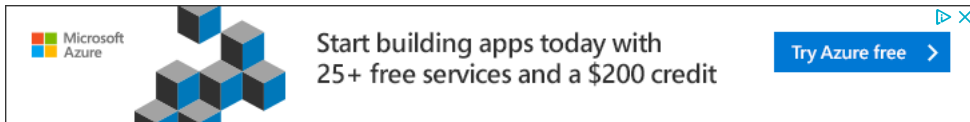


# An Introduction to Apache Spark with Java

By  Chandan Singh () • February 01, 2019 • 0 Comments (/an-introduction-to-apache-spark-with-java/#disqus\_thread)



## What is Apache Spark?

Apache Spark (<https://spark.apache.org>) is an in-memory distributed data processing engine that is used for processing and analytics of large data-sets. Spark presents a simple interface for the user to perform distributed computing on the entire clusters.

Spark does not have its own file systems, so it has to depend on the storage systems for data-processing. It can run on HDFS or cloud based file systems like Amazon S3 (<https://aws.amazon.com/s3>) and Azure BLOB (<https://azure.microsoft.com/en-us/services/storage/blobs/>).

Besides cloud based file systems it can also run with NoSQL databases like Cassandra (<http://cassandra.apache.org/>) and MongoDB (<https://www.mongodb.com/>).

Spark jobs can be written in Java, Scala, Python, R, and SQL. It provides out of the box libraries for Machine Learning, Graph Processing, Streaming and SQL like data-processing. We will go into detail about each of these libraries later in the article.

The engine was developed at the University of California, Berkeley's AMPLab and was donated to Apache Software Foundation in 2013.

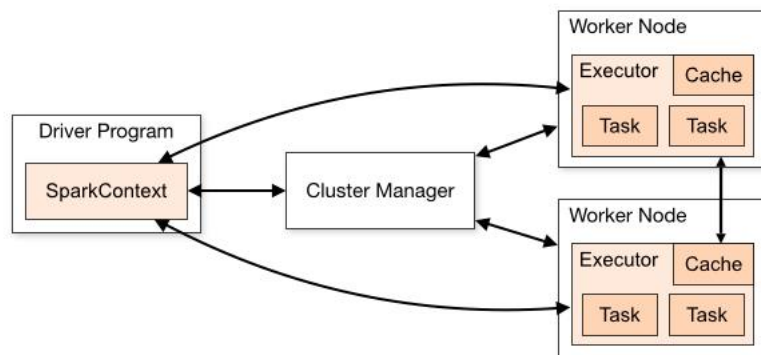
## Need for Spark

The traditional way of processing data on Hadoop (<https://hadoop.apache.org/>) is using its MapReduce framework ([https://hadoop.apache.org/docs/r1.2.1/mapred\\_tutorial.html](https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html)). MapReduce involves a lot of disk usage and as such the processing is slower. As data analytics became more main-stream, the creators felt a need to speed up the processing by reducing the disk utilization during job runs.

Apache Spark addresses this issue by performing the computation in the main memory (RAM) of the worker nodes and does not store mid-step results of computation on disk.

Secondly, it doesn't actually load the data until it is required for computation. It converts the given set of commands into a *Directed Acyclic Graph* (DAG) ([https://en.wikipedia.org/wiki/Directed\\_acyclic\\_graph](https://en.wikipedia.org/wiki/Directed_acyclic_graph)) and then executes it. This prevents the need to read data from the disk and writing back the output of each step as is the case with Hadoop MapReduce. As a result, Spark claims to process data at **100X** faster than a corresponding job using MapReduce for in-memory computation jobs.

