Amirkabir University of Technology
(Tehran Polytechnic)

Cloud Computing

Spark Architecture and Useful Links
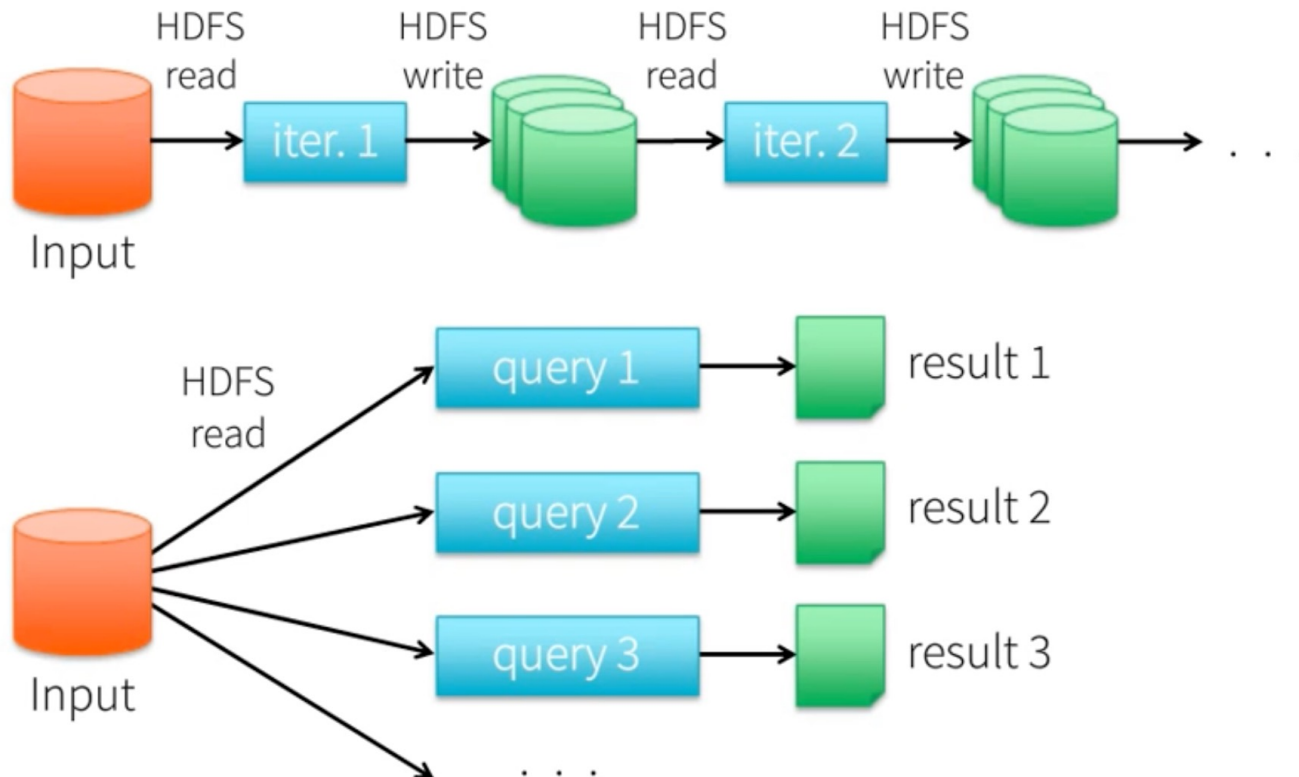
Seyyed Ahmad Javadi

sajavadi@aut.ac.ir

Fall 2022

A Quick Review

Few slides from Mr. Zahari's presentation available in the link below:

