

# Take Me to SSD: A Hybrid Block-Selection Method on HDFS based on Storage Type

Minkyung Kim  
Yonsei University  
50 Yonsei-ro, Seodaemun-gu  
Seoul, Korea  
+82 2 2123 7757  
goodgail@cs.yonsei.ac.kr

Mincheol Shin  
Yonsei University  
50 Yonsei-ro, Seodaemun-gu  
Seoul, Korea  
+82 2 2123 7757  
smanioso@cs.yonsei.ac.kr

Sanghyun Park<sup>1</sup>  
Yonsei University  
50 Yonsei-ro, Seodaemun-gu  
Seoul, Korea  
+82 2 2123 7757  
sanghyun@cs.yonsei.ac.kr

## ABSTRACT

As the era of Big-data has risen, the importance of big data technologies is also increasing day by day. Especially, Hadoop has become a critical part of the overall Big-data system because of its ability to store, process, and analyze thousands of terabytes of data. A major issue for supporting high performance on Hadoop is managing the growth of data while satisfying high storage I/O request. Hadoop's overall performance is largely influenced by the storage input/output(I/O). However, storage I/O technologies are still very limited. Therefore, now more than ever, studies on improving storage I/O on a distributed file system of Hadoop(HDFS) have been gaining popularity. To this end, latest trend in storage systems is to utilize hybrid storage devices. However, it is not easy to use the information of heterogeneous storage devices in HDFS. This is because, when reading data, HDFS is unable to exploit such heterogeneous storage type information yet.

In this paper, we propose a hybrid block-selection method on the HDFS, we consider the storage type such as SSD and HDD when reading data. Using this method, the Hadoop Eco System utilizes the high SSD bandwidth by priority. As a result, we certainly improve the Hadoop Eco System overall performance. In the experiments, we demonstrated that our new method efficiently reduced the execution time of select count(\*) query and TPCB benchmark up to 22% and 30% on average.

---

<sup>1</sup> Corresponding author. Tel: +82-2-2123-5714; Fax: +82-2-365-2579.; E-mail address: [sanghyun@cs.yonsei.ac.kr](mailto:sanghyun@cs.yonsei.ac.kr) (S.Park).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

SAC 2016, April 04-08, 2016, Pisa, Italy  
© 2016 ACM. ISBN 978-1-4503-3739-7/16/04...\$15.00  
DOI: <http://dx.doi.org/10.1145/2851613.2851658>

## CCS Concepts

• Information system → Database management system engines

## Keywords

Hadoop, SQL-on-Hadoop, SSD, HDD, Storage I/O

## 1. INTRODUCTION

Big-data is an evolving term for extremely large or complex data sets. It produces terabyte or petabyte quantities of data in a short period of time. As the data amount is increased, the need for big data analysis to extract meaningful information from a huge amount of data has increased significantly. However, traditional relational database management systems (RDMS) are inadequate for very-large data sets storage and processing. Consequently, studies on data store, access, manipulation and analysis specializing in Big-data have been actively carried out.

There are various applications that allow handling and processing of Big-Data, the basic framework has been that of Apache Hadoop. Hadoop comprises of two parts, which are the data storage part(HDFS) and the other being the data processing part(MapReduce). For data processing, MapReduce divides tasks and allows their execution in parallel. MapReduce is limited in that it is not suitable for real-time processing and does not support the SQL language. To this end, different methods of processing Big-Data, such as SQL-on-Hadoop, have been developed. Over the years various attempts have been made to improve Big-Data's real time processing performance on SQL-on-Hadoop with the combination of HDFS.

HDFS is a distributed file system of Hadoop. HDFS divide Big-data into small unit of data. The small data are replicated, distributed and stored in a large number of storage devices. Recently, HDFS optimization techniques have been proposed using SSDs with high I/O performance[1,8,13,16,19]. However, it is still difficult to replace all existing HDD devices with SSD devices. New solutions, that simultaneously use both SSDs and HDDs, are being suggested to solve the problem.

In this paper, we suggest a hybrid block selection method on HDFS based on storage type. We consider the type of storage device and select optimal location to read block. If the same block exists in both HDD and SSD among different candidate