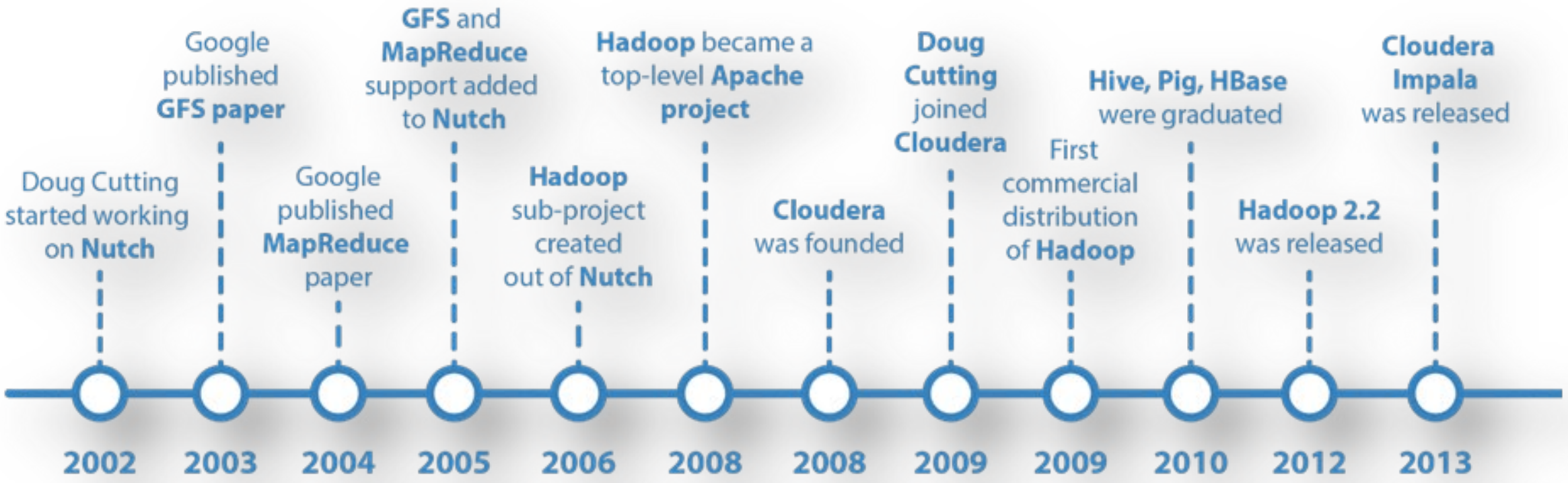
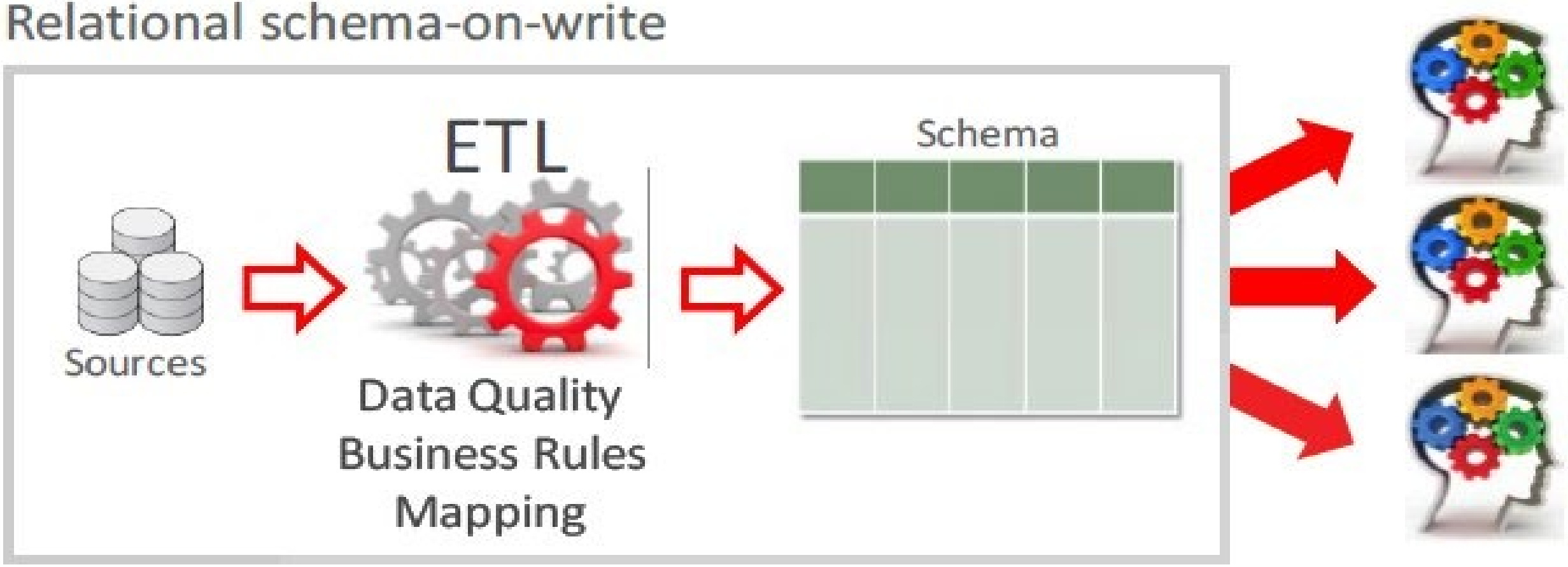
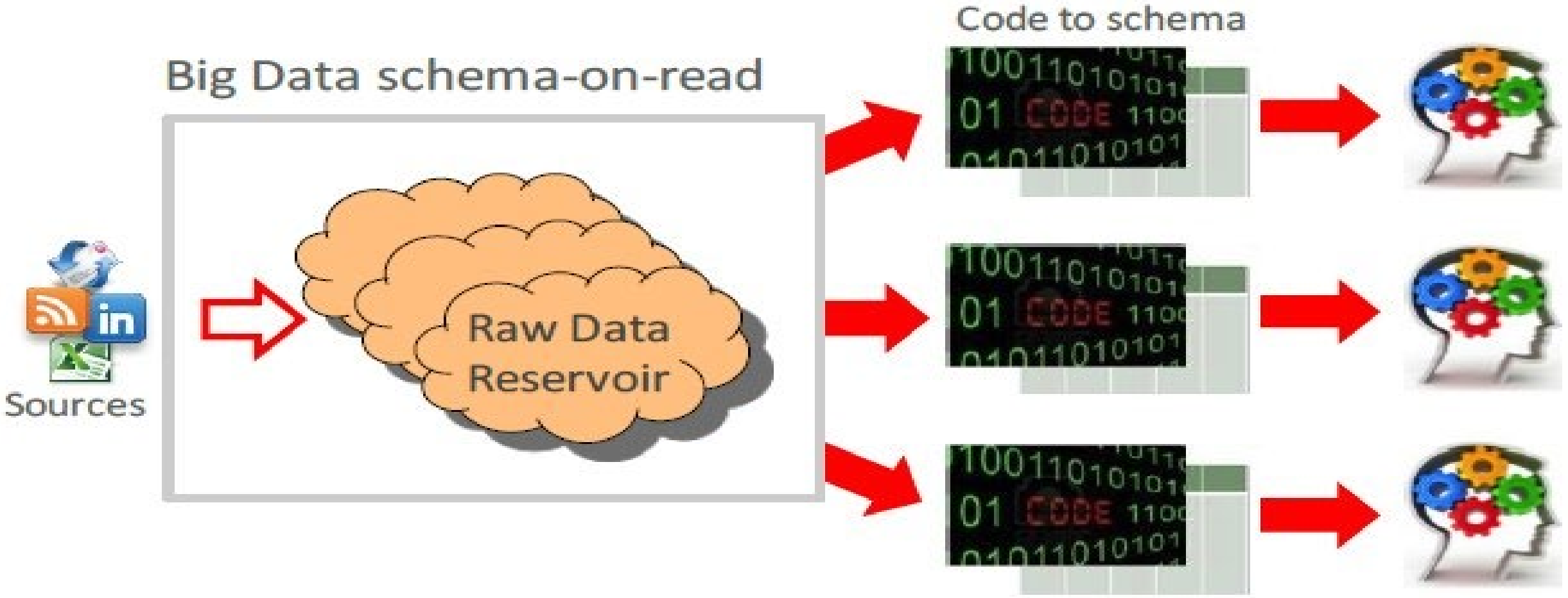
Hadoop "Fasi dell'evoluzione"



