



APACHE SPARK: QUICK INTRO & DEMO

Novembre 2016

ANDREA BECCARIS



ABOUT ME

ANDREA BECCARIS

**FULL-STACK DEVELOPER @ TOP-IX
NETWORK & SOFTWARE ENGINEER
#CODENINJA AND #SPORTADDICTED**

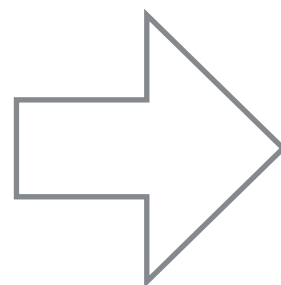
andrea.beccaris@top-ix.org

<https://it.linkedin.com/in/andreabeccaris>

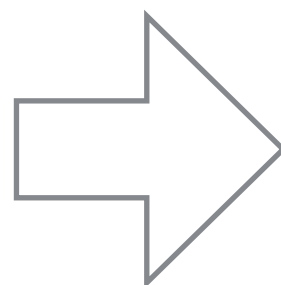




DALLO SCRIPTING AI FRAMEWORK



PYTHON + MRJOB





APACHE SPARK IN A NUTSHELL

Apache **Spark** è un framework open-source per l'analisi di grandi quantità di dati su cluster, nato per essere **veloce** e **flessibile**.

/ **Veloce** nel configurare un **CLUSTER** (riduce i tempi di kickstart)

/ **Veloce** nello scrivere applicazioni (supporta diversi linguaggi: Python, Scala, Java,...)

/ **Veloce** nell'eseguire applicazioni (nasce con questo focus)



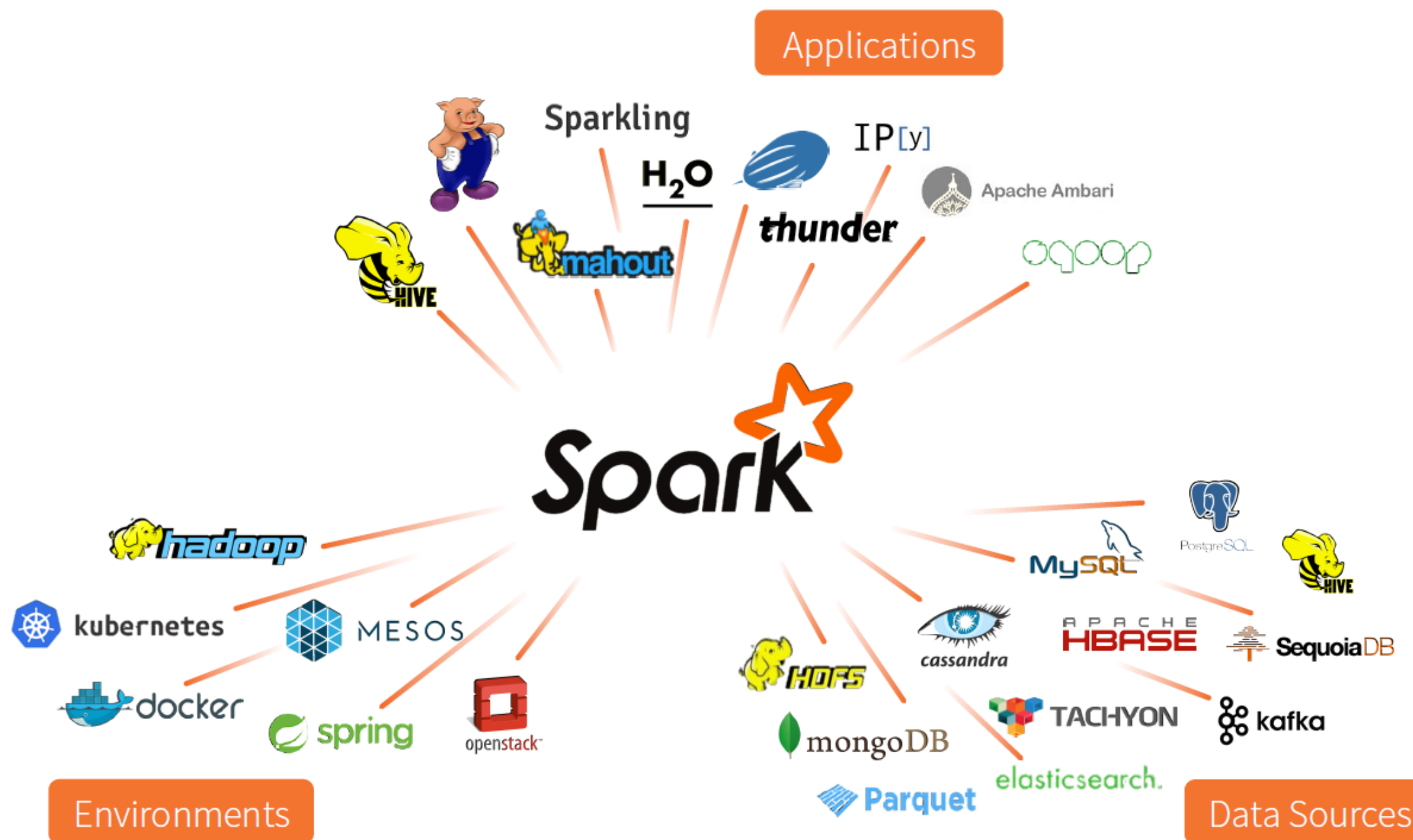
APACHE SPARK

**IL PROGETTO DI DATA PROCESSING
OPEN-SOURCE PIÙ ATTIVO**

**COMMUNITY DI SVILUPPATORI
IN CRESCITA**

	2014	2015
Contributors	255	730
Push/month	75	135
Lines of Code	175,000	400,000

APACHE SPARK ECOSYSTEM





APACHE SPARK

APACHE HADOOP “VS” APACHE SPARK

MAPREDUCE (MRJOB) “VS” APACHE SPARK



APACHE SPARK: HADOOP vs SPARK

I due framework di casa Apache vengono spesso descritti come mutualmente esclusivi, **ma così non è.**

Spark è costruito a partire da Hadoop con il dichiarato obiettivo di migliorarne le prestazioni nel campo MapReduce.



