

KI Cloud 플랫폼을 위한 데이터 저장소 설계 및 구현

조혜영°, 정기문, 홍태영
한국과학기술정보연구원 슈퍼컴퓨팅본부
e-mail:{chohy°, kmjeong, tyhong}@kisti.re.kr

Design and Implementation of The Data Storage for KI Cloud Platform

Hyeyoung Cho°, Gi-Mun Jeong, TaeYoung Hong
Division of Supercomputing
Korea Institute of Science and Technology Information

요 약

최근 베어메탈 시스템에 스케줄러 시스템 기반 인프라 시스템에서 벗어나 사용자 맞춤형 클라우드 인프라가 관심을 받고 있다. KI Cloud 플랫폼은 고성능 컴퓨팅(HPC) 사용자를 위한 클라우드 기반 IaaS, PaaS 통합 플랫폼이다. 본 논문에서는 KI Cloud를 위한 데이터 저장소 설계 및 구현에 대해 기술한다. 오픈 소스 Ceph 분산 스토리지 시스템을 기반으로 구축하였고, 프로젝트의 목적에 맞게 블록 스토리지, 오브젝트 스토리지, 파일 스토리지 방식으로 구현하였다. Openstack 기반의 서비스와 Kubernetes 기반의 서비스가 하나의 스토리지를 통해 다양한 타입의 서비스를 제공할 수 있도록 개발하였다.

1. 서론

인공지능, 빅데이터 등의 IT 기술이 발전하면서 고성능 컴퓨터(HPC)를 사용자들이 다양해졌다. 따라서 같은 소프트웨어 환경의 인프라를 스케줄러를 통해 공유하던 전통적인 방식에서 벗어나 개인 맞춤형 인프라를 유연하게 제공하기 위한 클라우드 환경이 각광을 받고 있다. 이에 한국과학기술정보연구원(KISTI)에서도 고성능 컴퓨팅(HPC) 사용자를 위한 플랫폼인 KI Cloud를 개발하여 서비스하고 있다[1].

클라우드 IaaS, PaaS 통합 플랫폼을 위해서는 데이터 저장소는 중요 필수적인 요소이다. 본 논문에서는 KI Cloud 플랫폼의 데이터 저장소 설계 및 구현에 관한 내용을 기술한다. 클라우드 IaaS, PaaS 통합 플랫폼에 필요한 스토리지의 역할 등을 기술하고 이에 맞게 오픈 소스 Ceph 분산 스토리지를 기반으로 데이터 저장소를 설계 및 구현하였다.

본 논문의 구성은 다음과 같다. 2장에서는 KiCloud 플랫폼 구조를 기술하고, 3장에서는 KI Cloud 플랫폼을 위한 스토리지 설계 및 구현에 관한 내용을 기술한다. 마지막으로 4장에서는 결론에 대하여 기술한다.

2. KI Cloud 플랫폼

AI, 빅데이터 등의 IT 기술이 발전하면서 고성능 컴퓨터(HPC)를 사용자들이 다양해졌다. 전통적으로 같은 소프트웨어 환경인 하나의 큰 슈퍼컴퓨터에 다수의 사용자가 로그인 로드에서 스케줄러로 작업을 제출하여 사용하던 방식을 벗어나 더욱 유연하고 효율적인 방식으로 인프라를 사용하고자 하는 요구가 늘어나고 있다. 이러한 시대의 흐름에 따라 한국과학기술정보연구원(KISTI)에서 클라우드 기반 인프라를 제공하기 위한 KI Cloud 플랫폼을 개발하였다.

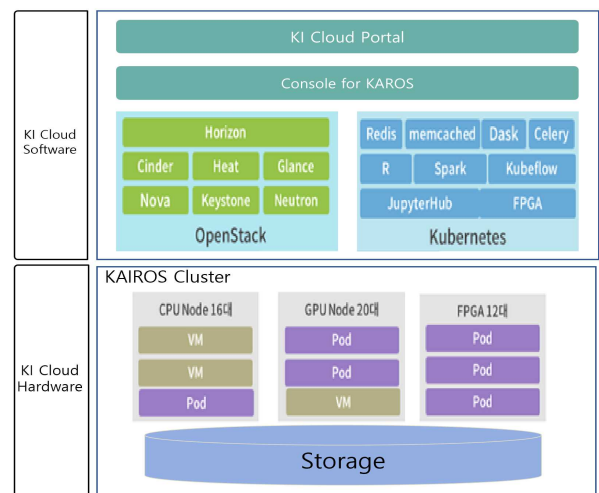


그림 1 KI Cloud 플랫폼 구조

KI Cloud는 KISTI 슈퍼컴퓨터 사용자 및 국내 연구자들에게 IaaS(Infrastructure-as-a-Service)에서 PaaS(Platform-as-a-Service)까지 다양한 서비스를 제공한다. IaaS는 가상 머신을 기반으로 인프라를 제공하고, PaaS 서비스는 컨테이너를 기반으로 제공하도록 개발되었다[2].

