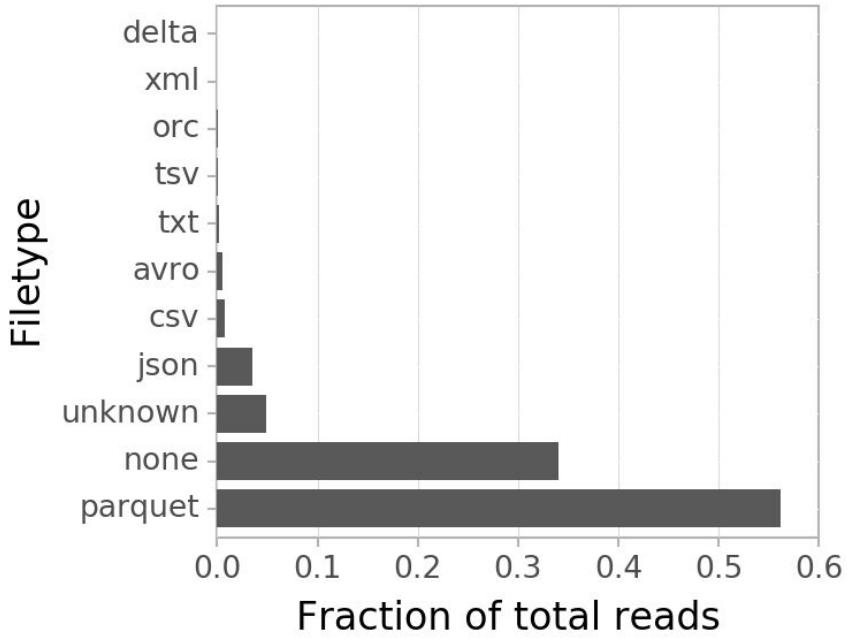



**Optimize cluster to read “not really big” data**

# Parquet files are most popular

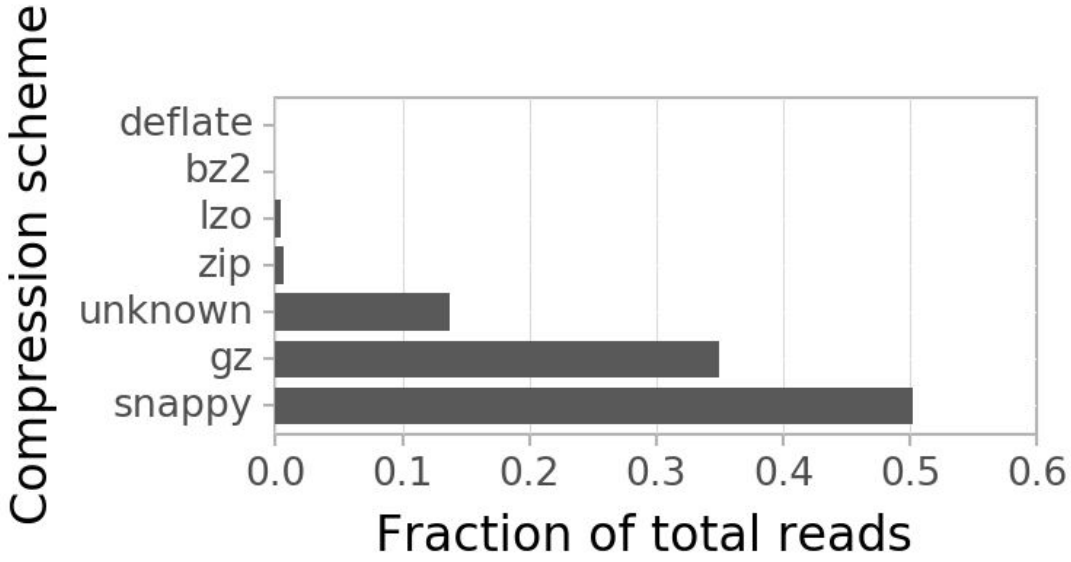


# Parquet files are most popular




**Need more parquet libraries, and reliable ways to read and write parquet**

# Snappy compression is the most popular



# Summary

1. Weekly and Daily trends
2. Sporadic bursts via Hurst parameter
3. Most (90%) reads are small
4. Reuse times logarithmically distributed
5. Most (75%) cluster read few files
6. Most (75%) clusters read median amount of data
7. Parquet is the most popular file format
8. Snappy is the most popular compression format

# Summary

1. Weekly and Daily trends
2. Sporadic bursts via Hurst parameter
3. Most (90%) reads are small
4. Reuse times logarithmically distributed
5. Most (75%) cluster read few files
6. Most (75%) clusters read median amount of data
7. Parquet is the most popular file format
8. Snappy is the most popular compression format

Questions?