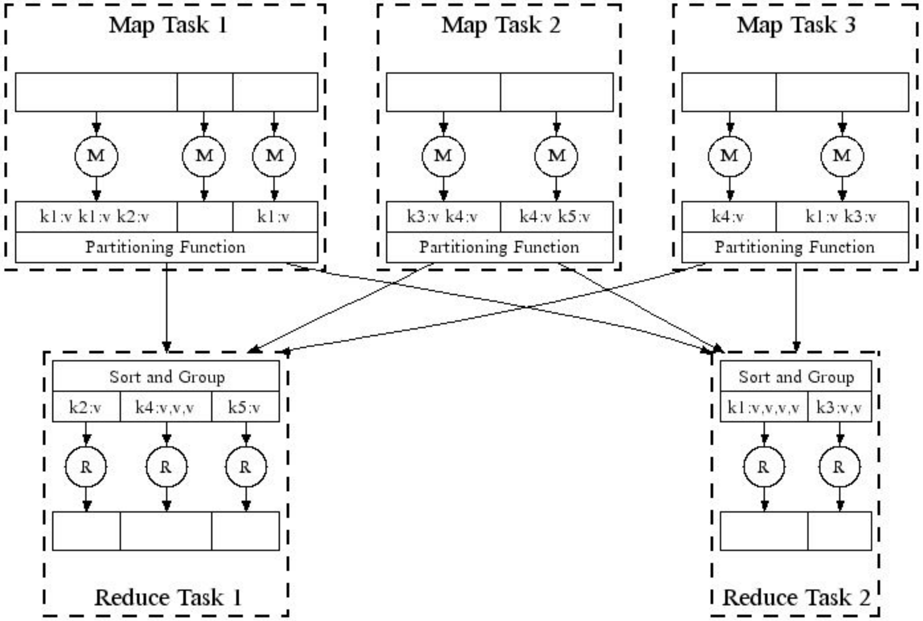
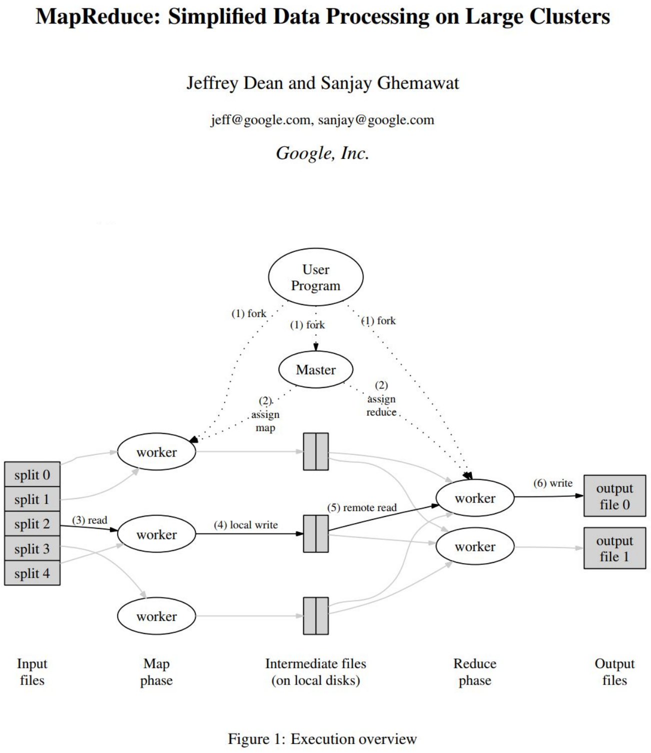
What Can Decade 2 of the Big Data Era Learn from Decade 1?

