이종 저장장치 환경에서의 하둡 분산 파일 시스템 성능 연구

이종백O 이상원
성균관대학교
hundredbag@skku.edu, swlee@skku.edu

Performance Evaluation of Hadoop Distributed File System with Heterogeneous Storage

Jongbaeg LeeO Sang-won Lee
Sungkyunkwan University

요 약

하둡 분산 파일시스템은 네임노드와 데이터노드로 구성되는 클러스터에 대용량의 데이터를 분산하여 저장한다. 하둡 분산 파일시스템은 범용 저장장치인 하드디스크의 사용을 가정하여 설계되었으나 다양한 저장장치 지원의 필요성이 증가함에 따라 이종의 저장장치로 클러스터를 구성할 수 있는 저장장치 정책 기능이 추가되었다. 이 기능을 통해 하둡 분산 파일시스템에서는 데이터 블록 복제본을 저장할 때 각 블록 복제본이 저장될 저장장치를 결정할 수 있다. 본 논문에서는 하둡에서 제공하는 저장장치 정책 중 All_SSD, Hot, One_SSD 정책을 사용하는 경우의 분산 파일시스템의 성능을 측정하며, One_SSD 정책을 사용할 때 성능 문제가 발생할 수 있음을 보인다.

1. 서 론

개인화 서비스와 소셜 서비스의 부상에 따라 대량의 데이터에 대한 효율적인 저장과 처리에 대한 필요성은 갈수록 증가하고 있다. 이에 따라 많은 양의 데이터를 다루기 위한 저장 시스템에 관하여 다양한 연구가 진행되었고, 그 예로 병렬 DBMS, 구글 파일 시스템, 하둡 등을 들 수 있다. 하둡[1]은 많은 양의 데이터의 저장과 처리를 제공하는 대표적인 기술로, 대량의 자료 처리를 위하여 여러 노드로 구성된 클러스터 환경에서 데이터를 분산하여 저장하고 처리하는 오픈소스 프레임워크이다.

하둡 분산 파일시스템(Hadoop Distributed File System, 이하 HDFS)은 하둡에서 데이터를 저장하기 위해 사용하는 프레임워크이다. HDFS는 여러 노드로 구성되는 클러스터 환경에 대용량의 데이터를 분산 저장하며, HDFS 클러스터는 파일 네임스페이스를 관리하는 네임노드와 실제로 데이터를 저장하는 역할의 데이터노드로 이루어지는 마스터-슬레이브 구조로 이루어진다. 하둡은 클러스터를 구성하는 노드가 범용 하드웨어로 구성됨을 가정하여 설계되었기 때문에 HDFS에서는 저장장치로 사용되는 하드웨어의 실패를 대비하여 데이터블록을 여러 데이터노드에 복제하여 저장하며, 블록의 크기를 128MB로 크게 설정하여 하드디스크의 데이터 탐색의 비중을 줄이는 특징을 가진다.

다양한 저장장치 사용의 요구에 따라 하드디스크의 사용만을 가정하던 하둡에서 데이터노드가 다양한 저장장치로 이루어진 환경을 지원하기 시작했다. 하드디스크 이외에 추가된 저장장치는 ARCHIVE, SSD, RAM_DISK이며 각 저장장치 특징을 활용하기 위하여 하둡에서는 저장장치 정책을 선정할 수 있는 기능을 추가로 제공한다. 저장장치 정책 기능은 HDFS 상의 파일이나 디렉토리가 저장 될 저장장치를 지정하기 위한 것으로 데이터 블록의 복제본이 데이터노드에 저장될 때 각 데이터블록이 어떤 장치에 쓰일 것인지를 결정하는 역할을 한다.

본 연구에서는 다양한 저장장치 정책을 사용하여 이종 저장장치 환경과 단일 저장장치 환경에서의 HDFS의 성능을 측정하고 비교한다. 또한 One_SSD 정책을 사용하는 경우 발생하는 문제와 그 원인에 대하여 분석한다.

본 논문은 다음과 같이 구성된다. 2장에서는 관련연구로 하둡 분산 파일 시스템과 저장장치 정책에 관하여 설명한다. 3장에서는 TestDFSIO 벤치마크를 수행하여 얻은 이종 저장장치 환경에서의 HDFS의 성능을 측정하고 평가한다. 마지막으로 4장에서는 결론과 향후연구로 본 논문을 마무리한다.

