About the Tutorial

Apache Pig is an abstraction over MapReduce. It is a tool/platform which is used to analyze larger sets of data representing them as data flows. Pig is generally used with Hadoop; we can perform all the data manipulation operations in Hadoop using Pig.

Audience

This tutorial is meant for all those professionals working on Hadoop who would like to perform MapReduce operations without having to type complex codes in Java.

Prerequisites

To make the most of this tutorial, you should have a good understanding of the basics of Hadoop and HDFS commands. It will certainly help if you are good at SQL.

Copyright & Disclaimer

© Copyright 2015 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the Tutorial</td>
<td>i</td>
</tr>
<tr>
<td>Audience</td>
<td>i</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>i</td>
</tr>
<tr>
<td>Copyright &amp; Disclaimer</td>
<td>i</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>ii</td>
</tr>
</tbody>
</table>

## PART 1: INTRODUCTION

1. Apache Pig – Overview ......................................................... 2
   - What is Apache Pig? ......................................................... 2
   - Why Do We Need Apache Pig? .............................................. 2
   - Features of Pig .............................................................. 2
   - Apache Pig Vs MapReduce .................................................. 3
   - Apache Pig Vs SQL .................................................................. 3
   - Apache Pig Vs Hive ................................................................ 4
   - Applications of Apache Pig .................................................. 4
   - Apache Pig – History ............................................................ 5

2. Apache Pig – Architecture .................................................... 6
   - Apache Pig – Components ...................................................... 7
   - Pig Latin – Data Model .......................................................... 7

## PART 2: ENVIRONMENT

3. Apache Pig – Installation ....................................................... 10
   - Prerequisites ....................................................................... 10
   - Download Apache Pig .............................................................. 10
   - Install Apache Pig .................................................................. 13
   - Configure Apache Pig ............................................................ 14

4. Apache Pig – Execution ............................................................. 16
   - Apache Pig – Execution Modes ............................................... 16
   - Apache Pig – Execution Mechanisms ....................................... 16
   - Invoking the Grunt Shell ...................................................... 16
   - Executing Apache Pig in Batch Mode ..................................... 17

5. Grunt Shell ............................................................................. 18
   - Shell Commands .................................................................... 18
   - Utility Commands .................................................................. 19

## PART 3: PIG LATIN

6. Pig Latin – Basics ................................................................. 26
   - Pig Latin – Data Model .......................................................... 26
   - Pig Latin – Statements ........................................................... 26
   - Pig Latin – Data types ............................................................ 27
   - Null Values ............................................................................ 27
   - Pig Latin – Arithmetic Operators .......................................... 28
LOWER ( )  ........................................................................................................... 136
REPLACE ( ) ......................................................................................................... 137
STRSPLIT ( ) ........................................................................................................ 138
STRSPLITTOBAG ( ) ............................................................................................ 139
Trim ( ) ................................................................................................................ 141
LTRIM ( ) ............................................................................................................. 142
RTRIM ..................................................................................................................... 143

28. date-time Functions ....................................................................................... 145
   ToDoDate ( ) .................................................................................................... 147
   GetDay ( ) ....................................................................................................... 148
   GetHour ( ) .................................................................................................... 149
   GetMinute ( ) ................................................................................................. 150
   GetSecond ( ) ............................................................................................... 151
   GetMilliseconds ( ) ....................................................................................... 152
   GetYear ............................................................................................................ 153
   GetMonth ( ) .................................................................................................. 154
   GetWeek ( ) ................................................................................................... 156
   GetWeekYear ( ) ............................................................................................ 157
   CurrentTime ( ) ............................................................................................. 158
   ToString ( ) ................................................................................................... 159
   DaysBetween ( ) ............................................................................................ 160
   HoursBetween ( ) .......................................................................................... 161
   MinutesBetween ( ) ....................................................................................... 161
   SecondsBetween ( ) ....................................................................................... 162
   MillisecondsBetween ( ) ................................................................................ 163
   YearsBetween ( ) ........................................................................................... 164
   MonthsBetween ( ) ....................................................................................... 165
   WeeksBetween ( ) .......................................................................................... 166
   AddDuration ( ) .............................................................................................. 167
   SubtractDuration ( ) ...................................................................................... 168

