



Apache Spark

Χρήστος Γκόγκος

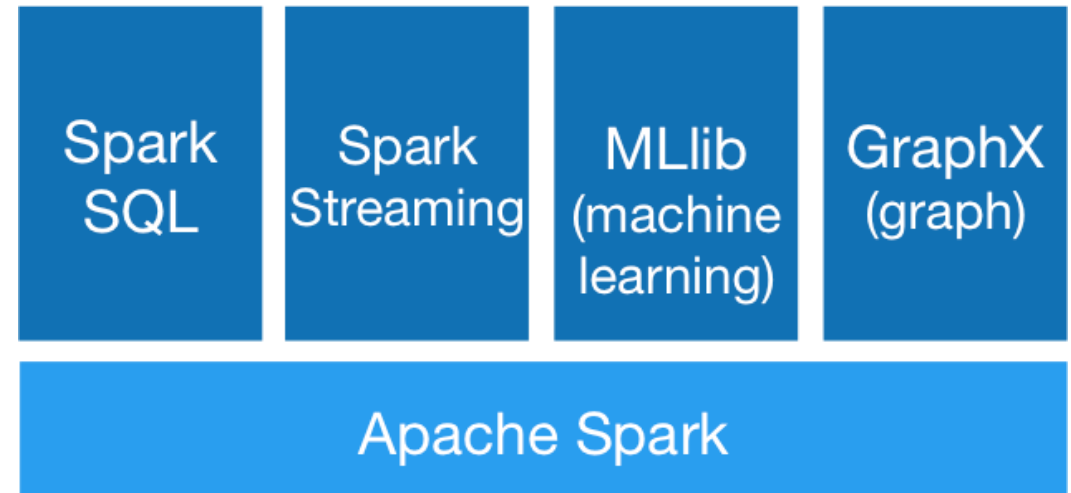
18/2/2019

https://github.com/chgogos/big_data

Apache Spark Stack



- **Spark SQL:** πρόσβαση σε δομημένα δεδομένα – συμβατότητα με HiveQL
- **Spark Streaming:** fault tolerant χειρισμός data streams (Flume, Kafka,...)
- **MLlib:** έλεγχος υποθέσεων, κατηγοριοποίηση, παλινδρόμηση, συσταδοποίηση, ανάλυση κύριων συνιστωσών
- **GraphX:** ανάλυση γραφημάτων (π.χ. pagerank), μέσω του Pregel API



Τι είναι το Apache Spark;

Το Apache Spark είναι ένα framework γενικού σκοπού που επιτρέπει κατανεμημένη επεξεργασία σε ομάδες υπολογιστών

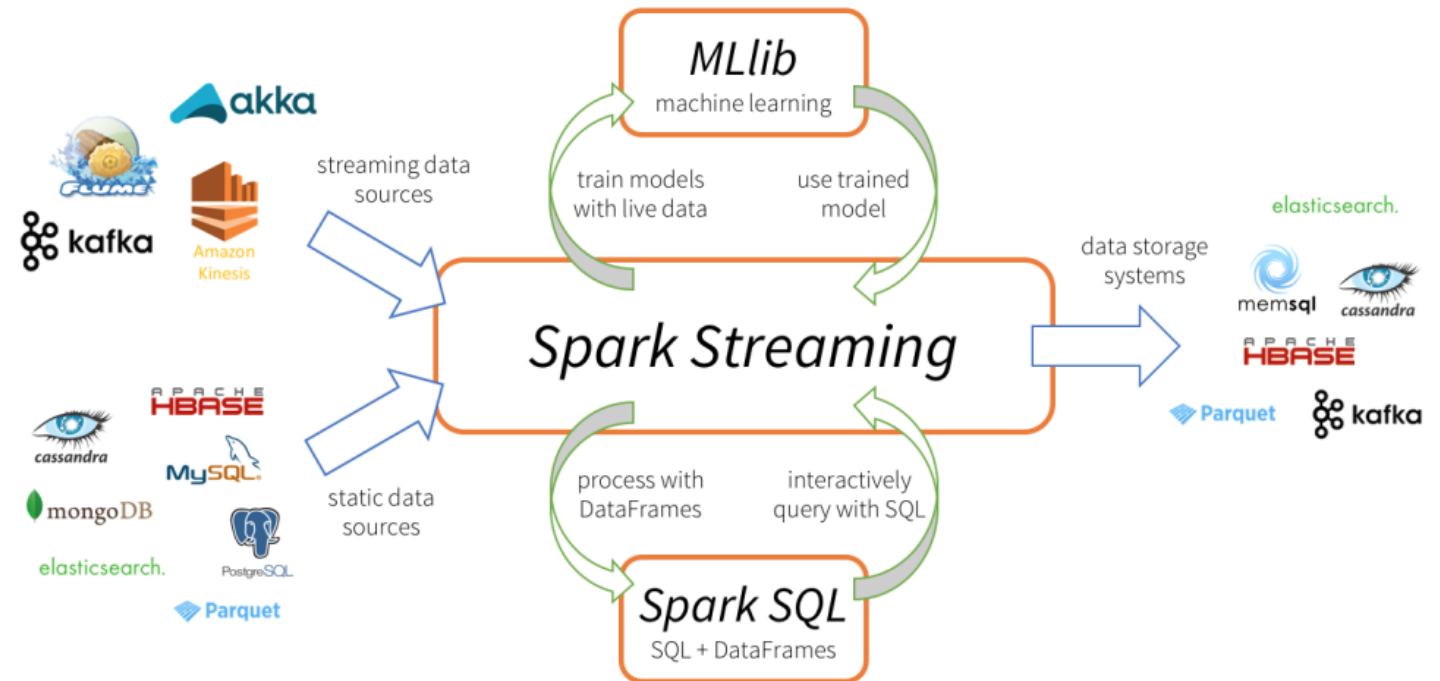
Μπορεί να θεωρηθεί ως μηχανή επεξεργασίας (processing engine) που δίνει έμφαση σε: ταχύτητα, ευκολία χρήσης και σε προχωρημένη ανάλυση δεδομένων