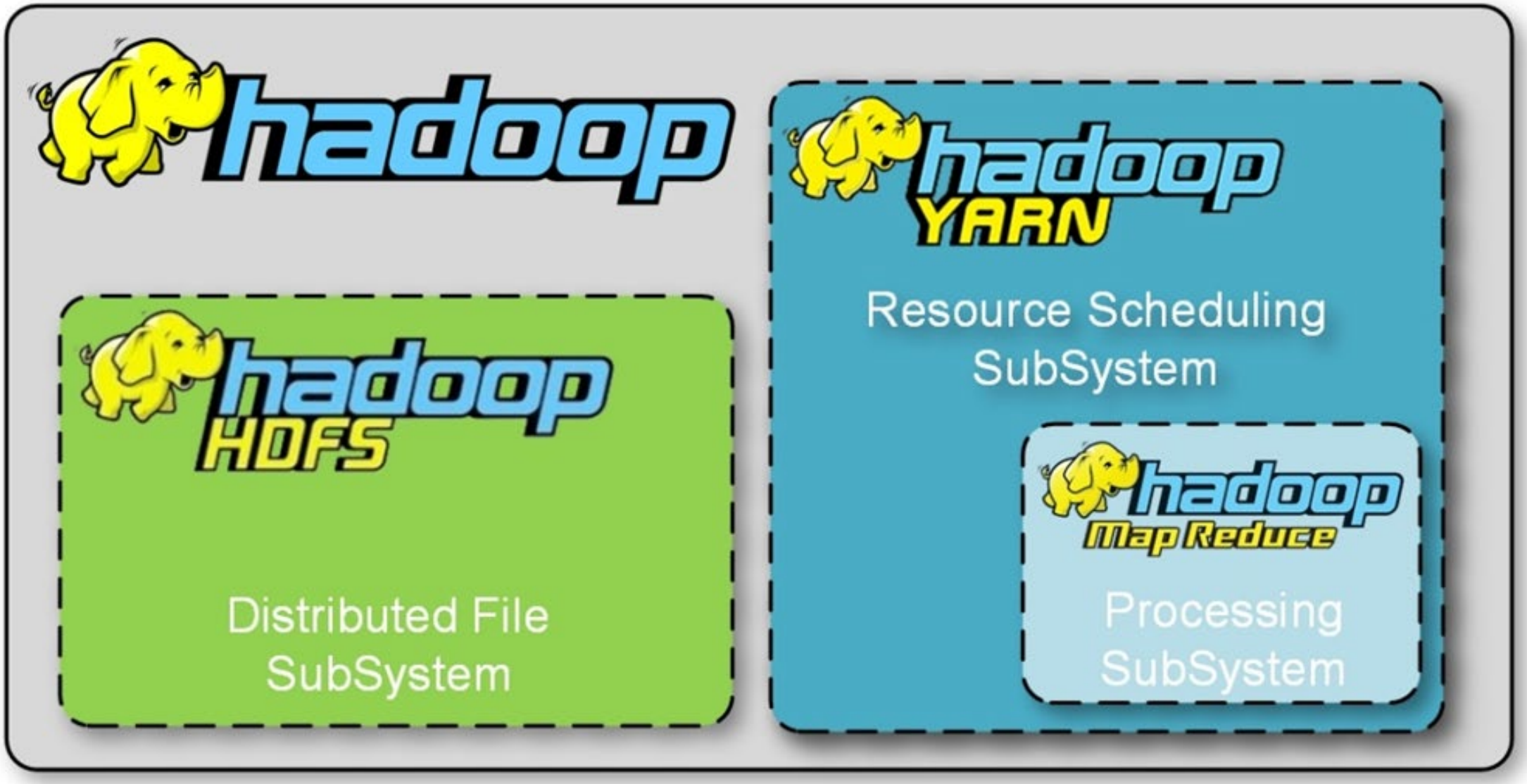


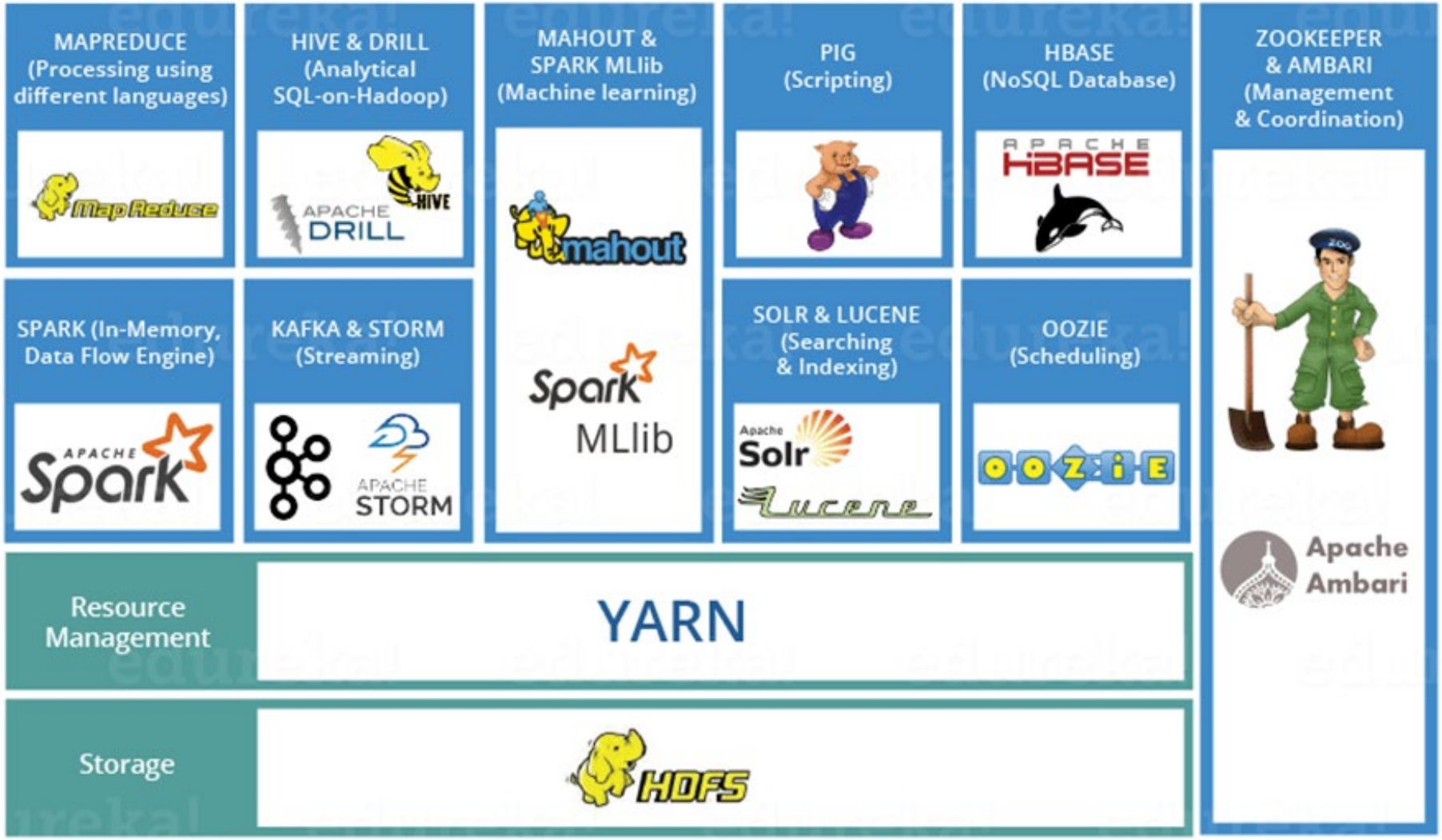
I sistemi che processano i dati in base allo Schema-on-write sono i database relazionali. Schema-on-write è definito come creazione di uno schema per i dati prima della scrittura nel database, dove i dati sono tipicamente fortemente tipizzati poiché è applicato uno schema predefinito su INSERT, UPDATE, o operazioni UPSERT.



