L1: Introduction to Hadoop
(Hadoop 简介)

Feng Li

feng.li@cufe.edu.cn

School of Statistics and Mathematics
Central University of Finance and Economics

Revised on November 30, 2016

Today we are going to learn... (本节知识要点)

1. **General Information（基本信息）**
2. **The Big Data background（大数据背景）**
3. **What is Hadoop?（Hadoop 概述）**
4. **Install Hadoop（安装 Hadoop）**

General Information（基本信息） I

- **Instructor**: Feng Li <feng.li@cufe.edu.cn>
- **Language**: The course is taught in Chinese. And all the assignments and examinations will be handed out in English. The students are free to choose English or Chinese to answer it.
- **Reception hours**: Questions concerned with this course are most welcome to ask during lectures.

  Of course they can be asked after **Thursday**'s lecture or via email.
- **Literature**
  - 《大数据分布式计算与案例》李丰 著中国人民大学出版社
- **Lecture notes** are available at

  http://feng.li/teaching/pc2016fall/
- **Case studies** are available at

  https://github.com/feng-li/Distributed-Statistical-Computing/

General Information（基本信息） II

- **Other references**
  - Holmes, Alex. Hadoop in practice. Manning Publications Co., 2012.
  - White, Tom. Hadoop: The definitive guide, Third Edition. “O’Reilly Media, Inc.”, 2012.
  - 陆嘉恒. Hadoop 实战. 机械工业出版社, 2012.
- **Working load**: It is a difficult course and I suggest you to study at least of equivalent lecture hours after each lecture to meet the minimal requirement of the exam.
- **Assignments and examinations**: Three sets of take-home group assignments (40% of total course scores).

The Big Data background (大数据背景)

- **Big data** means
  - **Volume**: the quantity of data
  - **Variety**: the category of data
  - **Velocity**: the speed of generation of data
  - **Variability**: the inconsistency of data
  - **Veracity**: the quality of the data
- Big data brings with it two fundamental challenges:
  - how to store and work with voluminous data sizes, and more important,
  - how to understand data and turn it into a competitive advantage.
- **Hadoop** fills a gap in the market by effectively storing and providing computational capabilities over substantial amounts of data. It's a **distributed system** made up of a **distributed filesystem** and it offers a way to parallelize and execute programs on a cluster of machines

What is Hadoop?（Hadoop 概述） I

- Hadoop is a platform that provides both distributed storage and computational capabilities.
- Hadoop proper is a distributed master-slave architecture consists of the Hadoop Distributed File System (**HDFS**) for storage and **MapReduce** for computational capabilities

What is Hadoop?（Hadoop 概述） II

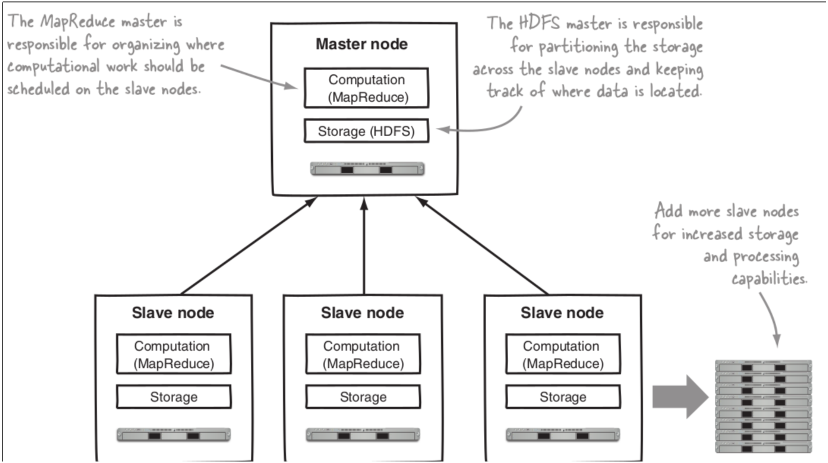

The MapReduce master is responsible for organizing where computational work should be scheduled on the slave nodes.
Master node
Computation (MapReduce)
Storage (HDFS)
The HDFS master is responsible for partitioning the storage across the slave nodes and keeping track of where data is located.
Add more slave nodes for increased storage and processing capabilities.
Slave node
Computation (MapReduce)
Storage
Slave node
Computation (MapReduce)
Storage
Slave node
Computation (MapReduce)
Storage