Το Spark δημιουργήθηκε αρχικά στο AMPLab του UC Berkeley (2009) και από το 2010 είναι open source ως Apache project

Δίνει έμφαση στη διατήρηση δεδομένων στη μνήμη (100x ταχύτερο από το Hadoop MapReduce για συγκεκριμένες εργασίες)

Για ποιες εφαρμογές είναι κατάλληλο το Apache Spark;

- Το Spark είναι κατάλληλο για:
- Διαδραστικά ερωτήματα σε μεγάλα δεδομένα
- Επεξεργασία streaming μεγάλων δεδομένων από αισθητήρες ή από άλλες πηγές
- Εκτέλεση εργασιών μηχανικής μάθησης σε μεγάλα δεδομένα



<https://databricks.com/blog/2016/06/22/apache-spark-key-terms-explained.html>

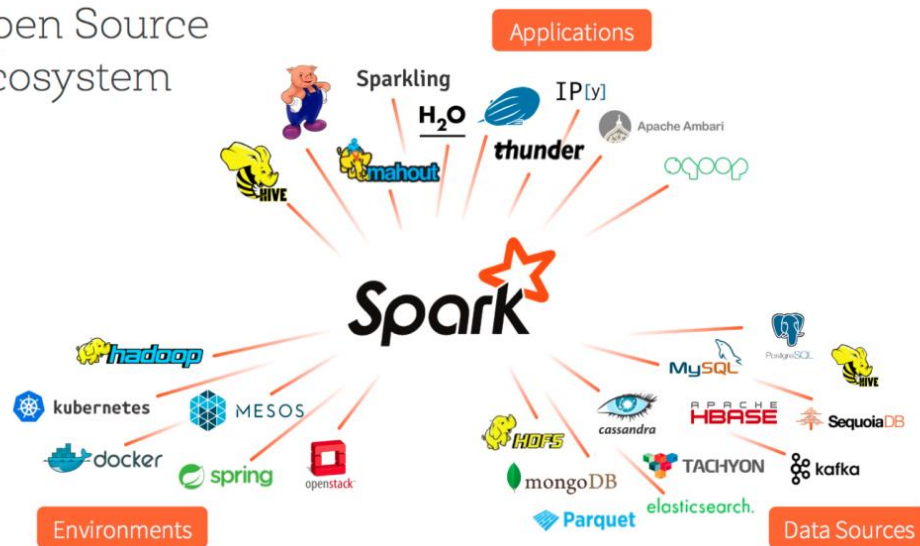
Χαρακτηριστικά του Spark

- Διατηρεί τα δεδομένα και τα ενδιάμεσα αποτελέσματα στη μνήμη, αντί να τα γράφει στο δίσκο
 - Παρέχει επεξεργασία «σχεδόν» πραγματικού χρόνου
 - Σε σχέση με το Hadoop MapReduce το Spark:
 - πραγματοποιεί λιγότερο ακριβά ανακατέματα (shuffles) κατά την επεξεργασία των δεδομένων
 - Παρέχει υψηλότερου επιπέδου API που διευκολύνει τους προγραμματιστές
- Έχει σχεδιαστεί ως **μηχανισμός εκτέλεσης εφαρμογών τόσο στη μνήμη όσο και στο δίσκο** (όταν η μνήμη δεν επαρκεί, οι λειτουργίες του Spark χρησιμοποιούν τους δίσκους)