Robert Saxby - Big Data Product Specialist

Google Cloud

Simplified Data Processing on Large Clusters





MapReduce: Simplified Data Processing on Large Clusters

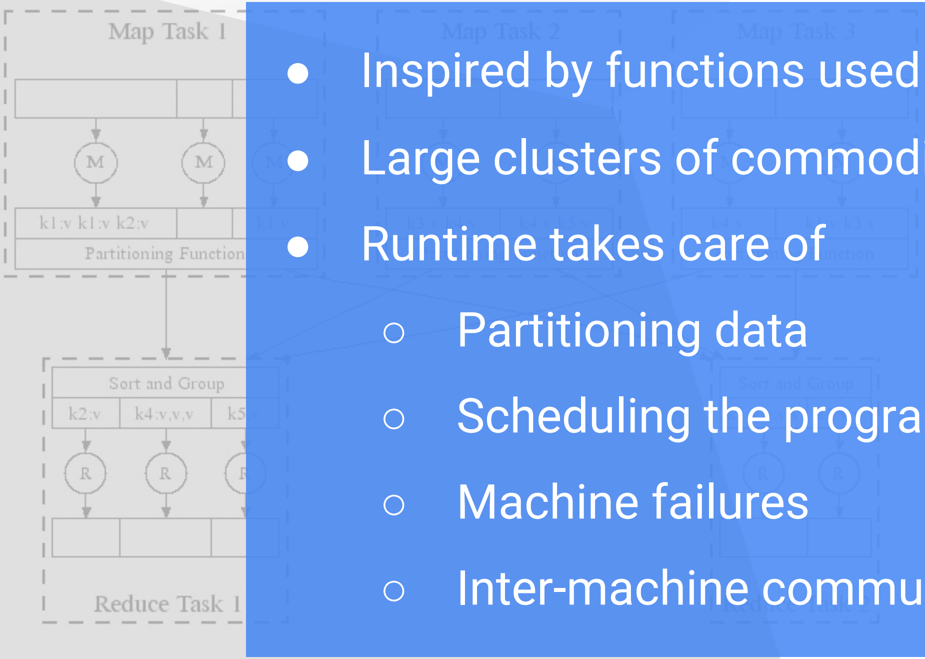
Jeffrey Dean and Sanjay Ghemawat

jeff@google.com, sanjay@google.com

Google, Inc.

Figure 1: Execution overview

Simplified Data Processing on Large Clusters



- Inspired by functions used in functional programming
- Large clusters of commodity machines
- Runtime takes care of
  - Partitioning data
  - Scheduling the program’s execution
  - Machine failures
  - Inter-machine communication

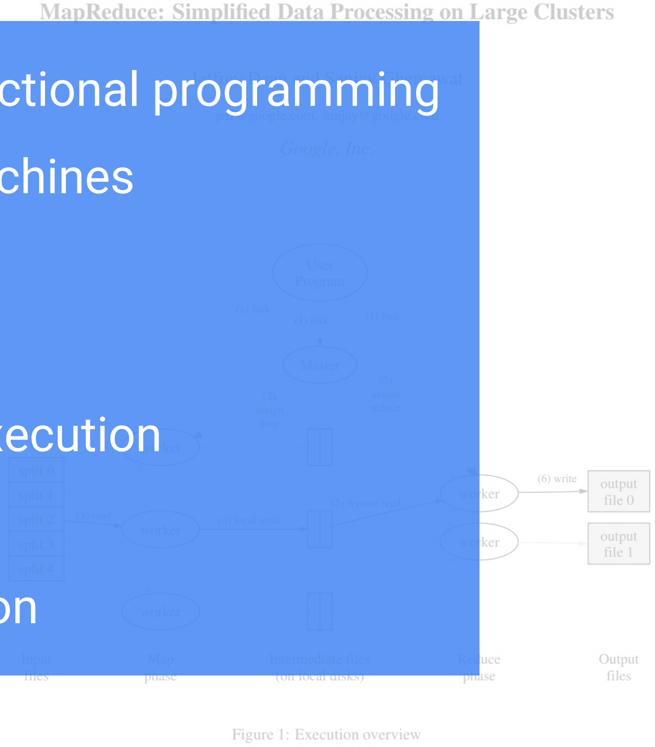
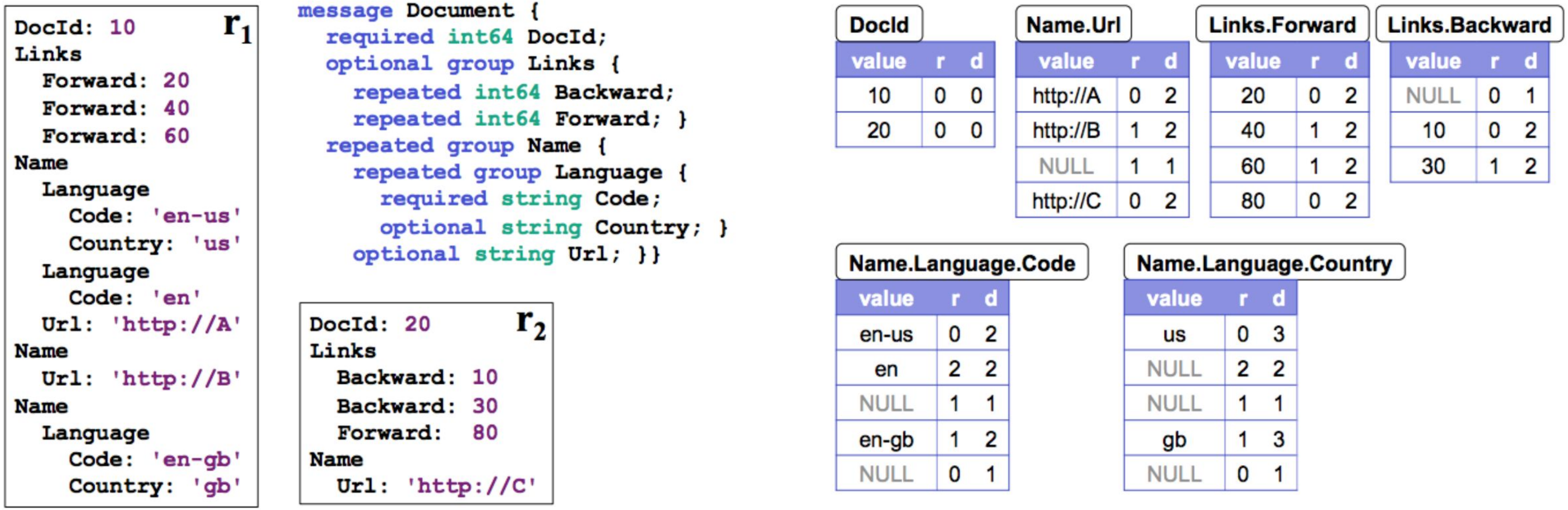