2. 관련 연구

2.1 하둡 분산 파일 시스템

HDFS는 하둡을 구성하는 코어 프로젝트 중 하나로, 범용 하드웨어로 구성된 클러스터에서 데이터를 분산하여 저장하고 관리하기 위해 설계된 파일시스템이다. HDFS는 마스터-슬레이브 구조로 이루어져있으며, 마스터 역할을 하는 네임노드와 슬레이브 역할을 하는 데이터노드 두 종류로 구성되는 클러스터 환경에서 동작한다. 네임노드는 파일시스템의 네임스페이스를 관리하고 클라이언트로부터의 데이터 접근을 통제하는 역할을 수행한다. 또한 주어진 파일에 대한 데이터 블록을 가지는 데이터노드의 정보를 관리한다. 데이터노드는 실제 데이터 블록을 저장하는 노드로, 클라이언트나 네임노드의 요청에 의해 데이터 블록을 저장 및 탐색하는 역할을 수행한다. 또한 저장하고 있는 데이터 블록에 대한 정보를 네임노드에 주기적으로 보고하여 네임노드가 데이터 블록의 위치정보를 알 수 있도록 한다. 데이터노드에서의 데이터 읽기/쓰기는 블록 단위로 이루어

지며, 블록은 기본 128MB로 디스크 블록에 비해 크게 설정된다. 이는 전체 데이터 전송 시간에서 디스크 탐색 시간의 비용을 줄이기 위함이다.

범용 하드웨어의 사용을 가정하여 설계된 하둡은 데이터의 내고장성을 보장하기 위해 하나의 데이터 블록을 여러 데이터 노드에 복제하여 저장하는 특징을 가진다. 블록 복제본의 수는 'Replication Factor'로 설정된 값에 따라 결정되며 기본적으로는 세 개의 복제본을 가진다. 이 때, HDFS는 세 개의 복제본을 위치시키는 방법으로 첫 번째 복제본은 로컬 랙의 노드에 저장하며, 두 번째 복제본은 첫 번째 복제본과 같은 랙의 다른 데이터노드에, 그리고 마지막 복제본은 다른 랙의 데이터노드에 저장하게 된다. 하나의 데이터 노드의 하드웨어 실패의 확률에 비해 랙의 하드웨어 실패의 확률은 매우 낮기 때문에 HDFS에서는 위의 방식으로 블록 복제본을 위치시킴으로써 분산 파일시스템의 내고장성을 제공한다.

2.2 HDFS의 이종 저장장치 정책

플래시 메모리 저장장치 사용의 증가, 자주 접근되지 않는 데이터를 위한 아카이브 저장소의 필요 등의 이유로 인하여 하둡의 2.6 버전부터 HDFS에서는 이종 저장장치 환경을 활용할 수 있는 기능이 추가적으로 제공된다. 이 기능에서 제공되는 저장장치는 ARCHIVE, DISK, SSD, RAM_DISK가 있다[2].

각각의 저장장치는 서로 구분되는 특징을 보이는데, 아카이브 저장소로 이용되는 저장장치는 데이터 접근속도가 느리지만 단위 가격 당 저장 용량이 크다는 것, SSD는 하드디스크에 비해 빠른 읽기/쓰기 성능을 보이며 특히 임의 데이터 접근 속도가 하드디스크에 비해 매우 빠른 특징을 보이는 것이 그 예이다. HDFS에서는 각 저장장치의 특징을 잘 활용하기 위하여 이종 저장장치 환경에서 데이터 복제본이 저장될 저장장치를 결정하는 여섯 가지의 정책을 제공하며, HDFS의 파일 또는 디렉토리 단위로 사용할 정책을 결정할 수 있다.

여섯 가지 정책 중 기본 값으로 설정되어있는 Hot 정책은 모든 데이터 복제본을 DISK에 저장하는 정책이다. Cold 정책은 모든 데이터 복제본을 ARCHIVE에 저장하는 정책으로 더 이상 이용되지 않는 데이터를 저장하기 위한 정책이다. Warm 정책은 하나의 복제본은 DISK에 저장하고 나머지 복제본은 ARCHIVE에 저장하는 정책으로, 자주 사용되는 데이터와 접근되지 않는 데이터가 함께 있을 경우를 위한 정책이다. Lazy_Persist는 하나의 복제본을 RAM_DISK에 저장하고 나머지 복제본을 DISK에 저장하는 정책이다. Lazy_Persist는 Replication Factor가 1일 때 효과적으로 동작하지만 2 이상일 때에는 DISK에 데이터 블록이 쓰이는 것을 기다려야하기 때문에 많은 성능 향상을 기대하기는 어렵다. All_SSD는 모든 복제본을 SSD에 저장하는 정책이며, One_SSD는 하나의 복제본을 SSD에 저장하고 나머지 복제본을 DISK에 저장하는 정책이다.

