

Ceph 스토리지 플랫폼 기반 항만 BIM 데이터 저장소 설계 및 구현

현근주*, 문현석*

Design and Implementation of The Data Storage for Port BIM Based on Ceph Storage Platform

Keun-Ju Hyun*, Hyoun-Seok Moon*

요약

항만 BIM 플랫폼의 클라우드 서비스를 위한 IaaS+PaaS 통합 환경에서 데이터 저장소는 필수적으로 구축되어야 한다. 스토리지를 도입함으로써 항만 BIM 플랫폼에서 생산 및 관리되는 모든 데이터를 효율적으로 자동 관리하며, 본 플랫폼내에서 라이브러리 업로드를 위한 스토리지로 구축되어 에러 자가 감지/치유, 대규모 성능확장, 모니터링, 보안 및 데이터 보호 기능, 유의미한 데이터 인사이트 제공 등의 기능을 보장한다. 스토리지와 함께 항만 BIM 기반기술을 클라우드에 적용하여 서버, 스토리지, 네트워크 등 ICT 자원을 필요한 만큼 탄력적으로 제공할 수 있는 기반을 제공한다. 본 논문에서는 Ceph 스토리지 플랫폼 기반으로 시스템을 구축하였으며, 크게 3가지 방식인 블록 스토리지, 파일 스토리지, 오브젝트 스토리지 방식으로 구현하였다. 이 Ceph 스토리지는 간단하고 대규모 확장이 가능한 기술로서 항만 BIM 플랫폼 클라우드 서비스를 안정적으로 운영할 수 있을 것으로 기대된다.

Key Words : BIM, Cloud Computing, Cloud Storage, Ceph Storage, Block/File/Object Storage

ABSTRACT

In the integrated IaaS+PaaS environment for cloud services of the port BIM platform, data storage must be built. With the introduction of storage, all data produced and managed on the port BIM platform is efficiently and automatically managed, and it is built as a storage for library upload within this platform, so that error self-detection/healing, large-scale performance expansion, monitoring, security and data protection functions, It ensures functions such as providing meaningful data insights. Along with storage, port BIM-based technology is applied to the cloud to provide a foundation to flexibly provide ICT resources such as servers, storage, and networks as needed. In this paper, the system was built based on the Ceph storage platform, and was implemented in three main ways: block storage, file storage, and object storage. This Ceph storage is a simple and massively expandable technology that is expected to stably operate the port BIM platform cloud service.

※ 본 논문은 2020년 해양수산부 재원으로 한국해양과학기술진흥원의 지원을 받아 수행된 연구(생애주기별 항만시설 통합 운영관리를 위한 BIM 기반기술 개발)입니다.

◆ First Author : INJE INC, Cloud Development Team, Manager, kjhyun@injeinc.co.kr, 정희원

* Corresponding Author : Korea Institute of Civil Engineering and Building Technology, Korea BIM Research Center, hsmoon@kict.re.kr, 정희원

논문번호 : 202010-246-D-RU, Received October 4, 2020; Revised October 29, 2020; Accepted October 30, 2020

I. 서 론

1.1 연구의 목적 및 필요성

BIM (Building Information Modeling, 이하 BIM)은 국가 기간 토목·건설 분야 프로젝트의 생애주기 동안 3D 기반 기술과 정보 모델을 활용하여, 프로젝트와 관련된 정보를 프로젝트 참여자들 간에 생성, 교환, 활용하는 행위 또는 기술을 말한다.^[19,20] 최근의 BIM 구축 환경은 상호 협업을 통해 설계 및 시공을 효율화하며, 정보의 통합관리와 대용량 BIM 데이터 처리를 위해 클라우드 환경의 구축이 필수적이다. 최근 국내외의 BIM 서버, 데이터관리, 프로세스 처리, 프로젝트 관리, 설계 협업 등 BIM 서비스 기업들은 온라인으로 관련 업무지원을 위한 소프트웨어 서비스를 제공하고 있다.^[15]

클라우드 기반 환경으로의 전환을 포함하여 이러한 노력에도 불구하고 국내 BIM 기술이 내포하고 있는 문제점을 살펴보면 향만 BIM 관련 소프트웨어의 수급, 고사양·고비용의 컴퓨터 및 서버, 스토리지 관리 문제 등과 같은 인프라 구축 관련 문제를 들 수 있다. 이 중에서도 BIM 스토리지 인프라 구축을 위한 투자비용, BIM 스토리지 전문가의 부족, 협력업체 및 운영조직의 부족, 현업에 적용하기 위한 현재의 불충분한 기술 수준 등을 들 수 있다.^[11]

BIM으로 작성한 파일이 대다수 대용량이기 때문에, 이를 클라우드 상에서 효율적으로 제공 및 관리하기 위한 기술이 필요하다. 예를 들어, 일반적으로 BIM으로 구현된 구조물들은 설계, 시공 및 유지관리 단계 정보까지 담고 있기 때문에 엄청난 크기의 파일을 생산하고 있으며, 대표적인 표준포맷인 IFC (Industry Foundation Classes)에서도 객체 기반의 형상정보 및 속성정보를 포함하고 있어 대용량 처리에 대한 이슈가 발생하고 있다.^[25] 또한, BIM 대용량 데이터와 관련하여 이를 경량화하는 연구들이 진행되고 있다.^[26]

국가 기간산업의 여러 BIM 기술이 내포하고 있는 이러한 스토리지 분야 문제들은 향만 BIM 기술발전에도 많은 저해요인이 되고 있으며 BIM 자원을 통합하여 일괄 구축 및 공동 활용 관리할 수 있는 체계가 무엇보다 절실하다.^[1,7]

클라우드 환경 기반 BIM 구축을 위해서는 클라우드 필수 요소 기술인 스토리지 기술을 반드시 도입해야 한다. 클라우드에서의 스토리지 기술은 클라우드 컴퓨팅의 한 분야로서, 인터넷 상의 '어딘가'에 위치하는 저장 공간에 BIM 정보를 보관해 두고, 이것을 시스템이 필요할 때 시스템과 접속하고자 하는 각종

단말기를 통해서 불러올 수 있는 기술이다.^[22] 각종 클라우드 컴퓨팅 기술 중에 반드시 구축해야 하는 필수적 서비스 중 하나이다.

Figure 1은 본 연구에서 구축한 클라우드 IaaS와 PaaS를 통합한 향만 BIM 정보통합 플랫폼 전체 개념도 중 스토리지 요소 부분 일부를 보여 주고 있다. 향만 BIM에 이미 클라우드 컴퓨팅이 도입되어 스토리지가 필수적으로 적용되고 있다. 그러나 클라우드 구성을 위한 인프라 측면의 단순 기능확장 구성이라면 성능이나 용량에 많은 영향을 미칠 것이다.

대용량 BIM 데이터는 모델의 렌더링 속도를 저하시키는 등 BIM 업무 생산성을 저하시키는 요인이다.

따라서 본 연구에서는 클라우드 스토리지 기능을 도출하고, Figure 1에서 제안한 향만 BIM 정보통합플랫폼의 IaaS+PaaS 환경 구축에서 대용량 BIM 데이터를 효율적으로 처리할 수 있는 방안으로 오픈소스 소프트웨어 Ceph 스토리지 플랫폼을 구축하여 BIM 데이터 저장소를 설계 및 구현하였다. 또한, 별도 상용제품의 DAS(Direct Attached Storage, NAS(Network Attached Storage))를 구축하지 않고

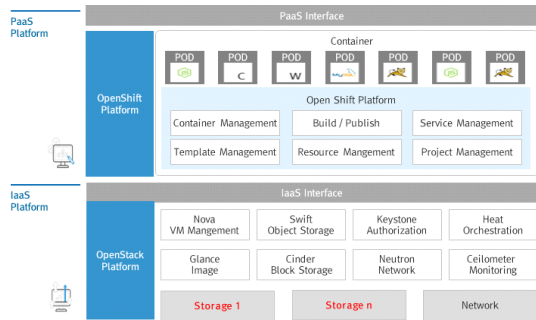


그림 1. 스토리지 기반 향만 BIM 클라우드 플랫폼
Fig. 1. Port BIM Cloud Platform with Storage

표 1. BIM을 위한 Ceph 스토리지 도입 효과
Table 1. Effect of Ceph Storage for BIM

항목	상세 설명 및 예시
기술적 효과	<ul style="list-style-type: none"> BIM에 최적화된 스토리지 환경 구성 BIM 프로젝트별 요구사항 변화에 따라 자유롭게 블록/파일/오브젝트 스토리지 방식 구현 및 확장 다운 타임 없는 스토리지 클러스터 확장/축소 BIM 시스템 전체 기능 구현 시간 단축 타 시스템에 비해 설치 과정 및 시간 1/3 축소 BIM의 방대한 비정형 데이터 관리
사회 경제적 및 기타 이득	<ul style="list-style-type: none"> 국가 기간산업으로써의 BIM 기술 성장 저비용의 인프라 구축 투자비용 고도의 전문가 불필요 협업체계 및 조직 구성의 용이성 확장 및 보급 등 SDS(Software Defined Storage) 도입의 편리성

Ceph 스토리지 플랫폼 기반으로 프로젝트 목적에 따라 블록(Block) 스토리지, 파일(File) 스토리지, 오브젝트(Object) 스토리지 저장방식으로 서비스를 제공하였다. Ceph 스토리지 플랫폼의 성능을 검토하기 위해, 타 상용제품과의 데이터 처리량을 비교하였으며, 처리량 성능 개선을 위해 GPU(Graphical Processing Unit)를 적용하였으며, 이에 따른 렌더링, 로딩 시간 및 처리량 성능을 검증하였다.