# Spark Architecture



Credit: <https://spark.apache.org/> (<https://spark.apache.org/>)

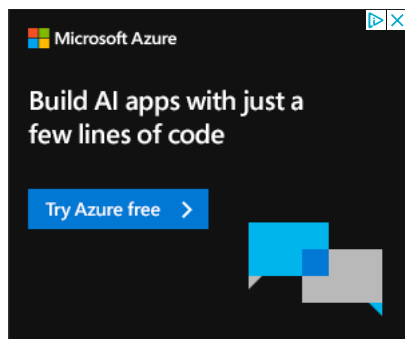
Spark Core uses a master-slave architecture. The Driver program runs in the master node and distributes the tasks to an Executor running on various slave nodes. The Executor runs on their own separate JVMs, which perform the tasks assigned to them in multiple threads.

Each Executor also has a cache associated with it. Caches can be in-memory as well as written to disk on the worker Node. The Executors execute the tasks and send the result back to the Driver.

The Driver communicates to the nodes in clusters using a *Cluster Manager* like the built-in cluster manager, Mesos (<http://mesos.apache.org/>), YARN (<https://yarnpkg.com/en/>), etc. The batch programs we write get executed in the Driver Node.

## Simple Spark Job Using Java

We have discussed a lot about Spark and its architecture, so now let's take a look at a simple Spark job which counts the sum of space-separated numbers from a given text file:



```
32 23 45 67 2 5 7 9
12 45 68 73 83 24 1
12 27 51 34 22 14 31
...
```

We will start off by importing the dependencies for Spark Core which contains the Spark processing engine. It has no further requirements as it can use the local file-system to read the data file and write the results:

```
<dependency>
  <groupId>org.apache.spark</groupId>
  <artifactId>spark-core_2.10</artifactId>
  <version>2.2.3</version>
</dependency>
```

With the core setup, let's proceed to write our Spark batch!

```
public class CalculateFileSum {
    public static String SPACE_DELIMITER = " ";
    public static void main(String[] args) {

        SparkConf conf = new SparkConf().setMaster("local[*]").setAppName("SparkFileSumApp");
        JavaSparkContext sc = new JavaSparkContext(conf);

        JavaRDD<String> input = sc.textFile("numbers.txt");
        JavaRDD<String> numberStrings = input.flatMap(s -> Arrays.asList(s.split(SPACE_DELIMITER)).iterator());
        JavaRDD<String> validNumberString = numberStrings.filter(string -> !string.isEmpty());
        JavaRDD<Integer> numbers = validNumberString.map(numberString -> Integer.valueOf(numberString));
        int finalSum = numbers.reduce((x,y) -> x+y);

        System.out.println("Final sum is: " + finalSum);

        sc.close();
    }
}
```

## Subscribe to our Newsletter

Get occasional tutorials, guides, and reviews in your inbox. No spam ever. Unsubscribe at any time.

Enter your email...

Subscribe

Running this piece of code should yield:

```
Final sum is: 687
```

The `JavaSparkContext` object we have created acts as a connection to the cluster. The Spark Context we have created here has been allocated all the available local processors, hence the `*`.

The most basic abstraction in Spark is `RDD`, which stands for *Resilient Distributed Datasets*. It is resilient and distributed since the data is replicated across the cluster and can be recovered if any of the nodes crash.

Another benefit of distributing data is that it can be processed in parallel thus promoting horizontal scaling. Another important feature of RDDs is that they are immutable. If we apply any action or transformation to a given RDD, the result is another set of RDDs.

In this example, we have read the words from the input file as `RDD`s and converted them into numbers. Then we have applied the `reduce` function on them to sum up the values of each of the RDDs before displaying them on the console.

## Introduction to Spark Libraries

Spark provides us with a number of built-in libraries which run on top of Spark Core.



## Spark SQL

Spark SQL (<https://spark.apache.org/sql/>) provides a SQL-like interface to perform processing of structured data. When the user executes an SQL query, internally a batch job is kicked-off by Spark SQL which manipulates the RDDs as per the query.

The benefit of this API is that those familiar with *RDBMS*-style querying find it easy to transition to Spark and write jobs in Spark.

## Spark Streaming

Spark Streaming (<https://spark.apache.org/streaming/>) is suited for applications which deal in data flowing in real-time, like processing Twitter feeds.