Δεδομένα που χειρίζεται το Spark

- Συχνά χρησιμοποιείται πάνω από το Hadoop που του παρέχει πρόσβαση σε δεδομένα τα οποία βρίσκονται στο HDFS ή στην HBase

Open Source
Ecosystem



- Επιπλέον:

- Μπορεί να διαβάσει δεδομένα και από άλλα συστήματα αποθήκευσης δεδομένων όπως Cassandra, MongoDB, CouchDB κ.α.
- Μέσω του υποσυστήματος Apache Spark SQL μπορεί να έχει πρόσβαση μέσω SQL σε σχεσιακές βάσεις δεδομένων
- Μπορεί να χρησιμοποιεί το Apache Mesos ως cluster manager και να εκτελείται εκτός Hadoop σε ομάδες υπολογιστών που τη διαχείρισή τους αναλαμβάνει το Mesos



Resilient Distributed Datasets (RDDs)

- Τα RDDs μοιάζουν με τους πίνακες των Βάσεων Δεδομένων
 - Τα RDDs είναι immutable (ένα RDD μπορεί να τροποποιηθεί μέσω ενός μετασχηματισμού αλλά σε αυτή την περίπτωση επιστρέφεται ένα νέο RDD και το αρχικό RDD παραμένει το ίδιο)
-
- Τα RDDs υποστηρίζουν κατανεμημένη αποθήκευση δεδομένων στις μνήμες των μηχανημάτων ενός cluster έτσι ώστε να επιτυγχάνεται
 - **ανοχή σε σφάλματα:** καταγράφοντας το ιστορικό των μετασχηματισμών που εφαρμόζονται στα δεδομένα
 - **υψηλή απόδοση:** Παραλληλισμός επεξεργασίας στους κόμβους του cluster

RDDs: Transformations - Actions

Από τη στιγμή που έχει δημιουργηθεί ένα RDD δύο βασικοί τύποι λειτουργιών μπορούν να γίνουν:

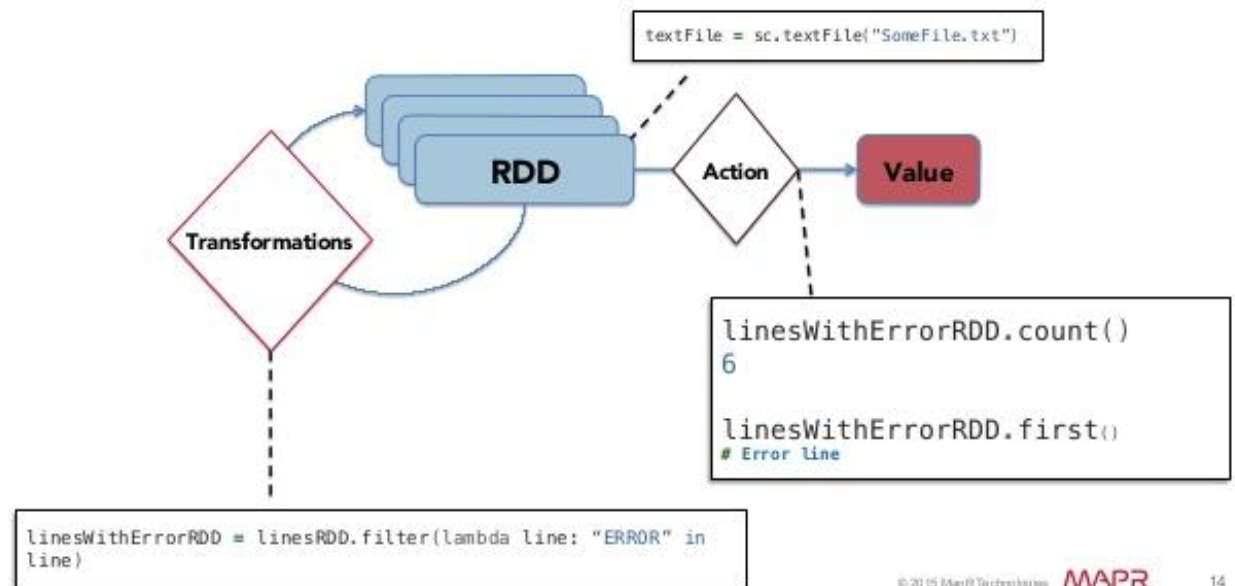
- **Μετασχηματισμοί (transformations)**
 - δημιουργούν ένα νέο RDD αλλάζοντας το αρχικό (π.χ. map, filter, flatMap, groupByKey, reduceByKey, aggregateByKey, pipe, coalesce)
 - οι μετασχηματισμοί δεν επιστρέφουν μια απλή τιμή αλλά ένα νέο RDD (lazy evaluation)
- **Ενέργειες (actions)**
 - υπολογίζουν μια ποσότητα αλλά δεν αλλάζουν τα δεδομένα (π.χ. reduce, collect, count, first, take, countByKey, foreach)
 - όταν καλείται μια ενέργεια σε ένα RDD, τότε εκτελούνται όλοι οι μετασχηματισμοί και επιστρέφεται το αποτέλεσμα