타 스토리지에 비해 Ceph 스토리지 플랫폼을 도입함으로써 Table1 과 같은 효과를 보장할 수 있다.

II. 클라우드 스토리지 관련 연구

2.1 클라우드 스토리지의 개요

Amazon, EMC, IBM, HP와 NetApp 등 많은 클라우드 서비스 공급자들이 클라우드 스토리지 서비스를 하고 있다. 또한 HDFS(Hadoop Distributed File System), GFS(Gluster File System, Amazon S3, EMC Atoms, Data ONTAP, HP Upline, Cloud NAS, FileStore 처럼 클라우드 스토리지 플랫폼들이 많이 증가하고 있는 추세이다. 이러한 다양한 클라우드 스토리지 플랫폼을 표준화하기 위해 SNIA(Storage Network Industry Association)에서는 2009년 클라우드 사용자, 서비스 공급자, 개발자 등 140개 이상의 기업들과 함께 클라우드 서비스 표준으로 CDMI(Cloud Data Management Interface)을 채택했다.^[5]

2.2 클라우드 스토리지와 클라우드 컴퓨팅 서비스와의 연관성

클라우드 스토리지는 클라우드 컴퓨팅 환경을 기반으로 하는 클라우드 서비스의 계층적인 구조와 동일한 구조로써 Figure 2와 같이 구성된다.

SaaS 클라우드 서비스는 클라우드 사용자가 인터넷을 이용하여 필요한 소프트웨어를 이용하는 응용 소프트웨어 계층이고, PaaS는 API interface와 스토리지 관리, 스토리지 가상화로 구성된 계층이며, IaaS는 서버와 하드웨어 자원과 서버 클러스터 기술을 제공하는 서비스이다.^[6]

DaaS는 클라우드 스토리지 시스템의 구조로써 데이터 스토리지로 구성되어진 스토리지 서비스 인터페이스에서 데이터 스토리지를 제공하고 추출, 전송함으로써 기존 클라우드 사용자에게 다양한 서비스 기능을 제공한다. 그리고 CIFS(Common interest File System), NFS(Network File System), WebDAV

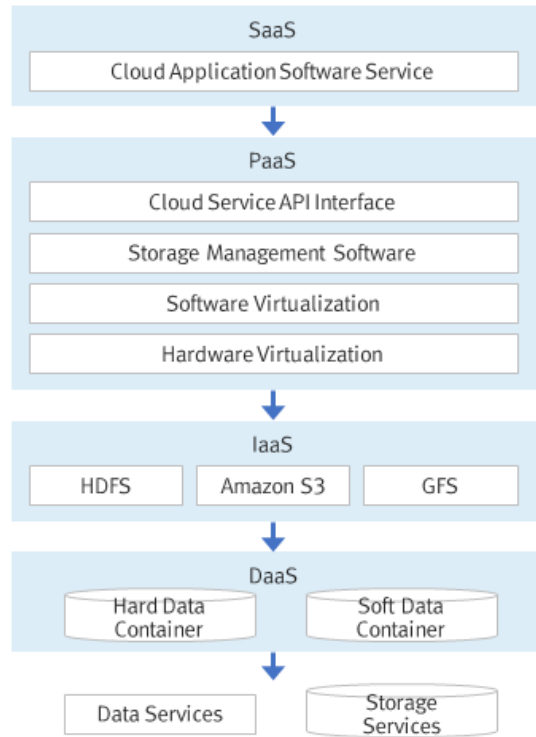


그림 2. 클라우드 스토리지 시스템 구조
Fig. 2. Structure of Cloud Storage System

(Web Distributed Authoring and Versioning) 같은 기존 표준 프로토콜을 이용하기도 한다.^[10]

클라우드 스토리지는 스토리지 미들웨어 시스템과 분산 파일시스템에 네트워크로 연결된 클러스터화로 많은 n대의 스토리지 장치들로 구성되어 사용자들에게 클라우드 서비스를 제공함과 동시에 SLA(Service Level Agreement), 분산파일시스템, 리소스 자원, 네트워크 장치들을 포함하고 있는 구조이다. 사용자에게 더 많은 호환기능 및 상호 작용을 제공하기 위해서 물리적인 기능과 논리적인 기능을 제공해야 한다.^[8,9]

2.3 클라우드 스토리지의 기술적 개요

클라우드 스토리지는 초기의 스토리지 기술인 NAS(Network Attached Storage)나 SAN(Storage Area Network)을 사용하였다. 이들은 스토리지 가상화 기술 등을 이용하여 스토리지 자원을 사용자의 요구에 따라 대규모 확장이 가능해야 하고, 특정 지리적 위치에 구애받지 않고 애플리케이션에 유연한 특징을 보유해야 한다. 특히 대규모의 확장이 가능해야 하는데, 전통적인 NAS 구조를 여러 대의 서버에서 동시에 실행되는 분산 파일 시스템을 이용한 클러스터-NAS(Clustered NAS)를 대표적인 예로 들 수 있다.

클러스터-NAS는 클러스터 노드나 기억장치에 데이터와 메타데이터(Metadata)를 분산시키는 것이고, 이때 사용되는 핵심 기술이 분산 파일시스템이다. 즉, 현재 클라우드 스토리지 기술의 핵심은 분산 파일 시스템이라 볼 수 있다.^[2,3]

클라우드 스토리지 관련 기술은 공개 소프트웨어와 상용 소프트웨어로 분류할 수 있는데, 공개 기술로는 하둡(Hadoop), 글러스터 FS(GlusterFS), 스위프트(Swift, Open Stack Object Storage) 등이 있고 상용 소프트웨어에는 IBM의 Scale-out File Services(SoFS)를 비롯하여 Microsoft, EMC, HDS(Hitachi Data System) 등이 있는데, 선형적인 확장을 지원하는 Scale-out 방식과 유니파이드 스토리지(Unified Storage) 방식으로 클라우드 스토리지 시장을 공략하고 있다.^[12]

2.4 클라우드 스토리지 핵심 기술

2.4.1 데이터 구성

클라우드 스토리지는 블록 스토리지, 파일 스토리지, 오브젝트 스토리지 등 세 개로 분류할 수 있다.

클라우드 스토리지의 파일 스토리지와 블록 스토리지를 통합 개발하여 파일 스토리지의 장점과 블록 스토리지의 장점의 서비스를 모두 지원하는 관련된 연구가 활발하게 이루어지고 있으며, EMC, NetAPP, IBM, HP 등에서 고확장형 유니파이드 스토리지 솔루션을 제공하고 있다.^[4,5,19]

표 2. 클라우드 스토리지 분류
Table 2. Classification of Cloud Storage

항목	상세 설명 및 예시
블록 스토리지	<ul style="list-style-type: none"> 클라우드 스토리지 노드와 같은 단일 스토리지 볼륨을 블록이라는 개별 인스턴스로 분할 엔터프라이즈 응용 소프트웨어에서 운영되는 가상화 환경에서 연동되는 서비스 고성능의 Read, Write I/O를 제공 고비용과 대용량 파일 스토리지에는 적합하지 않음. 빠르고 지연 시간이 짧은 스토리지 시스템으로서 고성능 워크로드에 적합
파일 스토리지	<ul style="list-style-type: none"> NAS 시스템에서 널리 사용 데이터를 체계화하여 사용자에게 표시 파일 레벨 스토리지에 기반을 두고 있음 계층구조로 되어 있어 데이터를 하향식으로 쉽게 탐색 웹 사이트, IPTV, 소셜, 앱 등의 대용량 파일들을 저장 가능 스토리지 노드의 제한 없는 확장이 가능 Read I/O에 집중하여 공유하는 파일 서비스에 적합 처리하는 데 시간이 오래 소요됨
오브젝트 스토리지	<ul style="list-style-type: none"> 데이터를 메타데이터라는 고유 식별자와 페어링함 오브젝트는 압축되거나 암호화되지 않으므로 대 규모로 신속하게 액세스 가능 클라우드 네이티브 애플리케이션에 적합함

선을 제공하고 있다.^[4,5,19]

2.4.2 스토리지 가상화

가상화 스토리지는 여러지역에 분산되어 있는 이기종 스토리지 들의 논리적인 볼륨 관리, 스토리지 관리, 다른 I/O 인터페이스로 스토리지 모니터링 및 배포와 아카이브를 통합하고 관리하기 위해서 스토리지를 가상화해서 스토리지 활용률을 높이고, 스토리지의 RAID-0 기술을 활용하여 높은 I/O 성능을 향상시킬 수 있으며, 장애가 발생하더라도 계속해서 서비스를 할 수 있다.^[23]

스토리지 가상화 구현에 크게 3가지 방법이 있는데, 호스트 기반, 가상화 기반, SAN 기반 등을 들 수 있다. 현재 추세로 SAN 기반에서 가상화 기술이 많이 개발되고 있으며 스토리지 중앙관리, 용량 등을 유연하게 관리할 수 있는 SAN 가상화를 많이 사용한다.

