Summary of “MapReduce: Simplified Data Processing on Large Clusters”

Hou, Jue

jue.hou@helsinki.fi

In the article “MapReduce: Simplified Data Processing on Large Clusters”, authors introduced a programming model called MapReduce. Aimed to handle large scale of raw input and get the information, the MapReduce is an abstraction for distributing computation. The whole abstraction can be divided into two parts: Map function and Reduce function. Map function will read from raw data and produce intermediate key/value pair. Reduce function will get its input from Map function and then take care of merging pairs and extracting the relevant information from it. And all the input will be split into tasks of the functions, and then assigned to different machine. About the implementation, the whole system is designed in hierarchical structure, which can be generally described as master and workers. Workers will finish their job according to the task they are assigned to and output the result to the local disk or the GFS. Meanwhile, besides assigning tasks, master also has to keep the consistency of the whole computation as well as maximize the efficiency. In order to do so, master will monitor both the workers and their tasks and keep records.

Overall, instead of describing the detail of the hardware and software, authors briefly but thoroughly introduced the idea of MapReduce and the way how it implements. And I think readers will have a bird’s eye view on MapReduce after their reading. I find there are some simple but effective implementation practices which are especially useful for me. Those practices can be applied to my future programming. Practice like backup tasks is also shown to be effective by authors through their evaluation of the performance.