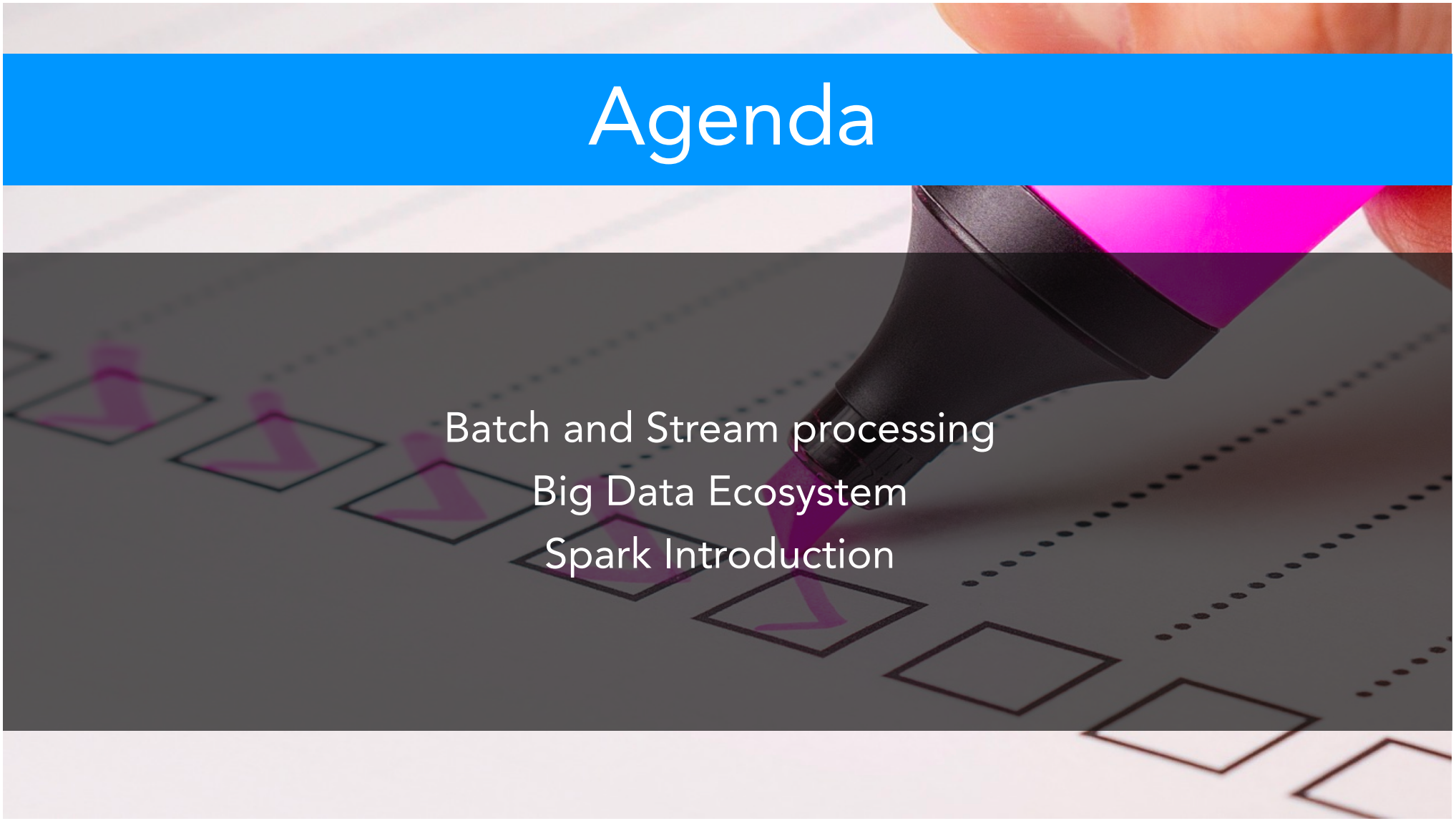
Google Cloud Platform

Cloud Dataproc

Agenda



Batch and Stream processing

Big Data Ecosystem

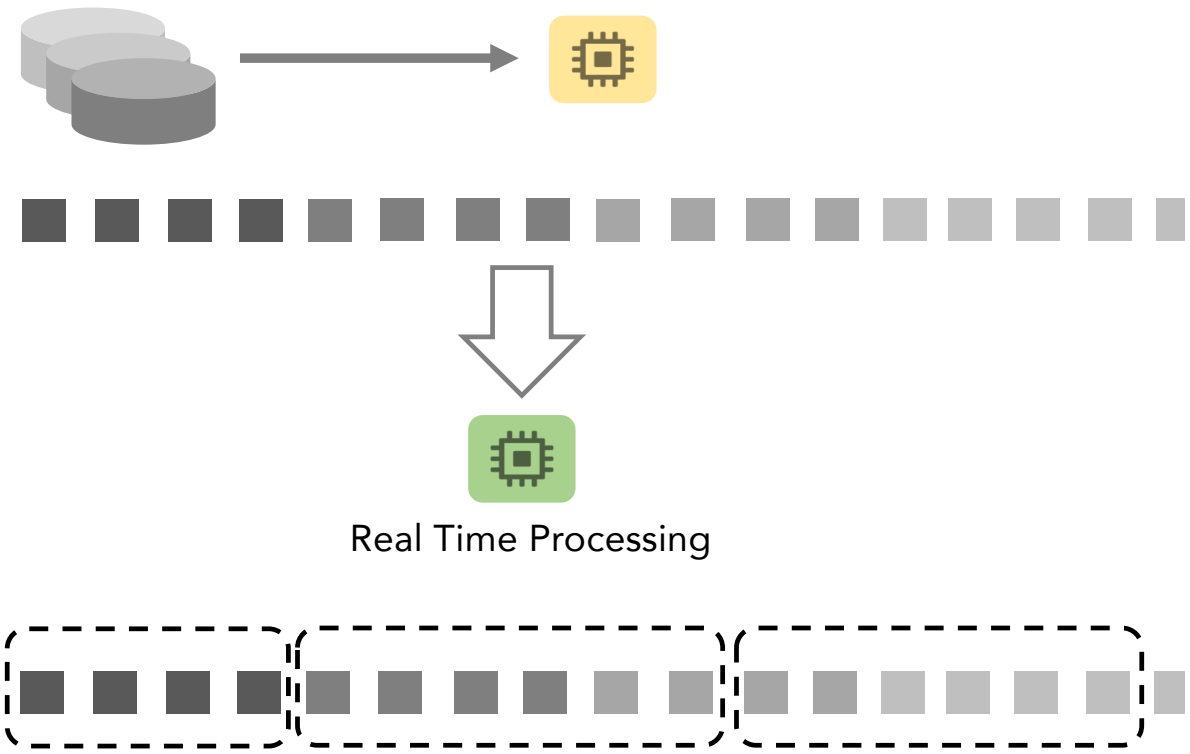
Spark Introduction

Batch and Streaming

- What is batch processing?
- Bounded data…

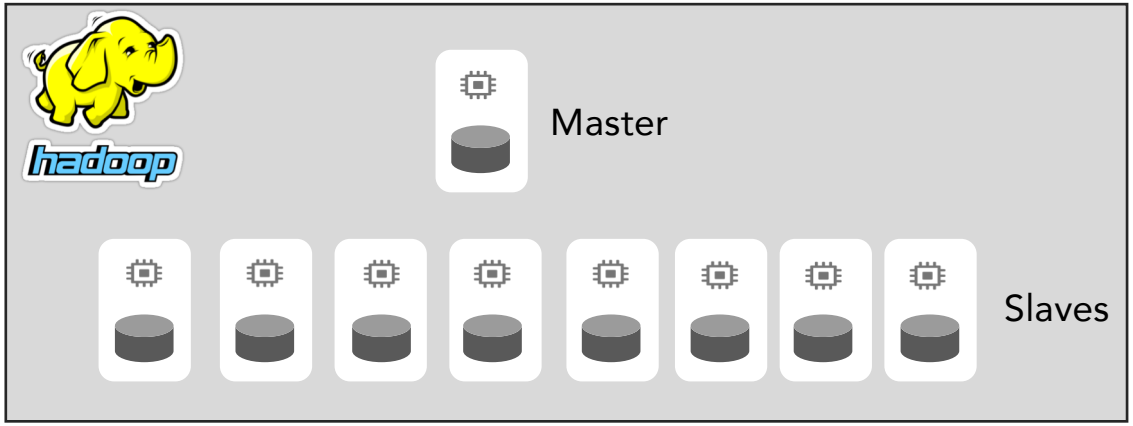
- Unbounded data…
  - Endless and Continuous
- Sources
  - IoT devices
  - Click Streams
  - Monitoring data
  - Fraud detection data
- What is stream processing?
  - Real time processing

- Micro-batches
  - Batch semantics
  - Re-run



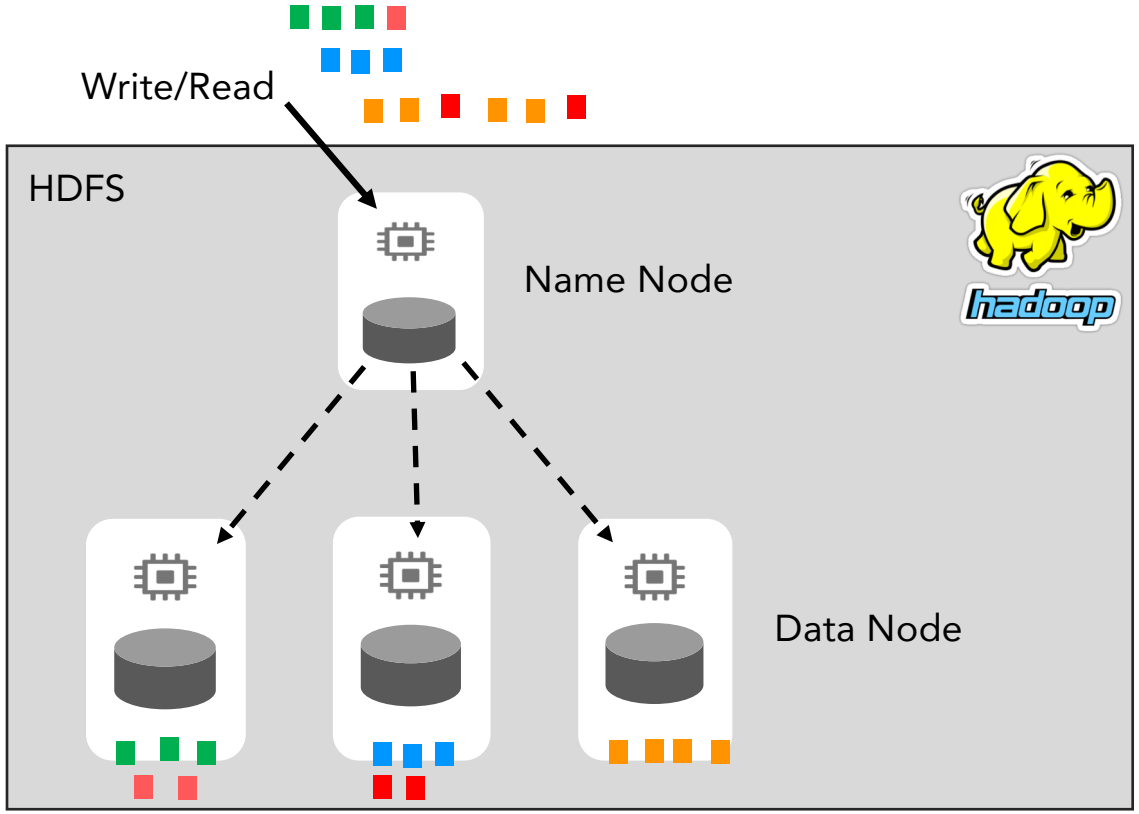
Batch Processing

- Hadoop
  - HDFS - Name Node & Data Node
  - MapReduce - map() + reduce()
  - YARN