MapReduce: Simplified Data Processing on Large Clusters

Figure 1: Execution overview

Interactive Analysis of Web-Scale Datasets



```
DocId: 10                    r1
Links
  Forward: 20
  Forward: 40
  Forward: 60
Name
  Language
    Code: 'en-us'
    Country: 'us'
  Language
    Code: 'en'
  Url: 'http://A'
Name
  Url: 'http://B'
Name
  Language
    Code: 'en-gb'
    Country: 'gb'
```

```
message Document {
  required int64 DocId;
  optional group Links {
    repeated int64 Backward;
    repeated int64 Forward; }
  repeated group Name {
    repeated group Language {
      required string Code;
      optional string Country; }
    optional string Url; }}
```

```
DocId: 20                    r2
Links
  Backward: 10
  Backward: 30
  Forward:  80
Name
  Url: 'http://C'
```

DocId

value	r	d
10	0	0
20	0	0

Name.Url

value	r	d
http://A	0	2
http://B	1	2
NULL	1	1
http://C	0	2

Links.Forward

value	r	d
20	0	2
40	1	2
60	1	2
80	0	2

Links.Backward

value	r	d
NULL	0	1
10	0	2
30	1	2

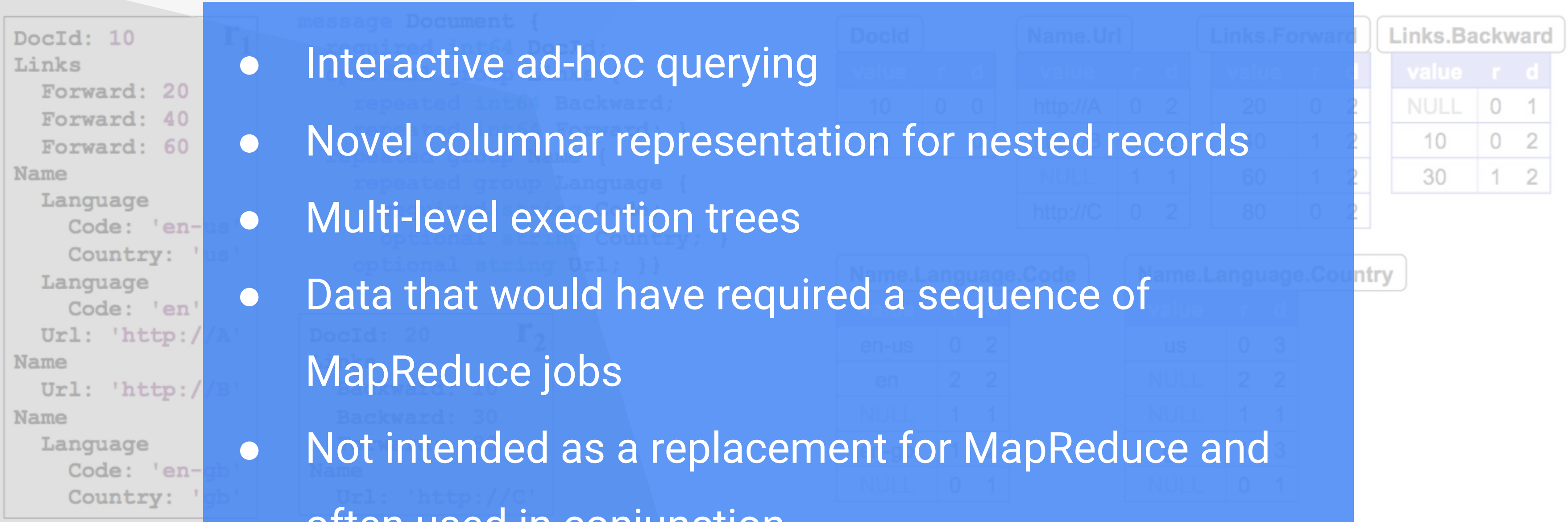
Name.Language.Code

value	r	d
en-us	0	2
en	2	2
NULL	1	1
en-gb	1	2
NULL	0	1

Name.Language.Country

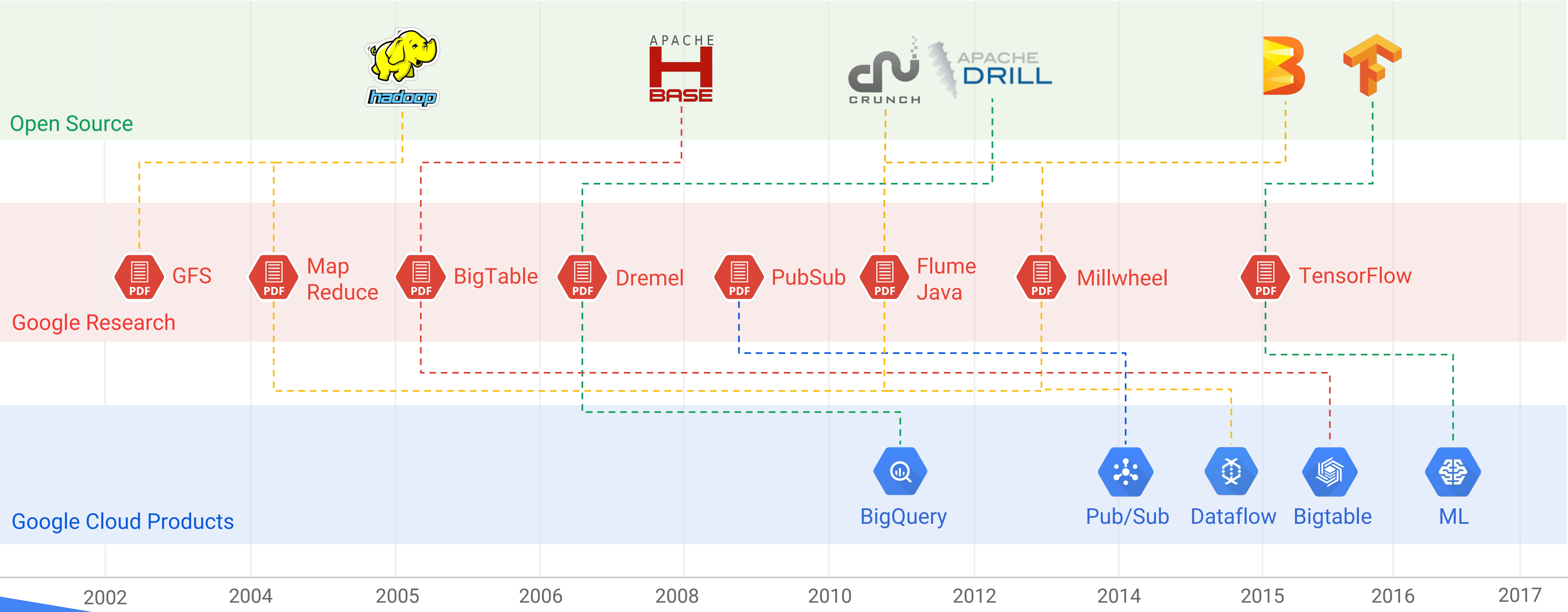
value	r	d
us	0	3
NULL	2	2
NULL	1	1
gb	1	3
NULL	0	1

Interactive Analysis of Web-Scale Datasets



- Interactive ad-hoc querying
- Novel columnar representation for nested records
- Multi-level execution trees
- Data that would have required a sequence of MapReduce jobs
- Not intended as a replacement for MapReduce and often used in conjunction

15+ Years of Solving Data Problems



2017, and Apache Spark and Hadoop are still too hard

Cost

On-prem Spark/Hadoop clusters are expensive to build, manage and grow.

Complexity

Operational overhead and locked-in resources make focusing on analytics difficult.