Hadoop è definito come un sistema schema-on-read. Hadoop è schemaless rispetto alle operazioni di scrittura. Questo significa che può archiviare ed elaborare una vasta gamma di dati: documenti di testo non strutturati, JSON semi-strutturati (JavaScript Object Notation), documenti XML, dati strutturati estratti/generati da sistemi di database relazionali

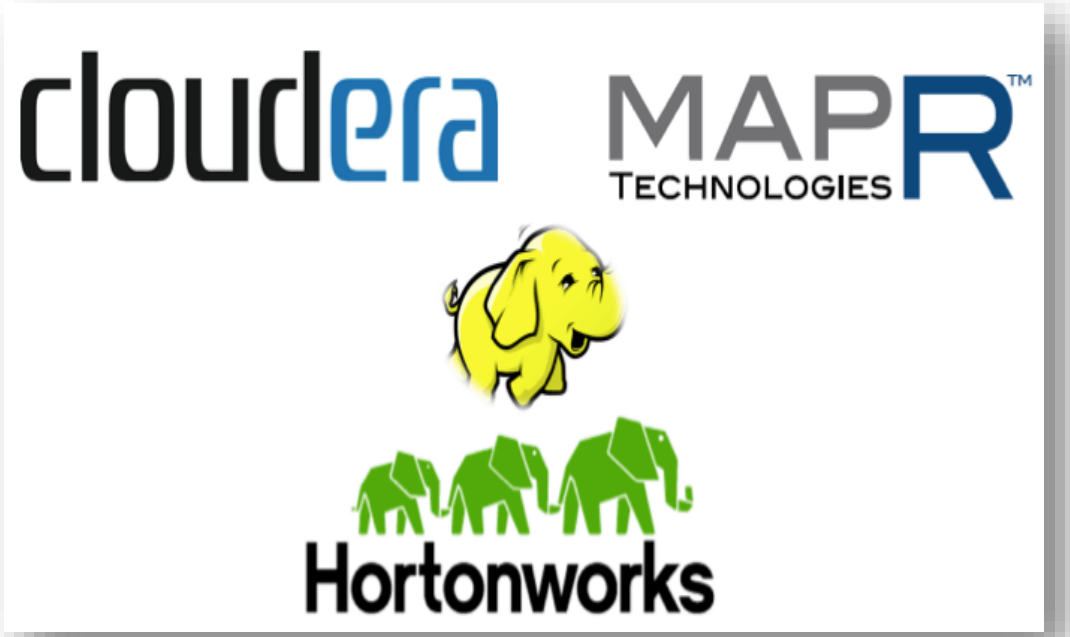
Hadoop "Core component"



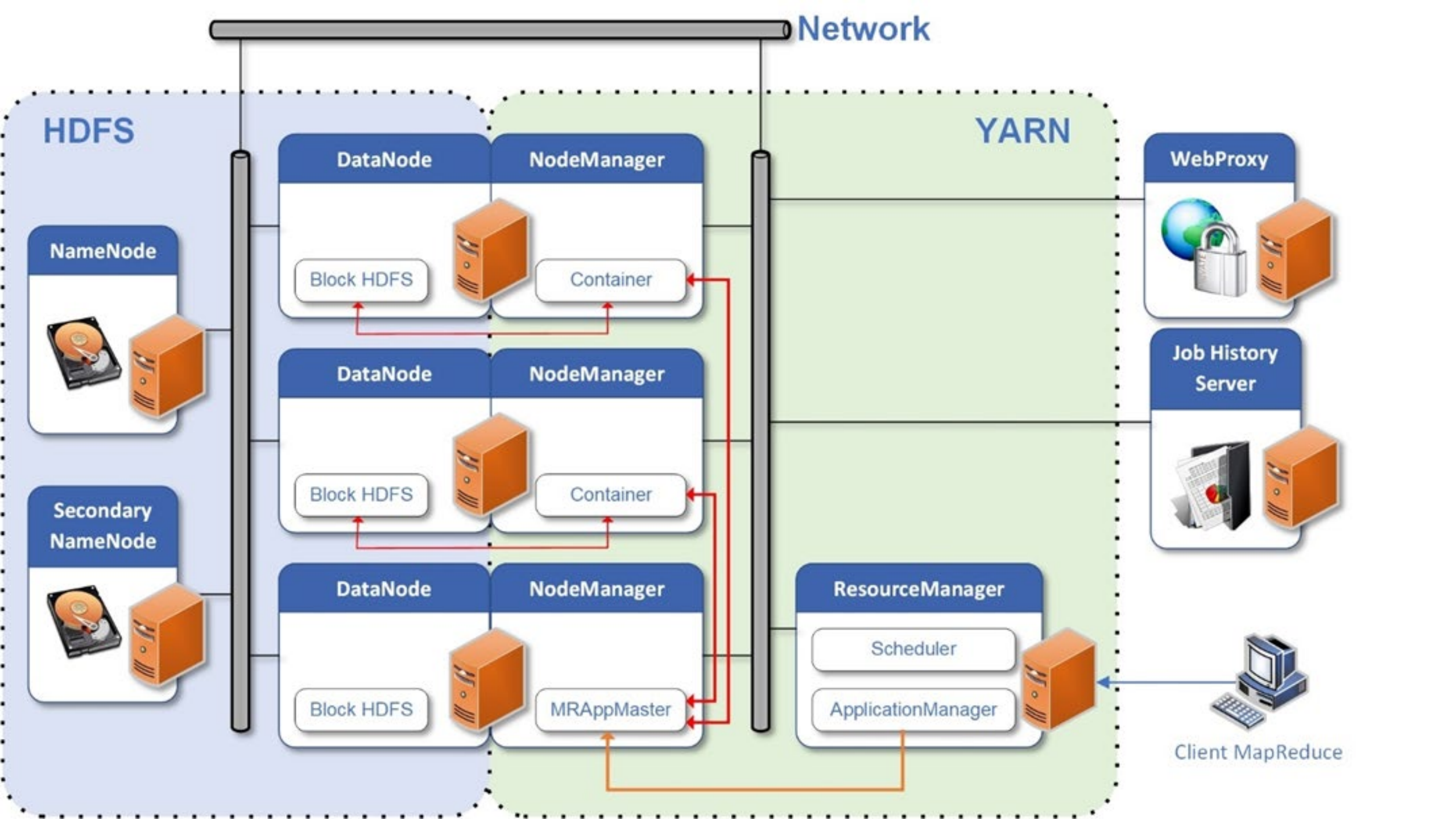


Hadoop "Ecosistema"