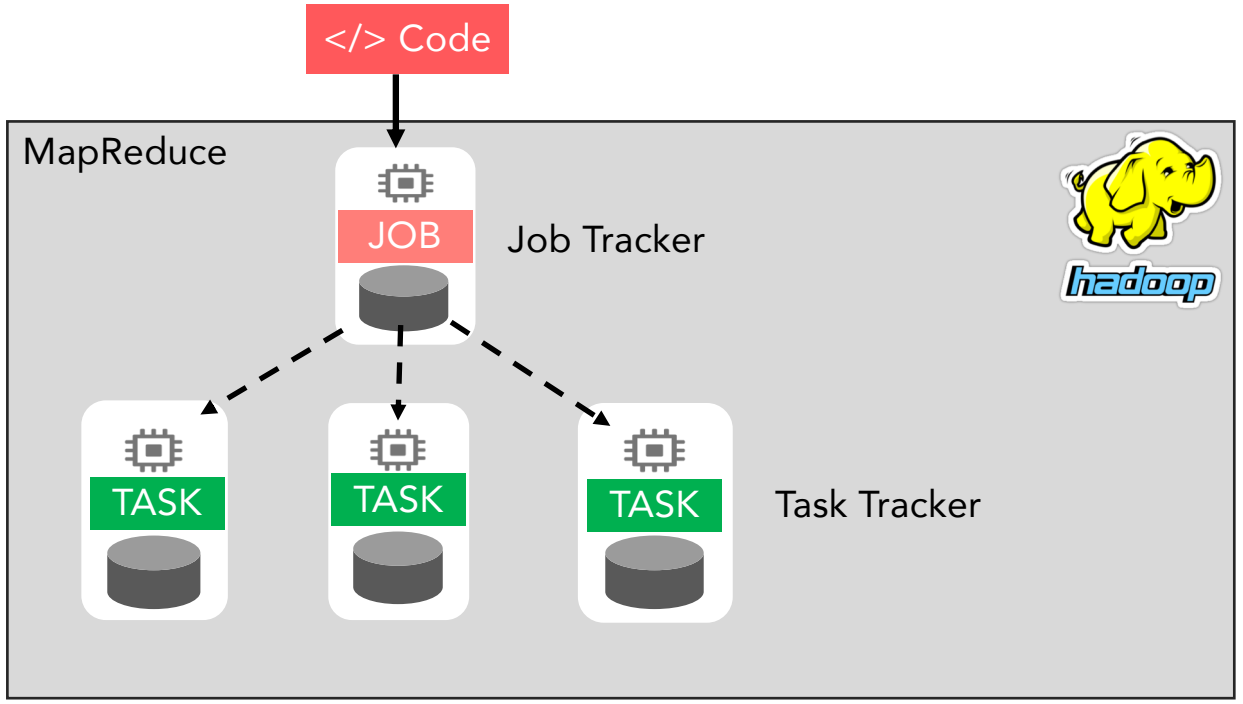
Batch Processing

- Hadoop
  - HDFS - Name Node & Data Node
  - MapReduce - map() + reduce()
  - YARN



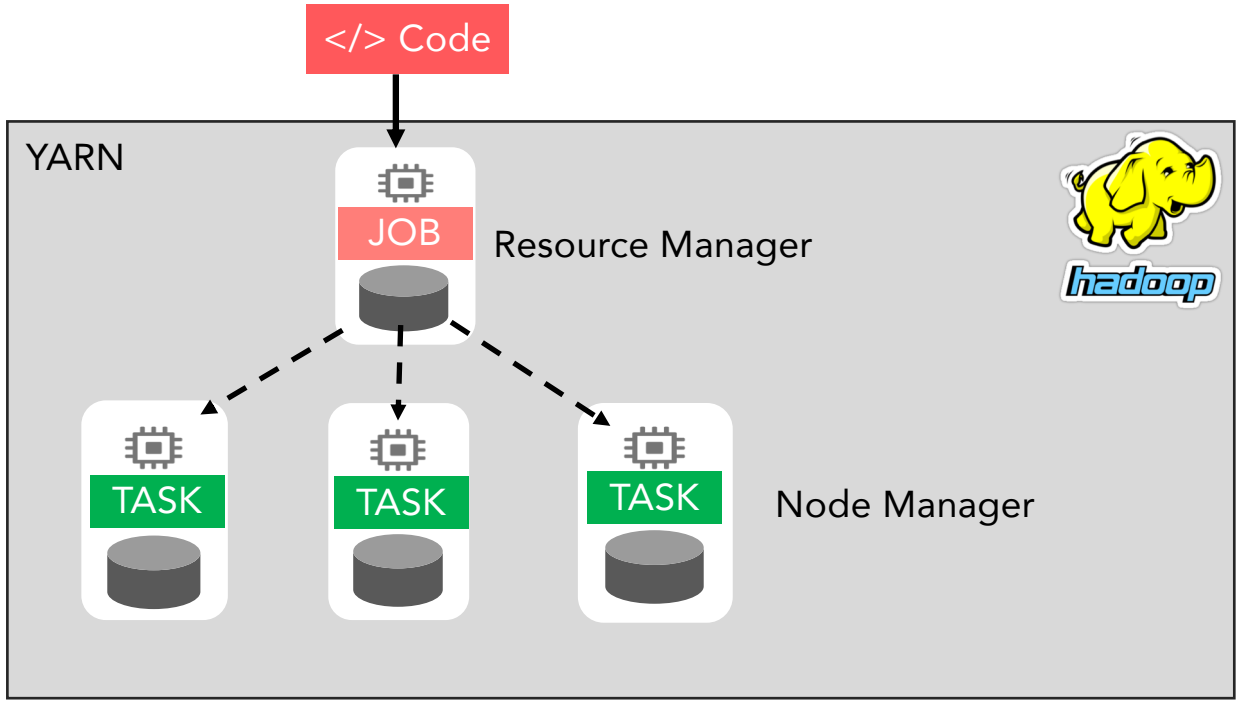
Batch Processing

- Hadoop
  - HDFS - Name Node & Data Node
  - MapReduce - map() + reduce()
  - YARN



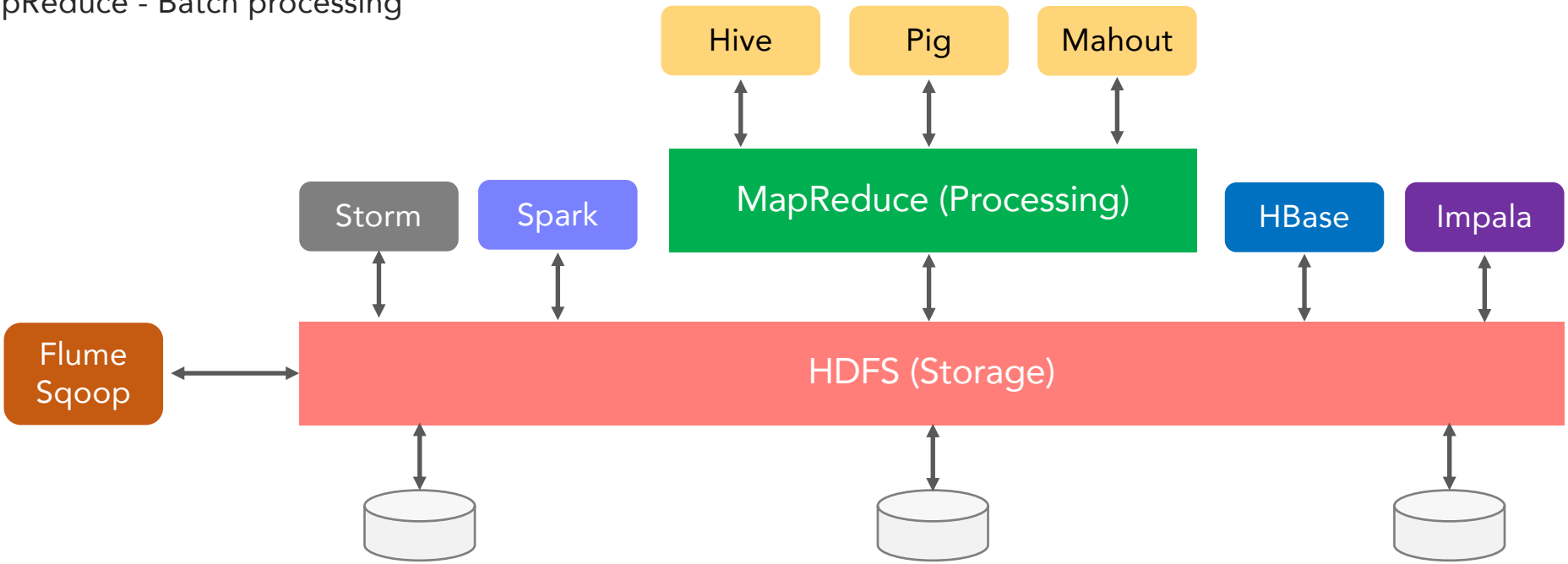
Batch Processing

- Hadoop
  - HDFS - Name Node & Data Node
  - MapReduce - map() + reduce()
  - YARN

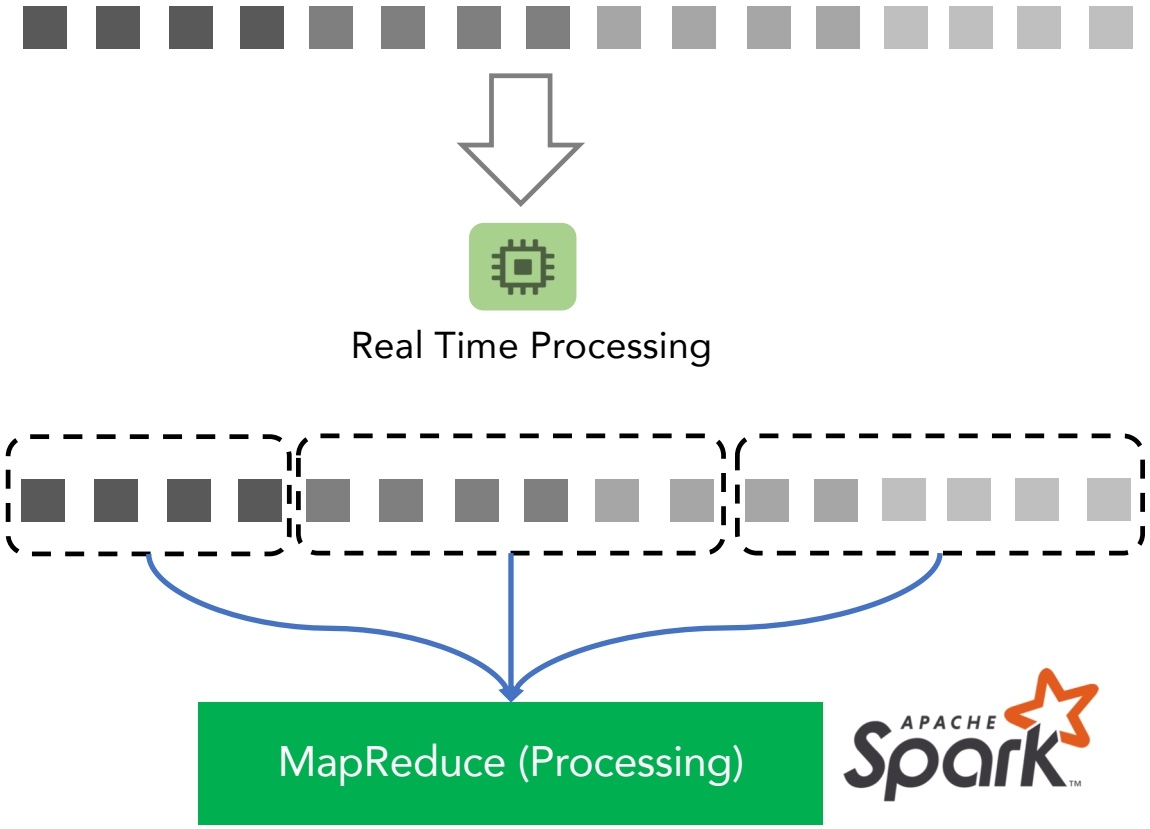


Big Data Ecosystem

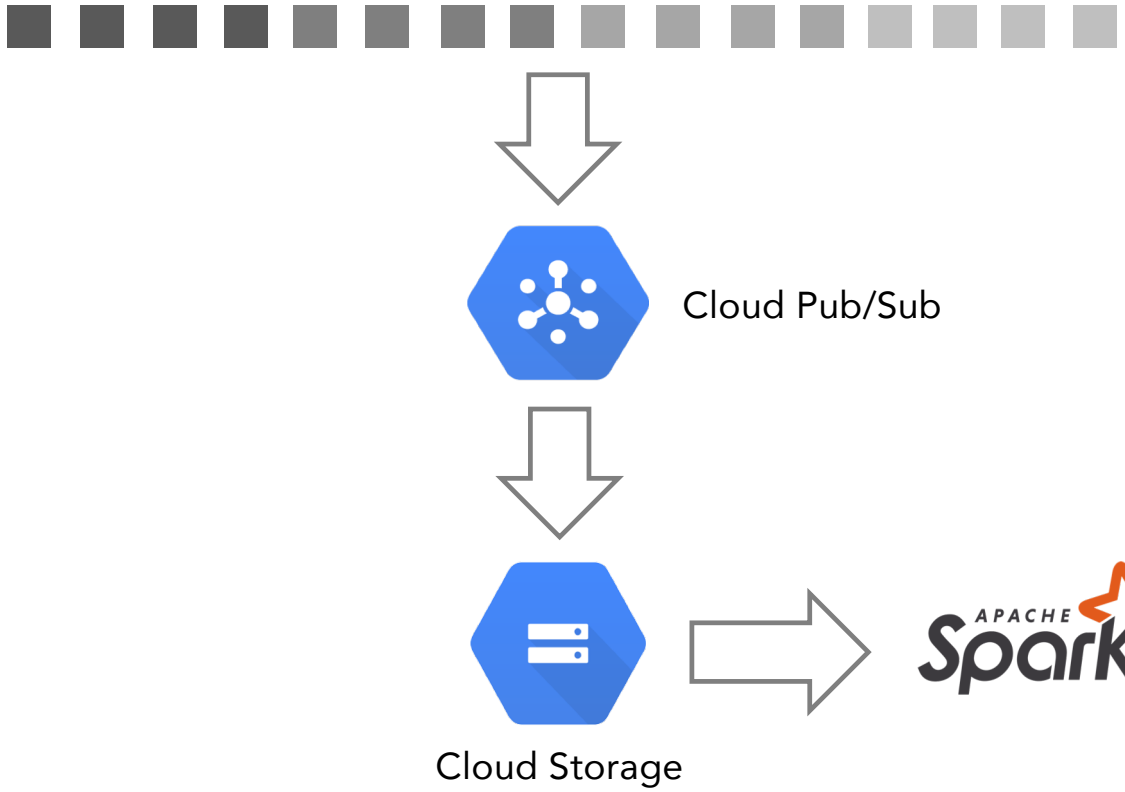
- HDFS - Distributed Storage
- MapReduce - Batch processing