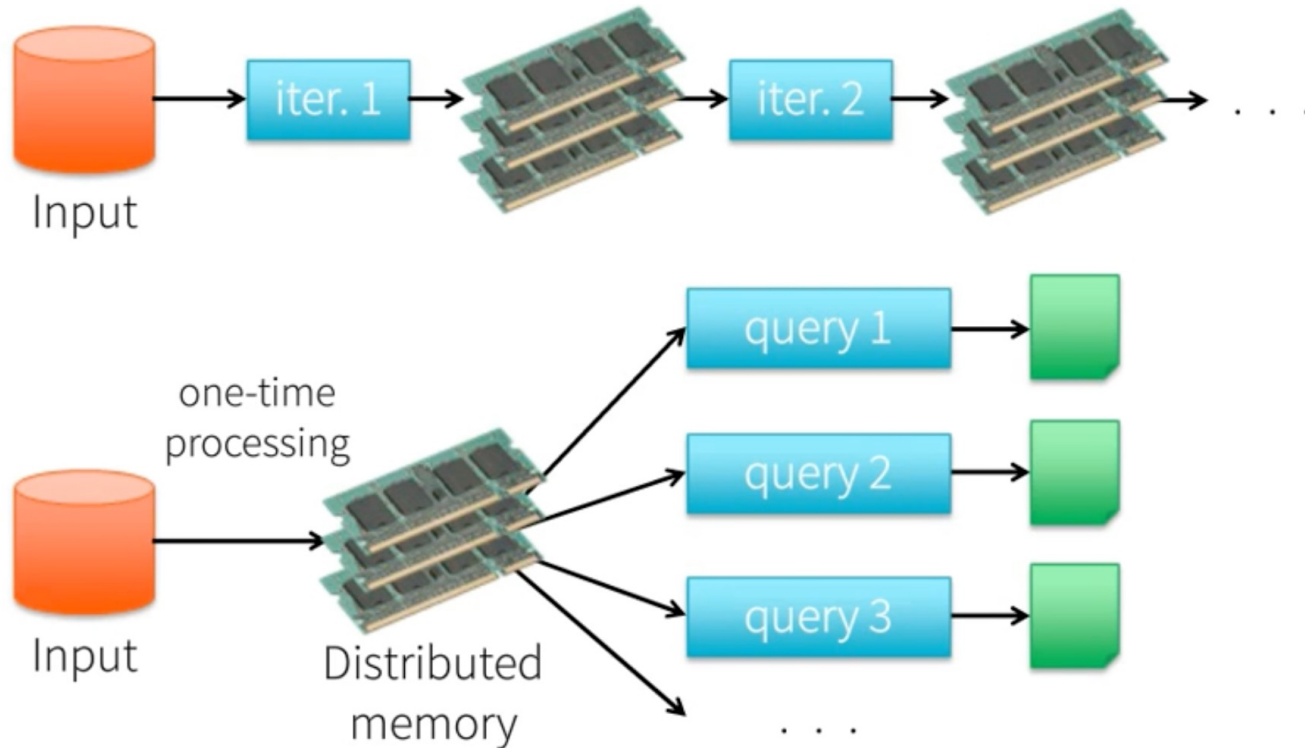https://www.youtube.com/watch?v=d9D-Z3-44F8&t=206s

Data Sharing in MapReduce

Slow due to replication and disk I/O

Source: Matei Zaharia's persentatioin

What We'd Like

10-100x faster than network and disk

Source: Matei Zaharia's persentatioin

Spark Programming Model

Resilient Distributed Datasets (RDDs)

- Collections of objects stored in RAM or disk across cluster
- Built via parallel transformations (map, filter, …)
- Automatically rebuilt on failure