Figure: The Hadoop architecture

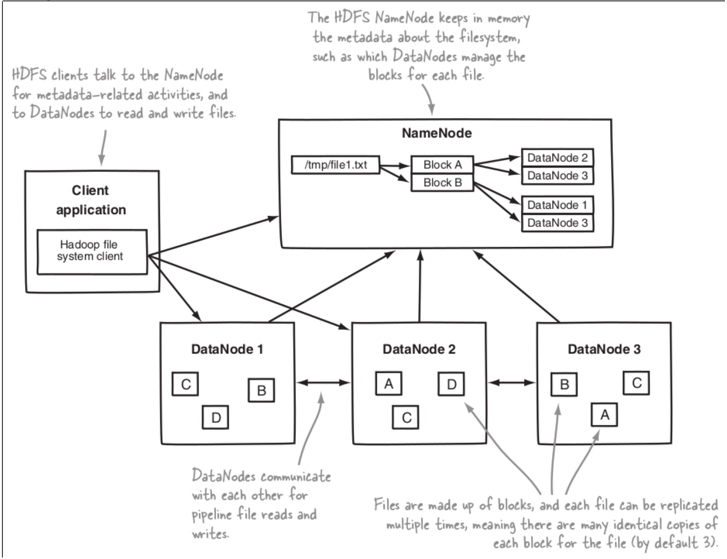
A Brief History of Hadoop（Hadoop 简史）

- Hadoop was created by Doug Cutting.
- At the time Google had published papers that described its novel distributed filesystem, the Google File System (GFS), and MapReduce, a computational framework for parallel processing.
- The successful implementation of these papers' concepts resulted in the Hadoop project.
- Who use Hadoop?
  - Facebook uses Hadoop, Hive, and HB ase for data warehousing and real-time appli- cation serving.
  - Twitter uses Hadoop, Pig, and HB ase for data analysis, visualization, social graph analysis, and machine learning.
  - Yahoo! uses Hadoop for data analytics, machine learning, search ranking, email antispam, ad optimization...
  - eBay, Samsung, Rackspace, J.P. Morgan, Groupon, LinkedIn, AOL , Last.fm...

Core Hadoop components: HDFS （Hadoop 核心组件：HDFS） I

- **HDFS** is the storage component of Hadoop
- It's a distributed file system.
- Logical representation of the components in HDFS : the **NameNode** and the **DataNode**.
- HDFS replicates files for a configured number of times, is tolerant of both software and hardware failure, and automatically re-replicates data blocks on nodes that have failed.
- HDFS isn't designed to work well with random reads over small files due to its optimization for sustained throughput

Core Hadoop components: HDFS (Hadoop 核心组件：HDFS) II



The HDFS NameNode keeps in memory the metadata about the filesystem, such as which DataNodes manage the blocks for each file.

HDFS clients talk to the NameNode for metadata-related activities, and to DataNodes to read and write files.

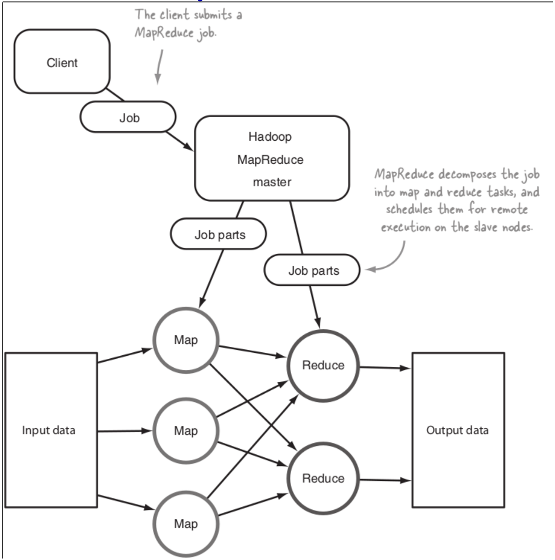
DataNodes communicate with each other for pipeline file reads and writes.