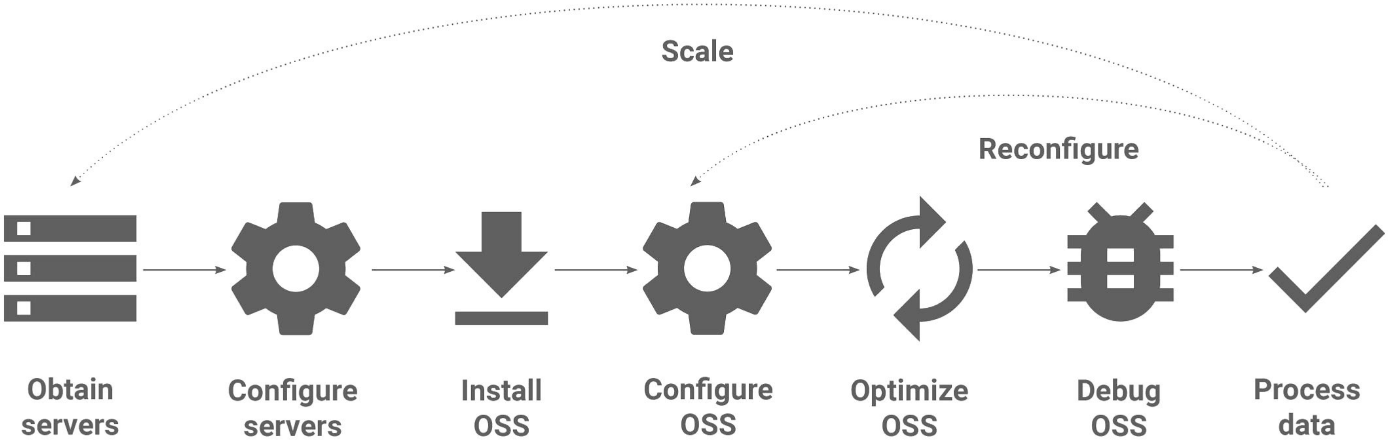
Inflexibility

Inability to independently scale compute and storage inhibit growth.

"Despite the variety of vendors, deployment environments, and geographic expansion, it is still challenging to get Hadoop-based projects beyond the pilot phase”

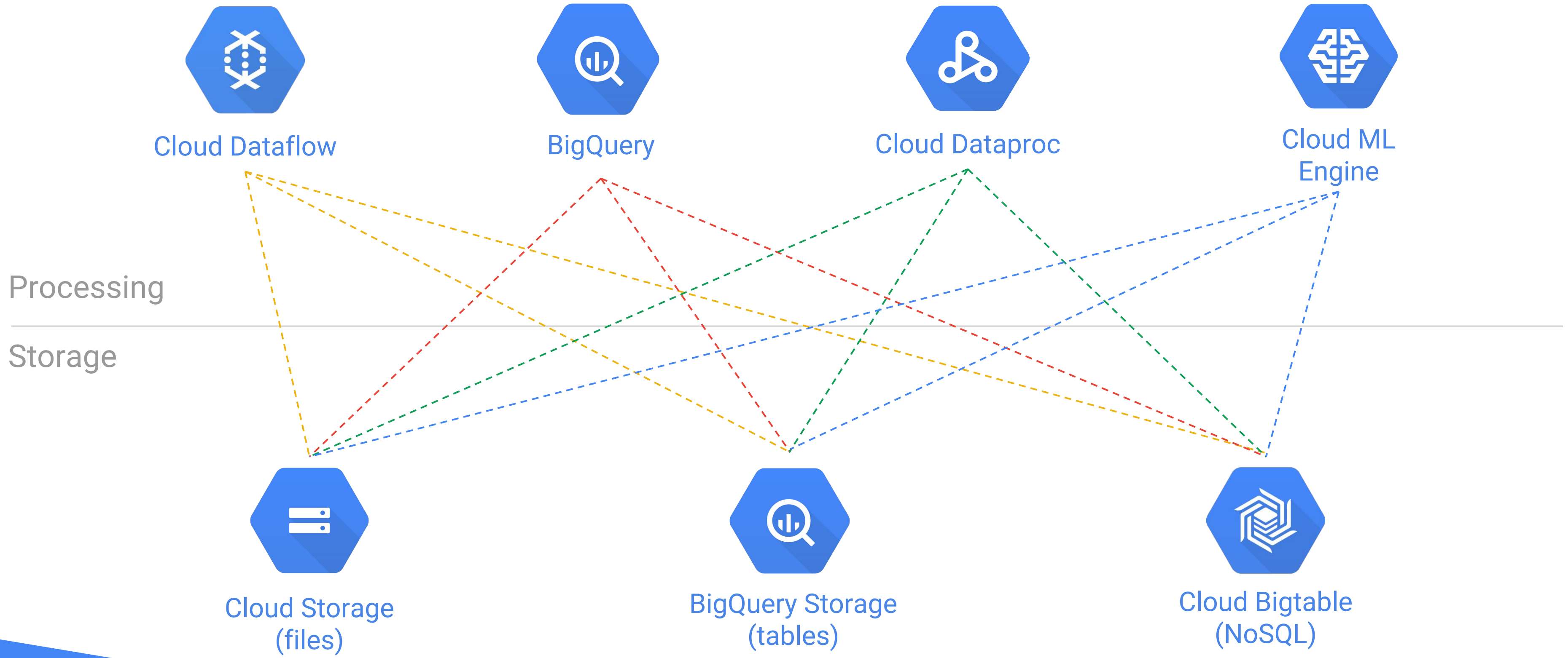
- *Market Guide for Hadoop Distributions* (2017), Gartner