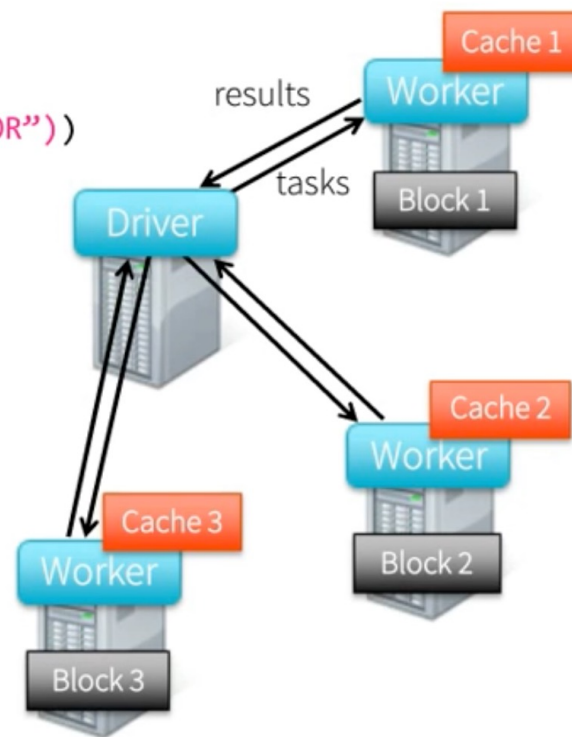Source: Matei Zaharia's persentatioin

Example: Log Mining

Load error messages from a log into memory, then interactively search for various patterns

```
lines = spark.textFile("hdfs://...")
errors = lines.filter(lambda s: s.startswith("ERROR"))
messages = errors.map(lambda s: s.split('\t')[2])
messages.cache()


messages.filter(lambda s: "MySQL" in s).count()
messages.filter(lambda s: "Redis" in s).count()

. . .
```

Example: full-text search of Wikipedia in 0.5 sec (vs 20s for on-disk data)

results
tasks

Driver

Cache 1
Worker
Block 1

Cache 2
Worker
Block 2

Cache 3
Worker
Block 3

Source: Matei Zaharia's persentatioin

On-Disk Performance

Time to sort 100TB

2013 Record:
Hadoop

2100 machines

72 minutes

2014 Record:
Spark

207 machines

23 minutes

Source: Matei Zaharia's persentatioin

Combining Processing Types

```
// Load data using SQL
points = ctx.sql("select latitude, longitude from tweets")

// Train a machine learning model
model = KMeans.train(points, 10)

// Apply it to a stream
sc.twitterStream(...)
    .map(lambda t: (model.predict(t.location), 1))
    .reduceByWindow("5s", lambda a, b: a + b)
```