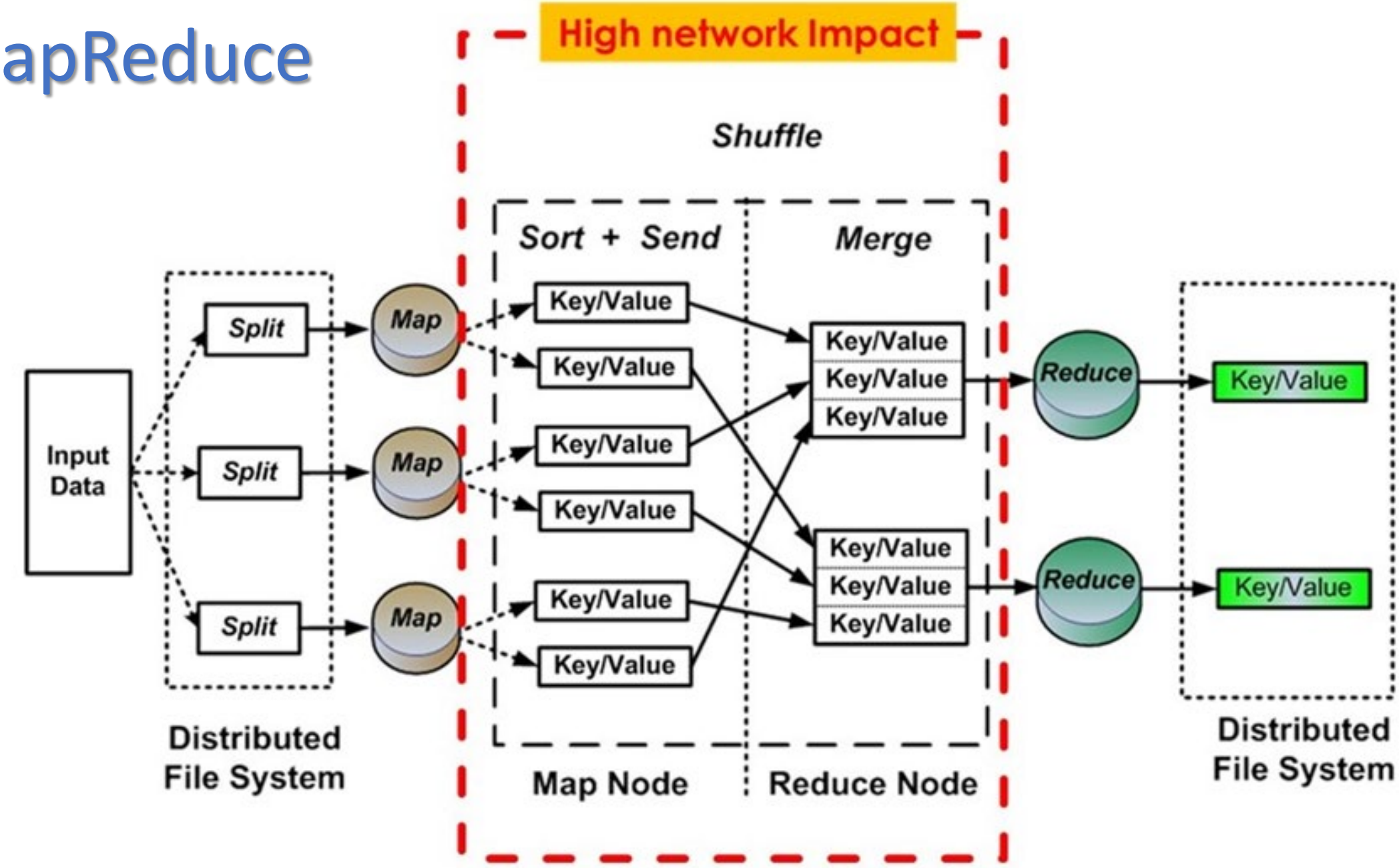
Hadoop "Distribuzioni"

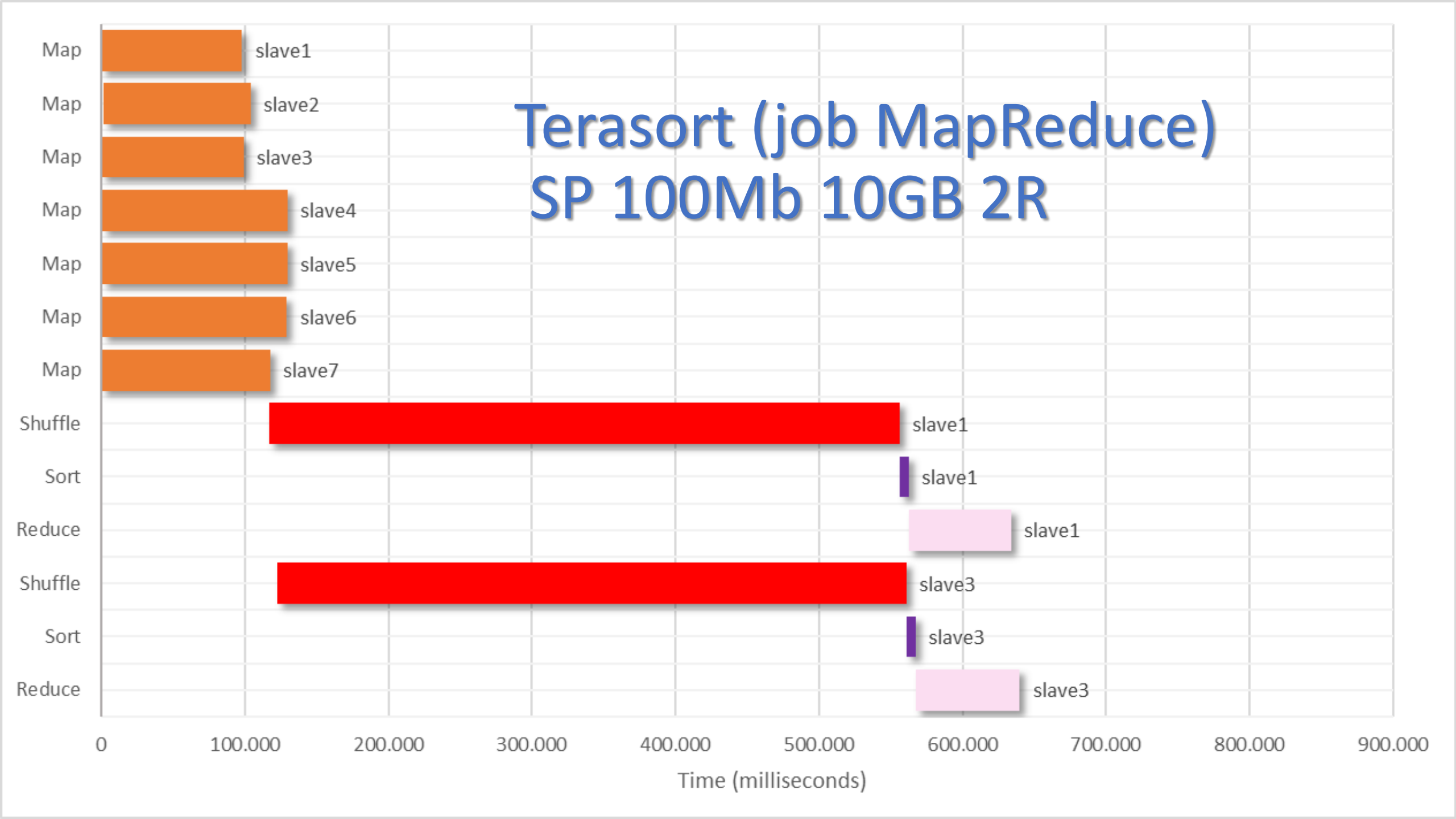


"Open Data Platform initiative (ODPi)"