객체기반 파일시스템과 클러스터 분산 파일시스템은 대칭형(symmetrical)과 비대칭형(asymmetrical) 구조로 메타데이터와 데이터 통신 채널을 같이 사용하는지와 분리하여 사용하는지에 따라 구분할 수 있다.^[16]

2.4.3 데이터 복제

데이터 복제에는 호스트 기반 복제, SAN 기반 복제, 백업 시스템 복제가 있으며 정리하면 Table 3 과 같다.^[13]

데이터 복제 개념은 복제될 스토리지 자원인 소스

표 3. 데이터 복제 방식
Table 3. Data Copy

항목	상세 설명 및 예시
호스트 기반 복제	<ul style="list-style-type: none"> 응용 소프트웨어 데이터베이스 파일 시스템 볼륨레벨
SAN 기반 복제	<ul style="list-style-type: none"> 디스크 어레이 가상화 어플라이언스 캐시 어플라이언스 스토리지 스위치
백업 시스템 복제	<ul style="list-style-type: none"> 블록 레벨 증분 백업 CDP 중복제거

표 4. 동기화 체계
Table 4. Scheme of Synchronization

항목	상세 설명 및 예시
동기식 (Synchronous) 복제	소스 노드에서는 타겟 노드에서 데이터가 전송 되었는지 확인응답(Acknowledge)후 write I/O 수행
비동기식 (Asynchronous) 복제	소스 노드에서 write I/O를 한 후 타겟노드로 전송함으로써 동기식 복제보다 성능저하가 발생하지는 않음 치명적인 오류로 인해 데이터의 손실이 발생할 수 있다

와 복제 스트림을 수신할 스토리지 자원인 타겟으로 구성이 되며 동기화 방식에는 Table 4와 같이 무손실 방식인 동기식(Synchronous) 복제와 비동기식(Asynchronous) 복제로 나누어진다.^[13]

2.4.4 데이터 중복제거

중복제거는 중복된 파일을 탐지해 중복된 파일을 삭제하여 유일(unique)한 데이터를 시스템에 저장하는 기술로 스토리지 시스템 내에 중복 데이터를 감소함으로써 스토리지 공간의 활용과 대역폭을 높일 수 있다.

데이터 중복 제거는 증분 백업에서 큰 차이점을 가진다. 일반적인 증분 백업은 변경된 파일 전체를 백업하는 반면 데이터 중복 제거는 변경된 파일의 데이터를 중복 제거하여 백업을 수행한다. 따라서 데이터 중복 제거는 변경된 파일에는 중복된 데이터가 존재하지 않게 된다.^[14,24] 데이터 중복 제거의 핵심은 효율적인 데이터 스토리지 감소와 데이터 인스턴스 유지이다.^[12]

일반적인 데이터 중복 제거는 데이터 압축 기술을 사용하여 중복 데이터를 제거한다. 하지만 이는 데이터 블록 또는 파일이 위치한 동일한 스토리지 시스템에서 압축 알고리즘을 사용하여 중복 제거를 해야하는 단점이 있다.^[8,14] 데이터 중복 제거를 간단히 표현하면 Figure 3과 같다.

중복 제거 소프트웨어에서 생성된 백업 데이터 스트림 인덱스를 저장소에 저장하여 데이터 블록 유무를 판단하고 디지털 서명(fingerprint)을 사용하여 오버헤드 없이 중복된 데이터를 찾을 수 있도록 한다. 이때 중복된 데이터는 원본 데이터 세그먼트의 포인

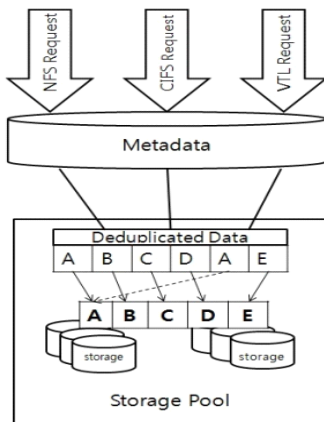


그림 3. 데이터 중복 및 중복 제거 구현 설계
Fig. 3. Design of Data Deduplication

터를 링크한다.

2.4.5 스토리지 보안

클라우드 스토리지는 네트워크를 통해 스토리지 서비스를 이용하기 때문에 네트워크를 통한 클라우드 사용자의 많은 데이터를 포함할 수 있어야 한다.

클라우드 특성상 가상화 센터와 클러스터 서버들은 하나의 지역에 국한되어있지 않고, 여러 지역으로 분산되어 있어 잘못된 접근으로 인해 개인 정보 보호 문제가 있기 때문에 반드시 클라우드 서비스 공급자로 접속을 해야 한다.^[15,11] 또한, 클라우드 네트워크 환경에서 실행되는 안전 위험 관리와 데이터 관리, 데이터 위치, 데이터 분리, 데이터 복구, 감사 및 서비스 연속성을 위해 관련된 업체에서 연구과제로 직면하고 있다.

III. Ceph 기반 항만 BIM 스토리지 설계 및 구현

3.1 Ceph 스토리지 플랫폼 개요

Ceph는 단일 분산 컴퓨터 스토리지 상에 오브젝트 스토리지를 구현한 오픈소스 스토리지 플랫폼(https://ceph.io/)이다. 블록, 파일, 오브젝트 레벨의 스토리지 인터페이스를 제공한다. 즉, Ceph는 Ceph

표 5. Ceph 특징 및 BIM 시스템 도입
Table 5. Introduction of Ceph and BIM System

항목	주요 특징 및 BIM 도입 내용
개념	<ul style="list-style-type: none"> • 데이터를 블록, 파일 및 객체 모드로 표시하는 일련의 게이트웨이 API가 있는 RADOS(Reliable Autonomic Distributed Object Store) 라는 객체 저장소 시스템 기반으로 구현함 • 단일 분산 컴퓨터 클러스터에서 object storage 를 수행하는 free-software 스토리지 플랫폼을 도입함 • 분산 object store이자 file system으로 분산 클러스터 위에서 object storage를 구현해 object, block, file level의 storage 인터페이스를 설계함 • Ceph는 Ceph Object Storage 서비스와 Ceph Block Device 서비스, Ceph File System 서비스 제공 가능함
주요기능 도입	<ul style="list-style-type: none"> • 다수의 Region에서 운영하는 클러스터 사이의 싱글 네임 스페이스와 데이터 동기화 기능 도입 • 액티브 디렉토리, LDAP 및 Keystone v3 등을 포함하는 openstack 인증시스템과 통합해 향상된 보안 기능 도입 • AWS v4 클라이언트 시그니처, object versioning 등에 대한 지원을 포함 • 간단한 내를 통해 스토리지 관리 및 모니터링을 지원하는 시스템인 redhat storage 콘솔 2를 포함해 구축, 운영 및 관리를 간소화 지원 • 용량을 페타바이트 수준으로 확장 가능하도록 함 • Ceph Block Device 서비스, Ceph File System 서비스, Ceph Object Storage 서비스 제공 기술 도입
시스템 안정성 보장	<ul style="list-style-type: none"> • Amazon S3 오픈소스 API, 오픈스택 Swift, NFS v4 프로토콜과 호환성 • 오픈스택, 리눅스, KVM 하이퍼바이저와 통합된 블록 스토리지 • 가변적인 워크로드 처리를 가능하게 함

Block Device 서비스, Ceph FileSystem 서비스, Ceph Object Storage 서비스를 제공한다. Table 5는 Ceph 에 대해 상세히 설명하고 있다.^[19-22]

3.2 Ceph 클러스터 아키텍처

Ceph는 하나의 통합 시스템에서 블록 및 파일, 객체 스토리지를 고유하게 제공한다. Ceph는 신뢰성이 높고 관리하기가 용이하다. Ceph는 다수 n개의 클라이언트가 페타 바이트에서 엑사 바이트의 데이터에 액세스 가능한 확장성을 제공한다. Ceph Node는 상용 하드웨어와 프로세스 대몬(Daemon)을 활용하고 Ceph Storage Cluster는 서로 통신하여 데이터를 동적으로 복제 및 재배포하는 많은 수의 노드를 수용한다.