APACHE SPARK

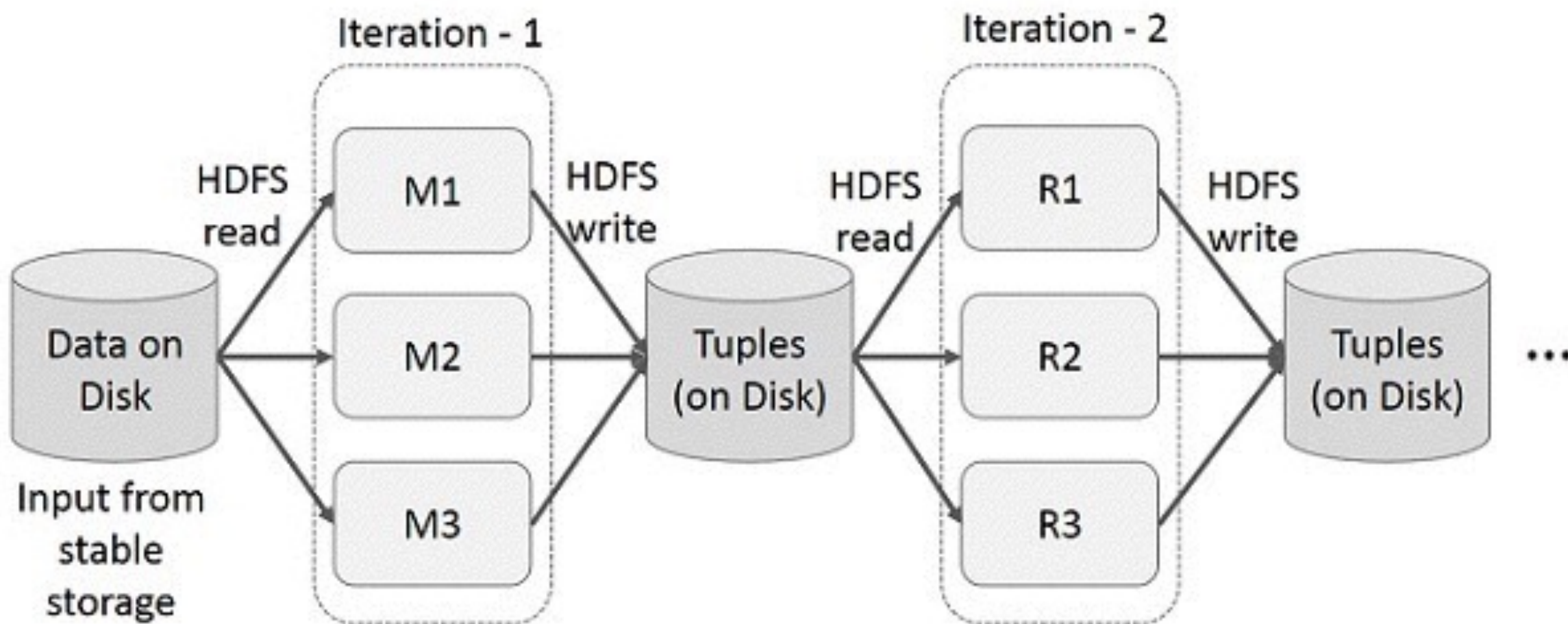
SPARK è ottimizzato per:

/ Minimizzare le operazioni di lettura e scrittura su disco

/ Migliorare le prestazioni di azioni ripetute, iterative, cicli

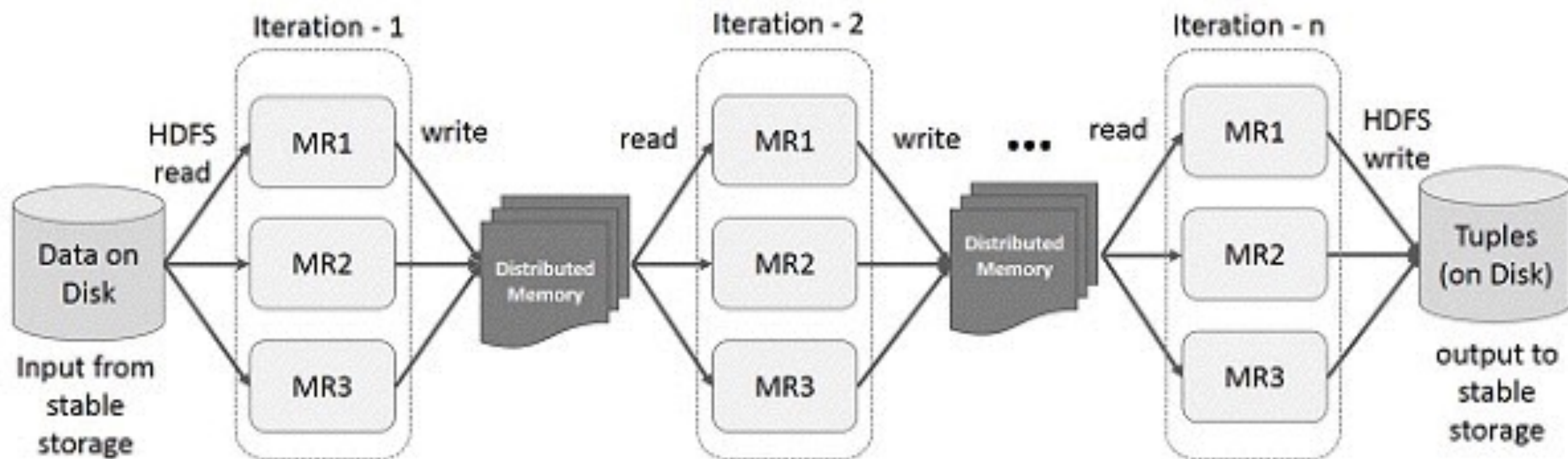


COME FUNZIONA MAPREDUCE STANDARD





COME FUNZIONA APACHE SPARK





APACHE SPARK: BENCHMARK

	Hadoop MR	Spark	Spark 1PB
Data size	102.5TB	100TB	1PB
Nodes	2100	206	190
Cores	50400 (phy)	6592 (virt)	6080 (virt)
Time	72 mins	23 mins	234 mins
Rate	1.42TB/min	4.27TB/min	4.27TB/min
Rate (node)	0.67GB/min	20.7GB/min	22.5GB/min



APACHE SPARK: COMPONENTI

Spark
SQL

Spark
Streaming

MLlib
(machine
learning)

GraphX
(graph)

Apache Spark



APACHE SPARK: COMPONENTI

Spark
SQL

Spark
Streaming

MLlib
(machine
learning)

GraphX
(graph)