Stream Processing

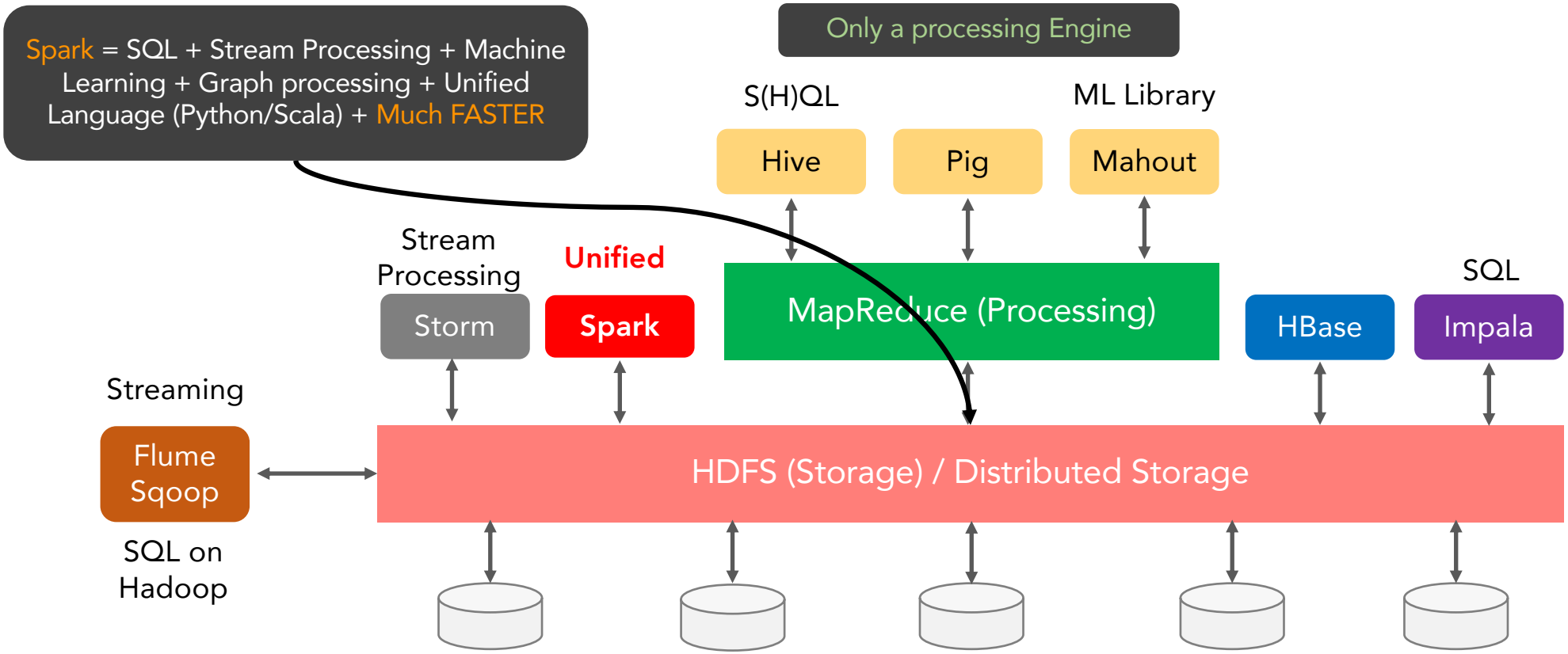


Stream Processing

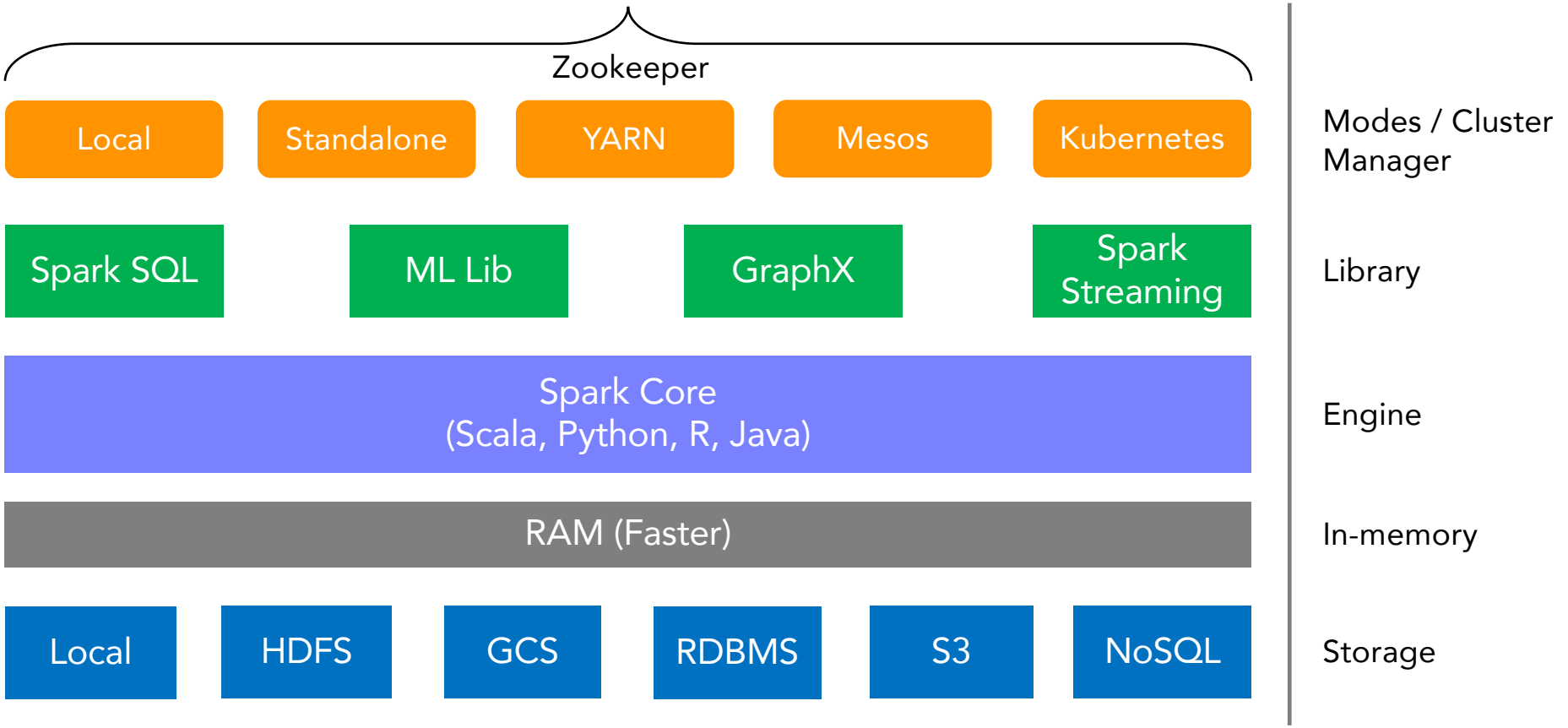


- Ingest
- Store
- Process

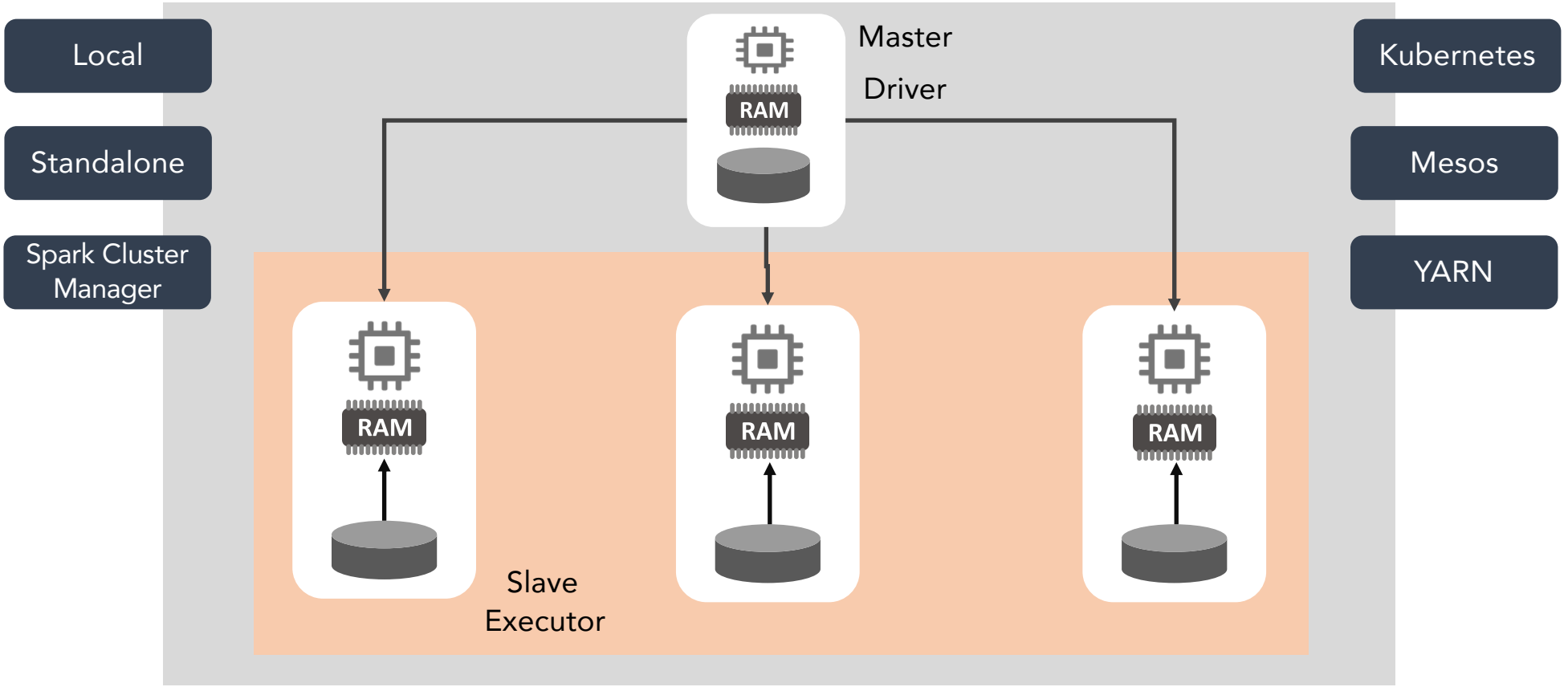
Spark Introduction



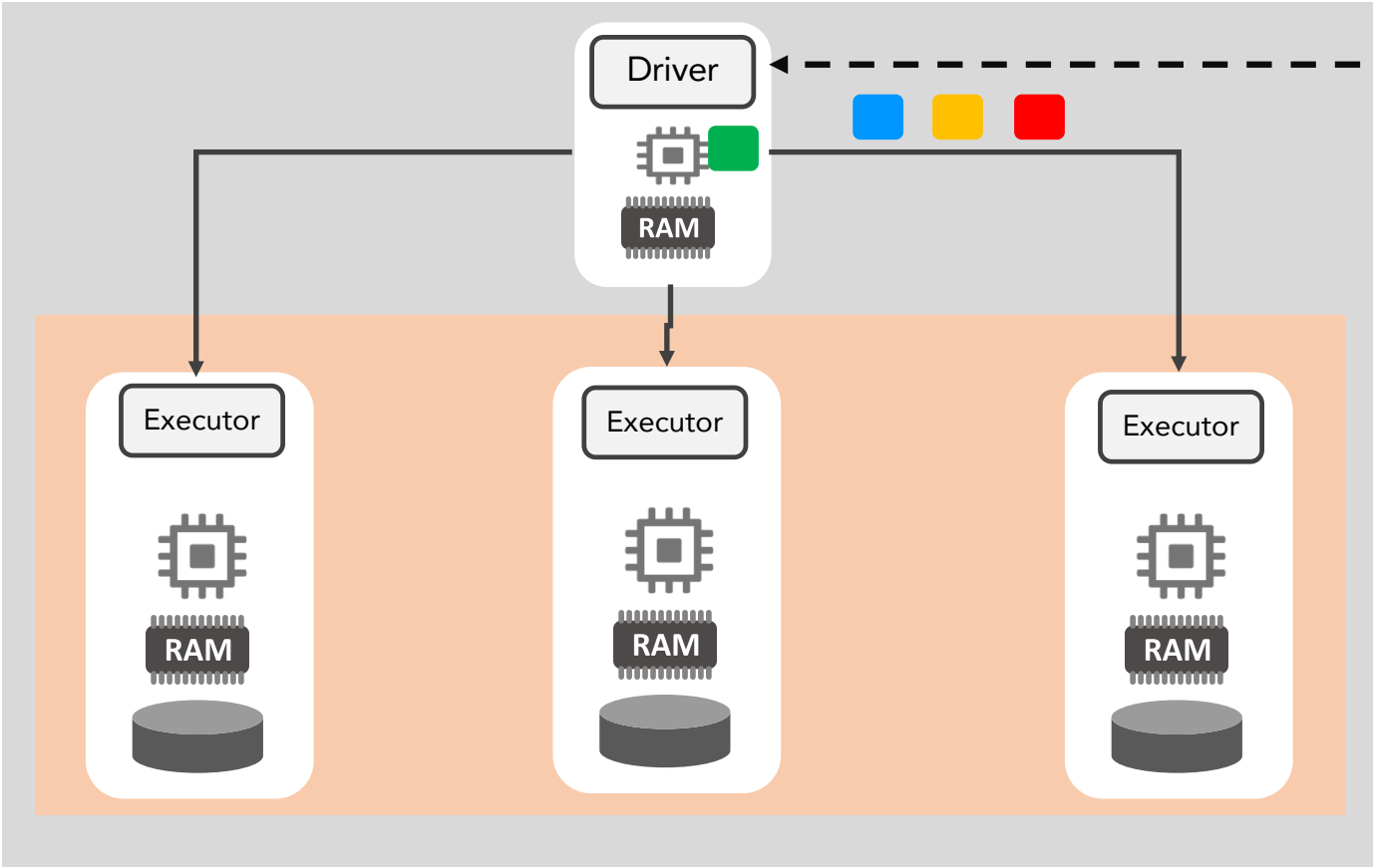
Spark Ecosystem



Spark Application Architecture



Spark Application Architecture



Spark Program (Scala, Python, Java, R, SQL)

- Interactive
- Batch

Driver

- Maintain information about the application
- Responding to user input
- Analyzing, distributing, scheduling work across executors