Source: Matei Zaharia's persentatioin

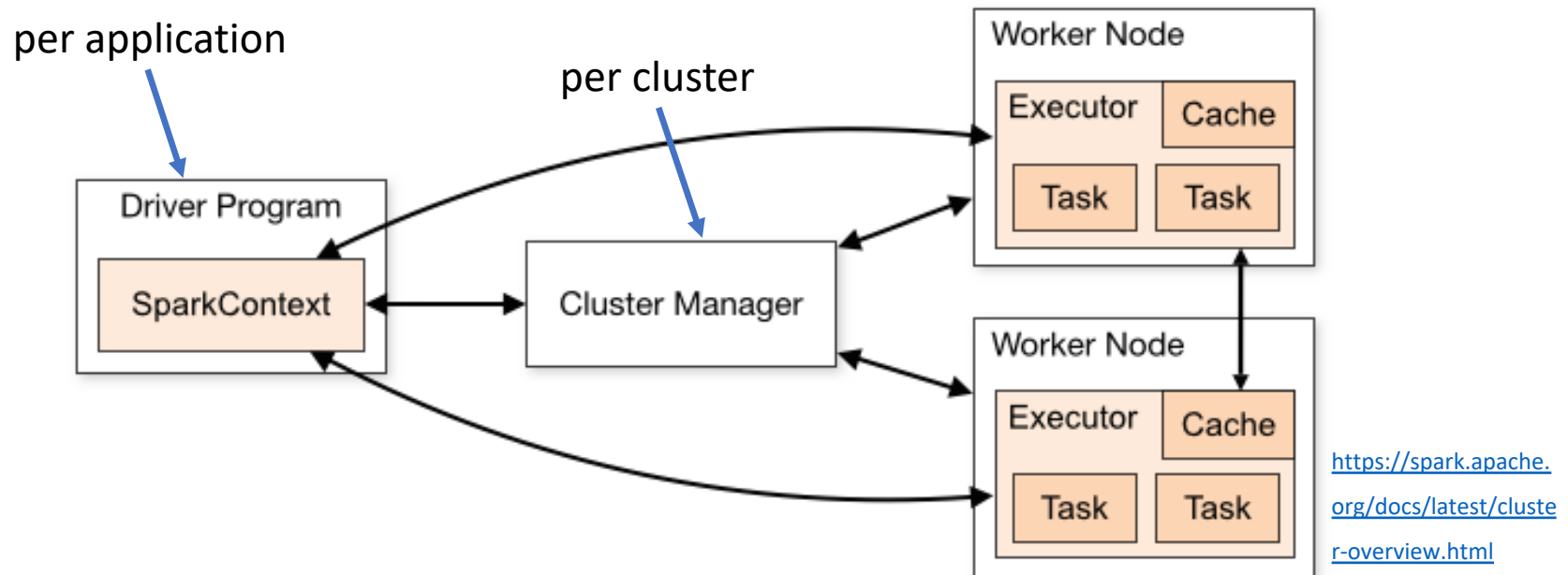# Combining Processing Types

Separate systems:

Spark:

Spark Architeture

Cluster Mode Overview

per application

per cluster

Worker Node
Executor   Cache
Task   Task

Worker Node
Executor   Cache
Task   Task

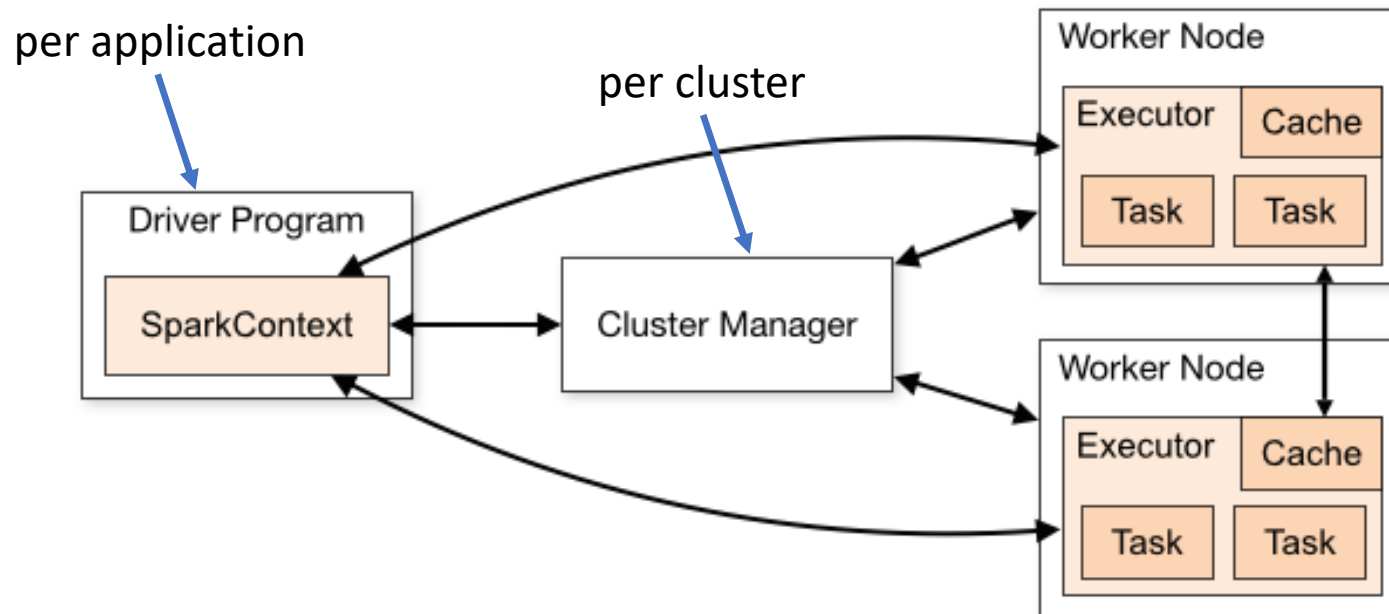Driver Program
SparkContext

Cluster Manager

https://spark.apache.org/docs/latest/cluster-overview.html

> SparkContext can connect to several types of *cluster managers:*

- Spark's own standalone cluster manager,
- Mesos,
- YARN or
- Kubernetes

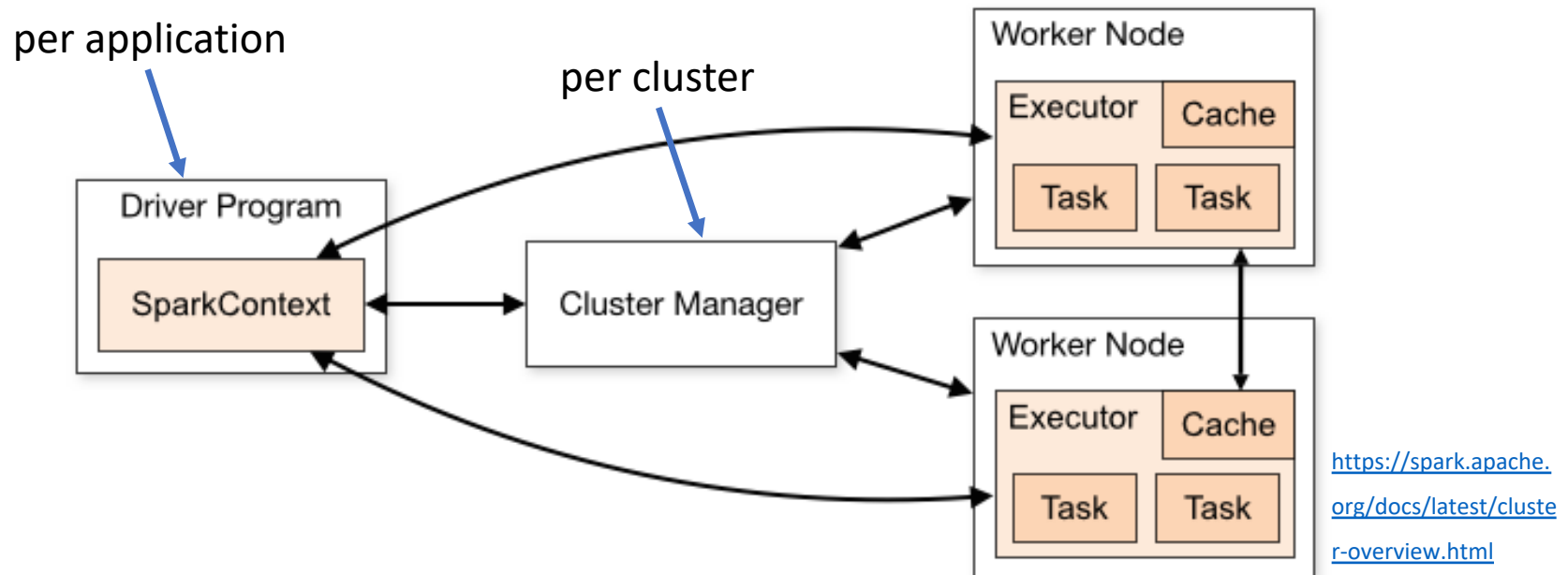> Cluster manager allocates resources across applications.

Cluster Mode Overview (cont.)

per application

per cluster

Driver Program

SparkContext

Cluster Manager

Worker Node

Executor | Cache

Task | Task

Worker Node

Executor | Cache

Task | Task

https://spark.apache.org/docs/latest/cluster-overview.html

➢ Each application gets its own executor processes.