Cluster deployment: The Hard Way

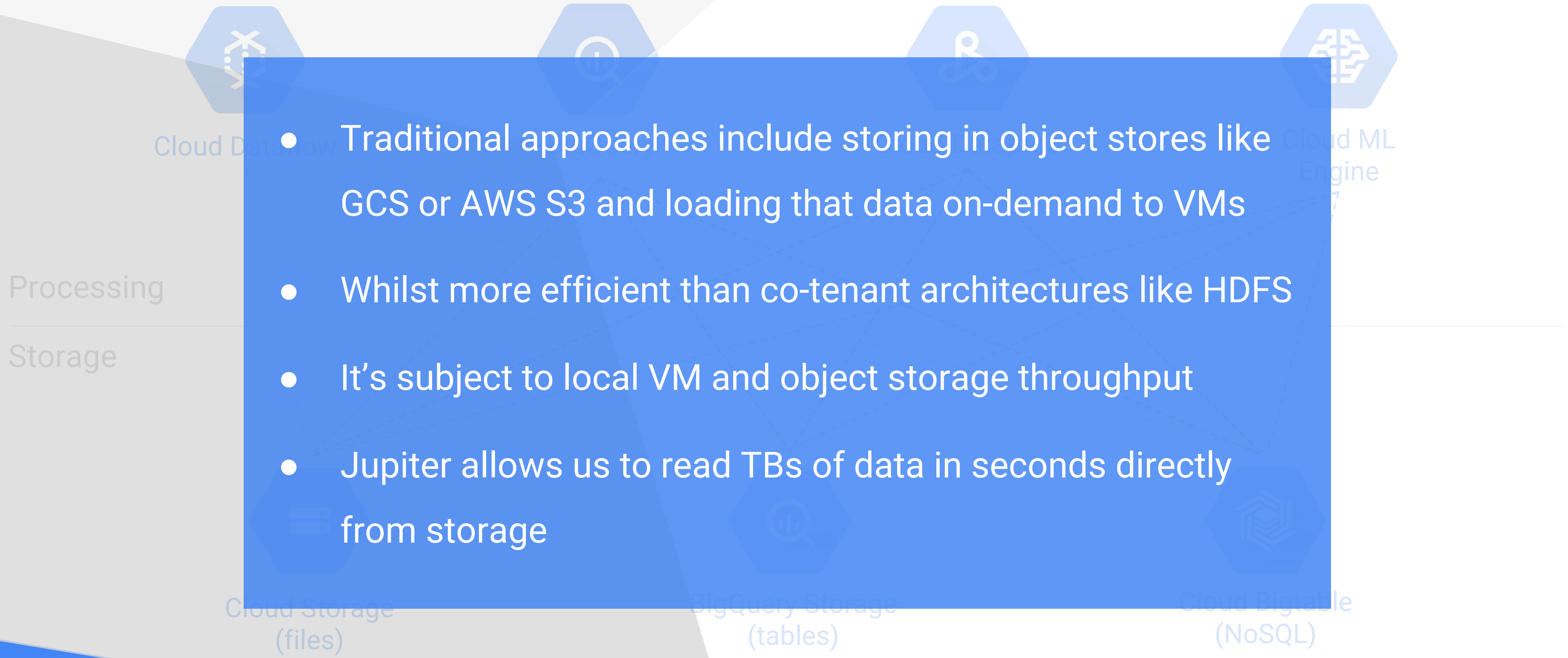


Total elapsed time: Hours or days

Separation of Storage and Compute

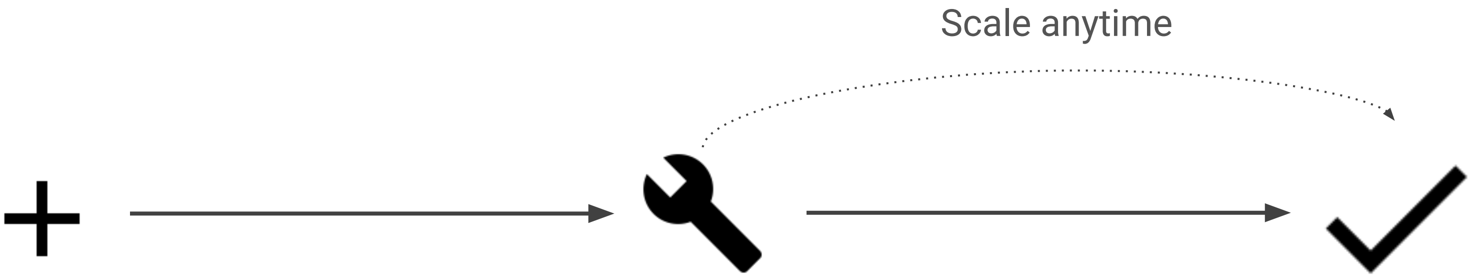


Separation of Storage and Compute



- Traditional approaches include storing in object stores like GCS or AWS S3 and loading that data on-demand to VMs
- Whilst more efficient than co-tenant architectures like HDFS
- It’s subject to local VM and object storage throughput
- Jupiter allows us to read TBs of data in seconds directly from storage

Cluster deployment: The Easy (Cloud Dataproc) Way



Create cluster

0 seconds

Configure cluster

20 seconds

Use cluster

90 seconds

Cluster deployment: The Easy (Cloud Dataproc) Way

- The same storage
- Match the right processing engine to the workload
- Cattle not pets
- Use only the resources that you need
- It’s about jobs and tasks