- Monitoring application

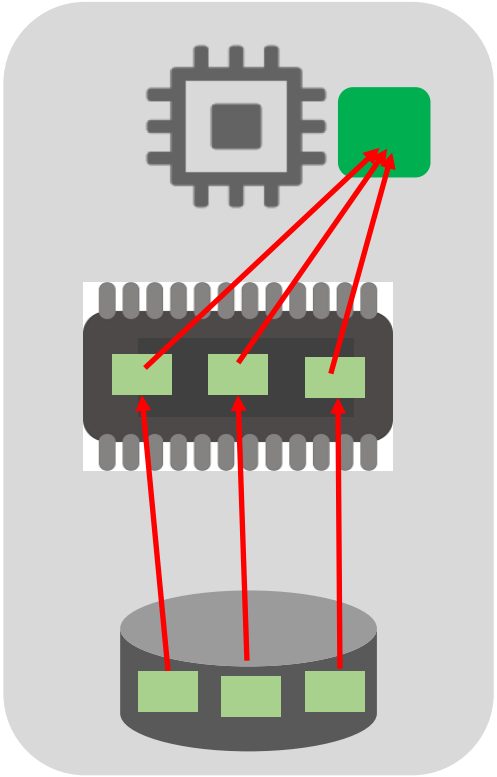
Executor

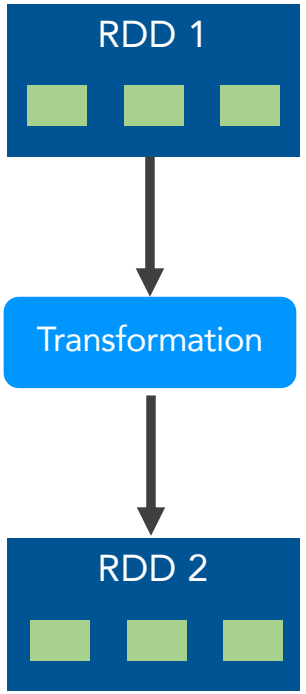
- Executing the code
- Reporting state of execution back to Driver process

RDD, Transformation, Action

- What is RDD
- Create RDD

- Action
  - countByValue
  - count
  - collect
  - Take(num)
  - Reduce(func)

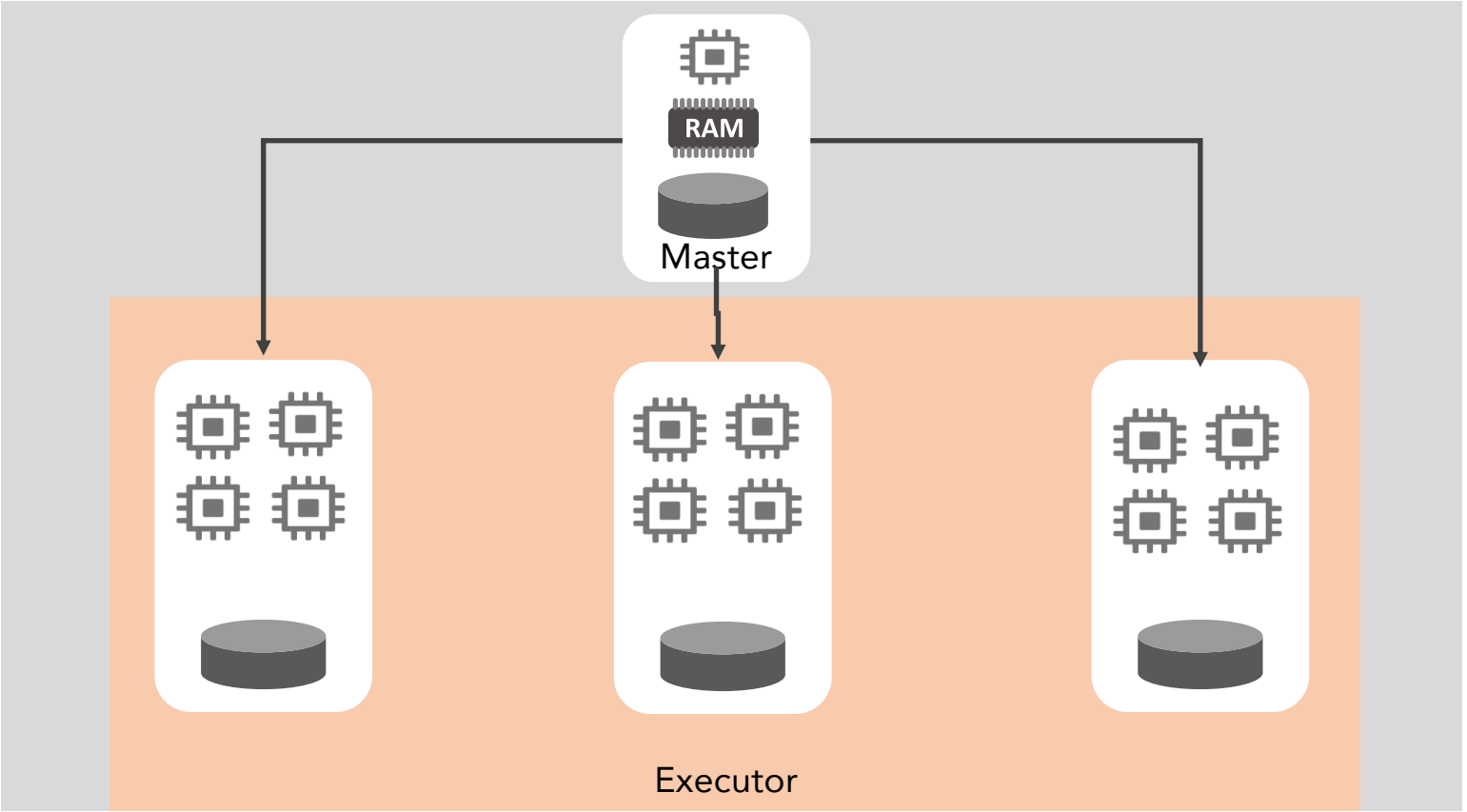




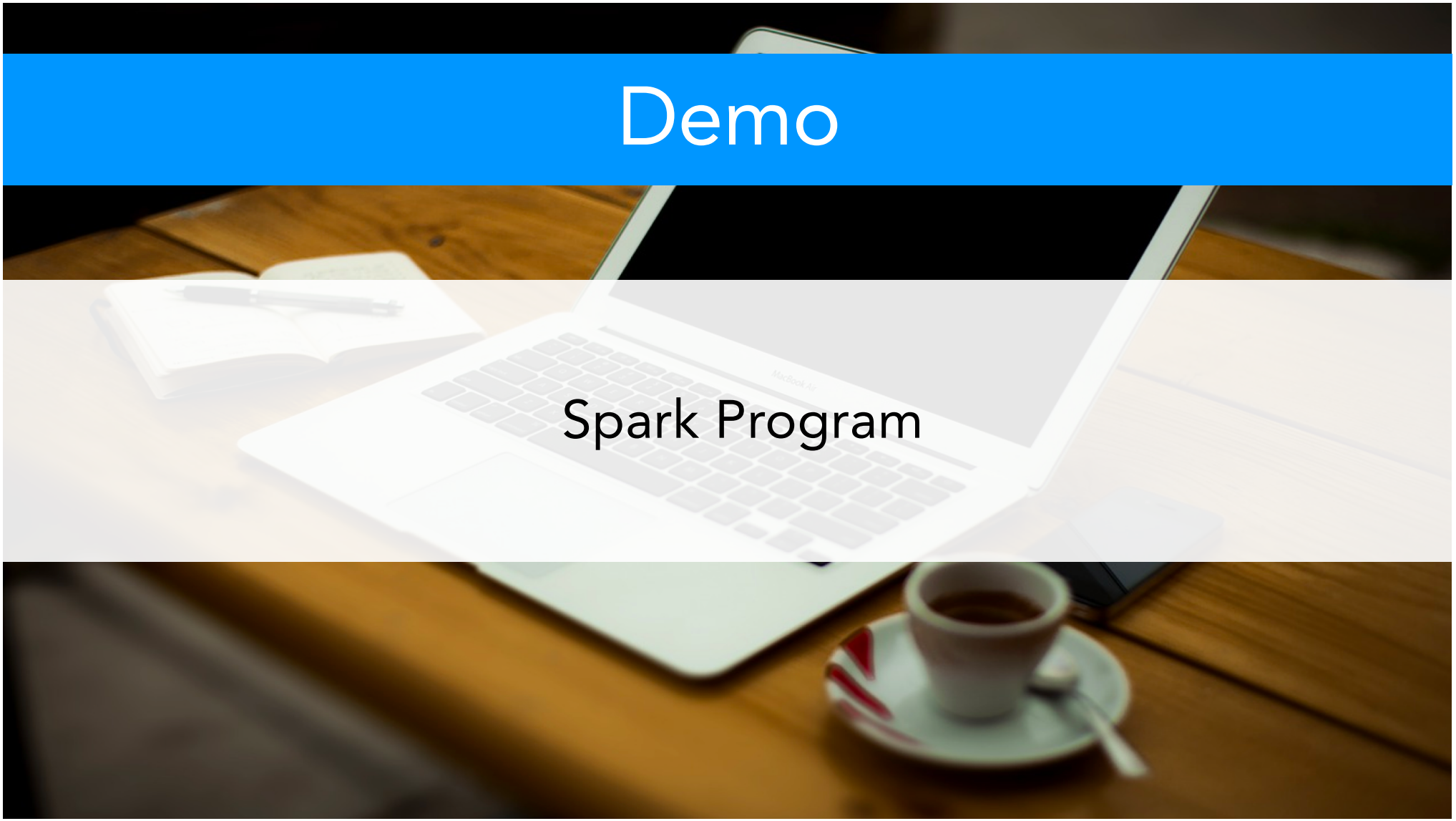
- Transformation
  - map
  - flatMap
  - filter
  - union
  - cartesian
  - distinct