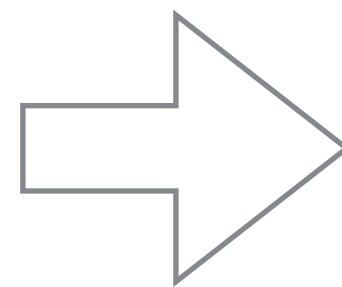
Apache Spark



APACHE SPARK CORE

Il CORE di SPARK si occupa di:

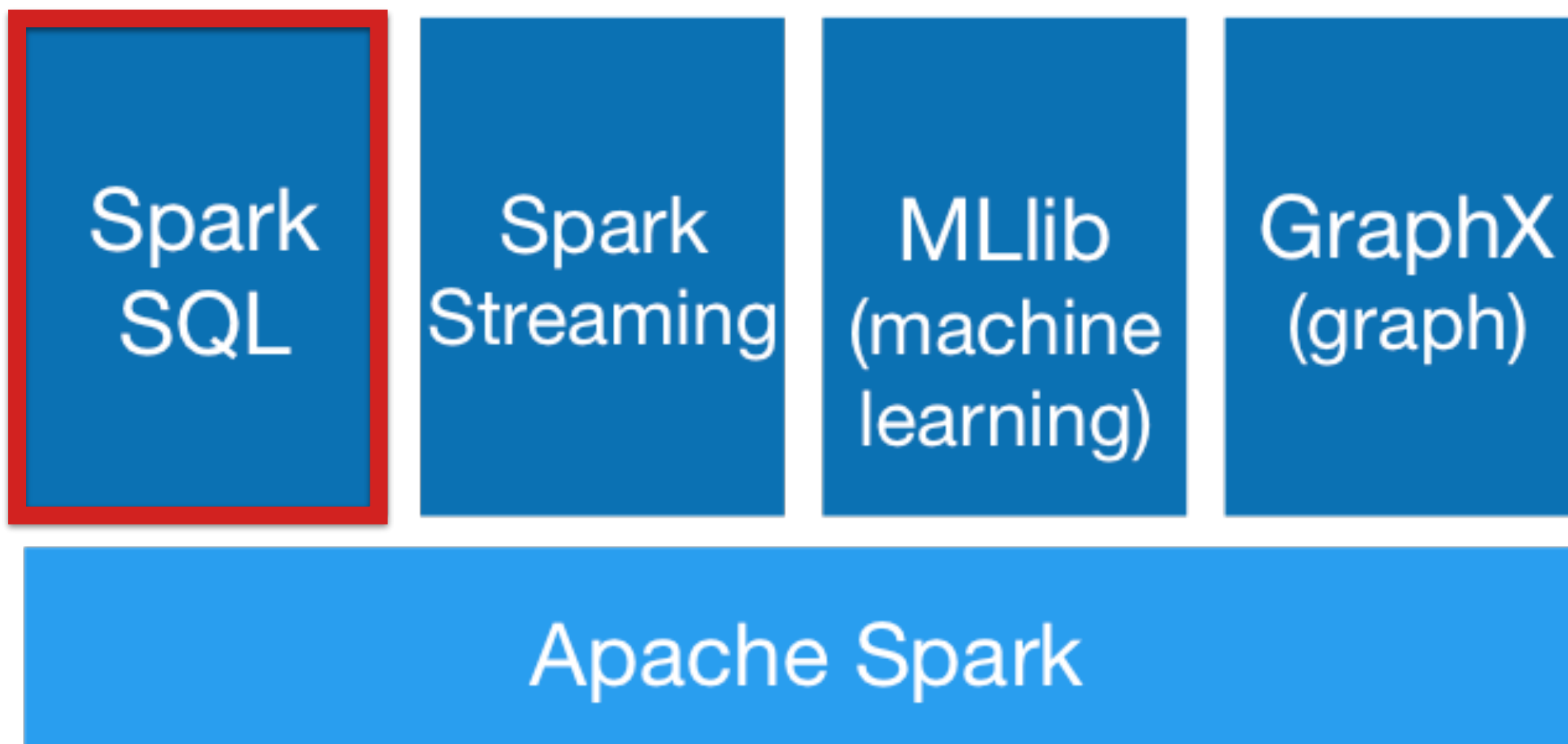
- / gestione della memoria**
- / gestione della rete**
- / gestione dello scheduling**
- / failure recovery**



GESTIONE RDD
(RESILIENT
DISTRIBUTED
DATASET)



APACHE SPARK: COMPONENTI





APACHE SPARK SQL

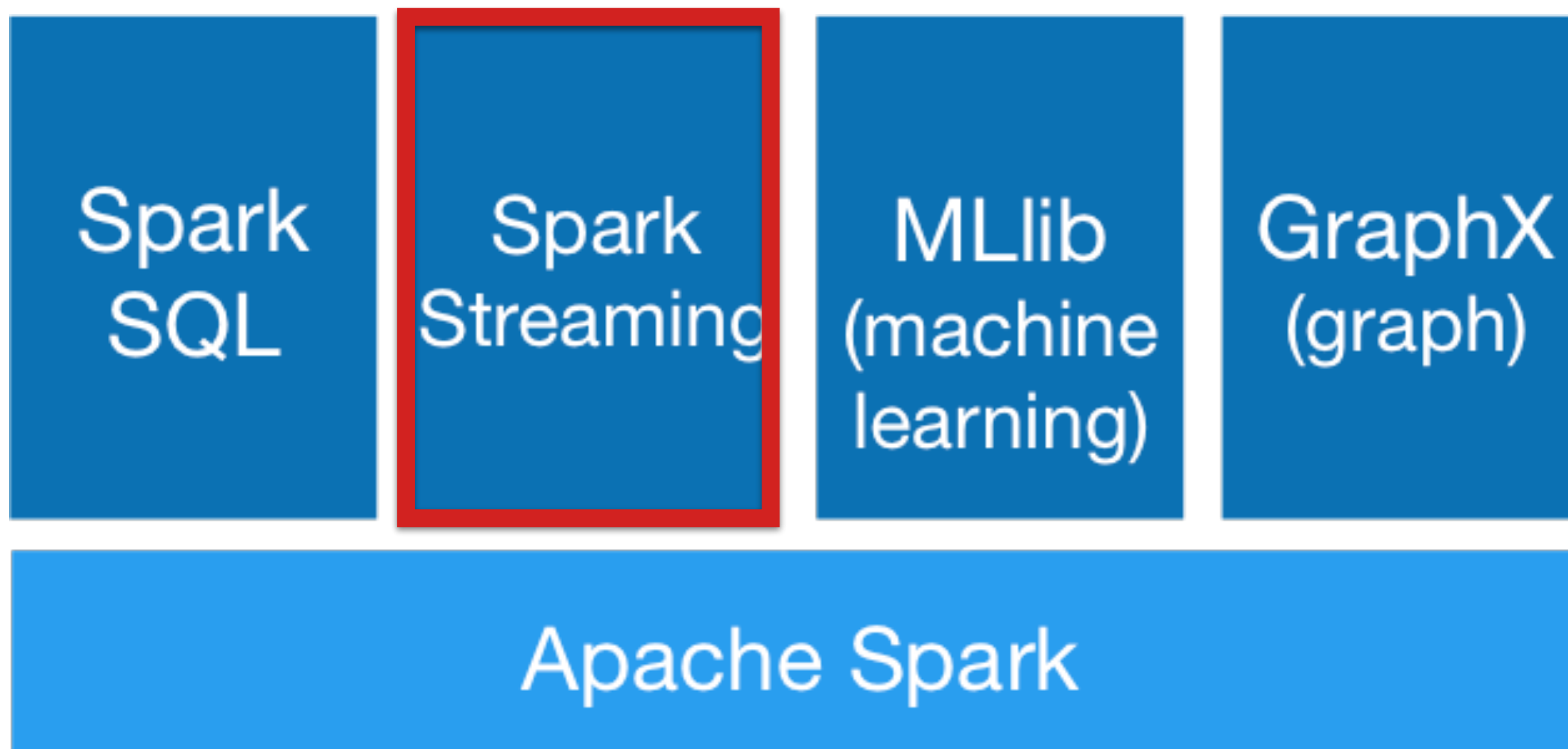
Supporta dati **strutturati e **non strutturati****

