옵테인 메모리 기반 OpenCAS 캐시 가속화를 활용한 올플래시 DataPond 클러스터 시험 구축

김호, 신준식, 김종원*

광주과학기술원

{ho.kim, jsshin, jongwon*}@smartx.kr

Prototype Implementation of All-flash DataPond Cluster employing OpenCAS Cache Acceleration with Optane Memory

Ho Kim, Jun-Sik Shin, and JongWon Kim*

Gwangju Institute of Science and Technology (GIST)

요 약

본 논문에서는 대량의 데이터를 유연하고 효율적으로 처리하는 클라우드-네이티브 방식의 하나인 Connected DataLake 개념과 이를 위한 DataPond 클러스터의 개념을 제안하고, 그중 DataPond 클러스터의 개념을 실체화하기 위해 옵테인 메모리에 기반한 OpenCAS 올플래시 캐시 가속화를 활용한 올플래시 DataPond 클러스터를 시험적으로 구축한다.

Ⅰ. 서 론

객체 인식, 자연어 처리, 자율주행 등의 지능형 서비스에는 데이터 처리 작업이 빠를수록 서비스가 데이터를 대기하는 데에 걸리는 지연 시간을 단축할 수 있으므로 데이터 처리 성능을 더욱 고속화할 필요성이 제기되고 있다[1]. 본 논문에서는 대량의 데이터를 유연하고 효율적으로 처리하는 Cloud-native Connected DataLake 개념과, 그중 종단 기기들(End-Boxes)로부터 수집한 데이터를 중앙의 DataLake로 전달하는 역할을 맡는 DataPond 클러스터의 개념을 제안한다. 그리고, 제안한 DataPond 클러스터의 개념을 실체화하기 위해 옵테인 메모리에 기반한 OpenCAS 캐시 가속화를 활용한 올플래시 DataPond 클러스터를 시험적으로 구축하고, 시험 구축한 DataPond 클러스터의 데이터 처리 성능을 측정하여 Cloud-native Connected DataLake 에서의 활용 잠재성을 평가한다.

Ⅱ. Cloud-native Connected DataLake 및 DataPond 클러스터 개념

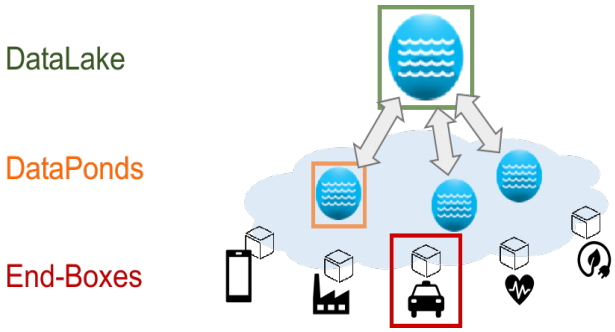

그림 1. Cloud-native Connected DataLake 및 DataPond의 개념도

DataLake는 다수의 종단 기기들로부터 비정형 데이터를 저장하고 처리하는데, 이를 최대한 자원 효율적으로 처리하기 위해서는 데이터를 최대한 신속하고 안정적으로 수신해야 할 필요가 있다[1]. 이는 Cloud-native Connected DataLake의 개념을 도입하여 해결할 수 있다. Cloud-native Connected DataLake란 하나의 크고 중앙집중화된 DataLake와 여러개의 작고 분산된 데이터 저장소인 "DataPond"로 구성할 수 있는데, 여기서 각 종단 기기는 각자에게 가장 가까운 DataPond로 데이터를 전송하고, DataPond는 수집한 데이터를 중앙의 DataLake로 전달하는 역할을 맡는다. 이를 단일한 DataLake를 두었을 때와 비교하면 종단 기기와 데이터 저장소 간의 거리가 가까워졌기 때문에 데이터 전송 속도가 개선될 수 있고, 복수개의 DataPond를 두기 때문에 한 DataPond에 고장이 발생해도 이를 대체할 여러개의 DataPond가 있기에 비교적 안정적이다.

DataPond는 각자의 데이터 처리 성능이 우수할수록 Cloud-native Connected DataLake 에서의 작업 수행 시간을 단축할 수 있다. 이러한 이유로, DataPond 클러스터 내의 스토리지를 모두 HDD가 아닌 플래시 메모리로 구성할 수 있는데, 이를 올플래시 DataPond 클러스터라 하겠다. 플래시 메모리는 HDD에 비해 최대 10배의 4K 랜덤 읽기, 최대 20배의 4K 랜덤 쓰기 작업 성능을 점하고 있는 것으로 알려져 있기에, 올플래시 DataPond를 채택하는 것은 HDD를 사용하는 것에 비교하여 데이터 처리 성능의 관점에서 적합하다고 할 수 있다[2, 3].