Partition, Persist

- Partition
  - 1 P - N Executor
  - 40 P - 1 Executor
- Persist



Demo

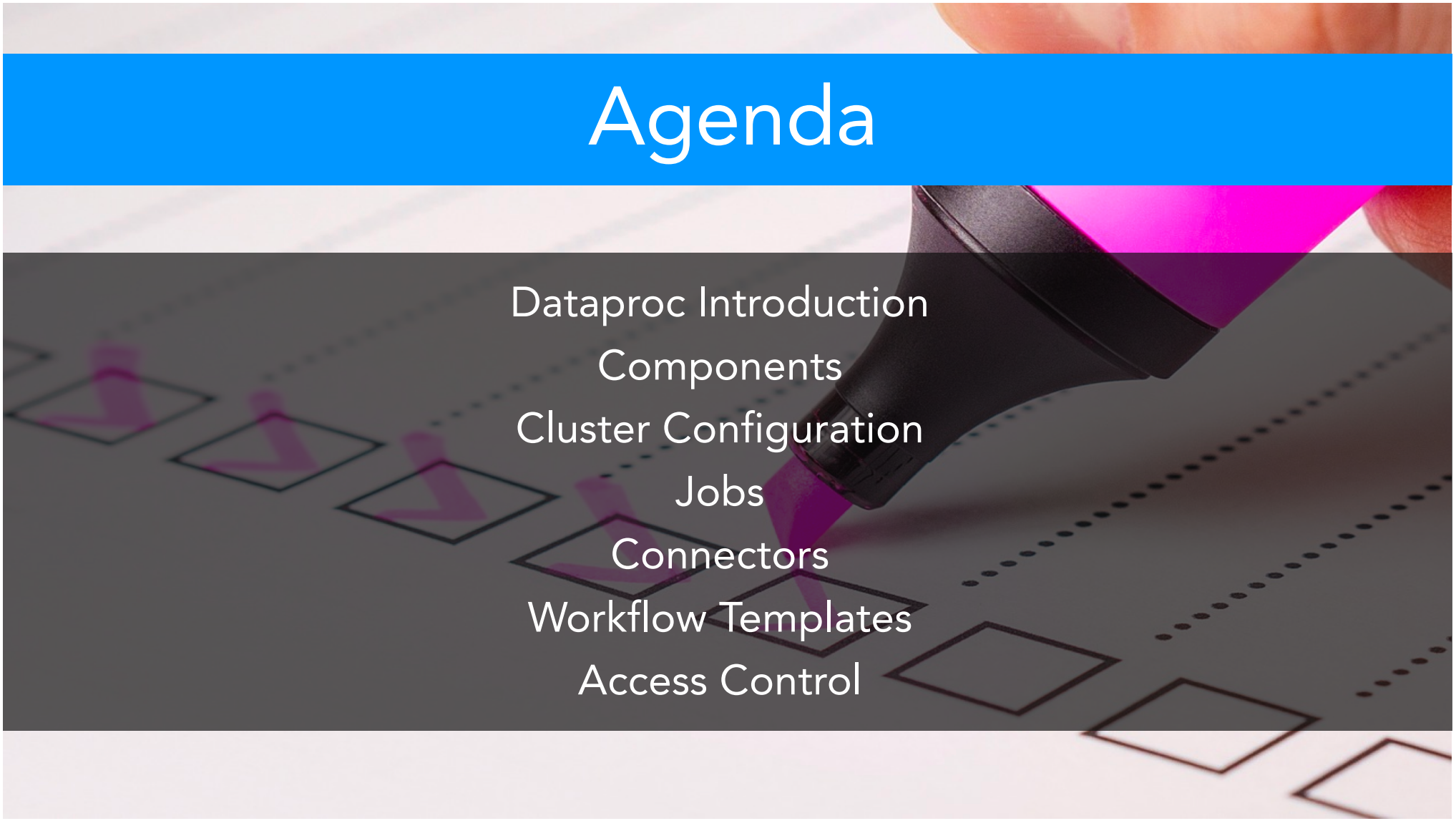


Spark Program

Google Cloud Platform

Cloud Dataproc

Agenda



Dataproc Introduction

Components

Cluster Configuration

Jobs

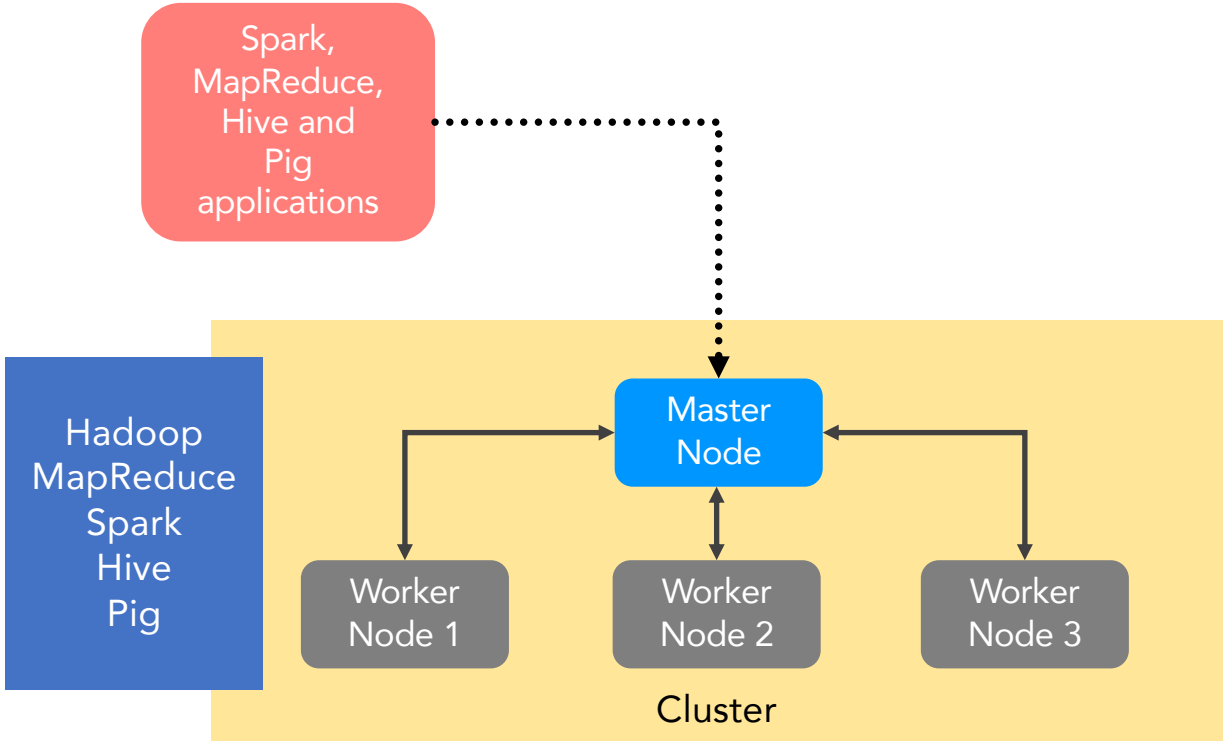
Connectors

Workflow Templates

Access Control

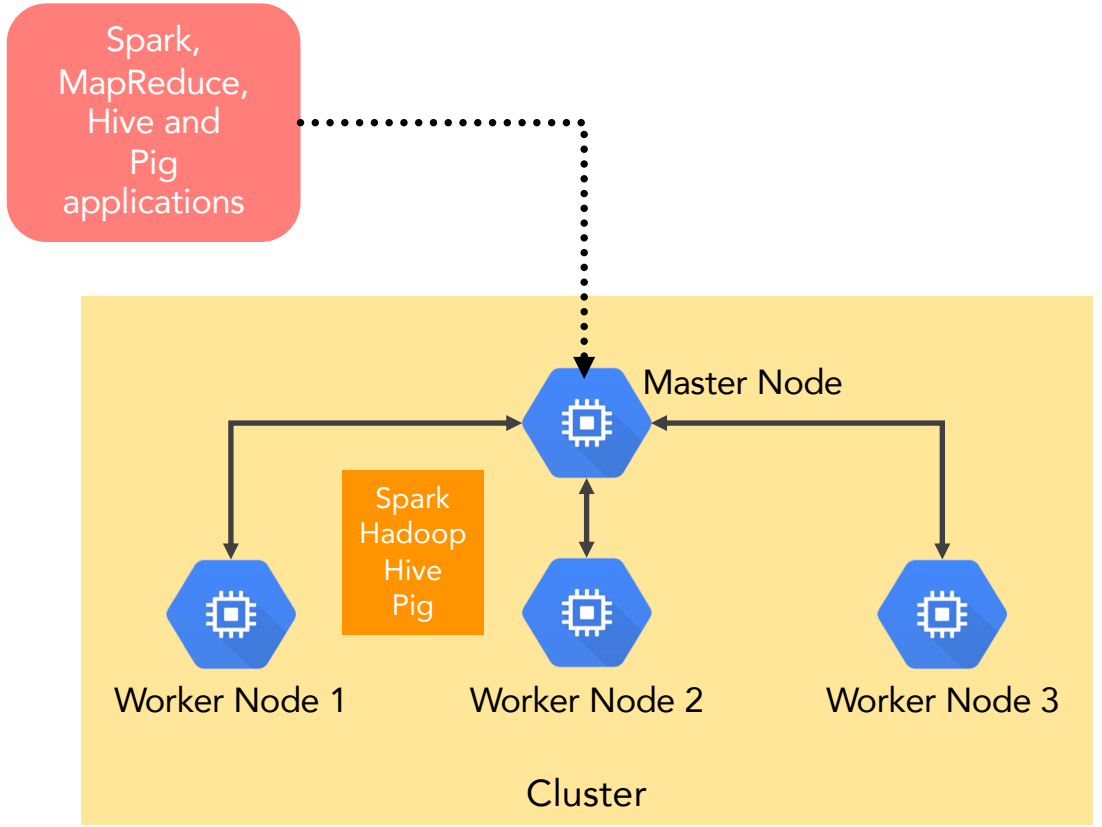
Dataproc Introduction

- What is Cloud Dataproc
- Install
  - ✓ Hadoop
  - ✓ Spark
  - ✓ Hive
  - ✓ Pig



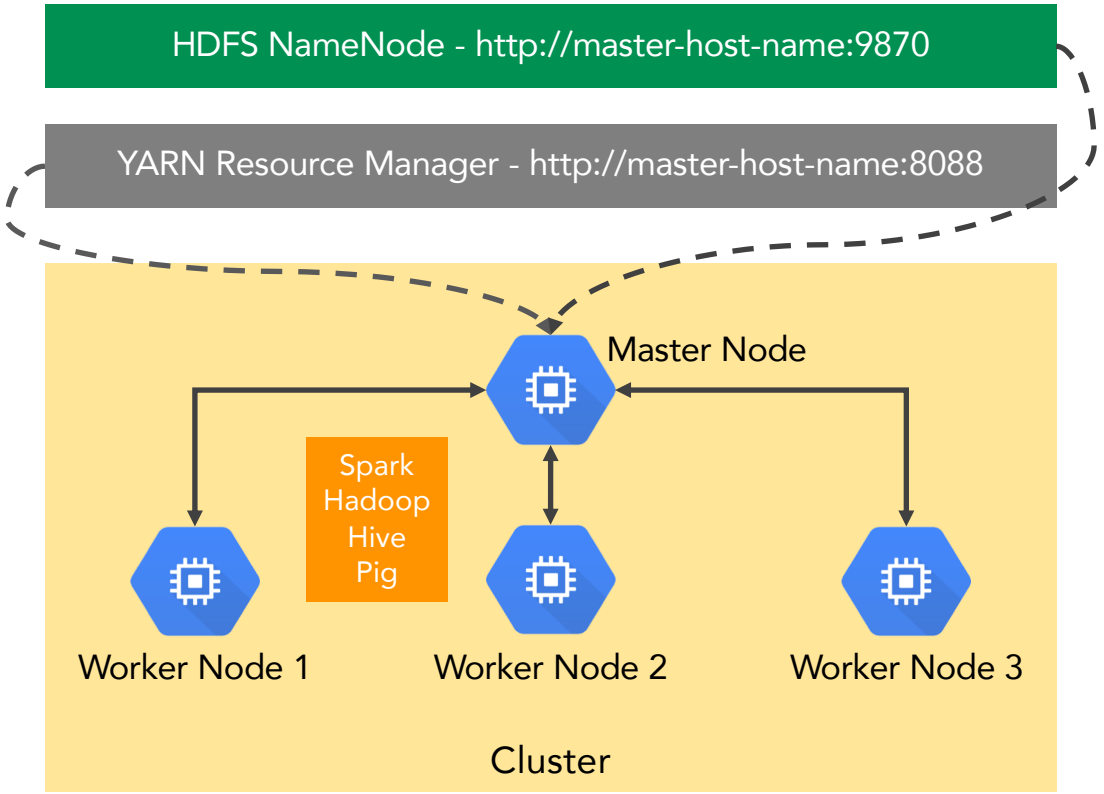
Dataproc Introduction

- What is Cloud Dataproc
  - ✓ Managed Cluster
- Pre-installed
  - ✓ Hadoop
  - ✓ Spark
  - ✓ Hive
  - ✓ Pig