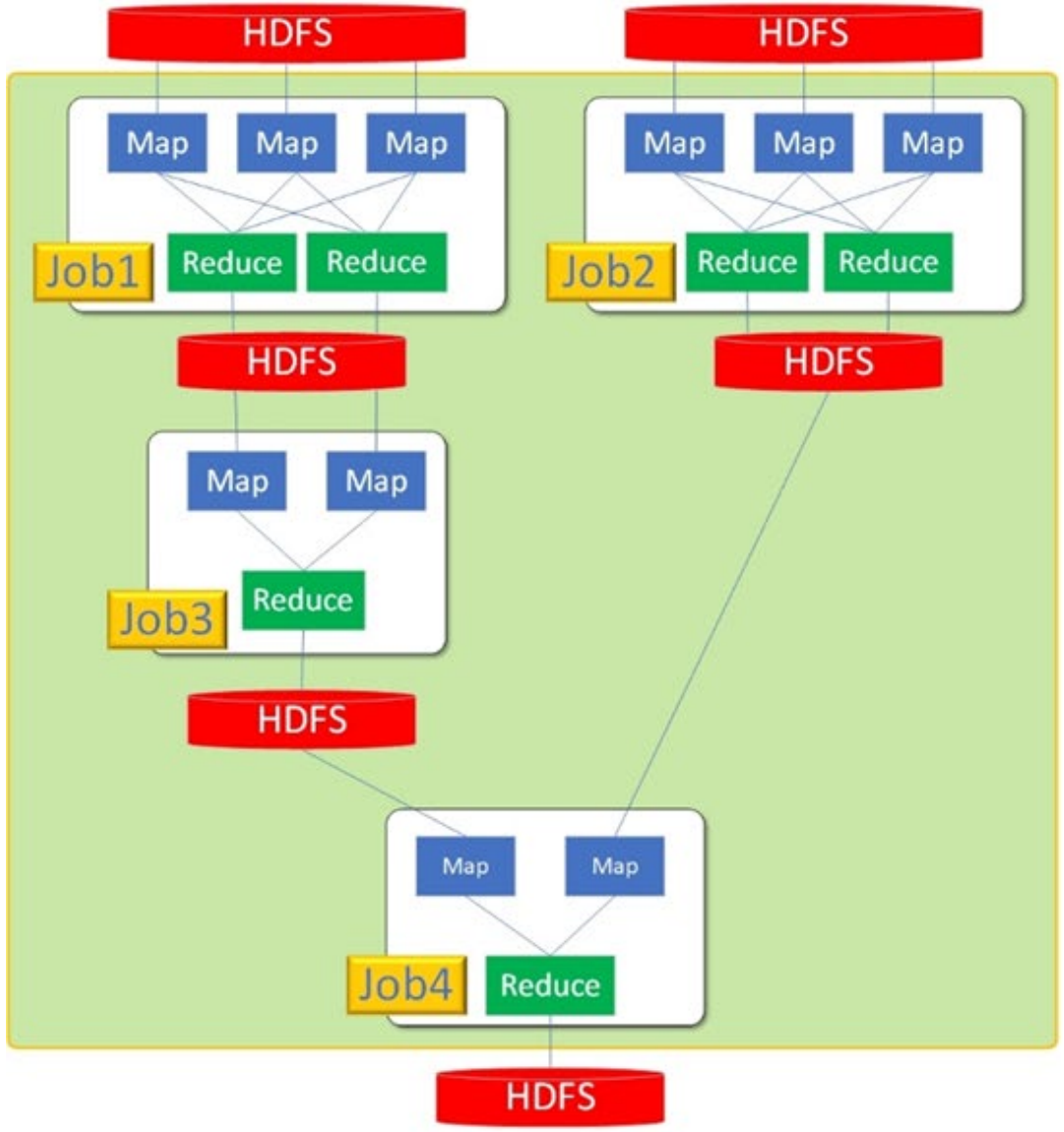

Network
HDFS
YARN
NameNode
Secondary NameNode
DataNode
NodeManager
Block HDFS
Container
DataNode
NodeManager
Block HDFS
Container
DataNode
NodeManager
Block HDFS
MRAppMaster
ResourceManager
Scheduler
ApplicationManager
WebProxy
Job History Server
Client MapReduce