Ceph의 주요 구조는 클러스터 모니터, 메타데이터 서버, 오브젝트 스토리지 데몬으로 구성되며 주요 내용은 Figure 4와 Table 6에 상세히 기술되어 있다.^[17,18]

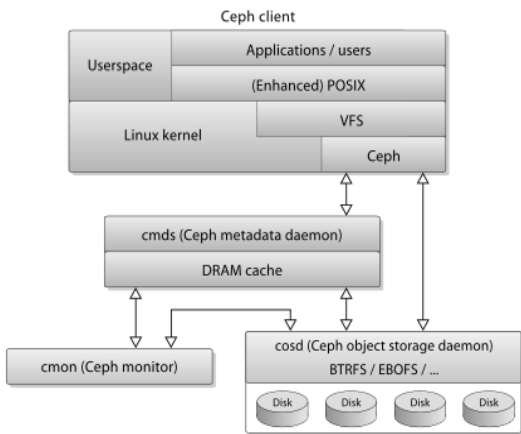


그림 4. Ceph 클러스터 아키텍처
Fig. 4. Architecture of Ceph Cluster

표 6. Ceph Cluster의 주요 구성요소
Table 6. Main Component of Ceph Cluster

항목	상세 설명 및 예시
클러스터 모니터 (ceph-mon)	<ul style="list-style-type: none"> Active와 Failed된 클러스터 노드를 관리 모니터는 클러스터의 상태를 체크 PG(Placement Group) 맵, OSD 맵 등을 관리
메타데이터 서버 (ceph-mds)	<ul style="list-style-type: none"> iNode와 디렉터리의 메타데이터 정보 관리 사용자가 Ceph 데이터를 검색과 점검을 위해 ls, find 등의 명령어를 사용할 때 metadata들을 저장하는 서버를 구축
오브젝트 스토리지 데몬 (ceph-osd)	<ul style="list-style-type: none"> 데이터를 저장하고, 복제, 부하분산 등의 역할 수행 OSD 디스크 1TB당 메모리 1G이상으로 구성해야 함

3.3 Ceph 스토리지 플랫폼 설계

3.3.1 블록 스토리지 설계

블록 스토리지는 BIM 데이터를 블록 단위로 파싱한다. 즉, 데이터를 별도의 세그먼트 단위로 분리해 저장한다. 각 데이터 블록은 고유 식별자를 부여받는데, 이는 스토리지 시스템이 더 작은 데이터 세그먼트를 원하는 곳에 배치할 수 있도록 해준다. 이를 이용하여 일부 데이터는 Linux 환경에 저장하고 일부는 Windows 장치에 저장 가능하도록 하였다.

블록 스토리지는 데이터를 사용자 환경에서 분리하여 이를 쉽게 활용할 수 있도록 다양한 환경 전반에 분산 관리되도록 설정되어 관리된다. 그 후 데이터가 요청되면 기본 스토리지 소프트웨어가 이러한 환경의 데이터 블록을 다시 조합해 사용자에게 제공한다. 이는 일반적으로 SAN(스토리지 영역 네트워크) 환경에 배포되며 가동되는 서버에 연결되도록 설정하였다.

블록 스토리지는 파일 스토리지와 같이 단일 데이터 경로에 의존하지 않으므로 신속하게 검색할 수 있다. 각 블록은 독립적으로 존재하며 파티션으로 분할될 수 있기 때문에 서로 다른 운영 체제에 액세스할 수 있다. 이 때문에 BIM 사용자는 자유롭게 데이터를 설정할 수 있으며 이는 데이터를 효율적이고 안정적으로 저장함과 동시에 사용과 관리에 편의성을 제공하도록 설계하였다.

대규모 트랜잭션을 수행하는 기능과 대용량 데이터 베이스를 배포하는 기능에서도 원활하게 동작되도록 설정하며 결론적으로 BIM 클라우드 데이터 측면에서 증가되는 데이터를 더 많이 저장하면 할수록 블록 스토리지를 사용하는 것이 더 유리한 장점을 적용하였다.

단점으로는 블록 스토리지는 비용이 많이 들 수 있고, 메타데이터를 처리하는 기능이 제한적이므로 애플리케이션 또는 데이터베이스 수준에서 취급해야 한다. 이 경우 개발자나 시스템 관리자의 업무 부담이 늘어나게 되며 이러한 점을 고려하여 설계하였다.

3.3.2 파일 스토리지 설계

파일 스토리지는 파일 수준 또는 파일 기반 스토리지 라고도 하며, 일반적으로 흔히 알려져 있는 스토리지 이다. 이 구조는 데이터가 폴더 안에 단일 정보로 저장되는 구조이다. 해당 데이터에 액세스해야 하는 경우 시스템은 그 데이터를 찾기 위해 경로를 알도록 설정해야 한다. 그리고 파일에 저장된 데이터는 제한된 양의 메타데이터를 사용해 구성 및 검색된다.

파일 스토리지는 계층 구조 스토리지라고도 하며, 저장되는 모든 문서가 논리적 계층 구조로 정리된다. 이는 직접 연결 및 네트워크 연결 스토리지 시스템으로도 사용되는 데이터 스토리지 시스템이다. BIM에서 사용되는 모든 문서에 대해 PC로 파일에 저장된 문서에 액세스 할 때마다 파일 스토리지를 사용하게 된다. 구현하고자 하는 파일 스토리지는 기능이 다양하며 거의 모든 데이터를 저장할 수 있다. 다수의 복잡한 BIM 데이터 파일을 저장하기에 용이하고 사용자가 신속히 탐색할 수 있는 장점이 있다.

단점으로는 파일 기반 스토리지 시스템은 더 많은 용량을 추가해 확장하는 것이 아니라 더 많은 시스템을 추가해 스케일아웃 해야 하며 이를 고려하여 동적으로 확장되도록 설계 하였다.

3.3.3 오브젝트 스토리지 설계

오브젝트 스토리지는 오브젝트 기반 스토리지로도 알려져 있으며, 파일들이 작게 나뉘어 여러 하드웨어에 분산되는 평면적(flat) 구조이다. 오브젝트 스토리지에서 데이터는 오브젝트라 불리는 개별 단위로 나뉘며, 서버의 블록이나 폴더에 파일을 보관하는 대신 단일 레포지터리(Repository)에 보관된다.

오브젝트 스토리지 볼륨은 모듈 단위로 동작한다. 각각은 독립적인 레포지터리이며, 데이터 및 오브젝트가 분산 시스템에 존재하도록 허용하는 고유 식별자, 그리고 해당 데이터를 설명하는 메타데이터를 보유하도록 설계하였다. 메타데이터는 사용 기간, 개인 정보/보안 및 액세스 비상 대책 등의 상세 사항을 포함하였다. 오브젝트 스토리지 메타데이터 내용은 매우 상세하게 구성하였으며 BIM 오브젝트의 위치, 사용된 장비 기종, 각 프레임에 나타난 도면 이름 등의 정보를 저장할 수 있다. 데이터를 검색하기 위해 스토리지 운영 체제는 메타데이터와 식별자를 사용하고, 로드를 보다 효율적으로 배포하여 관리자가 보다 강력한 검색 수행 정책을 적용하도록 하였다.

오브젝트 스토리지에는 단순한 HTTP API가 필요하며, 이는 대부분의 클라이언트에서 모든 언어로 사용된다. 오브젝트 스토리지는 사용한 만큼만 비용을 지불하면 되므로 비용면에서 효율적이다. 확장하기도 쉬우므로 퍼블릭 클라우드 스토리지에 적합하다. 이외에도 정적 데이터에 적합한 스토리지 시스템이며, 민첩성과 평면적 속성으로 인해 초대용량의 데이터로 확장할 수도 있다. 오브젝트에는 충분한 양의 정보가 있어서 애플리케이션이 신속하게 데이터를 검색할 수 있으며, 비정형 데이터를 저장할 수 있도록 설계하였다.

단점으로는 오브젝트는 수정이 불가능하다. 따라서 한 번에 오브젝트 작성을 완료해야 한다. 오브젝트 스토리지는 또한 전통적인 데이터베이스와 잘 연동되지 않는다. 오브젝트 작성이 느리게 진행되며, 오브젝트 스토리지 API를 사용하기 위해 애플리케이션을 작성하는 일이 파일 스토리지를 사용할 때 만큼 간단하지 않기 때문이다.