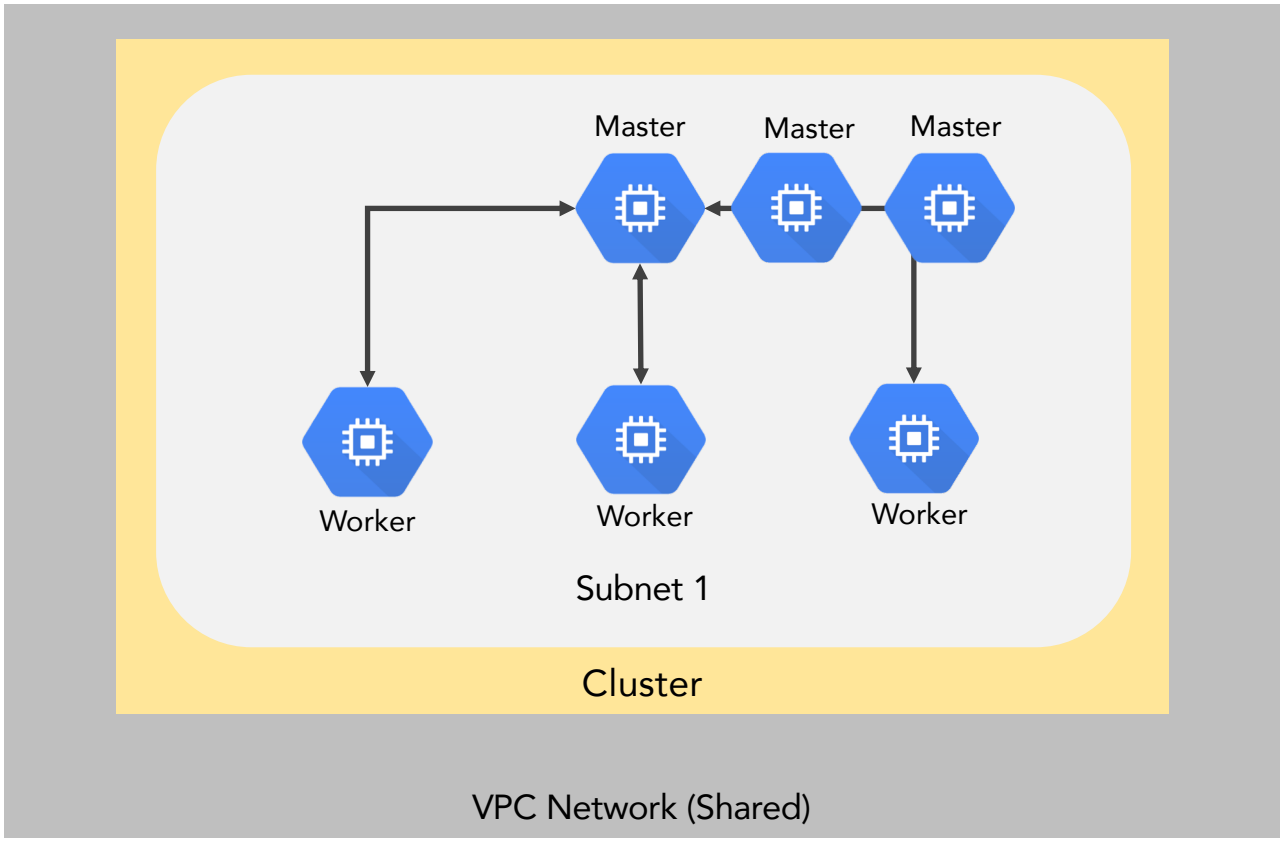
Dataproc Introduction

- What is Cloud Dataproc
  - ✓ Managed Cluster
- Pre-installed
  - ✓ Hadoop
  - ✓ Spark
  - ✓ Hive
  - ✓ Pig
- Web interface
  - ✓ Cloud Shell Web Preview
  - ✓ gcloud SSH Tunnel



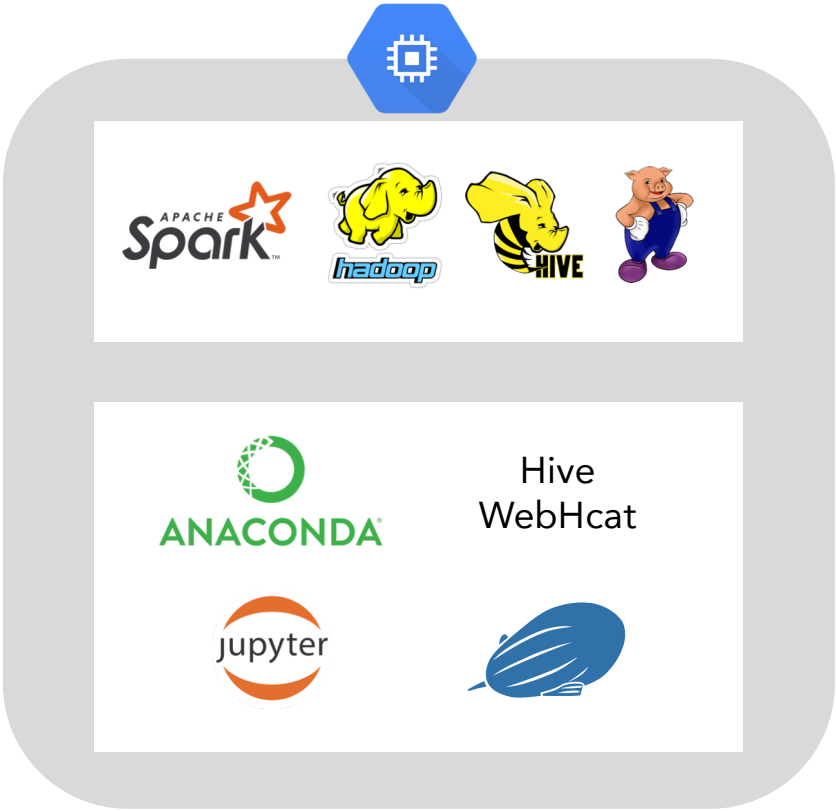
Dataproc Introduction

- What is Cloud Dataproc
  - ✓ Managed Cluster
- Pre-installed
  - ✓ Hadoop
  - ✓ Spark
  - ✓ Hive
  - ✓ Pig
- Web interface
  - ✓ Cloud Shell Web Preview
  - ✓ gcloud SSH Tunnel
- Network
  - ✓ Internal IP address only
- Multi-master



Components

- Essential components
- Optional components
  - ✓ Anaconda
  - ✓ Hive WebHCat (on port 50111)
  - ✓ Python Jupyter Notebook (on port 8123)
  - ✓ Zeppelin notebook (on port 8080)



Demo

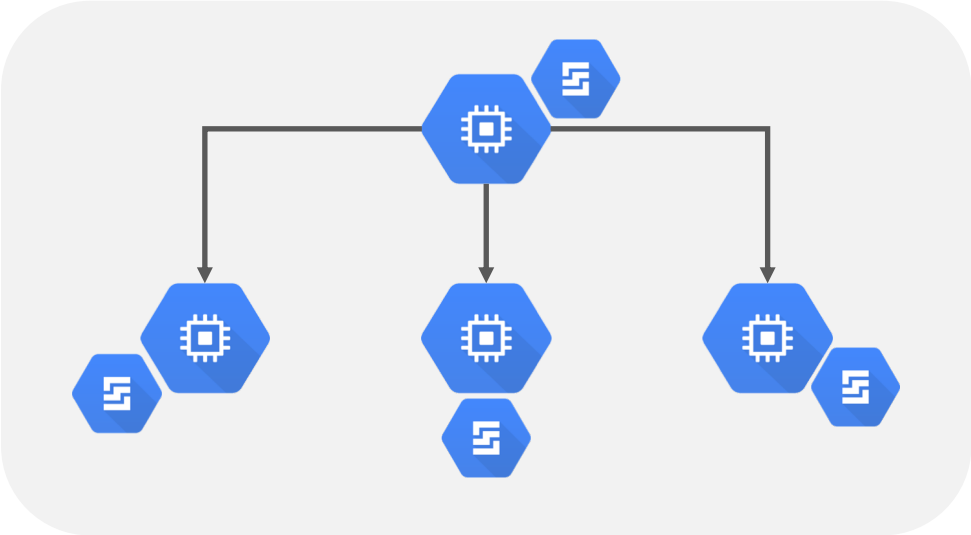
Create a VPC network

Create Cluster

Check various components

Web Interface

Cluster Compute Engines

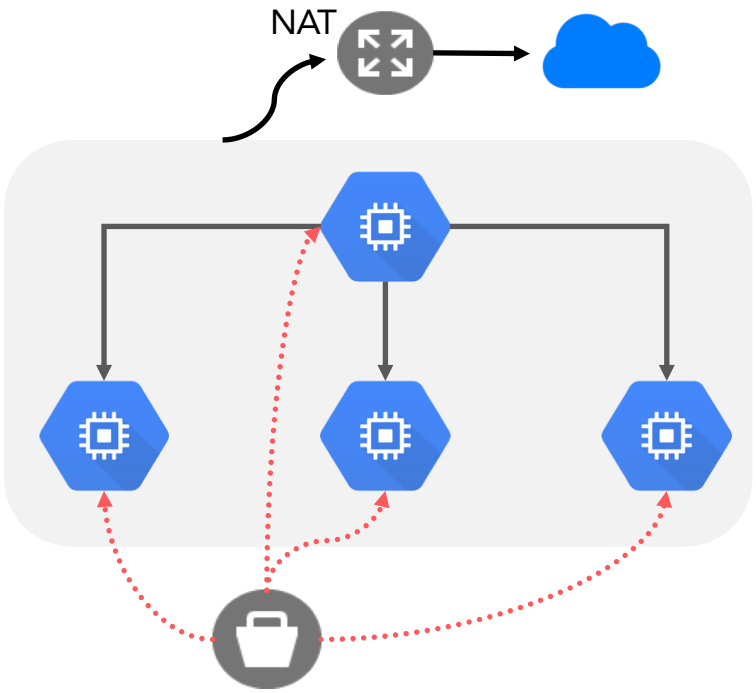


Demo - Various Compute Engine options while creating a cluster

- Custom Machine Types
- Persistent Disks
- Local SSDs
- Preemptible VMs
- Autoscaling Cluster
  - HDFS
  - Spark Structured Streaming
- Cluster Metadata
- Encryption using CMEK

`dataproc-bucket`
`dataproc-region`
`dataproc-worker-count`
`dataproc-cluster-name`
`dataproc-cluster-uuid`
`dataproc-role`
`dataproc-master`
`dataproc-master-additional`