29. Math Functions ............................................................................................. 170
   ABS ( ) ............................................................................................................ 171
   ACOS ( ) ......................................................................................................... 172
   ASIN ( ) ......................................................................................................... 174
   ATAN ( ) ......................................................................................................... 175
   CBRT ( ) ......................................................................................................... 176
   CEIL .................................................................................................................. 177
   COS ( ) ............................................................................................................ 178
   COSH ( ) ........................................................................................................ 179
   EXP ( ) ............................................................................................................ 180
   FLOOR ( ) ....................................................................................................... 181
   LOG ( ) .......................................................................................................... 181
   LOG10 ( ) ...................................................................................................... 182
   RANDOM ( ) .................................................................................................. 183
   ROUND ( ) ..................................................................................................... 184
   SIN ( ) ............................................................................................................. 185
   SINH ( ) ......................................................................................................... 186
   SQRT ( ) ......................................................................................................... 187
   TAN ( ) ............................................................................................................ 188
TANH ( ) .................................................................................................................. 189

PART 11: OTHER MODES OF EXECUTION ..................................................................... 191

30. User-Defined Functions ............................................................................................ 192
   Types of UDF’s in Java ............................................................................................... 192
   Writing UDF’s using Java .......................................................................................... 192
   Using the UDF .......................................................................................................... 196

31. Running Scripts ......................................................................................................... 198
   Comments in Pig Script ............................................................................................. 198
   Executing Pig Script in Batch mode .......................................................................... 198
   Executing a Pig Script from HDFS .......................................................................... 199
Part 1: Introduction
1. Apache Pig – Overview

What is Apache Pig?

Apache Pig is an abstraction over MapReduce. It is a tool/platform which is used to analyze larger sets of data representing them as data flows. Pig is generally used with Hadoop; we can perform all the data manipulation operations in Hadoop using Apache Pig.

To write data analysis programs, Pig provides a high-level language known as Pig Latin. This language provides various operators using which programmers can develop their own functions for reading, writing, and processing data.

To analyze data using Apache Pig, programmers need to write scripts using Pig Latin language. All these scripts are internally converted to Map and Reduce tasks. Apache Pig has a component known as Pig Engine that accepts the Pig Latin scripts as input and converts those scripts into MapReduce jobs.

Why Do We Need Apache Pig?

Programmers who are not so good at Java normally used to struggle working with Hadoop, especially while performing any MapReduce tasks. Apache Pig is a boon for all such programmers.

- Using Pig Latin, programmers can perform MapReduce tasks easily without having to type complex codes in Java.

- Apache Pig uses multi-query approach, thereby reducing the length of codes. For example, an operation that would require you to type 200 lines of code (LoC) in Java can be easily done by typing as less as just 10 LoC in Apache Pig. Ultimately Apache Pig reduces the development time by almost 16 times.

- Pig Latin is SQL-like language and it is easy to learn Apache Pig when you are familiar with SQL.

- Apache Pig provides many built-in operators to support data operations like joins, filters, ordering, etc. In addition, it also provides nested data types like tuples, bags, and maps that are missing from MapReduce.

Features of Pig

Apache Pig comes with the following features:

- Rich set of operators: It provides many operators to perform operations like join, sort, file, etc.

- Ease of programming: Pig Latin is similar to SQL and it is easy to write a Pig script if you are good at SQL.
• **Optimization opportunities:** The tasks in Apache Pig optimize their execution automatically, so the programmers need to focus only on semantics of the language.

• **Extensibility:** Using the existing operators, users can develop their own functions to read, process, and write data.

• **UDF’s:** Pig provides the facility to create **User-defined Functions** in other programming languages such as Java and invoke or embed them in Pig Scripts.

• **Handles all kinds of data:** Apache Pig analyzes all kinds of data, both structured as well as unstructured. It stores the results in HDFS.

### Apache Pig Vs MapReduce

Listed below are the major differences between Apache Pig and MapReduce.

<table>
<thead>
<tr>
<th>Apache Pig</th>
<th>MapReduce</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Pig is a data flow language.</td>
<td>MapReduce is a data processing paradigm.</td>
</tr>
<tr>
<td>It is a high level language.</td>
<td>MapReduce is low level and rigid.</td>
</tr>
<tr>
<td>Performing a Join operation in Apache Pig is pretty simple.</td>
<td>It is quite difficult in MapReduce to perform a Join operation between datasets.</td>
</tr>
<tr>
<td>Any novice programmer with a basic knowledge of SQL can work conveniently with Apache Pig.</td>
<td>Exposure to Java is must to work with MapReduce.</td>
</tr>
<tr>
<td>Apache Pig uses multi-query approach, thereby reducing the length of the codes to a great extent.</td>
<td>MapReduce will require almost 20 times more the number of lines to perform the same task.</td>
</tr>
<tr>
<td>There is no need for compilation. On execution, every Apache Pig operator is converted internally into a MapReduce job.</td>
<td>MapReduce jobs have a long compilation process.</td>
</tr>
</tbody>
</table>

### Apache Pig Vs SQL

Listed below are the major differences between Apache Pig and SQL.