Permette di **eseguire vere e proprie **query** come su un classico datawarehouse**

Implementa un livello di astrazione per i dataset chiamato **schemaRDD**



APACHE SPARK: COMPONENTI





APACHE SPARK: SPARK STREAMING

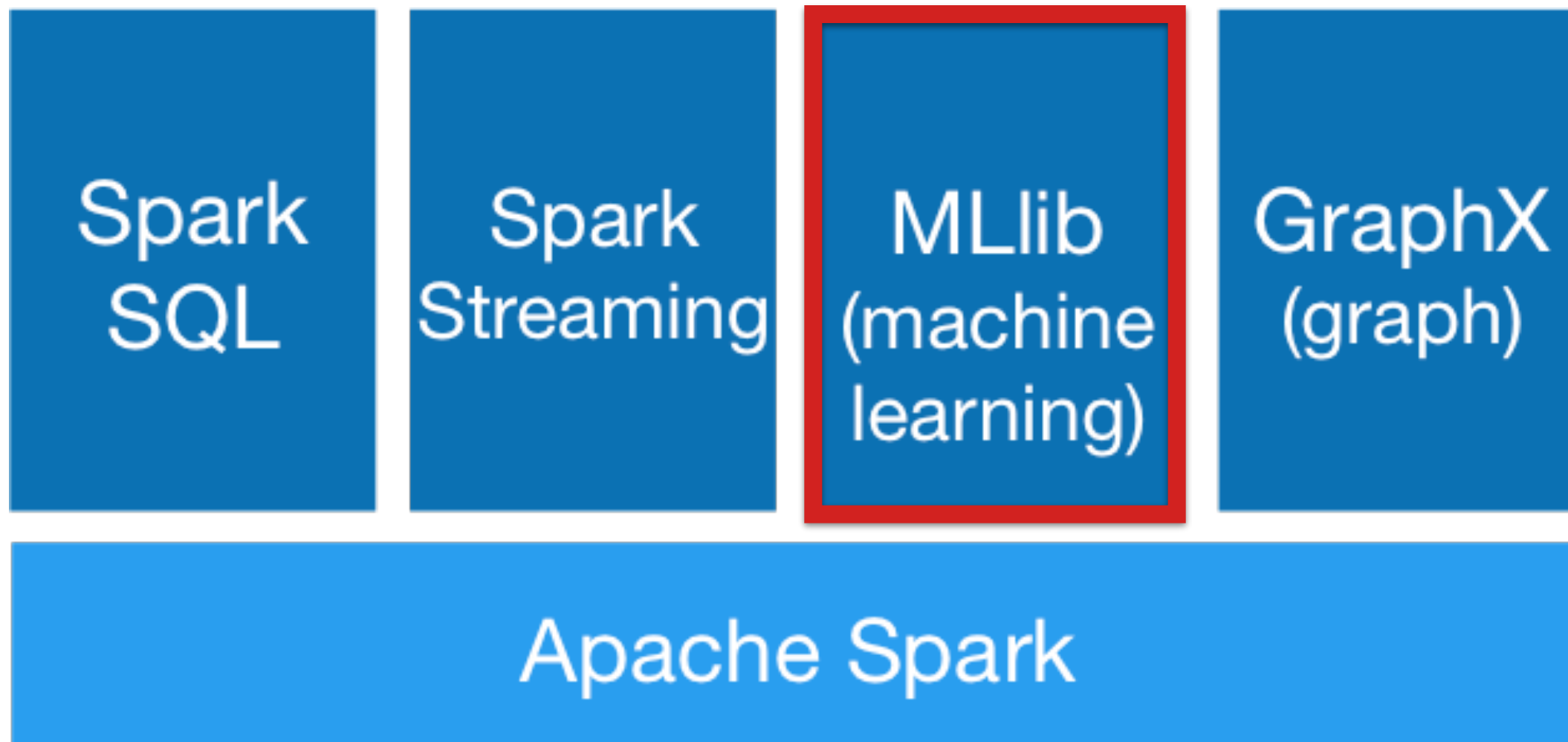
È possibile utilizzare spark per
l'analisi in **tempo reale** di dati o
flussi di dati.
(es. file di log)

The Spark Streaming logo is enclosed in a rounded orange border. It features the word 'Spark' in a bold, black, sans-serif font, with an orange star icon positioned above the 'k'. Below 'Spark', the word 'Streaming' is written in a black, italicized, sans-serif font.