Cluster Configuration

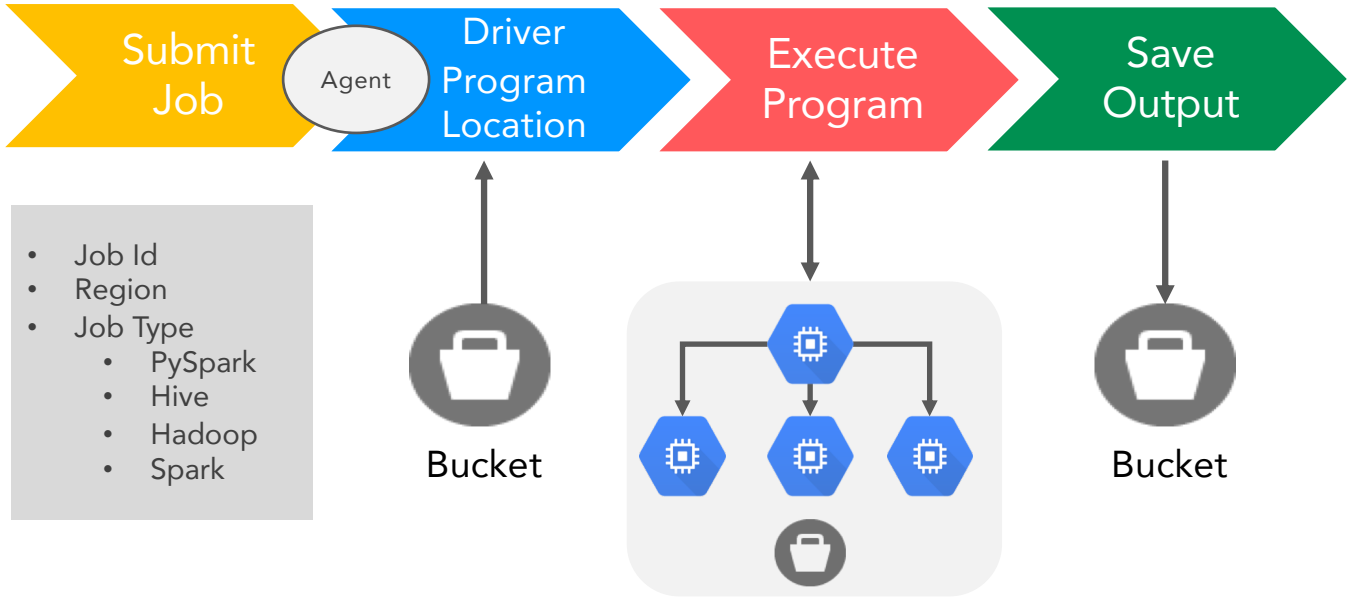


gs://dataproc-initialization-actions

- Initialization Actions
  - Executed on each node during creation
  - Newly created node using autoscaling
  - Executed as “root” user
  - Take necessary care when internet access is required
  - Jupyter notebaak
  - Datalab notebook
  - Presto
  - /var/log/dataproc-initialization-script-x.log

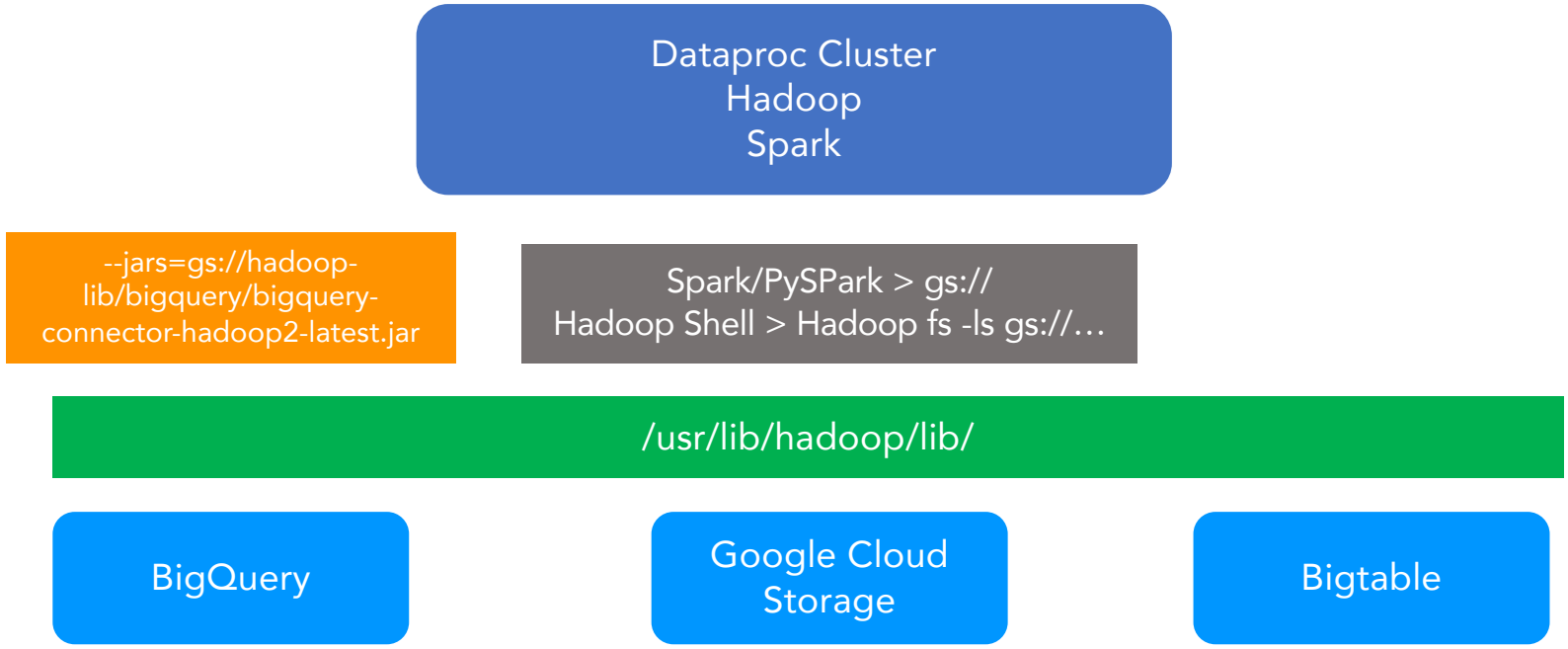
Demo - Initialize actions to install Jupyter notebook

Running Job

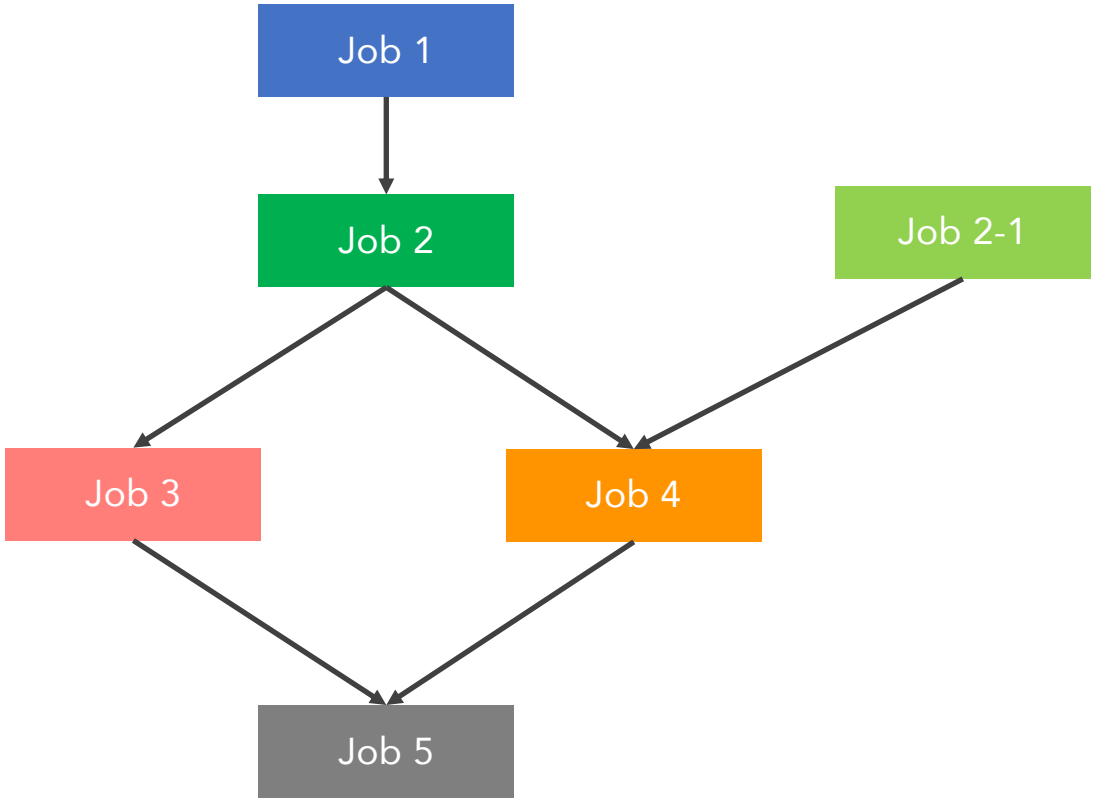


Demo – How to run a job

Dataproc Connectors



Workflow Templates



- Managed Cluster
- Cluster Selector
- Parameterized

Workflow Templates

Create Template

```
gcloud dataproc workflow-templates create gde-dataproc-template
```

Configure Cluster

```
gcloud dataproc workflow-templates set-managed-cluster gde-dataproc-template
        --master-machine-type machine-type
        --worker-machine-type machine-type
        --num-workers number
        --cluster-name gde-dataproc-temp-cluster

gcloud dataproc workflow-templates set-cluster-selector gde-dataproc-template --
cluster-labels [ cluster LABELs ]
```

Add Jobs

```
gcloud dataproc workflow-templates add-job pyspark --step-id step100 --
workflow-template gde-dataproc-template -- [ARGS]

gcloud dataproc workflow-templates add-job pyspark --step-id step200 --
start-after step100 --workflow-template gde-dataproc-template  -- [ JOB
ARGS ]
```

Running Workflow

```
gcloud dataproc workflow-templates instantiate gde-dataproc-template
```

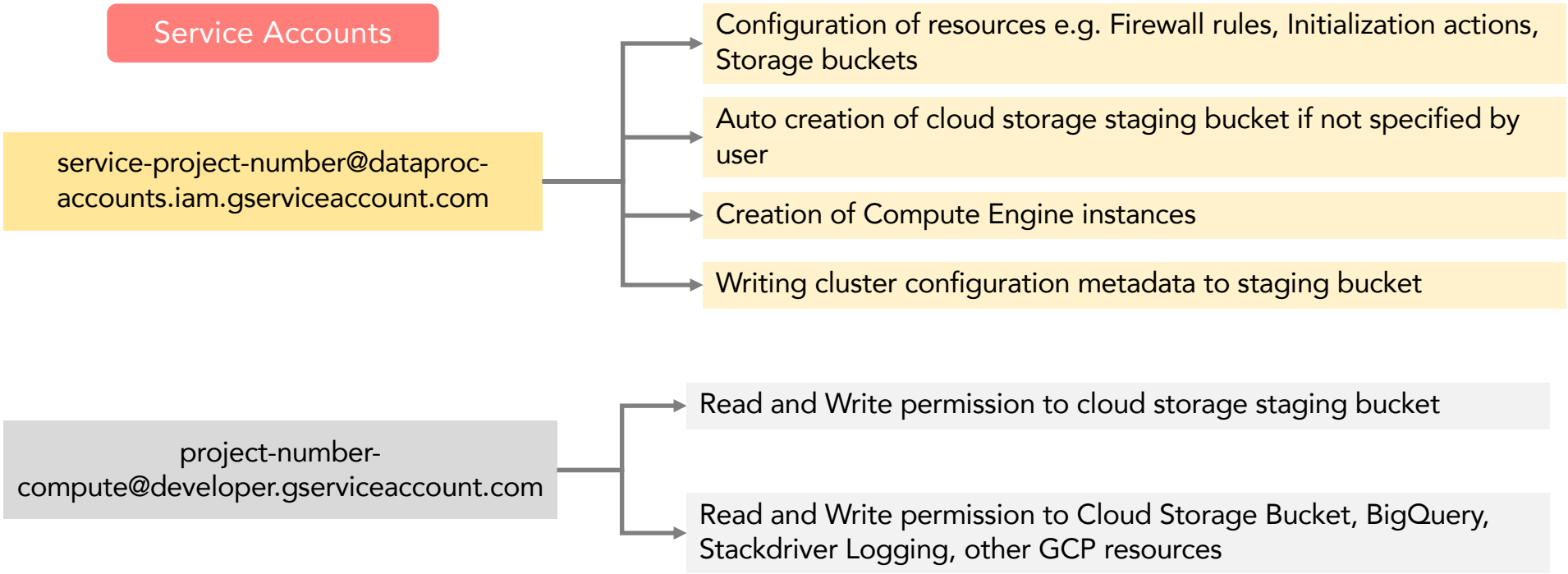
Demo

Create Workflow Template

Add Job

Execute with Dependencies

Access Control



Access Control – IAM permissions

Clusters Permissions

Method	Required Permission(s)
projects.regions.clusters.create [1, 2]	dataproc.clusters.create
projects.regions.clusters.get	dataproc.clusters.get
projects.regions.clusters.list	dataproc.clusters.list
projects.regions.clusters.patch [1, 2]	dataproc.clusters.update
projects.regions.clusters.delete [1]	dataproc.clusters.delete
projects.regions.clusters.diagnose [1]	dataproc.clusters.use