Scale anytime

Create cluster

0 seconds

Configure cluster

20 seconds

Use cluster

90 seconds

Google Cloud

We want more users querying the data, asking questions and developing insights

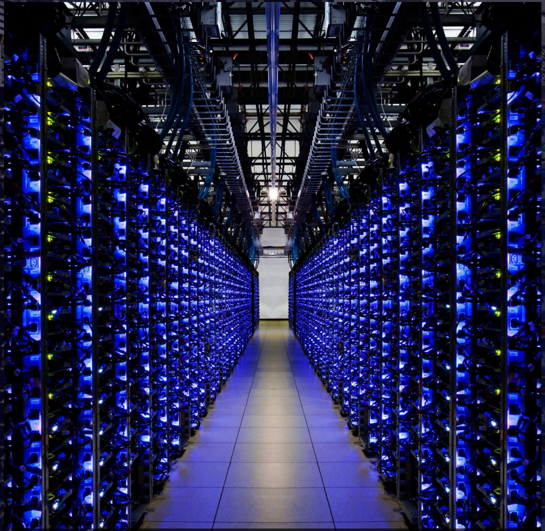
rather than

Having data siloed and locked down to the extent that users are impeded

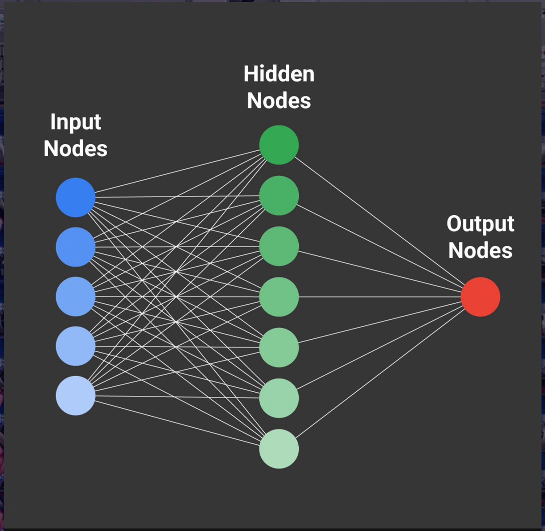
Discover

Discover

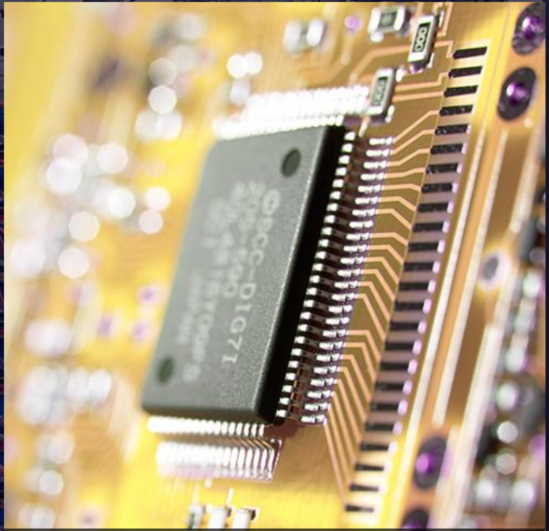
- What data do we have?
- Can you describe the data to me?
- Does it contain PII data?
- Where is it located?
- Where does it come from?
- Has it been manipulated? By whom or what?
- Who has access?
- Who is accessing it? And when?



Large Datasets



Good ML Models



Lots of Computation

From objective

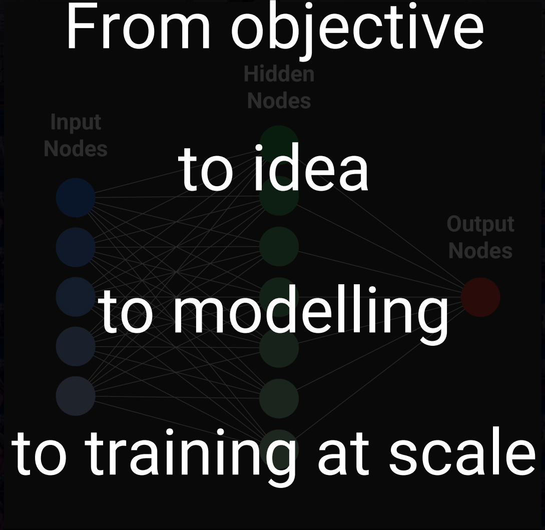
to idea

to modelling

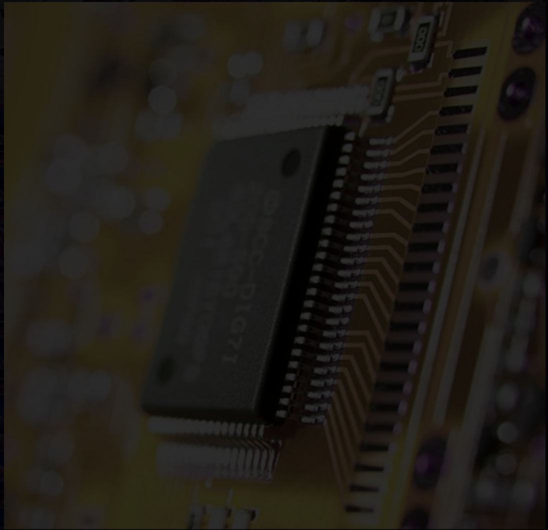
to training at scale

to serving in production

Large Datasets



Good ML Models



Lots of Computation

TensorFlow

Big data and the cloud

What can and should we expect?

