

Recommended Text:



Hadoop: The Definitive Guide Tom White O'Reilly

Big Data

- Large datasets are becoming more common
 - The New York Stock Exchange generates about one terabyte of new trade data per day.
 - Facebook deals with 350 million photo uploads per day
 - Ancestry.com, the genealogy site, stores around 2.5 petabytes of data.
 - The Internet Archive stores around 2 petabytes of data, and is growing at a rate of 20 terabytes per month.
 - The Large Hadron Collider near Geneva, Switzerland, will produce about 15 petabytes of data per year.

There is a need for a framework to process them.

Example: Find the largest marble for each color.



Sequential



Parallel. Step 1: Group by color for each jar



Parallel. Step 2: Collect all marbles of the same color



Parallel. Step 3: Find largest



Parallel. Step 3: Find largest



MapReduce

- A programming model for data processing
- Create by Google
- Assumes a cluster of commodity servers
- Fault tolerant

Programming Model

- Break processing into two (disjoint) phases
- Map
 - (K1, V1) \rightarrow list(K2, V2)
- Reduce
 - (K2, list(V2)) \rightarrow list(K3, V3)

Example – Weather Data

Logs from the National Climatic Data Center (NCDC)

Data files organized by year

1990.gz

1991.gz

- Highest recorded global temperature for each year in the dataset?
- Data stored using line-oriented ASCII format
 - Each line is a record
 - Includes many meteorological attributes
 - We will focus on temperature

332130 # USAF weather station identifier

19500101 # observation date

0300 # observation time

+51317 # latitude (degrees x 1000)

.....

10268 # atmospheric pressure (hectopascals x 10)

.

Example – Map Function

Input

- Key is offset of reading in file
- Value is a weather station's reading
- Sample input record in file:

0067011990999991950051507004...99999999N9+00001+999999999999... 0043011990999991950051512004...99999999N9+00221+999999999999... 0043011990999991950051518004...99999999N9-00111+99999999999...

• Presented to map function as key-value pairs:

(0, 0067011990999991950051507004...99999999N9+00001+999999999999...) (106, 0043011990999991950051512004...99999999N9+00221+99999999999...) (212, 0043011990999991950051518004...99999999N9-00111+99999999999...)

Extract year and air temperature from each record:

- (1950, 0) (1950, 22)
- (1950, -11)

Input

- Key is year
- Value is a list of temperature reading for that year
- (1949, [111, 78])

(1950, [0, 22, -11])

Iterate through list and pick up the maximum reading

(1949, 111)

(1950, 22)



Apache Hadoop

- Open source MapReduce implementation
- Created by Doug Cutting
- Hadoop is a made-up name.
 - Name Cutting's kid gave to his stuffed yellow elephant
- Supports different programming languages
 - Java, Ruby, Python, C++, others
 - Our focus will be on Java
- In February 2008, Yahoo! announced that its production search index was being generated by a 10,000-core Hadoop cluster
- April 2008, Hadoop broke the world record to sort a terabyte of data in 209 seconds.

Hadoop Job

Input data

- Stored on Hadoop Distributed File System (HDFS)
- MapReduce program
- Configuration information

Hadoop Job Execution

Divide input into fixed-sized input splits

• Typical split size is 1 HDFS block (64 MB)



Run a map task for each split

- Map tasks run in parallel
- Preference is given to running map on node where split is locally stored
- Map tasks write their output to local hard drive

Sort map output and send to reduce task (shuffle)

- All records for the same key are sent to the same reducer
- By default, keys are partition between reducers using a hash function
- Merge records on reducer from multiple mappers
- Run reduce task
- Write output to HDFS

Hadoop Data Flow



Hadoop 1 - System Architecture



Application code



http://hortonworks.com/hadoop/yarn/

Hadoop 2 - System Architecture

ResourceManager

- Scheduler
 - Global resource allocation
- ApplicationsManager.
 - Starts ApplicationMaster

NodeManager

- Per-machine
- Monitors resources

ApplicationMaster

- Per application
- Negotiating appropriate resource containers from the Schedule
- Tracking their status and monitors progress.



Failures

Maper or reducer tasks

- Expected to be common in large clusters
- Re-run on a different node up to a configurable number of times (default is 4)
- Possible to configure number of tasks that can fail before job is terminated

Tasktracker/NodeManager

- Expected to be common in large clusters
- Incomplete allocated job re-run on other nodes

Jobtracker/ApplicationMaster

- Happens infrequently
- Single point of failure
- Job fails
- Hadoop 2 runs multiple Application Masters in high availability mode

Amazon Elastic MapReduce (EMR)

- Amazon supports Hadoop versions 2.6-2.8
- Upload application JAR file to S3
- Upload input files to S3
- Create a new job flow on AWS Management Console



🗨 Feedback	🚱 English (US)		© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.	Privacy Policy	Terms of Use
× Q delete	A V Highlight All	Match Case Whole Word	s 1 of 1 match		

MRJob

Lets you write MapReduce jobs in Python 2.7/3.4+

Run them on several platforms

• Local, Hadoop, EMR

Installation

pip install mrjob

Documentation

https://mrjob.readthedocs.io/en/latest/index.html

Configuration

• mrjob.conf file

runners:

hadoop:

jobconf:

mapred.map.tasks: 2

mapred.reduce.tasks: 1

Example – Word Count

Reads text files and counts how often words occur

Mapper

- Implements map function
- Takes a line as input and breaks it into words. It then emits a key/value pair of the word and 1.

Reducer

- Implements reduce function
- Sums the counts for each word and emits a single key/value with the word and sum.

```
from mrjob.job import MRJob
 1
 2
 3
    class MRWordFrequencyCount(MRJob):
 4
 5
        def mapper(self, _, line):
 6
            for word in line.split():
 7
                 yield(word, 1)
 8
 9
        def reducer(self, word, counts):
            yield(word, sum(counts))
10
11
12
    if ___name___ == '___main___':
13
        MRWordFrequencyCount.run()
```