Directed Acyclic Graphs (DAGs)

- Το Spark επιτρέπει την ανάπτυξη σύνθετων εργασιών, που αποτελούνται από πολλά επιμέρους βήματα χρησιμοποιώντας το DAG pattern
- Το Spark διατηρεί τα ενδιάμεσα αποτελέσματα στη μνήμη αντί να τα εγγράφει στο δίσκο (ιδιαίτερα χρήσιμο όταν χρειάζεται να πραγματοποιηθούν εργασίες στα ίδια δεδομένα πολλές φορές)



Working With RDDs



Mlib (scalable machine learning library)

- Το API του MLib βασίζεται στα DataFrames
- Αλγόριθμοι μηχανικής μάθησης
 - Κατηγοριοποίηση
 - Παλινδρόμηση
 - Δένδρα απόφασης
 - Αλγόριθμοι συστάσεων
 - Συσταδοποίηση
 - ...

Streaming

- Τα streaming δεδομένα φθάνουν συνεχώς από διάφορες πηγές με μικρά μηνύματα
 - Υπάρχουν πολλές εφαρμογές της streaming τεχνολογίας (παρακολούθηση αισθητήρων, έλεγχος logs, παρακολούθηση χρηματοοικονομικών αγορών κ.α.)
-
- Λογισμικά ανάλυσης streams
 - Flink
 - Storm
 - Kafka
 - Spark
 - Samza
 - Kinesis
 - ...

Εξέλιξη του μοντέλου επεξεργασίας στο Apache Spark

- RDD
- Spark 1.3 ➔ DataFrame API (χρησιμοποιεί γλώσσα ερωτημάτων για να χειρίζεται τα δεδομένα ταχύτερα σε σχέση με τα RDD)
- Spark 1.6 ➔ DataSet API (δημιουργεί query plans για την εκτέλεση των ερωτημάτων, ταχύτερο σε σχέση με τα RDDs)

Απόδοση του Apache Spark (GraySortMetric, CloudSortMetric)

GraySort 2014	Hadoop MR Record	Spark Record
Data Size	102.5 TB	100 TB
Elapsed Time	72 mins	23 mins
# Nodes	2100	206
# Cores	50400 physical	6592 virtualized
Cluster disk throughput	3150 GB/s (est.)	618 GB/s
Sort Benchmark Daytona Rules	Yes	Yes
Network	dedicated data center, 10Gbps	virtualized (EC2) 10Gbps network
Sort rate	1.42 TB/min	4.27 TB/min
Sort rate/node	0.67 GB/min	20.7 GB/min

New CloudSort Benchmark

Cost to sort 100TB of data



<https://databricks.com/blog/2016/11/14/setting-new-world-record-apache-spark.html>

<https://databricks.com/blog/2014/11/05/spark-officially-sets-a-new-record-in-large-scale-sorting.html>

Demo: παράδειγμα επεξεργασίας με το Apache Spark

- Υπολογισμός πλήθους μοναδικών επισκεπτών ιστοσελίδας
- Εύρεση IP διευθύνσεων από τις οποίες συνδέθηκε ο κάθε μοναδικός χρήστης
- Χρήση αρχείων καταγραφής - weblogs (82.9MB)

Java magazine May/June 2016
Apache Spark 101: Getting up to speed on the popular big data engine

<http://www.oracle.com/technetwork/java/javamagazine>

id πελάτη

IP Διεύθυνση

3.94.78.5 - 69827 [15/Sep/2013:23:58:36 +0100]
"GET /KBD0C-00033.html HTTP/1.0" 200 14417
"http://www.loudacre.com"
"Loudacre Mobile Browser iFruit 1"