What Should a Cloud Offer?

An example with data processing pipelines

Apache Beam is a collection of SDKs for **building** streaming data processing pipelines.

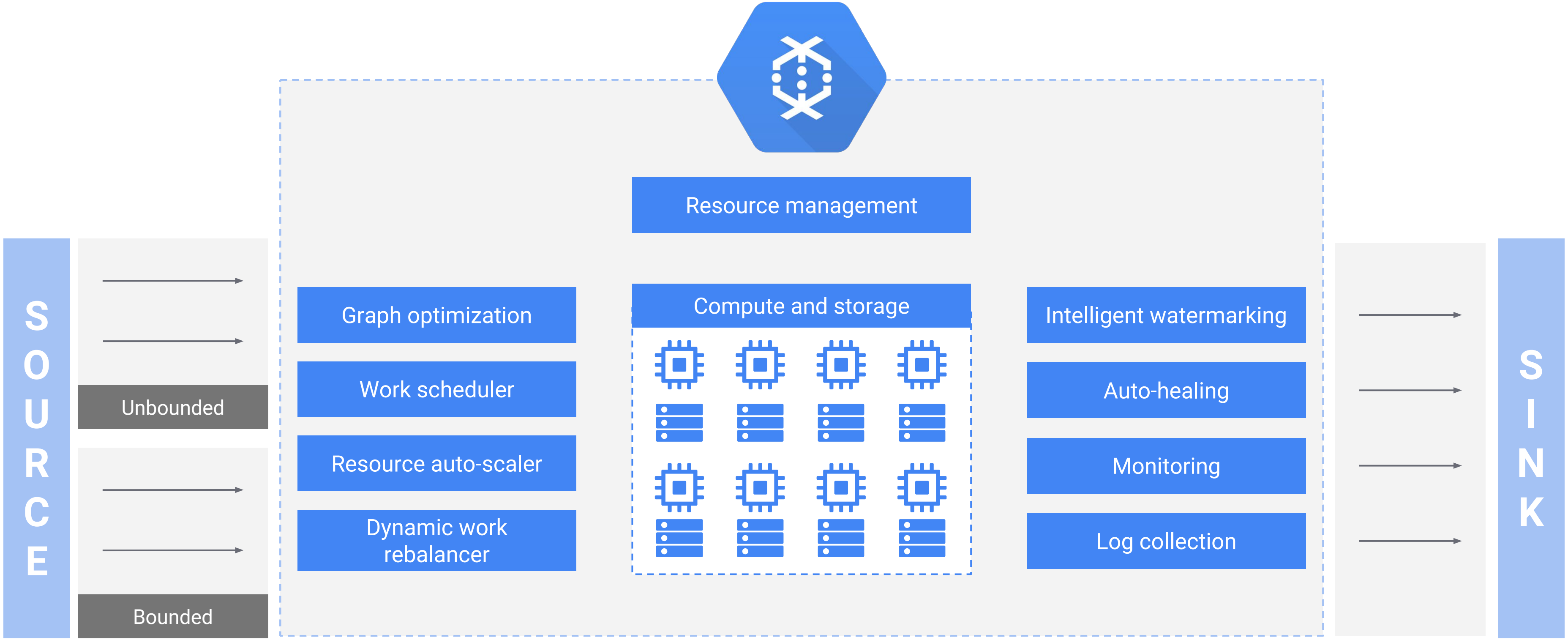
Cloud Dataflow is a fully managed (no-ops) and integrated service for **executing** optimized parallelized data processing pipelines.

What Should a Cloud Offer?

Apache Beam is a collection of SDKs for **building** streaming data processing pipelines.

Cloud Dataflow is a fully managed (no-ops) and integrated service for **executing** optimized parallelized data processing pipelines.

What Should a Cloud Offer?



What Should a Cloud Offer?

- OSS libraries and SDKs
- Managed services to run your OSS based software
  - Reduce operational overhead
- These services then compete on
  - Price
  - Performance
  - Additional non-functionals - e.g. execution optimisation
  - Integration with other systems

Apache Beam is a collection of SDKs for building streaming data processing pipelines.

Cloud Dataflow is a fully managed (no-ops) and integrated service for executing optimized parallelized data processing pipelines.

Thank you