Spark can integrate with Apache Kafka (<https://kafka.apache.org>) and other streaming tools to provide fault-tolerant and high-throughput processing capabilities for the streaming data.

## Spark MLlib

MLlib (<https://spark.apache.org/mllib/>) is short for *Machine Learning Library* which Spark provides. It includes the common learning algorithms like classification, recommendation, modeling, etc. which are used in Machine learning.

These algorithms can be used to train the model as per the underlying data. Due to the extremely fast data processing supported by Spark, the machine learning models can be trained in a relatively shorter period of time.

## GraphX

As the name indicates, GraphX (<https://spark.apache.org/graphx/>) is the Spark API for processing graphs and performing graph-parallel computation.

The user can create graphs and perform operations like joining and transforming the graphs. As with MLlib, Graphx comes with built-in graph algorithms for page rank, triangle count, and more.

## Conclusion

Apache Spark is the platform of choice due to its blazing data processing speed, ease-of-use, and fault tolerant features.

In this article, we took a look at the architecture of Spark and what is the secret of its lightning-fast processing speed with the help of an example. We also took a look at the popular Spark Libraries and their features.

---

🔖 [java \(/tag/java/\)](#), [spark \(/tag/spark/\)](#)

🐦 (<https://twitter.com/share?text=An%20Introduction%20to%20Apache%20Spark%20with%20Java&url=https://stackabuse.com/an-introduction-to-apache-spark-with-java/>)

f (<https://www.facebook.com/sharer/sharer.php?u=https://stackabuse.com/an-introduction-to-apache-spark-with-java/>)

🔗 (<https://plus.google.com/share?url=https://stackabuse.com/an-introduction-to-apache-spark-with-java/>)

in (<https://www.linkedin.com/shareArticle?mini=true%26url=https://stackabuse.com/an-introduction-to-apache-spark-with-java/%26source=https://stackabuse.com>)



(/author/chand31290-2/)

### About Chandan Singh (/author/chand31290-2/)

🏠 Mumbai 🌐 Website (<https://www.linkedin.com/in/chandan-singh-431a2340/>)

Chandan is a passionate software engineer with extensive experience in designing and developing Java applications. In free time, he likes to read fiction and write about his experiences.

## Subscribe to our Newsletter

Get occasional tutorials, guides, and reviews in your inbox. No spam ever. Unsubscribe at any time.

Enter your email...

Subscribe

0 Comments StackAbuse

Login

Recommend Tweet Share

Sort by Best



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

ALSO ON STACKABUSE

### JavaScript Convert String to Number

1 comment • a month ago



Эд Лесничий — what about Number('1') ?

Avatar

### Python Logging Basics

1 comment • 4 months ago



Simon SOUVANNARAT — Thanks for this article,I noticed the code below in section Logging to a File vs the Standard ...

Avatar

### Sorting Algorithms in Python

1 comment • 15 days ago



Randy Ram — Great article Marcus! Loving the posts so far. Just a note, the table in the article wasn't scrolling to ...

Avatar

### Getting Started with Selenium and Python

3 comments • 14 days ago



Przemek — Thanks. You write this Just\_In\_Timeand GrandPlusAward for great introduction

Avatar

Subscribe Add Disqus to your siteAdd DisqusAdd Disqus' Privacy PolicyPrivacy PolicyPrivacy

[< Previous Post : Linked Lists in Detail with Python Examples: Single Linked Lists \(/linked-lists-in-detail-with-python-examples-single-linked-lists/\)](#)[Next Post : Running SQL on CSV Data: Data Conversion and Extraction \(/running-sql-on-csv-data-data-conversion-and-extraction/\)](#)

## Ad

## Follow Us

-  [Twitter \(https://twitter.com/StackAbuse\)](https://twitter.com/StackAbuse)
-  [Facebook \(https://www.facebook.com/stackabuse\)](https://www.facebook.com/stackabuse)
-  [RSS \(https://stackabuse.com/rss/\)](https://stackabuse.com/rss/)

## Newsletter

Subscribe to our newsletter! Get occassional tutorials, guides, and reviews in your inbox.