그림 1은 KI Cloud 플랫폼의 구조를 보여준다. KI Cloud 플랫폼은 IaaS 서비스 제공을 위해서는 Openstack[3]을, PaaS를 위해서는 Kubernetes[4]를 기반으로 개발되고 운용되고 있다. 오픈 소스 Openstack 기반으로 가상 머신을 생성 삭제 가능하며 사용자는 이 인프라를 자신의 전용 인프라로 구성하여 사용할 수 있다. 또한 GPU를 장착한 가상 머신도 제공하고 있다. Kubernetes 기반 PaaS 서비스로는 Jupyter, RStudio 와 같은 웹 기반 대화형 서비스, Redis, Memcached와 같은 인메모리 데이터베이스 서비스, Spark, 분산 Tensorflow와 같은 대규모 데이터 분석 환경 서비스를 제공한다. KI Cloud의 인프라 하드웨어는 크게 슈퍼컴퓨터 5호기 Nurion 시스템과 KAIROS 시스템 2세트로 구성되어 있는데, 본 논문에서는 클라우드 전용으로 설계된 KAIROS 시스템을 기준으로 기술하였다.

3. KI Cloud 플랫폼을 위한 스토리지

3.1 클라우드 스토리지

KI Cloud 플랫폼의 스토리지 서비스는 분산 스토리지 Ceph을 기반으로 Openstack과 Kubernetes 시스템에 공통으로 사용하도록 구축·개발하였다. KI Cloud에서 클라우드 데이터 처리를 위해 필요한 스토리지 서비스를 표 1에 기술하였다. KI Cloud에는

표 1 KI Cloud 데이터 처리를 위한 필요 스토리지

서비스명	서비스 주요 기능
Cinder Backend	- VM Block Storage Service - VM 생성 및 운영을 위한 블록 스토리지 볼륨 생성·삭제·변경 관리
Glance Backend	- OS 이미지 관리 서비스 - Nova에서 생성하는 인스턴스 운영체제의 이미지를 관리
Swift 오브젝트 스토리지	- 오브젝트 스토리지 서비스 - AWS S3와 같이 RESTful APIs를 통해 사용 가능한 스토리
Manila	- OpenStack 파일시스템 공유 서비스
Kubernetes의 PV	- Kubernetes의 Persistent Volumes로 사용하는 서비스

Openstack 가상 머신에 블록스토리지를 제공하는 Cinder와 가상 머신의 이미지를 관리하는 Glance, 가상 머신 간의 파일 공유를 제공하는 Manila 서비스를 위한 스토리지가 필요하다[5-7]. 그리고 KI Cloud는 오브젝트 스토리지를 제공하는 Swift[8]와 쿠버네티스의 Pod의 디스크로 사용되는 PV(Persistent Volume)으로 스토리지가 제공되어야 한다.

3.2 Ceph 분산 스토리지

KI Cloud는 다양한 스토리지 서비스를 오픈 소스 기반의 Ceph 분산 스토리지를 이용하여 설계 및 구축하였다. Ceph는 Software-defined storage 시스템으로 오픈 소스이고 분산(distributed), 확장(scale-out)이 용이하다[9]. 또한 클라우드 서비스에서 필요로 하는 블록, 객체 및 파일 스토리지 모두 제공하는 장점이 있다. Ceph는 2019년 Openstack Survey에 따르면, 클라우드 영역에서 Openstack Block 스토리지로 가장 많이 사용되는 스토리지로 알려져 있다[9].

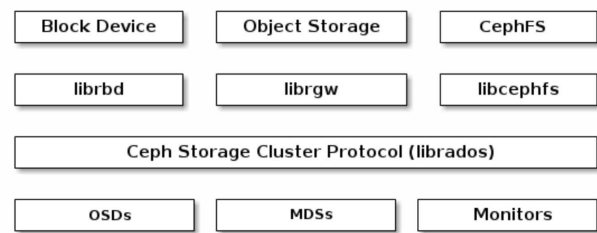


그림 2. Ceph 주요 아키텍처

그림 2는 Ceph의 주요 아키텍처를 보여준다[10]. Ceph은 단일 시스템에서 Block Storage, Object Storage, File Storage 3가지 타입의 스토리지를 제공하는 대표적인 Unified 분산 스토리지이다. Block 장치(RBD)는 스냅샷 및 클로닝과 함께 확장 가능한 썬 프로비저닝 블록 장치를 제공한다. Object Storage는 Amazon S3 및 OpenStack Swift와 호환되는 RESTful APIs 인터페이스를 제공한다. File Storage는 POSIX 호환 파일 시스템을 마운트하거나 사용자 공간의 파일 시스템(FUSE)으로 사용할 수 있다.