Files are made up of blocks, and each file can be replicated multiple times, meaning there are many identical copies of each block for the file (by default 3).

Core Hadoop components: MapReduce （Hadoop 核心组件：MapReduce） I

- **MapReduce** is a batch-based, distributed computing framework
- It allows you to parallelize work over a large amount of raw data.
- This type of work, which could take days or longer using conventional serial programming techniques, can be reduced down to minutes using MapReduce on a Hadoop cluster.
- MapReduce allows the programmer to focus on addressing business needs, rather than getting tangled up in distributed system complications.
- MapReduce doesn't lend itself to use cases that need real-time data access.

Core Hadoop components: MapReduce (Hadoop 核心组件：MapReduce) II



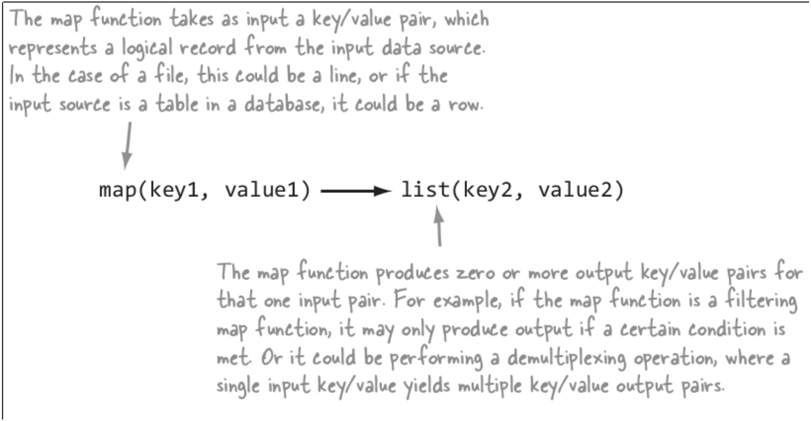
The client submits a MapReduce job.

MapReduce decomposes the job into map and reduce tasks, and schedules them for remote execution on the slave nodes.

The role of the programmer in Hadoop?（Hadoop 解放了程序员的双手） I

- Just need to define map and reduce functions.
- The map function outputs key/value tuples, which are processed by reduce functions to produce the final output.
- The power of MapReduce occurs in between the map output and the reduce input, in the shuffle and sort phases

The role of the programmer in Hadoop? (Hadoop 解放了程序员的双手） II



The map function takes as input a key/value pair, which represents a logical record from the input data source. In the case of a file, this could be a line, or if the input source is a table in a database, it could be a row.

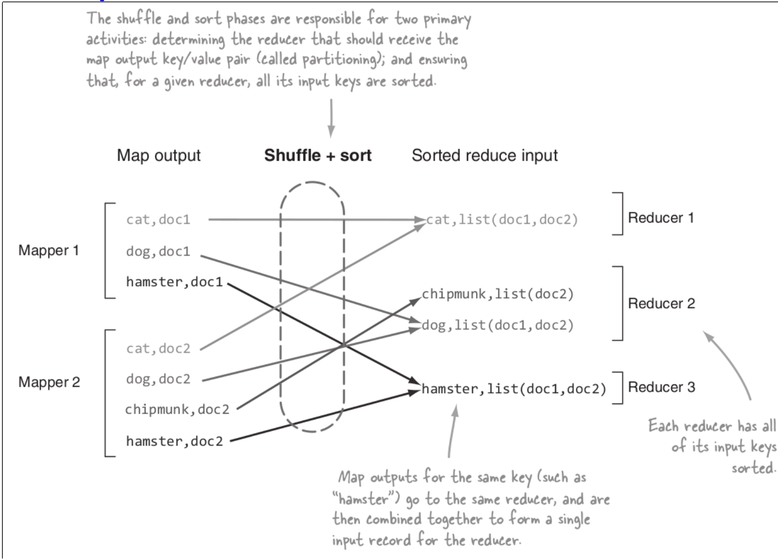
```
map(key1, value1) ——▶ list(key2, value2)
```

The map function produces zero or more output key/value pairs for that one input pair. For example, if the map function is a filtering map function, it may only produce output if a certain condition is met. Or it could be performing a demultiplexing operation, where a single input key/value yields multiple key/value output pairs.