3.4 Ceph 스토리지 플랫폼 구현

Ceph 스토리지는 다운타임 없이 클러스터 확장과 같은 대규모 확장이 가능하며, 설치과정 및 운영의 간소화를 통해 더 빠르게 구축하여 사업을 수행할 수 있으며, 데이터 레이크분석 등 이머징 워크로드 처리 기능을 제공한다.

이러한 Ceph 스토리지 플랫폼 기반으로 향만 BIM 데이터 저장소를 블록 스토리지, 파일 스토리지, 오브젝트 스토리지로 구성하였다.

먼저 블록 스토리지는 하드 디스크 방식으로써 VM에 ‘경로’ 형태의 외부 저장 공간을 제공하고 개인 외부 공간이기 때문에 1개의 VM만 연결 가능하다.

파일 스토리지는 네트워크 공유 방식으로써 BIM 사용자들이 네트워크 스토리지처럼 연결하여 사용 가능 하고 VM에 ‘[StorageIP]/경로’ 형태의 저장 공간을 제공한다.

오브젝트 스토리지는 BIM 프로젝트 관리자 계정을 통해 저장공간을 생성한 후 BIM 사용자가 URL 인증 형태로 스토리지에 연결하고 RestAPI(PUT/GET)를 통해 데이터 관리하는 방식이다.

업무 프로세스에 있어서도 블록 스토리지는 먼저 블록스토리지 사용신청을 한 후 생성된 볼륨 공간을 개인 파티션 공간처럼 사용할 수 있도록 VM에 Mount (/data1)를 한 후 BIM 사용자가 외장 스토리지를 단독으로 사용할 수 있다.

파일 스토리지의 경우 파일 스토리지 사용 신청을 하고 생성된 볼륨 공간을 네트워크 스토리지 형태로 사용할 수 있도록 VM에 Mount (192.168.x.x: BIM_instance_1)한 후 같은 네트워크 안에 있는 BIM 사용자들이 네트워크 드라이브로 스토리지를 연결하여 사용할 수 있다.

오브젝트 스토리지의 업무 프로세스는 오브젝트 스토리지 저장공간을 생성하고 향만 BIM 라이브러리 업로드 및 웹 링크 URL을 공유한다. 그리고 향만 BIM 라이브러리가 필요한 사용자들이 웹 링크를 통해 다운로드를 하여 사용할 수 있다. 이를 도시하여 나타내면 Table 7과 같이 나타낼 수 있다.

표 7. BIM을 위한 Ceph 스토리지 플랫폼 설계
Table 7. Ceph Storage Platform Design for BIM

항목	상세 설명 및 예시
블록 스토리지	<p>블록 스토리지 사용 신청</p> <p>↓</p> <p>생성된 볼륨 공간을 개인 파티션 공간처럼 사용할 수 있도록 VM에 Mount (/data)</p> <p>↓</p> <p>BIM 사용자가 외장 스토리지를 단독으로 사용</p>
파일 스토리지	<p>파일 스토리지 사용 신청</p> <p>↓</p> <p>생성된 볼륨 공간을 네트워크 스토리지 형태로 사용할 수 있도록 VM에 Mount (192.168.x.x: BIM_instance_1)</p> <p>↓</p> <p>같은 네트워크 안에 있는 BIM 사용자가 네트워크 드라이브로 스토리지를 연결하여 사용</p>
오브젝트 스토리지	<p>오브젝트 스토리지 저장공간 생성</p> <p>↓</p> <p>항만 BIM 라이브러리 업로드 및 웹 링크 URL 공유</p> <p>↓</p> <p>항만 BIM 라이브러리가 필요한 사용자들이 웹 링크를 통해 다운로드</p>

3.4.1 블록 스토리지 구현

블록 스토리지는 VM에 '[경로]' 형태로 저장되며 (예. '/data') 개인 외부 공간이기 때문에 1개의 VM만 연결이 가능하다.

BIM 사용자가 VM 생성 시 블록스토리지 사용을 선택하게 되면 블록스토리지가 할당된 VM이 생성되며 생성 완료 후 해당 VM에 접속하면 마운트된 스토리지를 확인할 수 있다.

생성된 볼륨 공간은 개인 파티션 공간처럼 사용할 수 있으며 BIM 사용자는 할당된 블록 스토리지 공간에 개인적으로 사용할 데이터를 업로드하여 단독으로 사용하게 된다.

3.4.2 파일 스토리지 구현

파일 스토리지는 '[스토리지IP]/경로' 형태의 저장 공간을 제공하며 (예. 192.168.50.231:vm_shared_vol/BIM_instance_1) 네트워크 스토리지처럼 다수의 VM이 접속하여 BIM 데이터를 공유할 수 있다.

BIM 사용자가 VM 생성 시 파일 스토리지 사용을 선택하게 되면 IP로 구성되어 있는 스토리지 경로를 마운트한 VM이 생성된 것을 확인할 수 있다.

생성된 볼륨 공간은 네트워크 드라이브처럼 사용할 수 있으며 BIM 데이터 보안을 위해 같은 네트워크 안

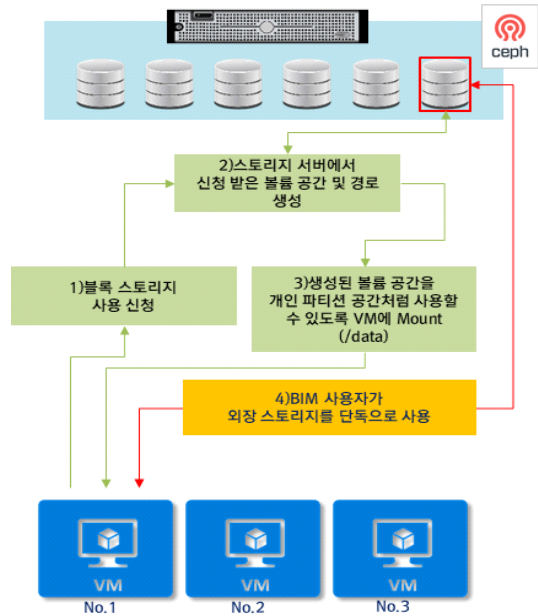


그림 5. 블록 스토리지 구현
Fig. 5. Implementation of Block Storage

에 다른 BIM 사용자 중 허가된 사용자만 접근할 수 있도록 IP 기반의 ACL(Access Control List) 설정을 할 수 있다.

접근이 허가된 다른 BIM 사용자들은 파일 스토리

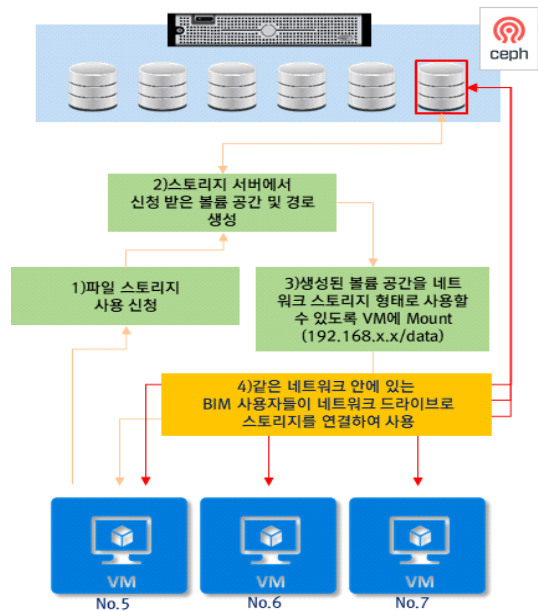


그림 6. 파일 스토리지 구현
Fig. 6. Implementation of Block Storage

지 경로를 자신들의 VM에 마운트한 후 접근하여 사용자들 간에 BIM 데이터를 공유한다.

3.4.3 오브젝트 스토리지 구현

오브젝트 스토리지는 관리자가 오브젝트 저장공간을 생성하고 사용자들이 BIM 데이터를 업로드하면 내부 혹은 외부 사용자들이 URL로 구성된 WebLink를 통해 데이터를 다운로드 후 사용할 수 있다.

BIM 사용자들은 BIM 플랫폼을 통해 관리자가 생성한 오브젝트 스토리지 저장공간을 사용할 수 있으며 BIM 데이터를 드래그 앤 드롭을 통해 업로드 할 수 있다.

업로드된 BIM 데이터는 WebLink URL를 제공하며 해당 URL을 다른 사용자들에게 공유하면 사용자들은 공유한 URL을 통해 BIM 데이터를 다운로드 한다.