MapReduce

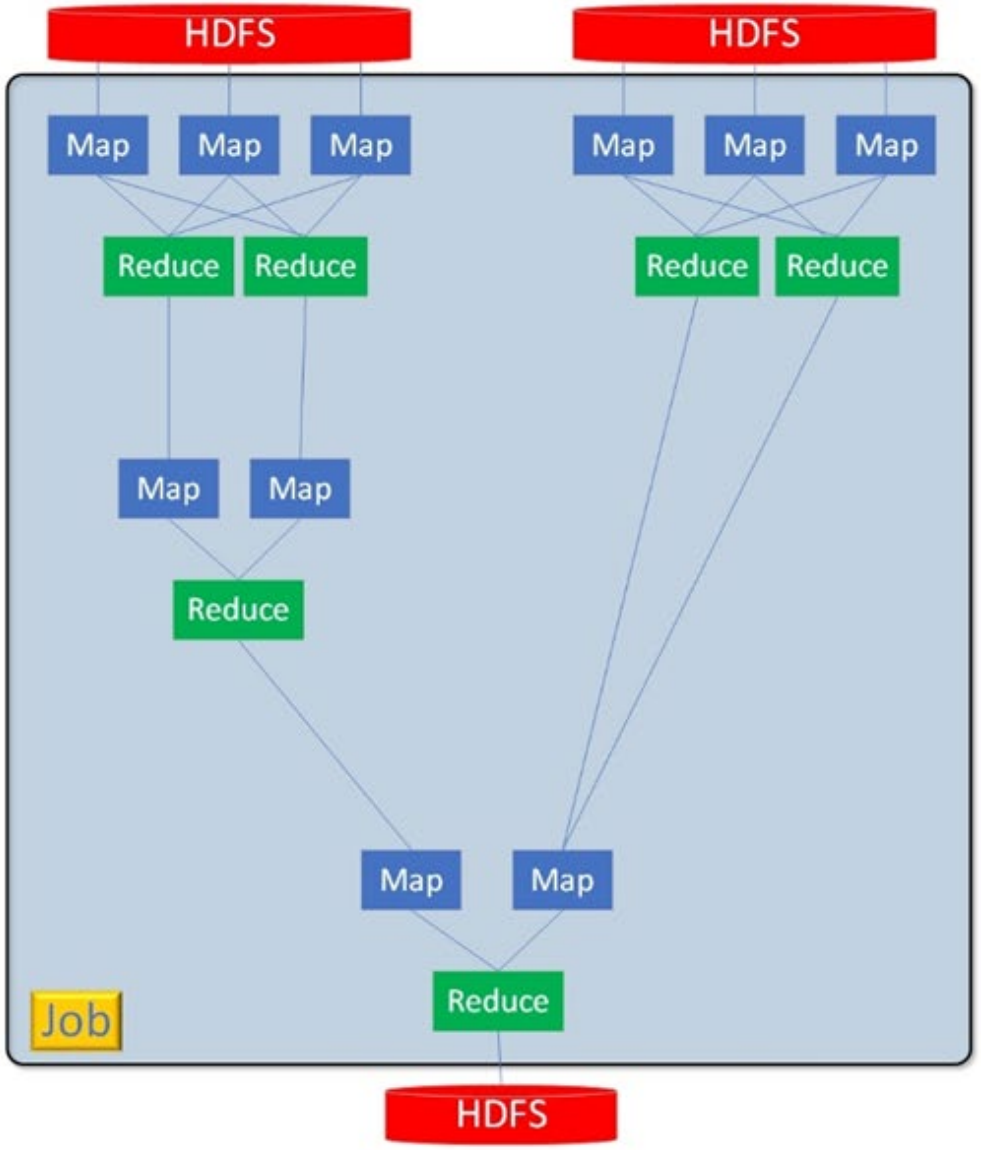

High network Impact
Shuffle
Sort + Send
Merge
Input Data
Split
Split
Split
Map
Map
Map
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Key/Value
Reduce
Reduce
Key/Value
Key/Value
Distributed File System
Map Node
Reduce Node
Distributed File System


Terasort (job MapReduce)
SP 100Mb 10GB 2R
Map
slave1
Map
slave2
Map
slave3
Map
slave4
Map
slave5
Map
slave6
Map
slave7
Shuffle
slave1
Sort
slave1
Reduce
slave1
Shuffle
slave3
Sort
slave3
Reduce
slave3
0
100.000
200.000
300.000
400.000
500.000
600.000
700.000
800.000
900.000
Time (milliseconds)

