ImportantConcepts

Most of the documentation on Hadoop assumes that you know some of the basic concepts and procedures involved in writing and running jobs in Hadoop. This can be very confusing at first because many of these concepts won't be things that you will have picked up unless you have used a system like Hadoop in the past.

Some notable terms that may confuse you:

- Hadoop - Hadoop itself refers to the overall system that runs jobs, distributes tasks (pieces of these jobs) and stores data in a parallel and distributed fashion.
- Map/reduce - Is the style in which most programs running on Hadoop are written. In this style, input is broken in tiny pieces which are processed independently (the map part). The results of these independent processes are then collated into groups and processed as groups (the reduce part). Follow the link for a much more complete description.
- Job - In Hadoop, the combination of all of the JAR files and classes needed to run a map/reduce program is called a *job*. All of these components are themselves collected into a JAR which is usually referred to as a *job file*. To execute a job, you submit it to a JobTracker. On the command line, this is done with the command:

```
hadoop jar your-job-file-goes-here.jar
```

This assumes that your job file has a main class that is defined as if it were executable from the command line and that this main class defines a JobConf data structure that is used to carry all of the configuration information about your program around. The wordcount example shows how a typical map /reduce program is written. Be warned, however, that the wordcount program is not usually run directly, but instead there is a single example driver program that provides a main method that then calls the wordcount main method itself. This added complexity decreases the number of jars involved in the example structure, but doesn't really serve any other purpose.

- Task - Whereas a job describes all of the inputs, outputs, classes and libraries used in a map/reduce program, a task is the program that executes the individual map and reduce steps. They are executed on TaskTracker nodes chosen by the JobTracker.
- HDFS - stands for Hadoop Distributed File System. This is how input and output files of Hadoop programs are normally stored. The major advantage of HDFS are that it provides very high input and output speeds. This is critical for good performance for highly parallel programs since as the number of processors involved in working on a problem increases, the overall demand for input data increases as does the overall rate that output is produced. HDFS provides very high bandwidth by storing chunks of files scattered throughout the Hadoop cluster. By clever choice of where individual tasks are run and because files are stored in multiple places, tasks are placed near their input data and output data is largely stored where it is created. An HDFS cluster is built from a NameNode and one or more DataNode instances.