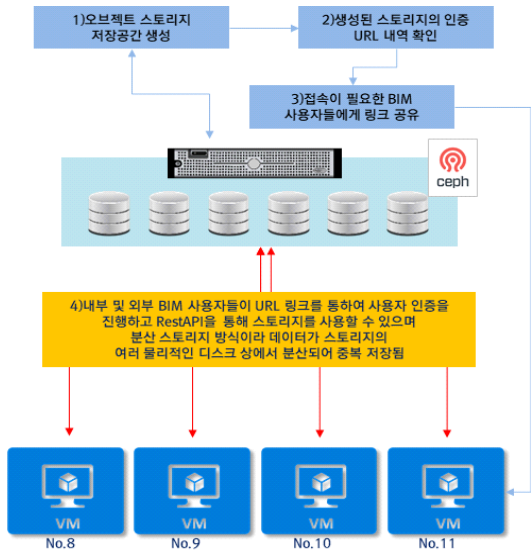


그림 7. 오브젝트 스토리지 구현
Fig. 7. Implementation of Object Storage

IV. 시스템 실행결과 및 성능평가

4.1 시스템 실행결과

구현된 프로그램의 주요 기능은 프로젝트의 목적에 따라, 블록 스토리지 생성, 파일 스토리지 생성, 오브젝트 스토리지 관리, 오브젝트 스토리지에서 BIM 데이터를 다운로드 하는 등의 기능이다.

VM 생성 시 블록스토리지 사용을 선택하고 생성된 VM에 접속 후 '/data1'으로 보이는 개인 파티션 공간이 마운트 되는 것을 볼 수 있다.

사용자는 FTP 프로그램을 통해 IFC 파일을 업로드

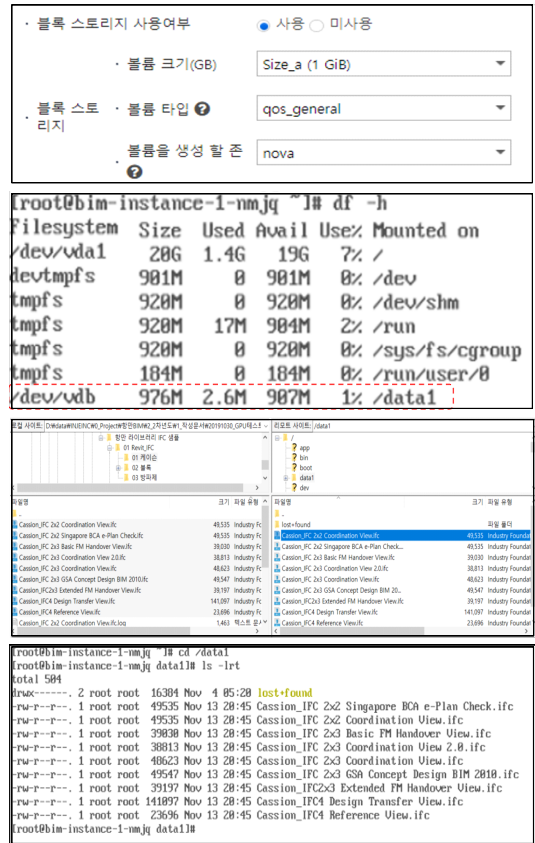


그림 8. 블록 스토리지 생성 구현
Fig. 8. Application of Block Storage

하고 블록 스토리지 공간에 접근해서 파일이 정상적으로 업로드 되었는지 확인할 수 있다.

파일 스토리지 사용을 선택하면 VM 생성 시 192.168.50.231의 스토리지 IP를 가진 경로를 마운트 하는 것을 볼 수 있다.

사용자는 할당받은 파일 스토리지 공간에 IFC 파일을 업로드 한 후 해당 스토리지 경로를 다른 BIM 사용자에게 공유하고 공유받은 사용자는 자신의 VM 공간에 파일 스토리지를 마운트 한 후 IFC 파일을 공유하는 것을 확인할 수 있다.

오브젝트 스토리지의 경우 할당받은 오브젝트 스토리지 공간에 IFC 파일을 드래그앤드롭 방식으로 업로드를 하고 업로드한 IFC 파일의 WebLink URL을 확인할 수 있다.

확인된 WebLink URL을 IFC 데이터가 필요한 BIM 사용자에게 공유하고 공유받은 BIM 사용자는 해당 WebLink URL에 접속해서 데이터를 다운로드 받은 후 BIM IFC 파일 뷰어 프로그램으로 열람하는

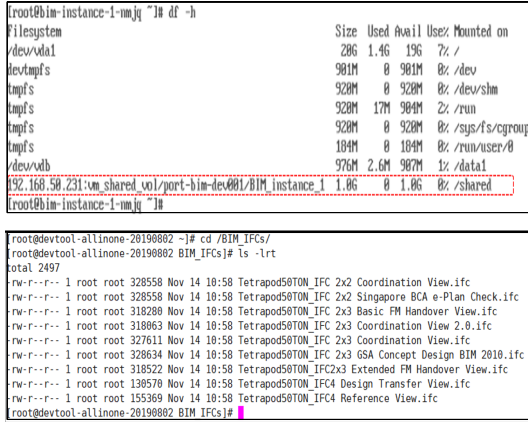
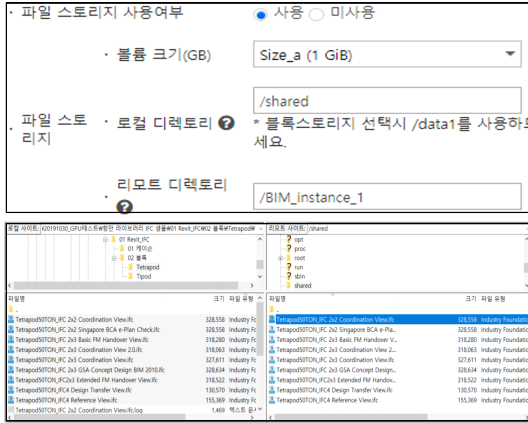


그림 9. 파일 스토리지 생성 구현
Fig. 9. Application of File Storage

것을 확인할 수 있다.



그림 10. 오브젝트 스토리지 생성 구현
Fig. 10. Application of Object Storage

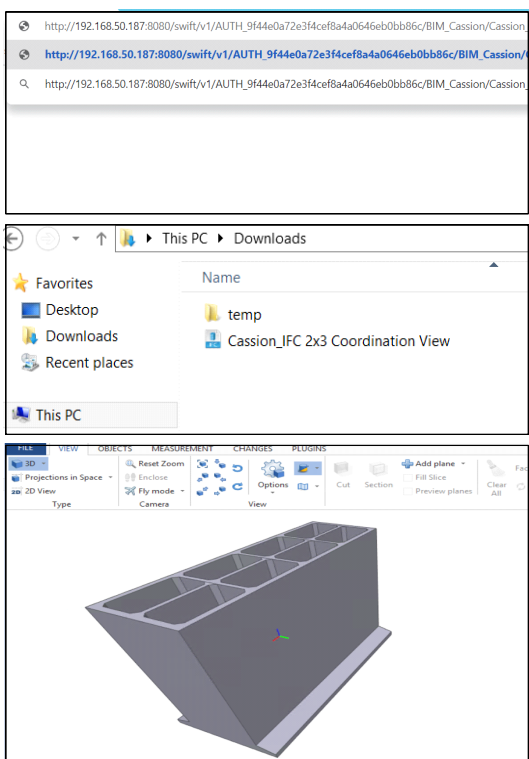


그림 11. 오브젝트 스토리지로부터 BIM 데이터 다운로드
Fig. 11. Download BIM Data form Object Storage

4.2 시스템 성능 평가

본 연구에서는 BIM 데이터인 IFC 파일을 상용 NAS와 SAN 구성에서의 read/write 처리량 속도와 Ceph 스토리지 플랫폼 기반에서의 블록 스토리지와 파일 스토리지 read/write 성능을 비교하였다. 테스트 구성 정보는 Table 8과 같다.

표 8. 테스트 구성정보
Table 8. Test Information

항목	상세 설명 및 예시
플랫폼	<ul style="list-style-type: none"> • 오픈스택(OpenStack) 커뮤니티 : Stein • 오픈서프트(OKD) : 3.11 • Ceph 스토리지 : 10.2.11
OS	• Cent OS 7.8
CPU	• CPU: Intel® Xeon® Gold 6126 (12Core/2.6GHz)
GPU	• NVIDIA Tesla V100 32GB
Memory	• 256GB DDR4
Ceph 스토리지	• 200 TB
NAS	• IBM ESS GL4C
SAN	• IBM Flashsystem 9200
측정 대상	• IFC file : 15.3 MB
성능 측정툴	• IOzone 3.4 (Read/Write 처리량 성능 측정)

성능 측정 방법은 블록 스토리지와 파일 스토리지를 새로 생성된 VM에 마운트해서 IOzone 측정툴로 처리량 측정값을 확인하였다. IOzone은 직접 임시 파일을 생성해 테스트하므로, 15 MB 크기의 IFC 임시 파일을 생성하고 단일 스트레드에서 레코드 크기 256 Kbytes로 Read/Write 처리량 값을 비교하였다. 오브젝트 스토리지는 URL 기반으로 동작하므로 VM에 마운트해서 사용할 수 없어 테스트 환경이 다르기에 측정 평가에서 제외하였다.