Differenze tra MapReduce e DAG

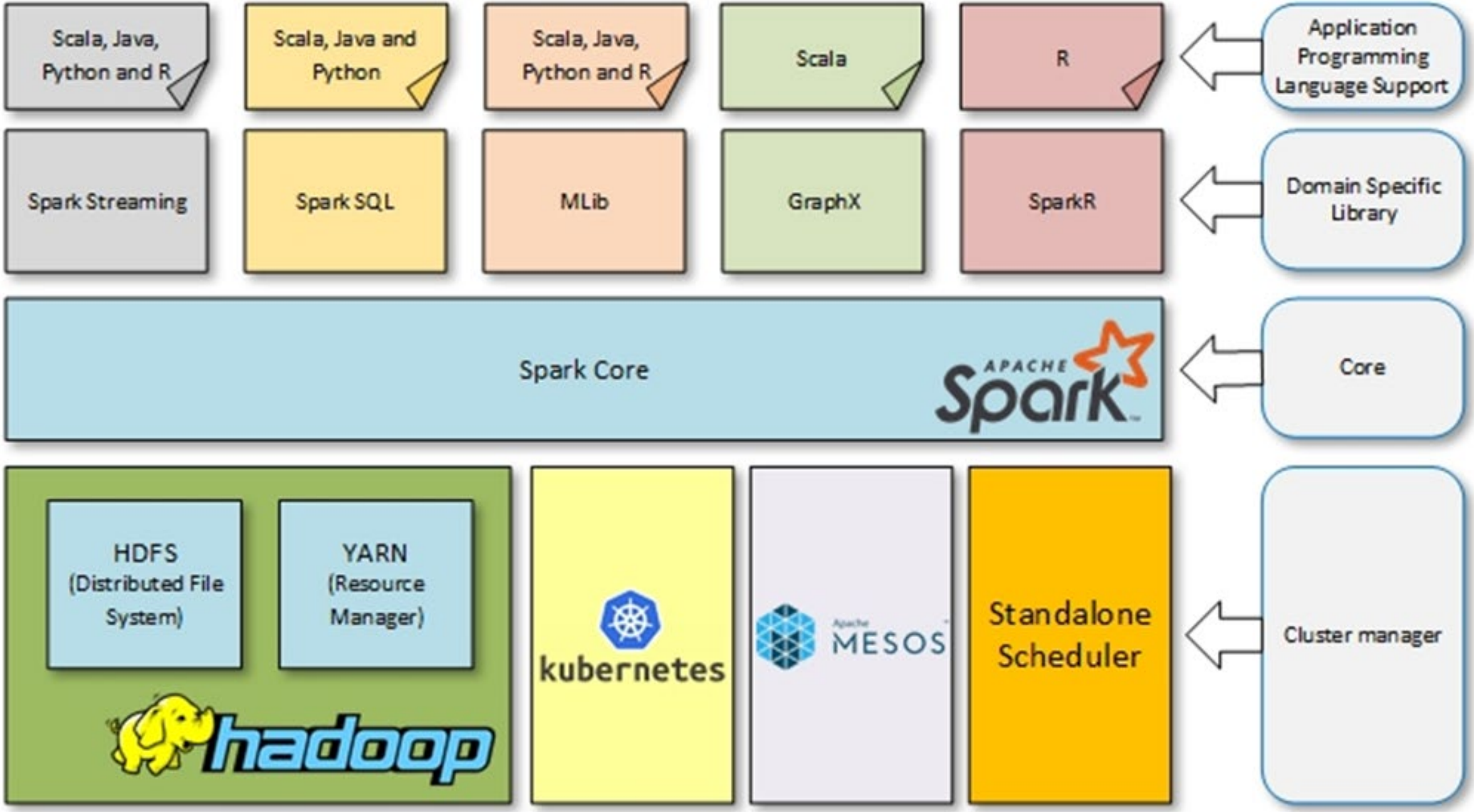


MapReduce

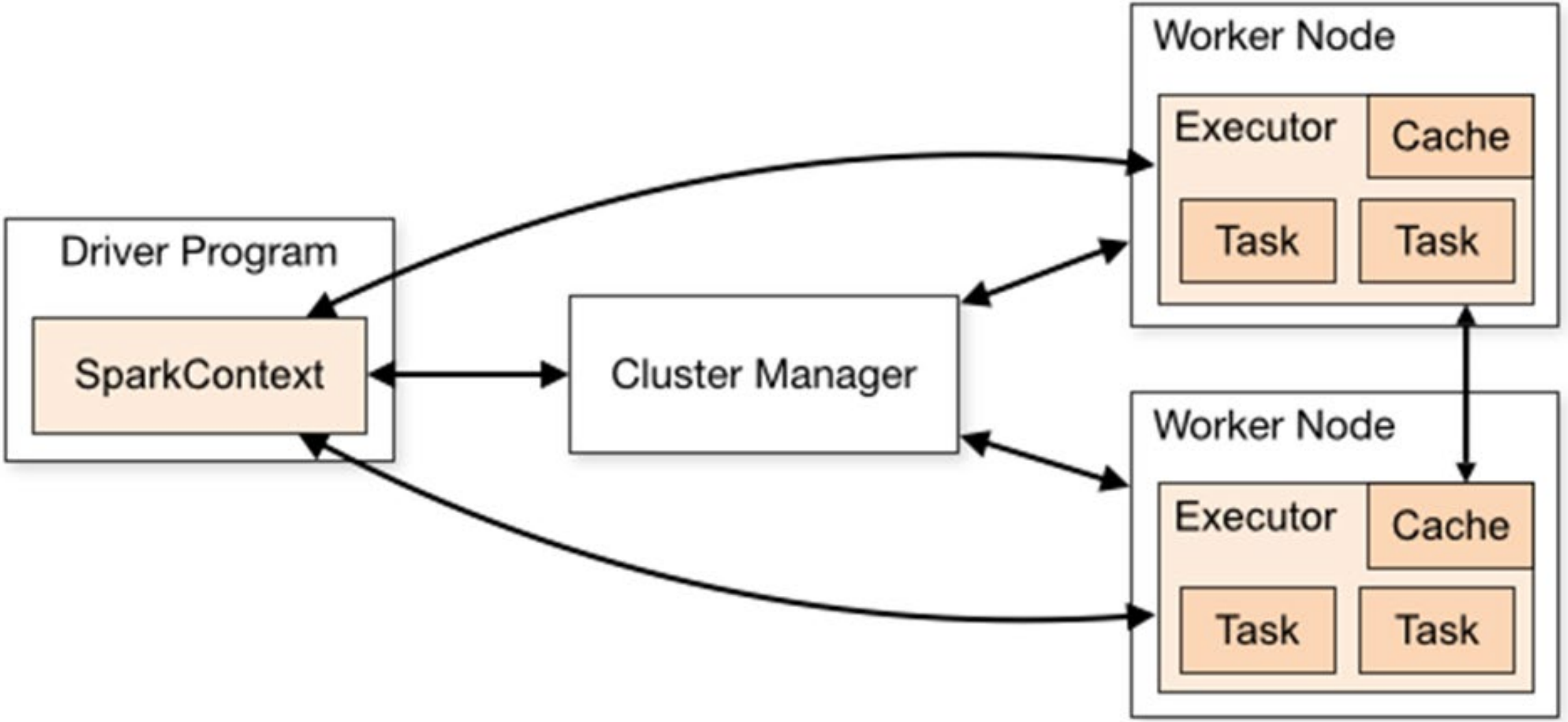


DAG

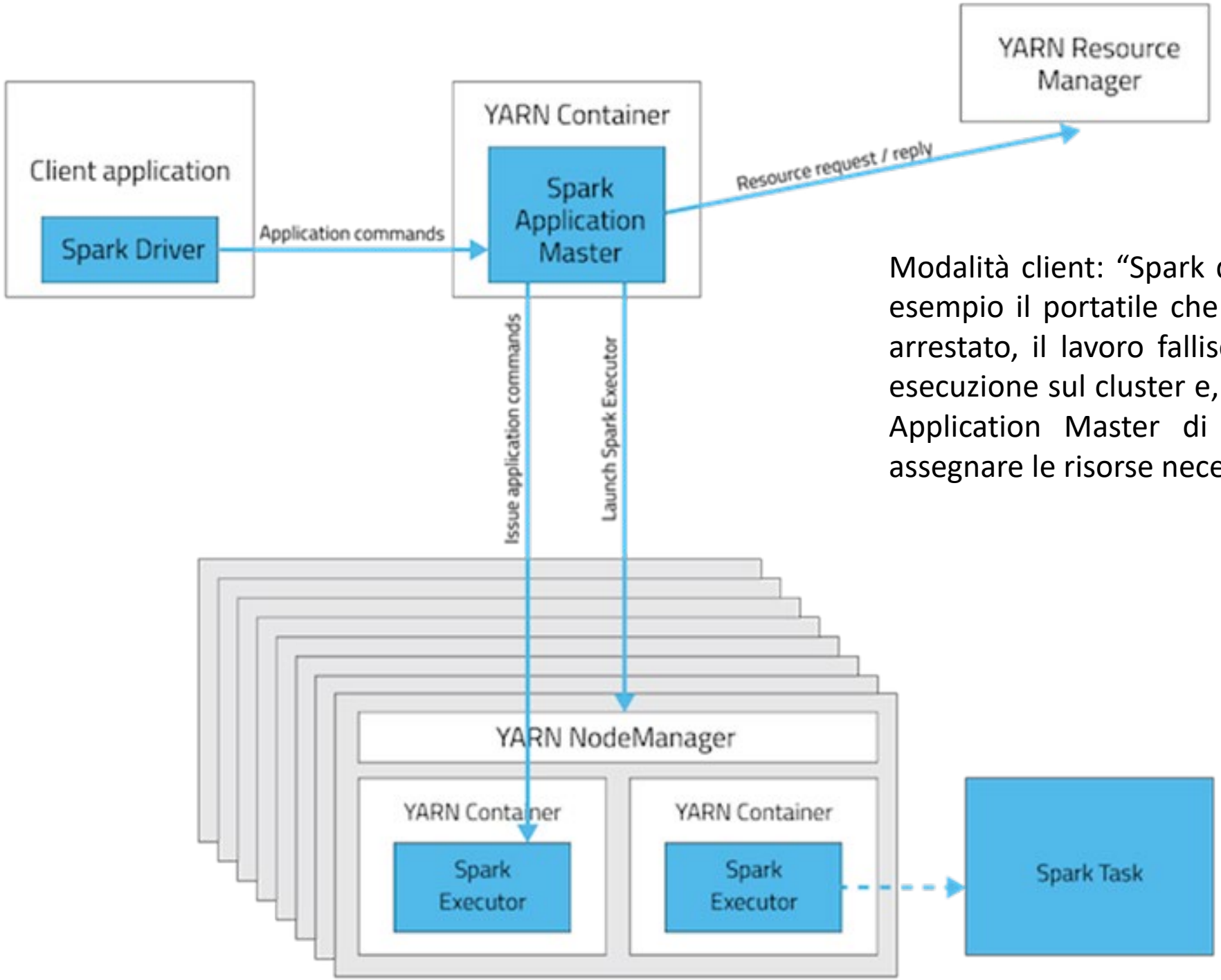
Spark Stack


Scala, Java, Python and R
Scala, Java and Python
Scala, Java, Python and R
Scala
R
Application Programming Language Support
Spark Streaming
Spark SQL
MLib
GraphX
SparkR
Domain Specific Library
Spark Core
APACHE Spark
Core
HDFS (Distributed File System)
YARN (Resource Manager)
hadoop
kubernetes
MESOS
Standalone Scheduler
Cluster manager