<table>
<thead>
<tr>
<th>Pig</th>
<th>SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pig Latin is a <strong>procedural</strong> language.</td>
<td>SQL is a <strong>declarative</strong> language.</td>
</tr>
</tbody>
</table>
In Apache Pig, **schema** is optional. We can store data without designing a schema (values are stored as $01, $02 etc.)

<table>
<thead>
<tr>
<th>Apache Pig</th>
<th>Hive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema is mandatory in SQL.</td>
<td>The data model used in SQL is <strong>flat relational</strong>.</td>
</tr>
<tr>
<td>The data model in Apache Pig is <strong>nested relational</strong>.</td>
<td>Apache Pig provides limited opportunity for <strong>Query optimization</strong>.</td>
</tr>
<tr>
<td>There is more opportunity for query optimization in SQL.</td>
<td></td>
</tr>
</tbody>
</table>

In addition to above differences, Apache Pig Latin;

- Allows splits in the pipeline.
- Allows developers to store data anywhere in the pipeline.
- Declares execution plans.
- Provides operators to perform ETL (Extract, Transform, and Load) functions.

**Apache Pig Vs Hive**

Both Apache Pig and Hive are used to create MapReduce jobs. And in some cases, Hive operates on HDFS in a similar way Apache Pig does. In the following table, we have listed a few significant points that set Apache Pig apart from Hive.

<table>
<thead>
<tr>
<th>Apache Pig</th>
<th>Hive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Pig uses a language called <strong>Pig Latin</strong>. It was originally created at <strong>Yahoo</strong>.</td>
<td>Hive uses a language called <strong>HiveQL</strong>. It was originally created at <strong>Facebook</strong>.</td>
</tr>
<tr>
<td>Pig Latin is a data flow language.</td>
<td>HiveQL is a query processing language.</td>
</tr>
<tr>
<td>Pig Latin is a procedural language and it fits in pipeline paradigm.</td>
<td>HiveQL is a declarative language.</td>
</tr>
<tr>
<td>Apache Pig can handle structured, unstructured, and semi-structured data.</td>
<td>Hive is mostly for structured data.</td>
</tr>
</tbody>
</table>

**Applications of Apache Pig**

Apache Pig is generally used by data scientists for performing tasks involving ad-hoc processing and quick prototyping. Apache Pig is used;

- To process huge data sources such as web logs.
• To perform data processing for search platforms.
• To process time sensitive data loads.

Apache Pig – History

In **2006**, Apache Pig was developed as a research project at Yahoo, especially to create and execute MapReduce jobs on every dataset. In **2007**, Apache Pig was open sourced via Apache incubator. In **2008**, the first release of Apache Pig came out. In **2010**, Apache Pig graduated as an Apache top-level project.
2. Apache Pig – Architecture

The language used to analyze data in Hadoop using Pig is known as Pig Latin. It is a high-level data processing language which provides a rich set of data types and operators to perform various operations on the data.

To perform a particular task Programmers using Pig, programmers need to write a Pig script using the Pig Latin language, and execute them using any of the execution mechanisms (Grunt Shell, UDFs, Embedded). After execution, these scripts will go through a series of transformations applied by the Pig Framework, to produce the desired output.

Internally, Apache Pig converts these scripts into a series of MapReduce jobs, and thus, it makes the programmer’s job easy. The architecture of Apache Pig is shown below.

![Apache Pig Architecture Diagram]

Figure: Apache Pig Architecture
Apache Pig – Components

As shown in the figure, there are various components in the Apache Pig framework. Let us take a look at the major components.

Parser

Initially the Pig Scripts are handled by the Parser. It checks the syntax of the script, does type checking, and other miscellaneous checks. The output of the parser will be a DAG (directed acyclic graph), which represents the Pig Latin statements and logical operators.

In the DAG, the logical operators of the script are represented as the nodes and the data flows are represented as edges.

Optimizer

The logical plan (DAG) is passed to the logical optimizer, which carries out the logical optimizations such as projection and pushdown.

Compiler

The compiler compiles the optimized logical plan into a series of MapReduce jobs.