4.2.1 Ceph 플랫폼과 타 스토리지 성능 비교

Ceph 스토리지 플랫폼 기반의 블록 스토리지와 파일 스토리지의 데이터 처리량과 타 제품의 NAS 및 SAN 기반에서의 데이터 처리량 비교는 Table 9과 Figure 12와 같다.

측정결과, SAN 기반의 성능이 우수하지만, NAS 측정값과 비교해보면 별도 상용 스토리지 구성 없이 Ceph 스토리지 플랫폼을 통해서도 유사한 성능을 확인하였다. Initial write의 경우 데이터를 추적하기 위한 오버헤드(메타데이터) 정보가 필요하므로 이를 처리하는 과정에서 타 스토리지 성능 보다는 다소 떨어지는 것을 확인할 수 있다. 반면 Rewrite (다시 쓰기)

표 9. Ceph 플랫폼과 타 상용스토리지 처리량 결과
Table 9. Result of Ceph Platform and Other Commercial storage throughput

항목		측정결과 처리량(GB/S)			
		Initial write	Re-write	Read	Re-read
상용 사	NAS	1.12	2.58	4.17	4.67
	SAN	1.01	3.28	5.51	5.57
Ceph 플랫폼	블록	0.45	2.52	4.48	4.53
	파일	0.57	2.17	4.13	2.64

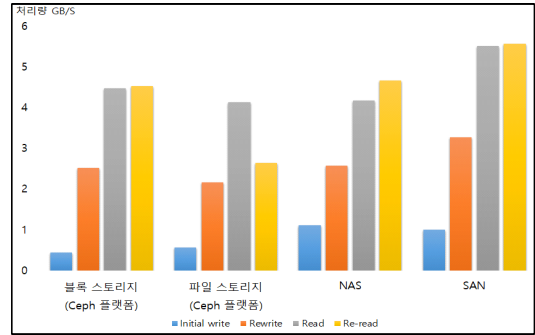


그림 12. Ceph 플랫폼과 타 상용스토리지 처리량 비교
Fig. 12. Comparison of Ceph Platform and Other Commercial Storage Throughput

의 경우 이미 만들어진 오버헤드를 참조하므로 성능이 개선되는 것을 확인할 수 있다.

Read (읽기)의 경우 Re-Read와 큰 성능 차이가 없는데 Re-read (다시 읽기)의 경우 캐시를 활용하면 기존보다 처리량 성능을 높일 수 있다

4.2.2 GPU 기반의 블록/파일 스토리지 성능 비교

본 항만 BIM 플랫폼은 디지털화된 항만지형 및 구조물을 포함한 대용량의 BIM 데이터를 다루므로 이를 고속의 가시화 처리를 위해 클라우드 서버에 별도

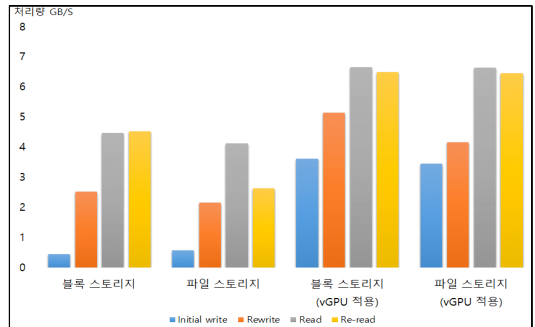


그림 13. GPU 기반의 블록/파일 스토리지 처리량 비교
Fig. 13. Comparison of Block/File Storage Throughput based on GPU

표 10. GPU 기반의 블록/파일 처리량 결과
Table 10. Result of Block/File Storage Throughput based on GPU

항목		측정결과 처리량(GB/S)			
		Initial write	Re-write	Read	Re-read
vGPU 미적용	블록	0.45	2.52	4.48	4.53
	파일	0.57	2.17	4.13	2.64
vGPU 적용	블록	3.61	5.15	6.67	6.49
	파일	3.46	4.16	6.65	6.46

의 GPU(Grp)를 탑재하여 각 VM들이 그래픽 자원을 공유할 수 있도록 할당하였다. 이는 BIM 데이터의 그래픽 가지화 처리와 사용자 뷰어로의 로딩속도, 데이터 처리량 개선을 목적으로 활용되었다.

‘본 연구에서는 GPU가 적용되기 전과 항만 BIM 플랫폼에서 제공하는 vGPU가 적용된 경우의 블록/파일 스토리지를 상호 비교하였다. 처리량 성능 비교는 Table 10과 Figure 13과 같다.

측정결과, GPU의 적용으로 Ceph 스토리지 플랫폼 데이터 처리량 성능이 상당히 개선 됨을 알 수 있었다. 특히, Initial write의 경우 3배 이상의 성능이 개선되었으며, 타 스토리지 구성보다도 처리량 성능이 우수하였다.

또한, BIM 라이브러리 모델(테트라포드)의 렌더링 및 페이지 로딩시간 처리를 검증하였다. Figure 14의 측정값은 크롬 브라우저에서 개발자 도구를 사용하여 성능을 측정하였으며, GPU를 제외하고 동일한 H/W 환경에서 테스트한 것이다.

Ceph 스토리지는 설치 및 운영이 간단하여 설치 과정이 기존 설치 과정에 비해 1/3 수준으로 간소화되어 더 빠르게 배포할 수 있으며 간소화된 운영, 모니터링 및 용량 관리를 통해 방대한 BIM 비정형 데이터 관리가 가능하다.

스토리지 종류도 블록/파일/오브젝트 스토리지를 모두 지원하기 때문에 단독 파티션을 통해 BIM 데이터를 개인 용도로 활용하고 싶을 경우에는 블록 스토리지, 네트워크 공유 방식으로 같은 네트워크에 있는 사용자와 BIM 데이터를 공유하고 싶을 경우에는 파일 스토리지, WebLink를 통한 외부 사용자에게 BIM 데이터를 공유하고 싶을 경우에는 오브젝트 스토리지를 사용하여 BIM 데이터의 활용성을 높일 수 있다. 이런 다양한 방식의 스토리지 지원은 BIM 데이터 라

표 11. Ceph 와 상용 스토리지와의 기술적 차이
Table 11. Technical difference between Commercial Storage and Ceph

항목	F사	H사	제안시스템
블록/파일/오브젝트 스토리지 동시지원	★★☆	★★★★	★★★★★
오브젝트 확장성	★★★★	★★★★	★★★★★
설치 소요시간	★★★★	★★★★☆	★★★★★
전문성 의존도 낮음	★★	★☆☆	★★★★
구축비용	★★☆	☆	★★★★★

이브리리를 이용한 협업 시 실무자의 업무 효율성을 높여줄 수 있을 것이다.

이외에도 AI/ML(Artificial Intelligence/Machine Learning), 데이터 레이크 분석, 클라우드 오브젝트 스토리지, 기타 이머징 애플리케이션의 요구 사항을 충족하고, 데이터 연구 및 사용자에게 기존 스토리지 서비스 제공을 그대로 유지할 수 있다.

타 상용 제품의 스토리지와의 비교에 있어서도 기술 및 성능 면에서도 유사한 결과가 도출되었다. 특히 블록/파일/오브젝트 스토리지 동시지원과 오브젝트의 확장성, 구축비용 부문에 있어서 많은 차이를 보였다.

V. 결 론

항만 BIM 플랫폼의 클라우드 서비스를 위한 IaaS+PaaS 통합 환경에서 데이터 저장소는 필수적으로 구축되어야 한다. 스토리지 도입으로 항만 BIM 플랫폼에서 생산 및 관리되는 모든 데이터를 효율적으로 관리하며, 본 플랫폼 내에서 BIM 라이브러리 업로드를 위한 스토리지로 구축하여 대규모 성능확장, 모니터링, 프로젝트 목적에 맞는 데이터 저장 방식 등의 기능을 보장한다.

본 연구에서는 이를 위해 Ceph 스토리지 플랫폼 기반으로 시스템을 구축하였으며, 크게 3가지 방식인 블록 스토리지, 파일 스토리지, 오브젝트 스토리지 방식으로 구현하였다. 프로젝트 목적에 맞게 BIM 데이터를 개인 용도로 활용하는 방안은 블록 스토리지 방식으로, 네트워크에 있는 사용자와 BIM 데이터를 공유하고 싶을 경우에는 파일 스토리지 방식으로, WebLink를 통한 외부 사용자에게 BIM 데이터를 공유하는 방안은 오브젝트 스토리지 방식으로 BIM 데이터 저장소의 운용성 및 편의성 확보를 확인하였다.