Spark Architettura

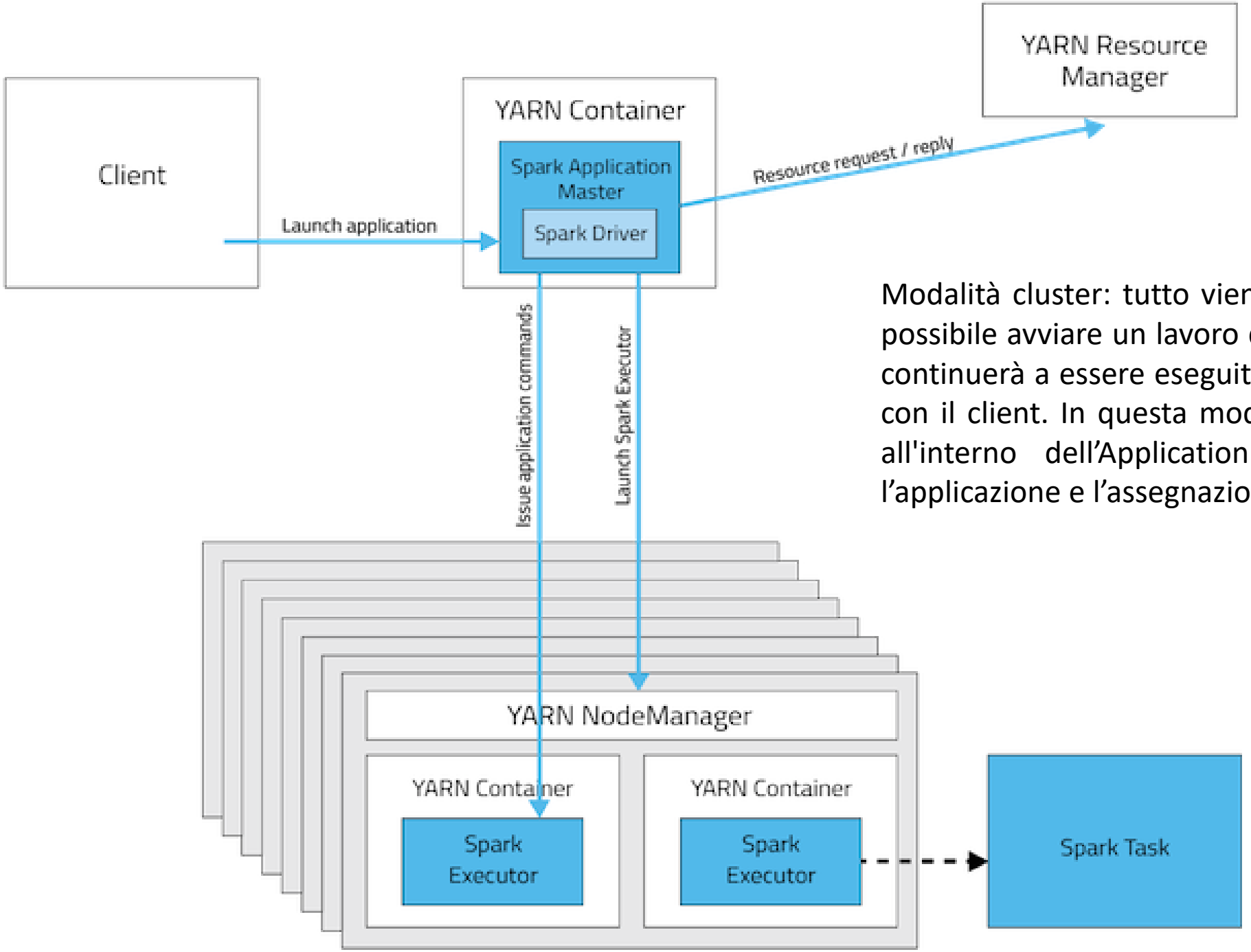


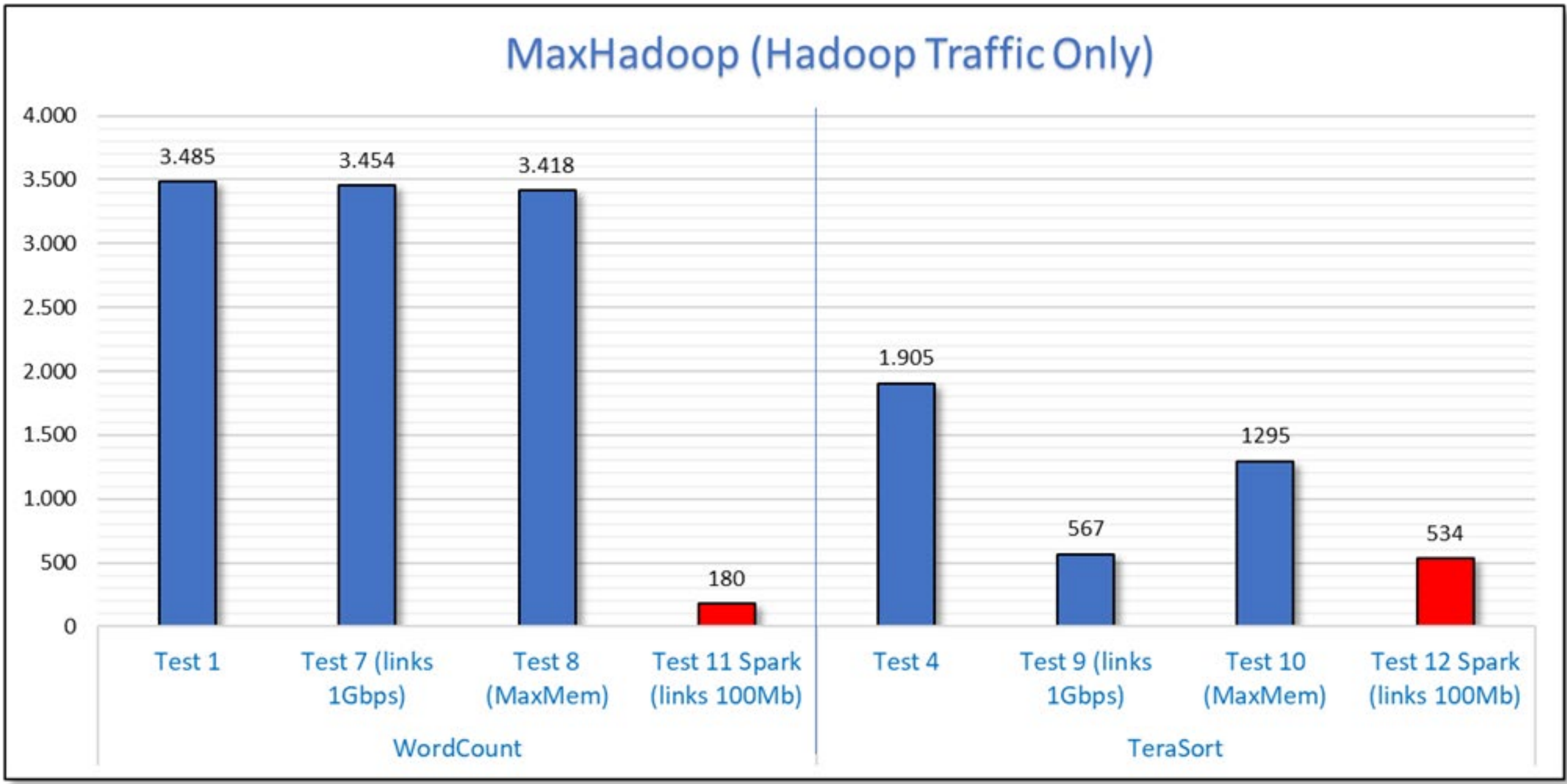
Spark su Yarn



Modalità client: “Spark driver” viene eseguito su un client, ad esempio il portatile che effettua la richiesta. Se il client viene arrestato, il lavoro fallisce. Gli Spark Executor sono ancora in esecuzione sul cluster e, per pianificare tutto, è stato creato un Application Master di YARN in versione ridotta solo per assegnare le risorse necessarie all’esecuzione del job.

Spark su Yarn



Modalità cluster: tutto viene eseguito all'interno del cluster. È possibile avviare un lavoro dal client esterno al cluster ed il job continuerà a essere eseguito anche se si chiude la connessione con il client. In questa modalità, lo Spark Driver è incapsulato all'interno dell'Application Master di YARN che gestisce l'applicazione e l'assegnazione di risorse.

Confronto con Spark