또한, SCM(storage class memory)과 OpenCAS를 활용하여 플래시 메모리의 데이터 처리 성능을 향상시킬 수 있다. OpenCAS는 고성능/소용량 스토리지를 캐시(cache)로 활용하여 저성능/대용량 스토리지에 대한 데이터 읽기/쓰기 성능을 향상시키는 소프트웨어 라이브러리, 어댑터, 도구를 통합 제공하는 오픈소스 프로젝트이다[4]. 이들을 올플래시 DataPond에서는 SCM을 플래시 메모리의 캐시로 활용함으로써 데이터 처리 성능을 향상시킬 수 있다.

Ⅲ. 올플래시 DataPond 클러스터의 시험 구축

본 절에서는 DataPond의 개념을 실체화하기 위해 그림 2 와 같이 올플래시 DataPond 클러스터를 시험적으로 구축한다. 제안하는 클러스터의 구성은 크게 하드웨어 구성, OpenCAS를 이용한 캐시 가속화, 그리고 쿠버네티스 클러스터 구축의 3가지 측면으로 나누어 볼 수 있다.

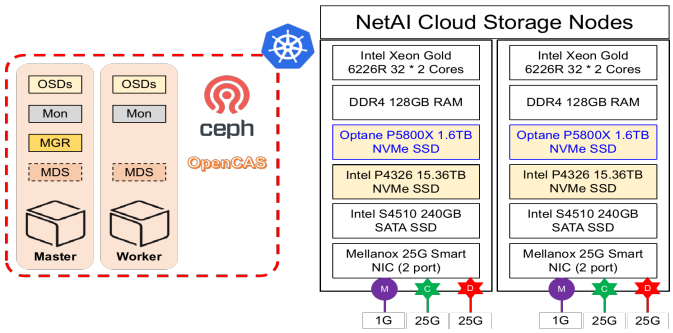

그림 2. 옵테인 메모리 기반 OpenCAS 캐시 가속화를 활용한 시험적인 올플래시 DataPond 클러스터

클러스터는 그림 2 의 우측과 같이 1개의 마스터 노드와 1개의 작업 노드로 하드웨어를 구성하였다. 각 노드는 Intel(R) Xeon(R) Gold 6226R CPU와 128GB의 메모리로 구성하였고, 두 노드 간의 데이터 통신을 위해 대역폭이 25G인 데이터 플레인 네트워크를 구성하였다. 또한, 각 노드에 플래시 메모리인 Intel(R) P4326 15.36TB NVMe SSD와 SCM인 Intel(R) Optane P5800X 1.6TB 옵테인 메모리를 장착해 스토리지를 구성하였다. 특히, 옵테인 메모리는 최근 Intel 사에서 선보인 플래시 메모리의 일종으로, 기존의 플래시 메모리보다 상대적으로 가격이 비싼 대신 약 136.4배의 쓰기 작업 성능을 점하고 있는 것으로 알려져 있다[3].

상기한 스토리지의 성능 가속화를 위해서는 OpenCAS를 이용하였다. 제안한 클러스터에서는 OpenCAS를 활용해 옵테인 메모리가 다른 플래시 메모리의 캐시 역할을 맡도록 하였다. 이로 인해 전반적인 스토리지의 입출력 성능 향상을 기대할 수 있으며, OpenCAS의 캐시 가속화가 적용된 가상의 디바이스를 획득할 수 있다.

마지막으로, 상기한 OpenCAS 캐시 가속화가 적용된 DataPond의 하드웨어 구성을 바탕으로 그림 2 의 좌측과 같은 쿠버네티스 클러스터를 구축하였다. 클러스터 스토리지의 원활한 관리를 위해 분산 저장 스토리지인 Ceph를 활용해 스토리지 자원들을 클러스터 단위로 통합하여 관리할 수 있는데, 이는 각 노드에서 Ceph 스토리지로 가용 가능한 자원들을 자동으로 탐색할 수 있는 쿠버네티스 서비스인 Rook을 활용해 구현하였다[5]. 이때, 각 노드의 OpenCAS 캐시 가속화가 적용된 가상의 디바이스가 Rook 서비스에 의해 OSD(object storage daemon)의 형태로 분할되어 가용될 수 있도록 하였다.