Spark
Streaming



APACHE SPARK: COMPONENTI





APACHE SPARK MLlib

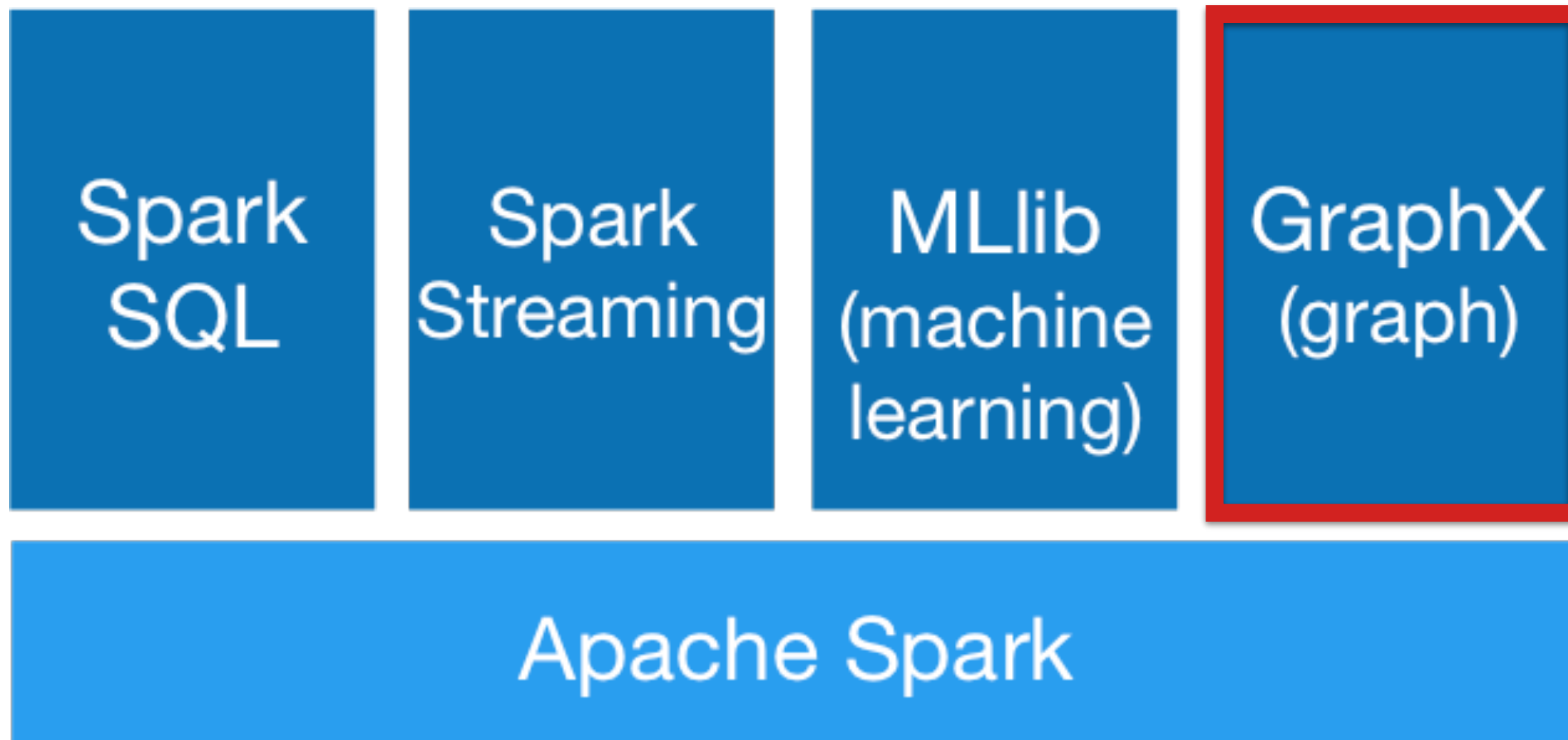
Include nativamente una libreria **altamente ottimizzata dedicata al machine learning**

9x (performance) rispetto all'analogia in Hadoop





APACHE SPARK: COMPONENTI





APACHE SPARK GraphX

Offre **NATIVAMENTE** la
possibilità di utilizzare
algoritmi tipici dell'analisi
dei grafi come il **pagerank**.



GraphX



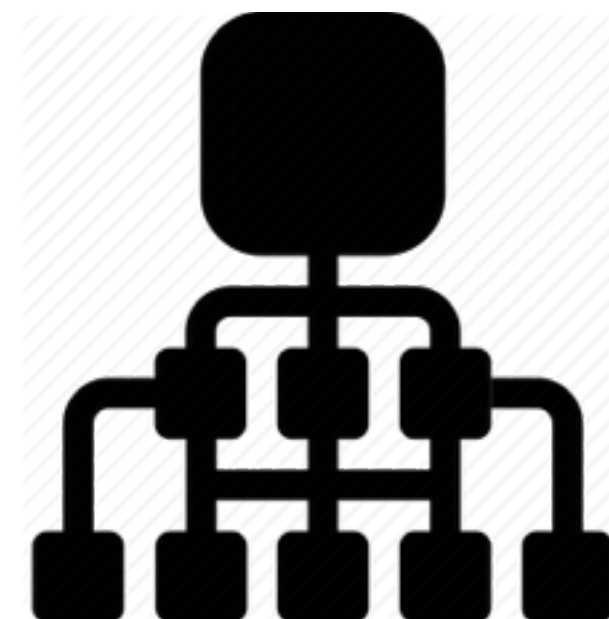
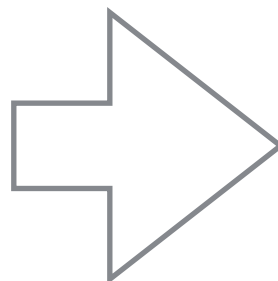
APACHE SPARK