Execution engine

Finally the MapReduce jobs are submitted to Hadoop in a sorted order. Finally, these MapReduce jobs are executed on Hadoop producing the desired results.

Pig Latin – Data Model

The data model of Pig Latin is fully nested and it allows complex non-atomic datatypes such as map and tuple. Given below is the diagrammatical representation of Pig Latin’s data model.

Atom

Any single value in Pig Latin, irrespective of their data, type is known as an Atom. It is stored as string and can be used as string and number. int, long, float, double, chararray, and bytarray are the atomic values of Pig.

A piece of data or a simple atomic value is known as a field.
Example: ‘raja’ or ‘30’

Tuple
A record that is formed by an ordered set of fields is known as a tuple, the fields can be of any type. A tuple is similar to a row in a table of RDBMS.

Example: (Raja, 30)

Bag
A bag is an unordered set of tuples. In other words, a collection of tuples (non-unique) is known as a bag. Each tuple can have any number of fields (flexible schema). A bag is represented by ‘{‘. It is similar to a table in RDBMS, but unlike a table in RDBMS, it is not necessary that every tuple contain the same number of fields or that the fields in the same position (column) have the same type.

Example: {(Raja, 30), (Mohammad, 45)}

A bag can be a field in a relation; in that context, it is known as inner bag.

Example: {Raja, 30, {9848022338, raja@gmail.com,}}

Relation
A relation is a bag of tuples. The relations in Pig Latin are unordered (there is no guarantee that tuples are processed in any particular order).

Map
A map (or data map) is a set of key-value pairs. The key needs to be of type chararray and should be unique. The value might be of any type. It is represented by ‘[]’

Example: [name#Raja, age#30]
Part 2: Environment
This chapter explains the how to download, install, and set up Apache Pig in your system.

Prerequisites

It is essential that you have Hadoop and Java installed on your system before you go for Apache Pig. Therefore, prior to installing Apache Pig, install Hadoop and Java by following the steps given in the following link:

http://www.tutorialspoint.com/hadoop/hadoop_environmet_setup.htm

Download Apache Pig

First of all, download the latest version of Apache Pig from the website https://pig.apache.org/.

Step 1

Open the homepage of Apache Pig website. Under the section News, click on the link release page as shown in the following snapshot.

---

Apache Pig is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs. The salient property of Pig programs is that their structure is amenable to substantial parallelization, which in turns enables them to handle very large data sets.

At the present time, Pig's infrastructure layer consists of a compiler that produces sequences of Map-Reduce programs, for which large-scale parallel implementations already exist (e.g., the Hadoop sub-project). Pig's language layer currently consists of a textual language called Pig Latin, which has the following key properties:

- **Ease of programming.** It is trivial to achieve parallel execution of simple, “embarrassingly parallel” data analysis tasks. Complex tasks comprised of multiple interrelated data transformations are explicitly encoded as data flow sequences, making them easy to write, understand, and maintain.

- **Optimization opportunities.** The way in which tasks are encoded permits the system to optimize their execution automatically, allowing the user to focus on semantics rather than efficiency.

- **Extensibility.** Users can create their own functions to do special-purpose processing.

News

Apache Pig 0.15.0 is released!

The highlights of this release includes Pig on Tez stabilization, improved tez auto-parallelism and invoking Hive UDFs from Pig. See details on the release page.
Step 2
On clicking the specified link, you will be redirected to the **Apache Pig Releases** page. On this page, under the **Download** section, you will have two links, namely, **Pig 0.8 and later** and **Pig 0.7 and before**. Click on the link **Pig 0.8 and later**, then you will be redirected to the page having a set of mirrors.
Step 3
Choose and click any one of these mirrors as shown below.

Step 4
These mirrors will take you to the Pig Releases page. This page contains various versions of Apache Pig. Click the latest version among them.
Step 5

Within these folders, you will have the source and binary files of Apache Pig in various distributions. Download the tar files of the source and binary files of Apache Pig 0.15, `pig-0.15.0-src.tar.gz` and `pig-0.15.0.tar.gz`.

![Index of /dist/pig/pig-0.15.0](image)

**Install Apache Pig**

After downloading the Apache Pig software, install it in your Linux environment by following the steps given below.

**Step 1**

Create a directory with the name Pig in the same directory where the installation directories of **Hadoop**, **Java**, and other software were installed. (In our tutorial, we have created the Pig directory in the user named Hadoop).

```
$ mkdir Pig
```