  ▪ Executor processes run tasks in multiple threads

  ▪ Benefit: isolating applications from each other

  ▪ Challenge: ***data cannot*** be shared across different Spark applications without writing it to an external storage system.
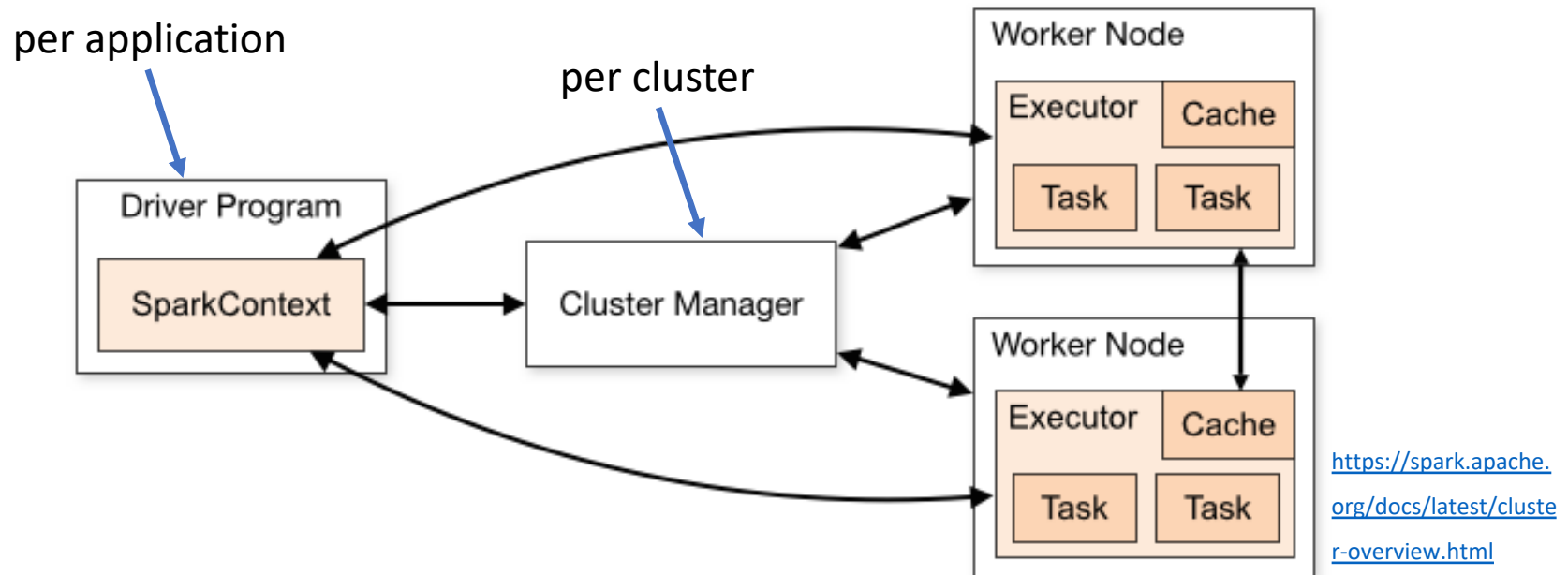
Cluster Mode Overview (cont.)

per application

per cluster

Worker Node

Executor | Cache

Task | Task

Driver Program

SparkContext

Cluster Manager

Worker Node

Executor | Cache

Task | Task

https://spark.apache.org/docs/latest/cluster-overview.html

➢ Spark is agnostic to the underlying cluster manager.

- As long as Spark can acquire executor processes & these communicate with each other.

Principles of Cloud Computing--S.A.Javadi

Cluster Mode Overview (cont.)

per application

per cluster

Driver Program

SparkContext

Cluster Manager

Worker Node

Executor | Cache

Task | Task

Worker Node

Executor | Cache

Task | Task

https://spark.apache.org/docs/latest/cluster-overview.html

➤ The driver program must listen for and accept incoming connections from its executors throughout its lifetime.

➤ As such, the driver program must be network addressable from the worker nodes.

Cluster Manager Types

➢ Standalone
- A simple cluster manager included with Spark making it easy to set up a cluster.

➢ Apache Mesos
- A general cluster manager that can also run Hadoop MapReduce and service applications. (Deprecated)

➢ Hadoop YARN
- The resource manager in Hadoop 2.

➢ Kubernetes
- An open-source system for automating deployment, scaling, and management of containerized applications.