ARCHITETTURA DI APACHE SPARK



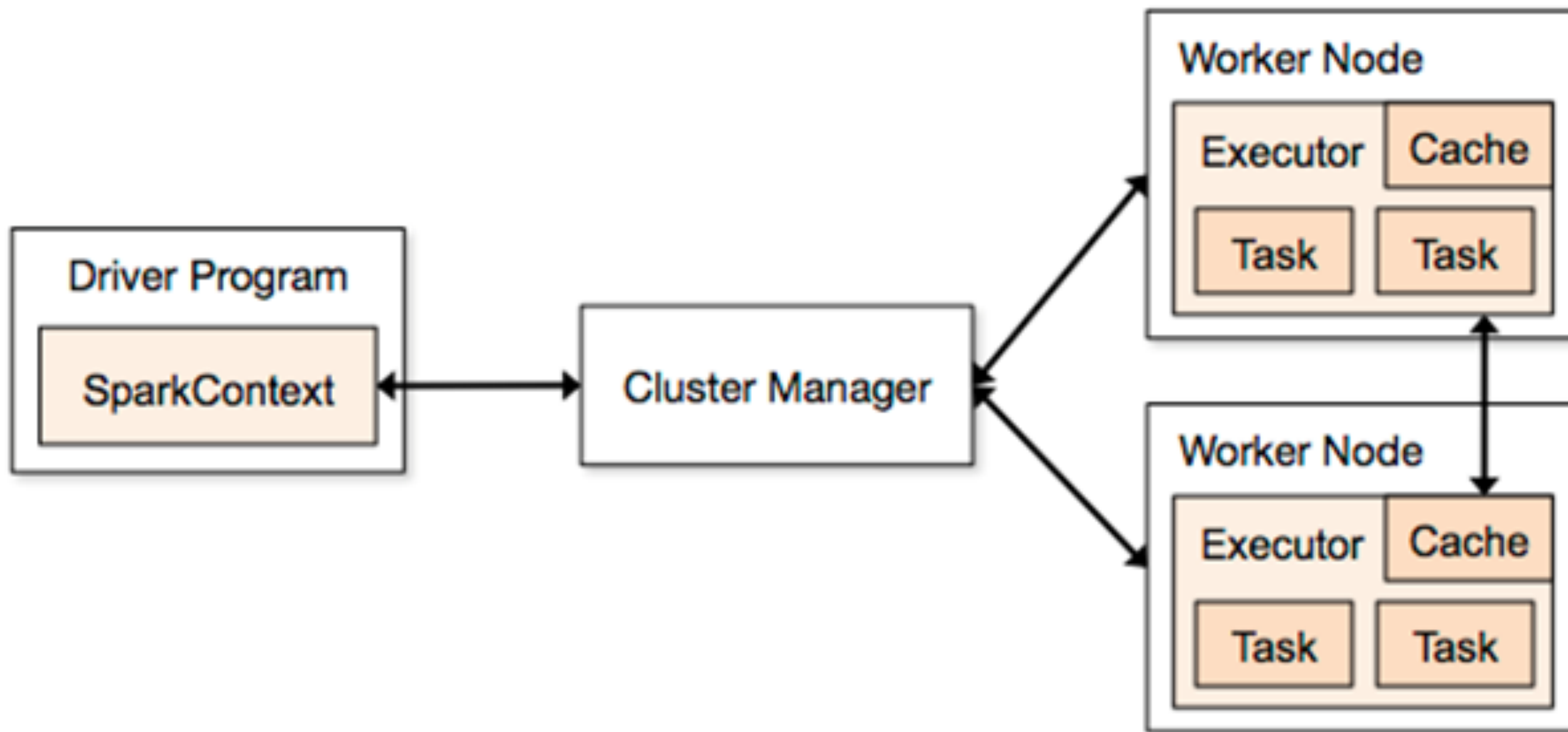
APACHE SPARK: ARCHITETTURA

Pensato perchè il codice scritto in locale funzioni immediatamente su un cluster



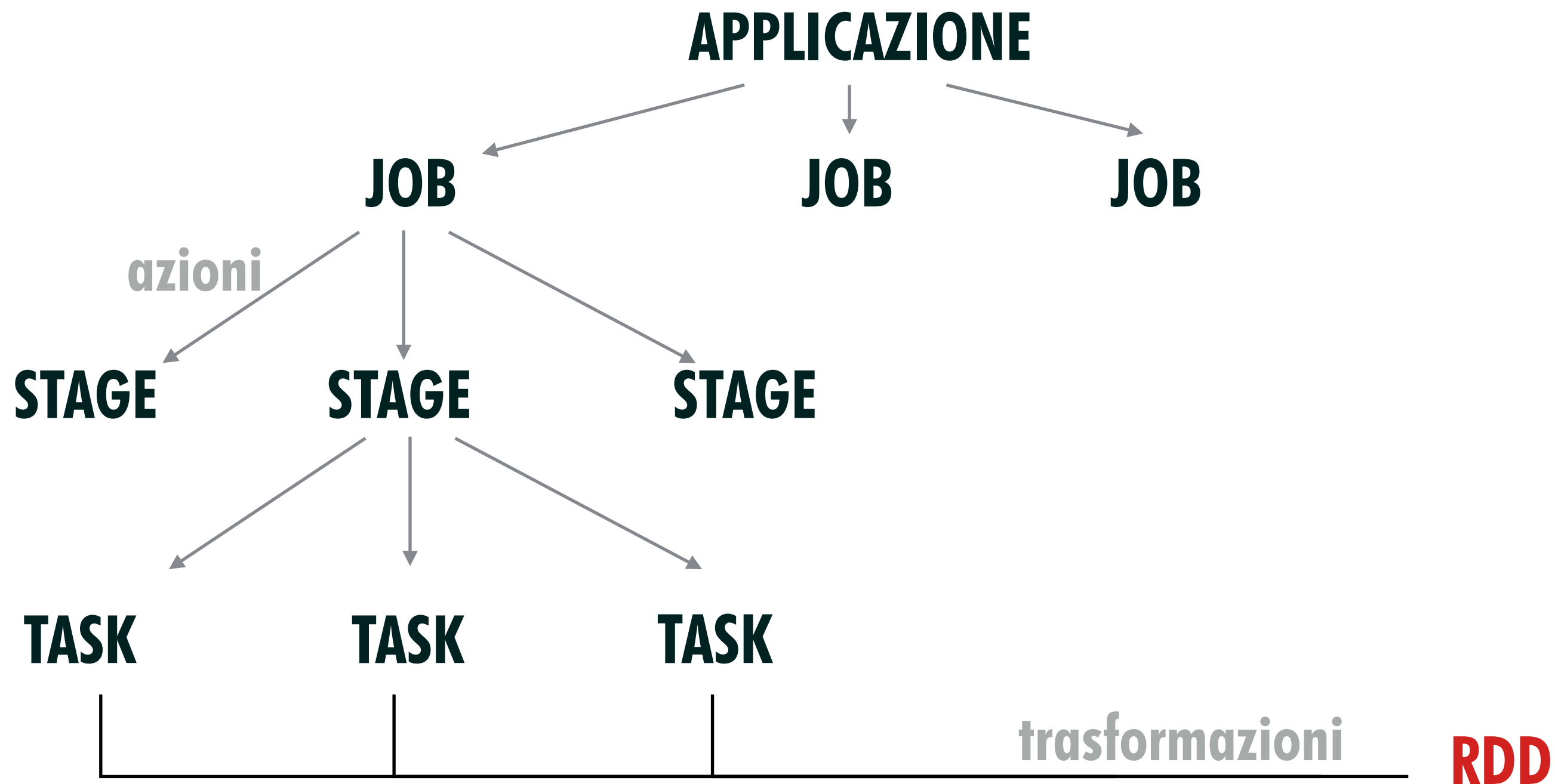


APACHE SPARK: ARCHITETTURA





APACHE SPARK: ARCHITETTURA





APACHE SPARK

RDD

RESILIENT DISTRIBUTED DATASET



APACHE SPARK: RDD

RESILIENT > > **FAILURE RECOVERY**

DISTRIBUTED > > **PARTIZIONATO SU DIVERSI NODI**

DATASET > > **COLLEZIONE DI ELEMENTI**



APACHE SPARK: RDD

RESILIENT >> FAILURE RECOVERY



**Per poter ricostruire il dato ho
bisogno di uno "storico"**