3. 이종 저장장치 환경에서의 HDFS 성능 연구

3.1 TestDFSIO

하둡에서 제공하는 TestDFSIO는 HDFS의 성능을 측정하는 데에 유용하게 사용되는 벤치마크로, HDFS에 읽기/쓰기를 수행한다. TestDFSIO는 HDFS의 동작에 있어 운영체제 설정이나 하둡의 클러스터 설정이 정상적인지 확인하거나 데이터 접근 시 네트워크 병목의 발생 여부를 판단하는 등의 다양한 목적으로 사용이 가능하지만, 가장 주된 역할은 I/O 관점에서 분산 파일 시스템의 성능을 측정하는 것이다.

TestDFSIO 벤치마크는 읽기, 쓰기를 수행하는 하나의 파일 당 하나의 맵 태스크를 수행하며, 읽거나 쓰기를 수행할 파일의 수는 벤치마크 수행 명령 시 설정이 가능하다. TestDFSIO 벤치마크에서는 각각의 맵 태스크를 통해 읽기/쓰기를 수행하고 해당 작업의 수행 결과 정보를 생성하며, 맵을 통해 생성된 결과 정보는 하나의 리듀스 태스크를 통해 하나로 합쳐진다. TestDFSIO는 벤치마크 수행의 결과로 각 태스크에 대한 Throughput mb/sec, 평균 IO rate mb/sec, 총 수행시간 등의 유용한 정보를 제공한다.

3.2 성능평가 환경

TestDFSIO 벤치마크의 성능 평가를 위해 사용된 데이터는 1GB 크기의 128개의 파일로, 총 128GB에 대한 읽기/쓰기 테스트가 수행되었다. 벤치마크 성능 평가를 위한 HDFS 클러스터 환경은 표 1과 같다. 하둡 클러스터는 3개의 데이터노드로 구성되며 각 노드는 10Gbit/s 네트워크로 연결되었다. HDFS는 데이터 블록의 크기를 128MB로 사용하도록 설정되었으며, Replication Factor는 기본 값인 3으로 설정되었다.

각 노드에서 데이터를 저장하기 위해 삼성 SSD 850 Pro 256GB, WDC WD10EZEX 7200RPM 하드디스크가 사용되었다. All_SSD 정책을 사용하는 경우 2개의 SSD를, Hot 정책을 사용하는 경우 2개의 하드디스크를, One_SSD 정책을 사용하는 경우 1개의 SSD와 1개의 하드디스크를 사용하였다.

3.3 성능 평가 및 분석

그림 1은 All_SSD, Hot, One_SSD 각각의 저장장치 정책을 사용하는 HDFS에 대하여 TestDFSIO 벤치마크를 통해 측정한 분산 파일 시스템의 읽기/쓰기 Throughput을 나타낸다. 측정 결과 SSD 2개를 이용하는 All_SSD 정책의 읽기/쓰기 Throughput은 각각 1935MB/s, 835MB/s로 높은 성능을 보이는 것을 확인할 수 있다. 하드디스크 2개를 이용하는 Hot 정책의 읽기/쓰기 Throughput은 각각 377MB/s, 248MB/s로 측정되었다. 마지막으로 하드디스크 1개와 SSD 1개를 이용하는 One_SSD 정책의 읽기/쓰기 Throughput은 각각 271MB/s, 202MB/s로 측정되었다.