Figure: The map function pseudo code.

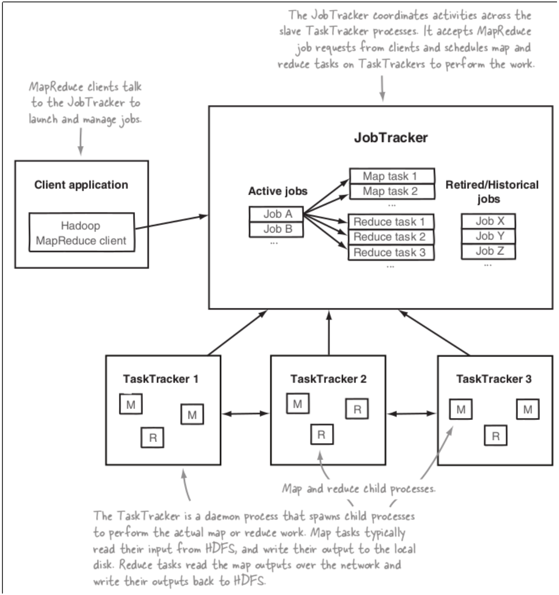
The role of the programmer in Hadoop?（Hadoop 解放了程序员的双手） III

The shuffle and sort phases are responsible for two primary activities: determining the reducer that should receive the map output key/value pair (called partitioning); and ensuring that, for a given reducer, all its input keys are sorted.



Each reducer has all of its input keys sorted.

Map outputs for the same key (such as "hamster") go to the same reducer, and are then combined together to form a single input record for the reducer.



The JobTracker coordinates activities across the slave TaskTracker processes. It accepts MapReduce job requests from clients and schedules map and reduce tasks on TaskTrackers to perform the work.

MapReduce clients talk to the JobTracker to launch and manage jobs.

Map and reduce child processes.

The TaskTracker is a daemon process that spawns child processes to perform the actual map or reduce work. Map tasks typically read their input from HDFS, and write their output to the local disk. Reduce tasks read the map outputs over the network and write their outputs back to HDFS.

Figure: The MapReduce architecture.

The Hadoop modes（Hadoop 运行模式）

- **The standalone mode** In this mode, you do not need to start any Hadoop daemons. Instead, just call `/Hadoop-directory/bin/hadoop` that will execute a Hadoop operation as a single Java process. This is recommended for testing purposes. This is the default mode and you don't need to configure anything else.
- **The pseudo mode**: In this mode, you configure Hadoop for all the nodes. A separate Java Virtual Machine (JVM) is spawned for each of the Hadoop components or daemons like mini cluster on a single host.
- **The full distributed mode:** In this mode, Hadoop is distributed across multiple machines. Dedicated hosts are configured for Hadoop components. Therefore, separate JVM processes are present for all daemons.

Install Hadoop on a pseudo mode （安装伪分布式 Hadoop） I

- Prerequisites
    - Linux OS
    - JDK
    - Dedicated Hadoop system user
    - Configuring SSH
        - Install Open SSH Server
        - Configuring keys
- The configure files are at `hadoop/ect/hadoop/`.
- Hadoop documetation is available at `http://hadoop.apache.org/docs/current/`.

Assignment (I)

- Install Hadoop on your own computer (pseudo mode)
- Try some commands to upload, download, copy, copy from local, move files in HDFS.
- If you encounter anything, start the help document.
- Familiar with Hadoop administrative system (configuration files, logs, http interfaces...)