APACHE SPARK: RDD

RESILIENT >> FAILURE RECOVERY



**Per poter ricostruire il dato ho
bisogno di uno "storico"**



DAG



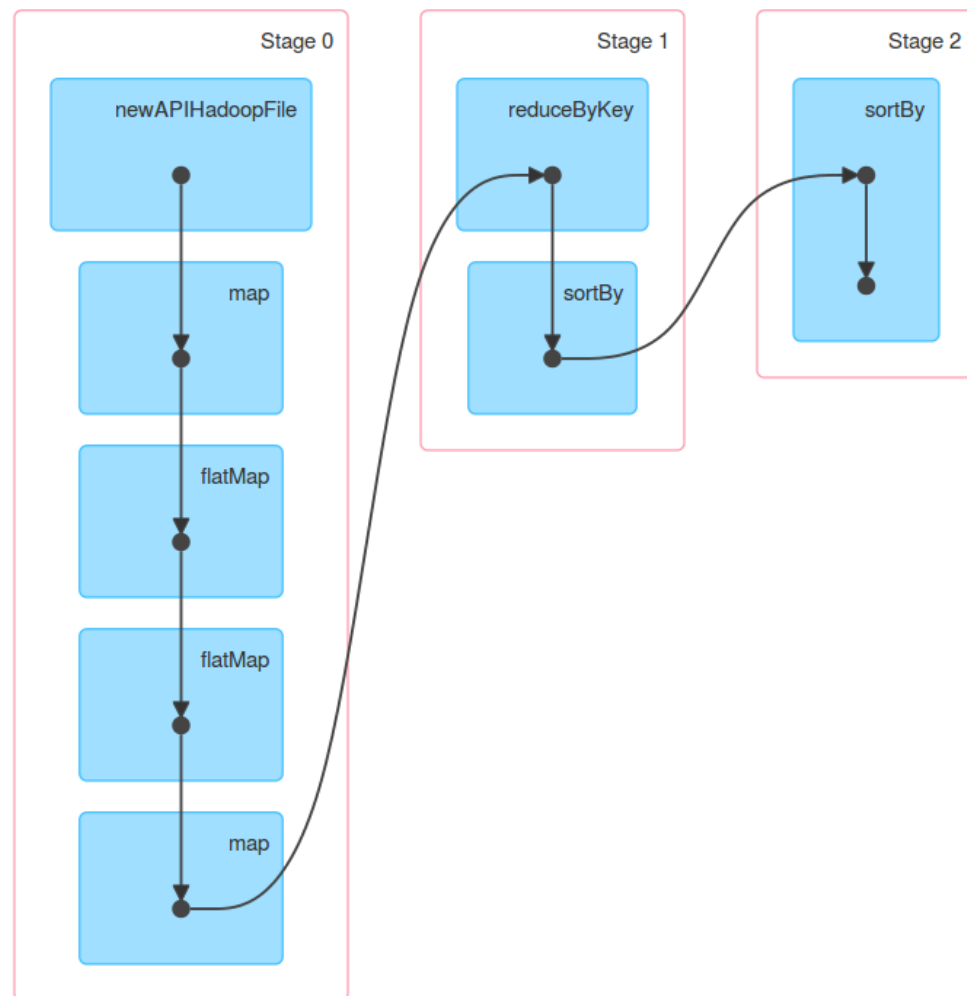
APACHE SPARK

DAG



APACHE SPARK: DAG

DIRECT ACYCLING GRAPH

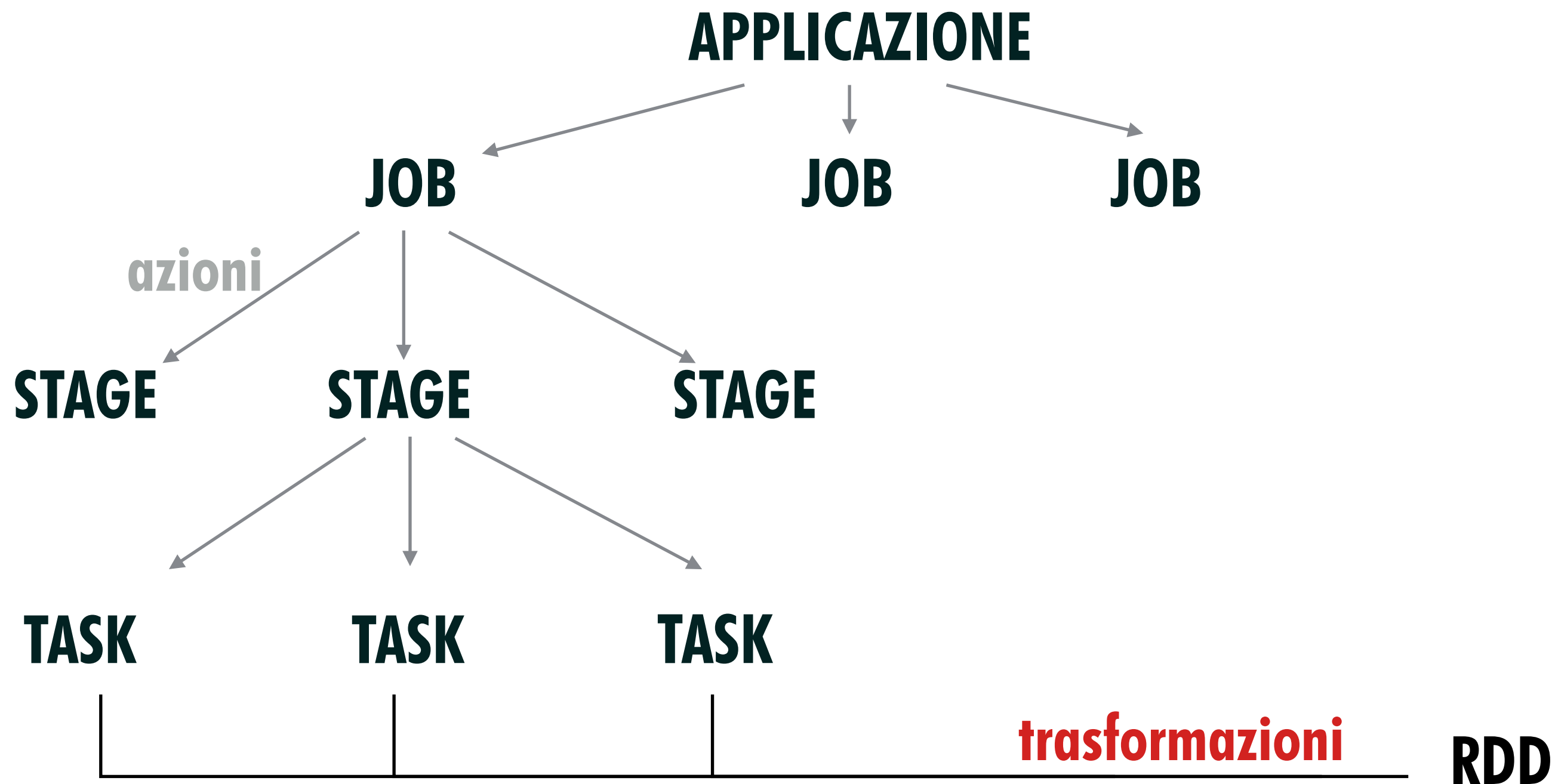


il **DAG** è un grafo in cui i nodi sono gli RDD/trasformazioni e le frecce sono le dipendenze

ogni **job** ha il suo dag!



APACHE SPARK: ARCHITETTURA





APACHE SPARK

TRASFORMAZIONI

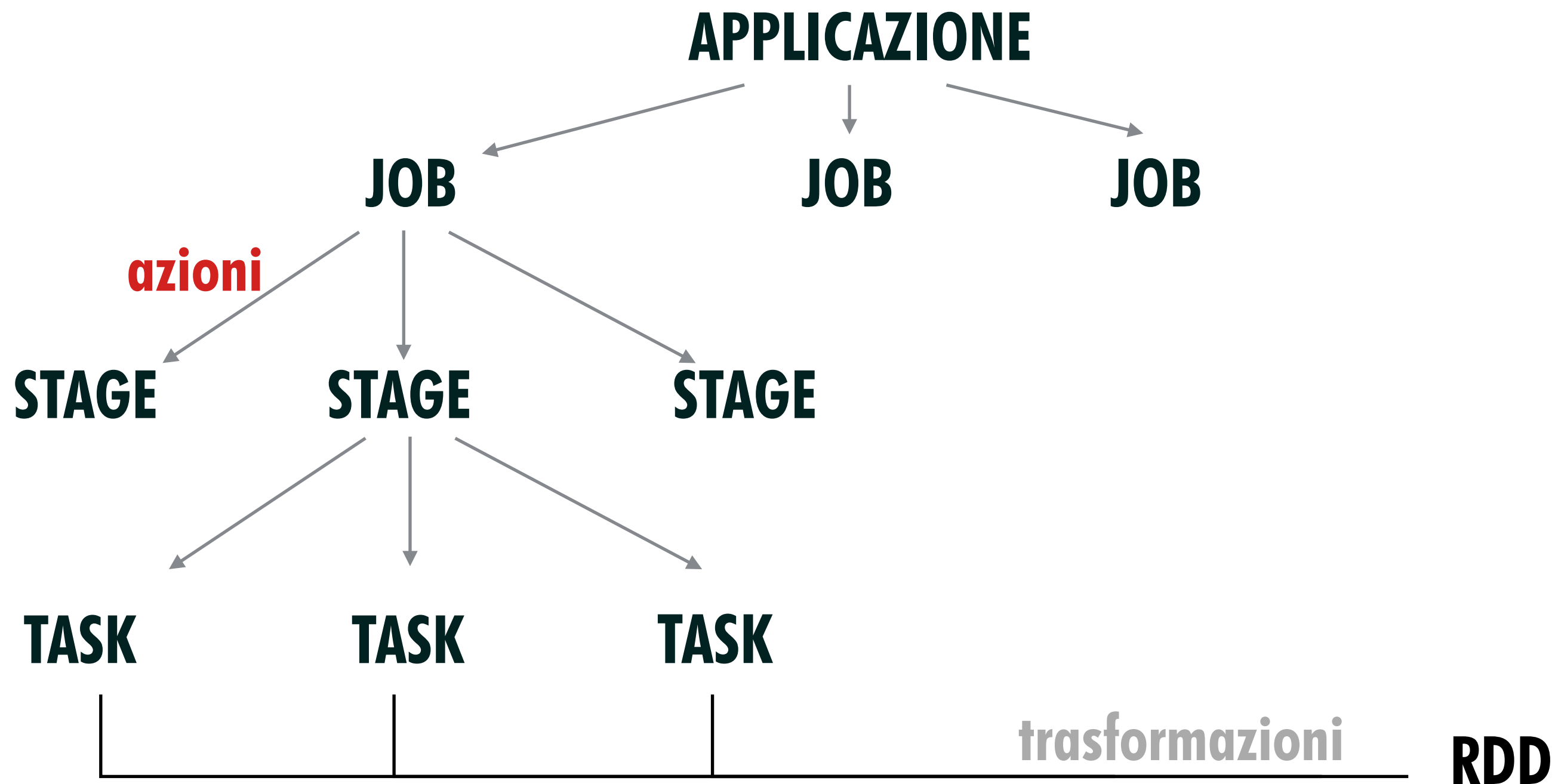


APACHE SPARK: TRASFORMAZIONI

- / Operano su RDD “modificandone” il contenuto**
- / Non vengono eseguite fino alla richiesta di output (**LAZY**)**
- / Gestione dello scheduling**
- / Possono essere invocate in cascata**
- / Ad ogni trasformazione viene aggiunto un nodo al DAG**



APACHE SPARK: ARCHITETTURA





APACHE SPARK

AZIONI



APACHE SPARK: AZIONI

- / Materializzano l'RDD scatenando risultati immediati**
- / Vengono usate per operazioni di OUTPUT (e.g. scrivere il dato su disco)**
- / Innescano l'esecuzione dell'intero DAG**



APACHE SPARK

DUBBI

???



APACHE SPARK

DEMO!!!



Contatti

WWW.TOP-IX.ORG

twitter: @A_Becca88

**email: andrea.beccaris@top-
ix.org**