벤치마크의 수행 결과 세 경우 모두에서 쓰기보다 읽기의 성능이 높게 측정이 되는 것은 HDFS에 쓰기를 수행하는 경우

표 1 성능평가 환경 설정

데이터노드	설명
CPU	Intel Xeon E5-2670V3 2.3GHz 24 Core (48 Thread)
메모리	64GB
저장장치	삼성 SSD 850 Pro 256GB * 2 WDC WD10EZEX 7200RPM * 2
복제본 수	3
HDFS 블록 크기	128MB

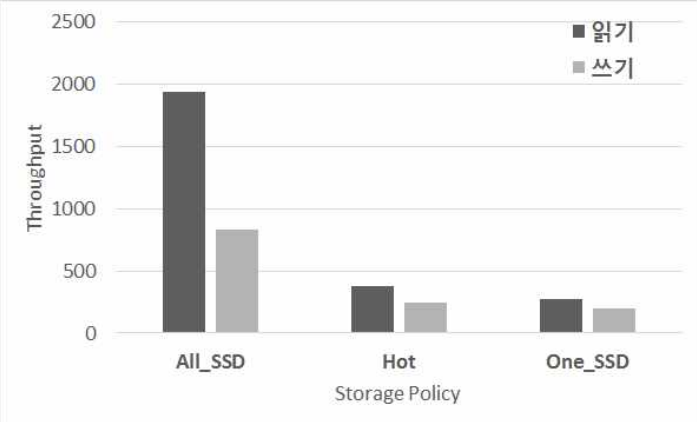

그림 1 이종 저장장치 정책에 따른 HDFS의 Throughput

여러 개의 복제본을 생성하지만 읽기를 수행하는 경우 여러 복제본으로부터 동시에 읽기를 수행할 수 있기 때문이다. 또한 결과에서 All_SSD 정책을 사용할 때가 Hot 정책을 사용할 때보다 크게 좋은 성능을 보이는 것은 SSD와 하드디스크의 성능 차이가 원인임을 알 수 있다.

또한 측정 결과 One_SSD 정책을 사용할 때 Hot 정책보다 느린 성능을 얻은 것을 볼 수 있다. HDFS에서 데이터를 읽을 때 저장장치를 고려하지 않고 네트워크상의 거리만을 기준으로 데이터를 읽어올 노드를 선정하기 때문에, 데이터 접근 과정에서 느린 하드디스크가 병목이 되어 또 다른 저장장치인 SSD의 속도가 빠름에도 이를 충분히 활용할 수 없는 문제를 일으키는 것이 느린 성능의 원인임을 확인할 수 있다.

4. 결론

본 논문에서는 이종 저장장치로 구성된 HDFS에서 저장장치 정책으로 All_SSD, Hot, One_SSD를 사용하는 경우에 대하여 성능을 측정하고 평가하였다. 성능 측정 결과 저장장치로 SSD를 사용하는 All_SSD 정책이 가장 빠른 성능을 보였으며, 하나의 SSD와 하나의 하드디스크를 사용하는 One_SSD 정책의 성능이 가장 좋지 않음을 보였다. 또한 One_SSD 정책에서 가장 느린 성능을 보이는 원인이 HDFS에서 데이터 접근 시 네트워크 거리만을 기준으로 데이터노드를 선택하기 때문에 느린 하드디스크가 병목을 일으키는 것임을 보였다.

향후 연구에서는 One_SSD에서 발생하는 하드디스크 병목 문제를 해결할 수 있는 방법에 관하여 연구를 진행할 것이다.

사사

이 논문은 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (R0126-16-1108, 비휘발성 메모리 기반 개방형 고성능 DBMS 개발)

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (10041244, 스마트TV 2.0 소프트웨어 플랫폼)

참 고 문 헌

[1] Apache Hadoop. [Online]. Available: http://hadoop.apache.org/.

[2] Archival Storage, SSD & Memory. [Online]. Available: https://hadoop.apache.org/docs/r2.6.0/hadoop-project-dist/hadoop-hdfs/ArchivalStorage.html.

[3] S.H. Kang, D.H. Koo, W.H. Kang, and S.W. Lee. "A case for flash memory ssd in hadoop applications". International Journal of Control and Automation, 6(1), 2013.

[4] K. Kambatla and Y. Chen. "The truth about mapreduce performance on ssds". In Proc. USENIX LISA, 2014.

[5] J. Lee, S. Moon, Y. suk Kee, and B. Brennan. "Introducing SSDs to the Hadoop MapReduce Framework", In NonVolatile Memories Workshop 2014.

[6] Krish K.R., A. Anwar, and A. R. Butt. "hatS: A heterogeneity-aware tiered storage for Hadoop". In Proceedings of IEEE/ACM CCGrid 2014.