Enter your email...

Subscribe

No spam ever. Unsubscribe at any time.

Ad

Want a remote job?

100% Remote Mid-Level Java Software Engineer  
TEKsystems 8 days ago



\$70K - \$100K  
([SR Java Developer \(remote\)  
\*\*100% Remote\*\* 6 days ago  
\\$120K - \\$130K  
\(\[Sr. Java Developer \\( Remote \\)  
\\*\\*Software Technology Inc\\*\\* 2 days ago  
\\\$103K - \\\$136K  
\\(\\[➡ More jobs \\\(<https://hireremote.io>\\\)\\]\\(https://www.ziprecruiter.com/clk/software-technology-inc-00000000-sr-java-developer-remote-4f8bc8b5?clk=fIXiFoYIfi7qUYUcFhTcrzFYJzsRqZ9es7BQq5-wbnodC8ljWLbZ4iifQxIPuRaMCZbmkd0880vm8TqCFVjsv4am8V6t\\_roUXPtV6Eco8mVRpZSvSyrqgQ-8SICzfABMTaHMo9I4-f8MM4QCPLMeesyt1Foc9dyGVuauYV0knvAAhNFySOqHaoXwofgxtqfIOg14YHFU-nBllpTAT\\_YZtTsf1HGGYsWq7jMg1YfNMaaWW4p144xoV9-EBCs12PMosxiRQsoujBeXWdUWRHMYfmSTqu3Fbu81Y4Lw3KiyJX68f3suU18Ymikelb-7RrTU7xeyNAWD5Bxoe-QPbtmUOnhvpbqLgHLR9RCKfslkhEpLhF5wYl1rzDBGhYCYUiRiWkYLFq973qlSP\\_vYsvDr9Yd7ruRLwmeGA-00YlZTSFRdO3Hgsi3q2Zsttq7SKgqxebVjjkvHvWL-WaL89hCFAs0vvaCNifMqZH6EUylaW4BR7osTbYKW2N3geuXEUWPatn30XJd18vi94Fpd6ZRv6sSy0B-QJXQG\\_NawSc\\_XTH\\_SFNwDKhU1DqD5UbCM5dfBDYRLMPxcW9tZlpa7Sa9JLG9\\_KeYRjRODvnTclHlBWrIDUjAb-bOpu6ZCNC-EasmMtDwD5XxYViMFlstsuVLXC5oDKwS8vCdXNwvpHew.f1b33ae5b0725898e6a9e3c5b178f14c\\)</a>\\)</p></div><div data-bbox=\\)\]\(https://www.ziprecruiter.com/clk/100-remote-00000000-sr-java-developer-remote-c1bf88b0?clk=r4a7dtHrcEx53taSBhQvjxtrSNzJYypJAesO9u00riGn6t7M\_1PD97viGOMBMWwSh5WYCFMaeXjzmZK2hgA2HXDEIWH5ponGGbdKrrGTMlIUU55Dbz-y7Z94X9LZgySevL1jOKKV8Hcv7mxXGxBzJApXEnAcf91jV5WiNKGAQAVy6KEh0BZ5WRmcU2lhqSVjmeEbPuBcqGgd2XsDYueDNtiMWABr665OQmj2MwswC62YVpFnKBUL-M4EZERA9KKFC7fxx5DXiokRSNC7Zy\_YedHCRm0XEcTh6Cf6ml198LBpz6uls9HCou1lzW\_hwpoaYlhzr3wYKURbxvPPZlPHuAoC5RyUYyvjgTXBvGCkfXB\_AzPoZrG2o4DZzmluCBryzD71gPg3nddvnpqmmamLrLryQQNPEz6Ry8Hvr58rdWvvqEshoK1LPVF437hOSBx8VYxt8q4h\_DIE9wQ7qvYr1eGz48qtH96176p9tYjg\_Nb1RTGas7KQoumGhNoG7N3fx14Y4kqm9mE21g88fbtk4vm4BBqepj9AMQPDUC2IEBFyg8x2lye3vgrkPTIDQ71g8uirBn38ZJ4J7Pvf2vLnFPXk1OCu1b5HS8501HPTU0LT-in5nDeKKNIYd\_XbA.894a1c1d0a1c94ac2bc9970de387ff47\)</a>\)</p></div><div data-bbox=\)](https://www.ziprecruiter.com/clk/teksystems-00000000-100-remote-mid-level-java-software-engineer-9c78bcf1?clk=bMhb-SvEDftQkWXvYep0cp0z4BJtxXYSylluu1-hVIRRVeXjhRFSmVitkNfAkhGeqLXtq-Qo2UEL_pxdD20q5SB26SSGLyCxZndjVu0BbeZYM5Y5ulro-YiluQg2Ffyu2EzvpjAfNaKKewgZBHmIGB08XqzCUmpkuZ0vd90KyE3O7dDTVoqArjLH_7LNaqM6eR_QBSCIAukZoiP7Wfp3kZSMPiEZWSVtly303luQOitBB7MF4SRHPbMcT9nMYDjX7S_bYtqWhEXcgFtusSKXgmVRY4vkRIKjUjoZT2oVoLamM4b21kzxyrnE2tpWf6kTaWegvhnaT2paW55W3UfdGABJXBeKTBDO55HIQrG277rZjkENuKqtBFMBdG42oknzsAUrbXHztmiiQoBFskh4ieaGJ9zDKIAcdU2X0leprgFHc9XEQ4HLWJ9eYYR-jjQZX7JcdnDC81zL4ybnReBI7_Kj6yoL4AQqKtEjkH_vWenmkXeqLWvWR8A1TUs0odoE6rkaeioim1MspWAc_KLUR7ZlpWppUhu7liiEwILON_-Oxvc0vALwpe56KJdi7tWKbGCCuWGxQwWh4sWLAvdGvllIRAr9w19-U9IAtOu3E2lZsN6fUxdoYqTQYw.b37bcd79287b75a60b3cfa60ae4086db)</a>)</p></div><div data-bbox=)