RADOS Cluster는 Monitor, MDS(Meta Data Server), OSD(Object Storage Daemon) 3가지로 구성되어 있다. Monitor는 RADOS Cluster를 운영에 필요한 Monitor Map, OSD Map, PG Map, CRUSH Map, MDS Map으로 구성되어 있다. Monitor는 고

표 2. KI Cloud를 위한 Ceph 스토리지 플랫폼 설계 및 구현

스토리지 타입	ceph interface	서비스 구분		용도 및 설명	
		openstack	Kubernetes		
블록 스토리지	RBD block storage	○		Cinder	- VM의 하디디스크로 사용됨 예) dev/vdx 형태로 장착되어 사용 - 한 개의 VM에만 연결 가능함
		○		Glance	- VM을 위한 OS 이미지 저장 - Nova에서 생성하는 OS이미지 관리
오브젝트 스토리지	RGW objects	○		Swift	- 오브젝트 스토리지 서비스 - RESTfull APIs로 스토리지 접근하여 사용 가능
파일 스토리지	Ceph FS	○		manila	- 다수의 VM간의 공유 파일 시스템 제공
			○	PV	- 쿠버네티스의 Persistent Volume 제공 - K8s Pod를 위한 스토리지로 사용

가용성을 보장하기 위해서 Cluster Map을 관리하고 유지하는 Daemon이다. OSD는 Object 형태로 디스크에 Data를 저장하는 Daemon이다. 데이터를 저장하고, 복제하고, 부하분산 시키는 역할을 한다. MDS는 POSIX 호환 File System를 제공할 때 필요한 메타 데이터를 관리하는 Daemon으로, Ceph가 File Storage로 동작할 때만 사용된다. Directory 계층 구조, iNode 같은 파일의 메타 정보들을 RADOS Cluster에 저장하고 관리한다.

3.3 KI Cloud를 위한 스토리지 설계 및 구현

KI Cloud를 위한 Ceph 스토리지 플랫폼 설계 및 구현 내용을 표 2에 기술하였다. 가상 머신의 하드디스크로 사용하는 스토리지는 RBD(RADOS Block Device)를 통해 서비스된다. VM을 위한 OS 이미지를 저장하는 스토리지 또한 RBD를 사용하도록 구현되었다. RESTfull Protocol을 이용하여 사용할 수 있는지 오브젝트 스토리지는 RGW(RADOS Gateway)를 사용한다. 가상 인스턴스 간에 공유 파일 시스템은 CephFS를 통해 구현되었다. 또한 쿠버네티스의 Pod를 위한 스토리지인 Persistent Volume도 CephFS를 활용하여 구현하였다. 이렇게 여러 가지 타입의 스토리지를 필요로 하는 플랫폼을 오픈 소스 기반의 Ceph 분산 스토리지에 통합하여 구현함으로써 스토리지 자원을 효율적으로 활용할 수 있는 기반을 마련하였다.

4. 결론

본 논문에서는 KI Cloud 플랫폼을 위한 데이터 저장소 설계 및 구현에 대해 기술하였다. 오픈 소스 Ceph 분산 스토리지 시스템을 기반으로 구축하였

고, 프로젝트의 목적에 맞게 블록 스토리지, 오브젝트 스토리지, 파일 스토리지 방식으로 구현하였다. 여러 가지 스토리지 타입을 하나의 Ceph 분산 시스템에 통합 구현함으로써 스토리지를보다 효율적으로 사용할 수 있도록 하였다.

앞으로 KI Cloud 플랫폼을 통하여 고성능 인프라가 필요한 국내 연구자들에게 다양한 IaaS 및 PaaS 서비스를 제공하고 사용자들의 편의성 향상에 힘쓸 예정이다.

ACKNOWLEDGMENTS

본 논문은 한국과학기술정보연구원에서 2020년도 정부(미래창조과학부)의 지원(No.K-21-L02-C07-S01, 차세대 슈퍼컴퓨팅 서비스 기술 연구)로 수행된 연구임.

참고문헌

- [1] KiCloud (KISTI Intelligence Cloud), <https://kicloud.ksc.re.kr/>
- [2] Hyeyoung Cho, et al, "Design and Development of KI Cloud Platform for High Performance Computing," KIPS fall conference, 2020.
- [3] Openstack, <https://www.openstack.org/>
- [4] kubernetes, <https://kubernetes.io/docs/home/>
- [5] Cinder, <https://docs.openstack.org/cinder/latest/>
- [6] Glance, <https://docs.openstack.org/glance/latest/>
- [7] Manila, <https://specs.openstack.org/openstack/manila-specs/>
- [8] Swift, <https://docs.openstack.org/swift/latest/>
- [9] openstack survey, <https://www.openstack.org/analytics>
- [10] Ceph, <https://ceph.io/>