Abstract

The performance of MapReduce greatly depends on its data splitting process which happens before the map phase. This is usually done using naive methods which are not at all optimal. In this paper, an Improved Input Splitting technology based on locality is explained which aims at addressing the input data splitting problems which affects the job performance seriously. Improved Input Splitting clusters data blocks from a same node into the same single partition, so that it is processed by one map task. This method avoids the time for slot reallocation and multiple tasks initializing. Experiment results demonstrated that this can improve the MapReduce processing performance largely than the traditional Hadoop implementation.
Improved Input Data Splitting in MapReduce

References

- Hadoop is released as source code tarballs with corresponding binary tarballs for convenience http://hadoop.apache.org/
- http://dailyhadoopsoup.blogspot.in/2014/02/mapreduce-inputs-and-splitting.html
- The paperwork for opening a business or getting unemployment http://www.openstack.org/
- http://www.revelytix.com/?q=content/hadoop-overview
- MarkLogic Connector for Hadoop Developer’s Guide http://docs.marklogic.com/hadoop/get-splits
- http://grepcode.com/file/repository.cloudera.com/content/repositories/releases/com.cloudera.hadoop.hadoop-core/0.20.2737/org/apache/hadoop/mapreduce/lib/input/FilesInputFormat.java
- Chunguang Wang; Qingbo Wu; Yusong Tan; Wenzhu Wang; Quanyuan Wu, "Locality Based Data Partitioning in MapReduce," in Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on, vol., no., pp. 1310,1317, 3-5
Dec. 2013

Index Terms

Computer Science
Information Sciences

Keywords
Hdfs  Improved Input Splitting  Mapreduce