연구수행 방법으로는 먼저 스토리지 분석을 통해

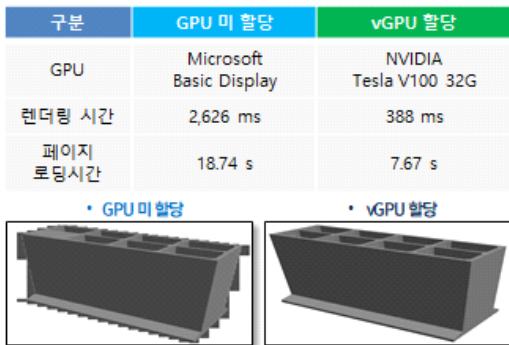


그림 14. GPU 기반의 렌더링 및 로딩시간 테스트
Fig. 14. Rendering & Loading Time Test based on GPU

클라우드 스토리지 종류를 분석하고 스토리지별 용도 및 특성을 파악하였다.

이 과정을 통해서 향만 연계 BIM 데이터 저장소의 방식을 선정하였으며, Ceph 스토리지 플랫폼 기반으로 대규모 확장성, 설치 및 운영 간소화, 스토리지 구매비용 절감 등을 보장하였다.

Ceph 스토리지 플랫폼의 성능을 검토하기 위해 타 상용제품과의 처리량 비교, GPU 적용 여부에 따른 향만 BIM 데이터에 대하여 렌더링, 페이지 로딩 시간 및 처리량을 테스트하였다. 그 결과 별도의 상용 제품 구성없이 오픈소스 기반의 Ceph 스토리지 플랫폼 구현을 통해서도 유사한 성능을 확인하였다.

GPU 적용 시 Initial write의 처리량은 3배 이상 성능이 개선되었으며, 타 상용 제품 대비 렌더링 시간 및 데이터 처리 성능이 높은 것으로 분석되었다. 이러한 결과 향만과 같이 대형 구조물 및 지형으로 구성된 향만 BIM 데이터 처리에 GPU의 적용은 필수적이다

향후, 본 Ceph 스토리지를 응용하여 향만 BIM 정보통합플랫폼의 SaaS 환경구축 및 BIM Browser 개발과 향만 BIM 플랫폼 저장소 구축의 요소기술로 활용할 예정이다.

References

[1] Gartner, "Gartner identifies the top 10 strategic technologies for 2011," <http://www.gartner.com>, 2010.

[2] S. A. Robison, "Bright Future in the Cloud," *Financial Times*, 2008.

[3] J. Ge, Y. Deng, and Y. Fang, "Research on storage virtualization structure in cloud storage environment," *ICMT*, pp. 1-4, 2010.

[4] Steve Lesem, "Cloud Storage and The Innovator's Dilemma," <http://cloudstoragestrategy.com/cloudecosystem/>, 2009.

[5] J. Farmer, "A crash course in wide area data replication," *SNIA*, 2009.

[6] Y. M. Bae, S. J. Jeong, L. Y. Jang, K. Seong, and W. Y. So, "Domestic status and prospect of cloud computing," *KICS Spring Conf.*, pp. 294-297, 2011.

[7] *SNIA CLOUD Storage Summit.*, <http://www.snia.org/events/wintersymp2009/cloud/>. Held at the WINTER SYMPOSIUM 2009.

[8] Q. He, Z. Li, and X. Zhang, "Analysis of the

key technology on cloud storage," *FITME, IEEE Conf. Publications 2010*, pp. 426-429, 2010.

[9] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data deduplication," *StorageSS'08*, Fairfax, Virginia, USA, Oct. 2008,

[10] B. Zhu, K. Li, and H. Patterson, "Avoiding the disk bottleneck in the data domain deduplication file system," in *Proc. 6th USENIX Conf. FAST*, pp. 269-282, SanJose, CA, USA, Feb. 2008.

[11] S. W. Yoon, B. K. Kim, J. M. Choi, and S. W. Kwon, "Aprototype BIM server based viewer for cloud computing BIM services," *J. KSCE*, vol. 33, no. 4, pp. 1719-1730, 2013.

[12] J. H. Lee and S. J. Eom, "Computing technology for mobile BIM based project management information system," in *Proc. KIBC Spring Conf.*, vol. 11, no. 1, pp. 145-148, 2011.

[13] L. Kang, H. S. Moon, S. B. Ji, and T. S. Lee, "Development of major functions of visualization system for construction schedule data in plant project," *Korea J. Construction Eng. and Manag.*, vol. 9, no. 1, pp. 66-76, 2008.

[14] K. H. Kim, E. J. Kim, and J. J. Kim, "A study to judge the importance of indices to estimate architectural competition on the apartment for using BIM," *J. The Architectural Inst. Korea (Planning & Design Section)*, vol. 25, no. 2, pp. 59-66, 2009.

[15] S. A. Kim, B. H. Yang, S. Chin, and S. W. Yoon, "A study on a development and current state of BIM in Korean focus on hype-cycle," in *Proc. KIBIM Annu. Conf. 2012*, vol. 2, no. 1, pp. 73-74, 2012.

[16] B. Kumar, J. Cheng, and L. Mcgibbney, "Cloud computing and its implications for construction it," in *Proc. Int. Conf., Computing in Civil and Building Eng.*, pp. 315-320, Nottingham Univ. Press, 2010.

[17] C. J. Lee, G. Lee, and J. S. Won, "An analysis of the BIM software selection factor,"

J. The Architectural Inst. Korea (Structure & Construction Section), vol. 25, no. 7, pp. 153-164, 2009.

[18] J. S. Lee, S. K. Moon, J. H. Kim, and J. J. Kim, "An analysis about factors affecting inactiveness of BIM (Building Information Modeling) introduction in the construction project," in *Proc. KICEM Annu. Conf. 2009*, pp. 757-762, 2009.

[19] J. D. Park and J. W. Kim, "A study on the ontology representation of the IFC based building information model," *J. The Architectural Inst. Korea (Structure & Construction Section)*, vol. 25, no. 5, pp. 87-94, 2009.

[20] J. W. Park, S. C. Kim, S. S. Lee, and H. Y. Song, "Suggesting solutions when applying building information modeling (BIM) to the Korean construction industry through case studies," *J. The Korea Inst. Building Construction*, vol. 9, no. 4, pp. 93-102, 2009.

[21] A. Redmond and B. Smith, "Exchanging partial BIM information through a cloud-based service: Testing the Efficacy of a Major Innovation," in *Proc. IBEA Conf. 2011*, vol. 31, no. 4, pp. 22-29, London South Bank Univ., 2011.

[22] J. K. Song and K. B. Ju, "Applying the cloud computing technology for utilizing mobile device of BIM-based maintenance systems," in *Proc. KICEM Annu. Conf. 2011*, pp. 311-312, 2011.

[23] K. H. Chae and G. Lee, "A study on the problems and the measurements for improving representations and drafting methods of architectural drawings by Adopting BIM," *J. The Architectural Inst. Korea (Planning & Design Section)*, vol. 27, no. 10, pp. 67-74, 2011.

[24] S. B. Han, *Feasibility study on strategic planning to build a cloud computing environment for smart education*, 2011 KERIS Issue Report, Korea Edu. and Res. Inf. Serv., vol. 3, no. 2, pp. 33-43, 2011.

[25] J. E. KIM and C. H. Hong, "A light weight

algorithm for large-scale BIM data for visualization on a web-based GIS platform," *J. Korea Spatial Inf. Soc.*, vol. 23, no. 1, pp. 41-48, 2015.

[26] T. W. Kang, "BIM geometry cache structure for data streaming with large volume," *J. Korea Academia-Ind. Cooperation Soc.*, vol. 18, no. 9, pp. 1-8, 2017.

현근주 (Keun-Ju Hyun)



1997년 2월: 청주대학교 정보통신공학과 졸업
 1999년 2월: 청주대학교 전자공학과 석사
 2014년 10월~현재: (주)인재 INC 클라우드팀 팀장

<관심분야> 정보통신, 클라우드, 건설정보모델

문현석 (Hyoun-Seok Moon)



2006년 2월: 경상대학교 토목공학과 (공학석사)
 2009년 8월: 경상대학교 토목공학과 (공학박사)
 2009년 8월~2011년 1월: Teesside University (UK), CCIR 센터 선임연구원

2012년 2월~2013년 1월: University of Michigan (Ann Arbor), Post-Doc.

2012년 12월~현재: 한국건설기술연구원 스마트건설 혁신본부 국가BIM연구센터, 수석연구원

<관심분야> BIM, 디지털트윈, AI 플랫폼, 건설관리
 [ORCID:0000-0002-2543-4070]