Ⅳ. 제안한 DataPond 클러스터의 성능 평가

Number of RBDs	20
Queue Depth	16
Duration	600 seconds
RBD volume size	64 GB

표 1. VdBench 파라미터 및 네트워크 환경

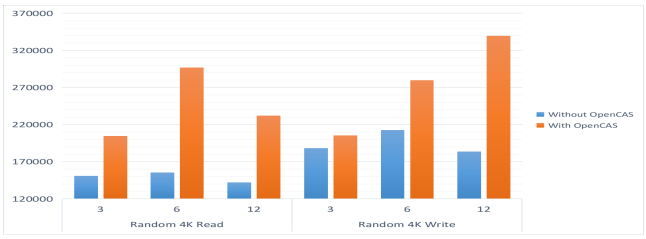

그림 3. 데이터 처리 성능 평가 실험 결과

옵테인 메모리에 기반한 올플래시 DataPond 클러스터에서 OpenCAS의 데이터 처리 성능 가속화 효과를 측정하기 위해, 본 논문에서 제안하는 DataPond 클러스터를 구성할 때에 OpenCAS를 사용한 경우와 그렇지 않은 경우의 초당 입출력 횟수(iops)를 측정하였다. 이때, OpenCAS를 사용하지 않는 경우의 클러스터를 구성할 때는 Ceph의 메타데이터 스토리지로 옵테인 메모리를 사용하였다. 성능 측정을 위해서는 스토리지에 입출력 부하를 주어 성능과 데이터 무결성을 검증하는 도구인 VdBench[6]를 이용하였다. 실험에 사용된 VdBench 파라미터는 표 1 과 같다.

실험의 결과는 그림 3 과 같다. 여기서 축의 'Random 4K Read'는 4K 랜덤 읽기, 'Random 4K Write'는 4K 랜덤 쓰기 작업의 초당 입출력 횟수를 의미하고, 바로 위의 숫자 3, 6, 12는 작업에 사용된 OSD의 수를 의미한다. 실험 결과, 사용된 OSD의 수가 6개일 때에 OpenCAS를 사용한 경우는 다른 경우와 비교하였을 때 읽기 작업의 초당 입출력 횟수가 가장 높았고, OSD의 수가 12개일 때에는 OpenCAS를 사용한 경우는 다른 경우와 비교하였을 때 쓰기 작업의 초당 입출력 횟수가 가장 높았다. 초당 입출력 횟수가 많을수록 단위 시간에 처리할 수 있는 데이터의 양이 많고, 따라서 데이터 처리 성능이 우수하다고 할 수 있다. 따라서, Connected DataLake의 올플래시 DataPond 클러스터를 구축할 때에 옵테인 메모리에 기반한 OpenCAS를 사용하는 것은 스토리지의 데이터 처리 성능을 개선할 수 있으므로 더욱 효율적임을 알 수 있다.

Ⅴ. 결론

본 논문에서는 Connected DataLake 및 DataPond 클러스터의 개념을 소개하고, 그중 DataPond 클러스터를 실체화하기 위해 옵테인 메모리에 기반한 OpenCAS 캐시 가속화를 활용한 올플래시 DataPond 클러스터를 시험적으로 구축하였다. 특히, 올플래시 DataPond 클러스터를 구축할 때에 옵테인 메모리에 기반한 OpenCAS 캐시 가속화를 활용하는 것이 더욱 효율적임을 밝혔다. 다만, 본 연구의 Connected DataLake를 구현하는 데에 사용된 25G의 네트워크 대역폭이 DataPond 클러스터의 입출력 작업 성능을 충분히 수용하지 못하는 것으로 보이므로, 네트워크 대역폭을 100G 이상으로 보완하는 등 이를 보완할 후속적인 연구가 필요하다.

ACKNOWLEDGMENT

본 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2021-0-01176, 클라우드 기반 융합형 자율주행 지능학습 데이터 생성/제공을 위한 데이터 수집 가공 핵심기술 개발). 또한, Intel 및 DS&G 사의 장비 지원에 감사드립니다.

참 고 문 헌

[1] N. Miloslavskaya and A. Tolstoy, “Big Data, Fast Data and Data Lake Concepts,” Procedia Computer Science, vol. 88, pp. 300-305.

[2] M. Oh, J. Eom, J. Yoon, J. Y. Yun, S. Kim, and H. Y. Yeom, “Performance Optimization for All Flash Scale-Out Storage,” in 2016 IEEE International Conference on Cluster Computing (CLUSTER).

[3] Intel Product Performance, "https://www.intel.com/content/www/us/en/products/performance/overview.html", 2021.

[4] OpenCAS, "https://open-cas.github.io/", 2021.

[5] Rook, "https://rook.io/", 2021.

[6] VdBench, "https://www.oracle.com/downloads/server-storage/vdbench-downloads.html", 2021.