

# Hadoop Ecosystem

## I. Hadoop là gì ?

- Hadoop là một open source framework cho phép lưu trữ và xử lý big data (lượng data lớn mà RDBMS truyền thống không thể xử lý hiệu quả).
- Thay vì sử dụng một máy tính lớn, Hadoop cho phép gom cụm nhiều máy tính
- Hadoop áp dụng kiến trúc master/slave cho cả distributed storage và distributed computation.
- Triết lý Move-code-to-data => Tránh bước truyền dữ liệu tốn kém khi làm việc với dữ liệu lớn
- Sự dư thừa dữ liệu (redundancy) cho phép Hadoop khôi phục lại nếu có một node bị lỗi
- So sánh Hadoop và SQL databases

Hadoop	SQL databases
Scale-out	Scale-up
Key/Value Pairs	Relational Tables
Functional Programming (MapReduce)	Declarative Queries (SQL)
Offline Batch Processing	Online Transactions

- 4 modules chính
  - Hadoop Distributed File System (HDFS)
  - Yet Another Resource Negotiator (YARN)
  - MapReduce
  - Hadoop Common

## II. Hadoop Distributed File System (HDFS)

- Hệ thống distributed storage gọi là Hadoop File System hoặc HDFS
- NameNode là master của HDFS chỉ đạo các slave DataNode thực hiện I/O tasks.
- Khi muốn đọc hoặc ghi một HDFS file, file được chia thành các blocks và NameNode sẽ chỉ cho client các DataNode chứa block tương ứng.
- Mỗi block được rep trên nhiều DataNode khác nhau thuộc Rack khác nhau
- DataNode report định kỳ tới NameNode
- NameNode là single point failure => Secondary NameNode giúp tối thiểu downtime và mất dữ liệu. SNN giao tiếp với NameNode để snapshot HDFS metadata định kỳ.

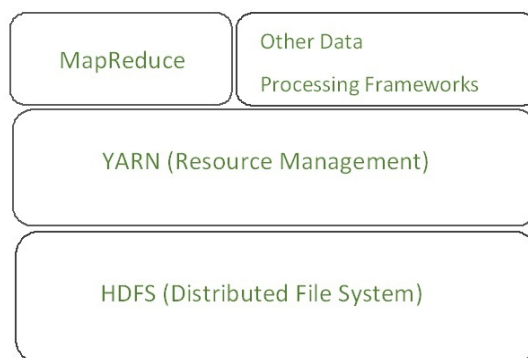
## III. MapReduce

- Là một software framework giúp viết các applications xử lý song song một lượng lớn dữ liệu trên các cluster lớn.
- Viết các applications xử lý song song trên cluster có rất nhiều vấn đề cần giải quyết: lưu trữ file trên nhiều máy, giới hạn RAM, partition data, shuffle partitions, chịu lỗi,...
  - ⇒ Khi viết application trong model MapReduce, Hadoop sẽ hỗ trợ các vấn đề trên.
- MapReduce programs chạy trong 2 pha chính, mapping và reducing.
- Mapper là “filter và transform” input thành “something” mà Reducer có thể tổng hợp lại
- Để mapping, reducing, partitioning, shuffling hoạt động liền mạch
  - ⇒ Cần một cấu trúc chung đủ mạnh và linh hoạt cho dữ liệu
  - ⇒ MapReduce dùng lists và (key/value) pairs

	Input	Output
map	$\langle k1, v1 \rangle$	$list(\langle k2, v2 \rangle)$
reduce	$\langle k2, list(v2) \rangle$	$list(\langle k3, v3 \rangle)$

#### IV. Yet Another Resource Navigator (YARN)

- Computing daemons cũng tuân theo kiến trúc master/slave. Ở Hadoop 1.0: JobTracker là master giám sát overall execution của 1 MapReduce job và TaskTrackers quản lý execution của từng task riêng biệt trên mỗi slave node.
- Ở Hadoop 2.0., YARN được giới thiệu để loại bỏ bottleneck trên JobTracker
- Kiến trúc YARN tách biệt resource management layer với processing layer.



Hadoop 2.0

- YARN cũng cho phép các processing engine khác chạy và xử lý dữ liệu được lưu trữ trong HDFS  $\Rightarrow$  Hệ thống hiệu quả hơn

- YARN Features: Scalability, Compatibility, Cluster Utilization, Multi-tenancy
- Đối với YARN, 1 MapReduce application sẽ sử dụng tài nguyên do Resource Manager cấp phát. Các node tính toán trong cluster sẽ chạy Node Manager quản lý các tiến trình chạy trên node.
- Application Master trên 1 ứng dụng là 1 thư viện cho phép yêu cầu tài nguyên từ Resource Manager và giao tiếp với Node Manager để chạy và thực thi các tasks.