Operations Permissions

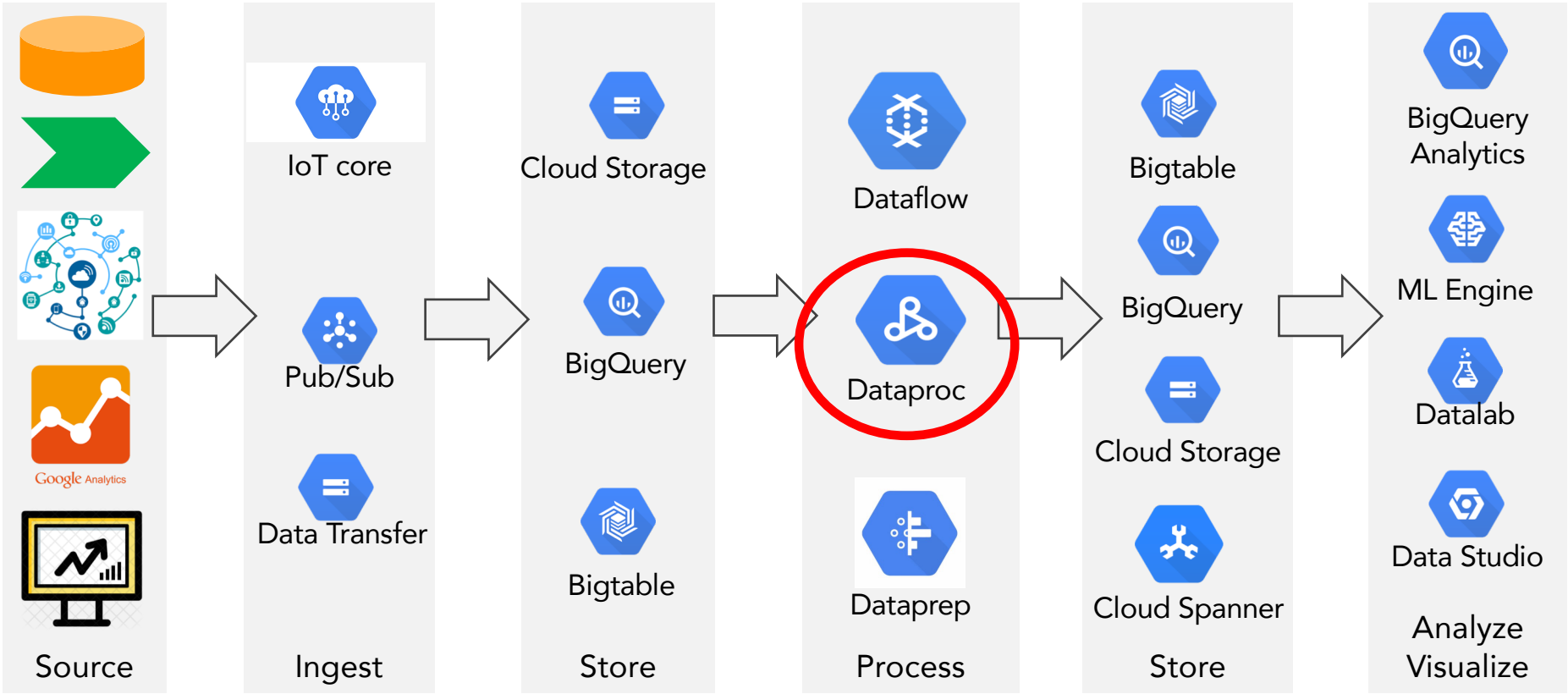
Method	Required Permission(s)
projects.regions.operations.get	dataproc.operations.get
projects.regions.operations.list	dataproc.operations.list
projects.regions.operations.cancel	dataproc.operations.cancel
projects.regions.operations.delete	dataproc.operations.delete

Workflow Template Permissions

Method	Required Permission(s)
projects.regions.workflowTemplates.instantiate	dataproc.workflowTemplates.instantiate
projects.regions.workflowTemplates.instantiateInline	dataproc.workflowTemplates.instantiateInline
projects.regions.workflowTemplates.create	dataproc.workflowTemplates.create
projects.regions.workflowTemplates.get	dataproc.workflowTemplates.get
projects.regions.workflowTemplates.list	dataproc.workflowTemplates.list
projects.regions.workflowTemplates.post	dataproc.workflowTemplates.update
projects.regions.workflowTemplates.delete	dataproc.workflowTemplates.delete

roles/dataproc.editor
roles/dataproc.viewer
roles/dataproc.worker

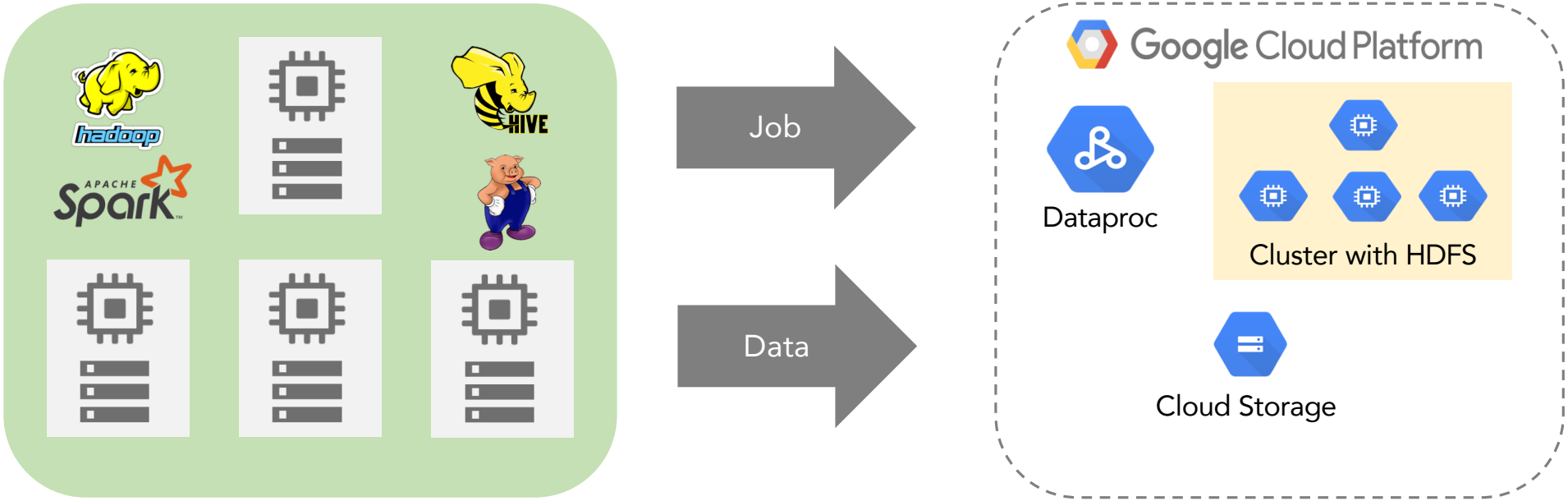
Data Analytics



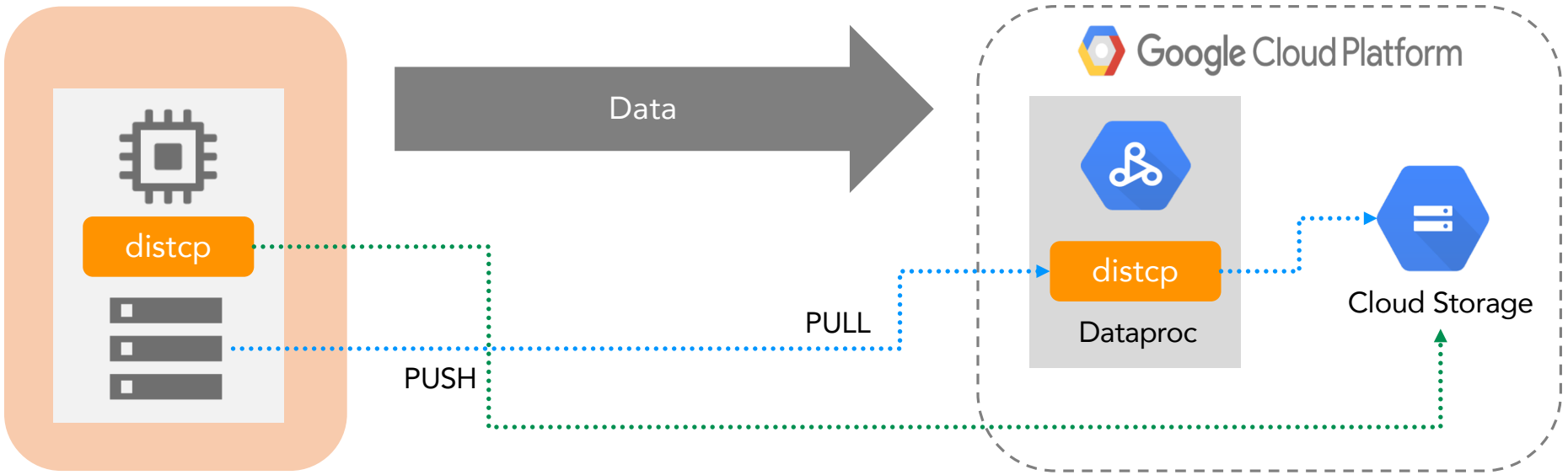
Thank You!!


Migration Consideration

Migrating Hadoop

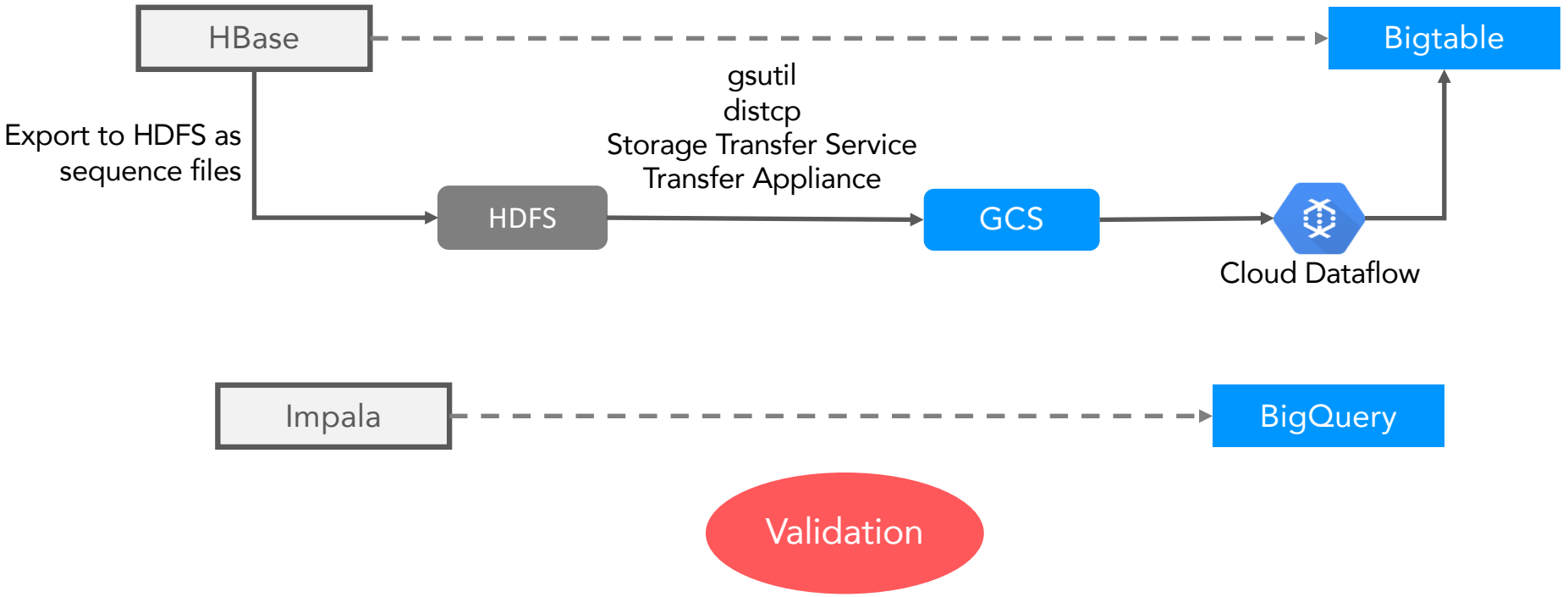


Migrating Hadoop - Data




```
hadoop distcp hdfs://source-name-node:9820/hddata/ gs://gcp-data-engineer-deep/hddata/
```

Migrating Hadoop - Data



Migrating Hadoop - Jobs

Dataproc

Cloud Storage

1. Update job to point to data stored in GCS
2. Hadoop, Spark, Hive, Pig
3. Initialization Actions
4. Cluster Configuration
5. Connectors
6. Additional libraries while running Spark jobs
   - Custom Images
   - Initialization Actions
7. Stackdriver logging and monitoring
8. Cost optimization – preemptible instances