Interviewing for a job?

(<http://stackabu.se/daily-coding-problem>)

- Improve your skills by solving one coding problem every day
- Get the solutions the next morning via email
- Practice on **actual problems** asked by top companies, like:

</> Daily Coding Problem (<http://stackabu.se/daily-coding-problem>)

## Recent Posts

---

Introduction to the Python Calendar Module (<https://stackabuse.com/introduction-to-the-python-calendar-module/>)

April 16, 2019

---

Python for NLP: Introduction to the TextBlob Library (<https://stackabuse.com/python-for-nlp-introduction-to-the-textblob-library/>)

April 15, 2019

---

How to fix: "WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED" on Mac and Linux (<https://stackabuse.com/how-to-fix-warning-remote-host-identification-has-changed-on-mac-and-linux/>)

April 12, 2019

---

## Tags

---


[python \(/tag/python\)](#) [nlp \(/tag/nlp\)](#) [ssh \(/tag/ssh\)](#) [unix \(/tag/unix\)](#) [security \(/tag/security\)](#) [java \(/tag/java\)](#) [xml \(/tag/xml\)](#) [scikit-learn \(/tag/scikit-learn\)](#) [spring boot \(/tag/spring-boot\)](#)  
[spring \(/tag/spring\)](#) [web scraping \(/tag/web-scraping\)](#) [algorithms \(/tag/algorithms\)](#) [postgresql \(/tag/postgresql\)](#) [spacy \(/tag/spacy\)](#)

## Follow Us

---

 Twitter (<https://twitter.com/StackAbuse>)

 Facebook (<https://www.facebook.com/stackabuse>)

 RSS (<https://stackabuse.com/rss/>)

---

Copyright © 2019, Stack Abuse (<https://stackabuse.com>). All Right Reserved

[Disclosure \(/disclosure\)](#) • [Privacy Policy \(/privacy-policy\)](#) • [Terms of Service \(/terms-of-service\)](#)