

클라우드 서비스를 위한 고가용성 대용량 데이터 처리 아키텍처 Implementation of Data processing of the High Availability for Software Architecture of the Cloud Computing

이병엽*, 박준호**, 유재수**

배재대학교 전자상거래학과*, 충북대학교 정보통신공학부 및 충북BIT연구중심대학육성사업단**

Byoung-Yup Lee(bylee@pcu.ac.kr)*, Junho Park(junhopark@chungbuk.ac.kr)**,
Jaesoo Yoo(yjs@chungbuk.ac.kr)**

요약

최근 많은 기관들로부터 클라우드 서비스가 향후 대세적인 IT서비스로의 확고한 기술로서 예견하고 있고, 실질적으로 IT를 리딩하는 많은 벤더 기업들을 주축으로 실질적인 클라우드 서비스를 제공 하고 있다. 따라서 클라우드 사용자는 서비스의 물리적인 위치나, 시스템 환경과 같은 부분들을 관여하지 않고, 스트리지 서비스, 데이터의 사용, 소프트웨어의 사용들을 제공하는 획기적인 서비스로 거듭나고 있다. 한편, 클라우드 컴퓨팅 기술들은 인프라스트럭처에서 요구되는 서비스의 수준, 다양한 시스템에서 요구되는 하드웨어적인 문제들을 벗어나서 자유롭게 원하는 만큼의 IT 리소스를 쉽게 사용할 수 있는 장점을 가지고 있지만, 고가용성 측면에서 반드시 기술적인 해결 방안을 모색하여야 한다. 따라서 본 논문 에서는 고가용성 측면에서의 클라우드 컴퓨팅을 위해 분산 파일 시스템이 갖추어야 할 사항들과 클라우드 컴퓨팅에서 활용 가능한 오픈소스 기반의 하둠 분산 파일 시스템, 메모리 데이터베이스 기술, 고가용성 데이터베이스 시스템을 소개하고 현재 클라우드 컴퓨팅 시장에서 활용되고 있는 분산 파일 시스템을 통한 분산처리 기술을 참고하여 고가용성 대용량 분산 데이터 처리 아키텍처를 클라우드 서비스 측면에서 구현하였다.

■ 중심어 : | 클라우드 | 클러스터 | DBMS | Grid | 분산처리 | 고가용성 |

Abstract

These days, there are more and more IT research institutions which foresee cloud services as the predominant IT service in the near future and there, in fact, are actual cloud services provided by some IT leading vendors. Regardless of physical location of the service and environment of the system, cloud service can provide users with storage services, usage of data and software. On the other hand, cloud service has challenges as well. Even though cloud service has its edge in terms of the extent to which the IT resource can be freely utilized regardless of the confinement of hardware, the availability is another problem to be solved. Hence, this paper is dedicated to tackle the aforementioned issues; prerequisites of cloud computing for distributed file system, open source based Hadoop distributed file system, in-memory database technology and high availability database system. Also the author tries to body out the high availability mass distributed data management architecture in cloud service's perspective using currently used distributed file system in cloud computing market.

■ keyword : | Cluster DBMS | High Availability | Grid |

* 본 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (2012R1A1A2A10042015)

접수번호 : #130108-003

접수일자 : 2013년 01월 08일

심사완료일 : 2013년 01월 18일

교신저자 : 유재수, e-mail : yjs@chungbuk.ac.kr

1. 서론

매년 굴지의 조사기관들은 그 다음해의 10대 기술 트렌드를 발표한다. 이는 기업들이 전략적으로 대응해야 할 기술 및 트렌드로서 향후 3년간 기업에 큰 영향을 미칠 수 있는 기술이라고 정의하고 있다. 영향력은 IT 또는 비즈니스에 대한 큰 변화의 발생 가능성, 막대한 투자가 발생할 가능성, 채택하지 않을 경우 겪을 수 있는 위험성 등이 포함된다. 2013년도의 트렌드로는 모바일 영역, 클라우드 영역, 빅데이터 및 분석 영역, 인메모리 및 통합시스템 영역으로 크게 분류할 수가 있는데, 해당 기술들은 소셜, 모바일, 클라우드 및 정보 등의 상호 융합되고 있는 힘의 결합과정에서 탄생한다고 볼 수 있다. 클라우드와 관련하여 가트너는 먼저 중요 트렌드로서 퍼스널 클라우드(Personal Cloud)를 언급하고 있다. 개인의 서비스와 데이터에 액세스하고 활용하는 터전이 PC에서 퍼스널 클라우드로 옮겨가고 있는 것을 주목하고 있으며, 다수의 사용자가 자신의 모바일 기기에 정보를 저장하지 않고 구글 드라이브, 드롭박스 등 퍼스널 클라우드 서비스를 활용하는 것이 확산되면서, 모바일 기기가 PC를 대체하는 속도가 가속화되고 있다고 보고 있다. 그동안 기업들이 회사의 정보를 어떻게 하면 클라우드로 옮길 수 있을 것인가를 고민해 왔다면, 이제는 얼마나 효율적으로 클라우드를 구축하여 정보를 관리 할 수 있을 것인지를 고민하고 있다는 것이 가트너의 설명이다. 클라우드 서비스 사업자의 입장에서 클라우드가 본격적인 비즈니스 모델로서 성장하기 위해서는 엔터프라이즈급 클라우드 서비스의 등장이 필수적인 것으로 보고 있으며, 이는 현재 기업에서 운용중인 업무가 클라우드에서도 그대로 운용 될 수 있는 수준의 성능, 안정성, 보안성관련 서비스 수준이 지켜질 수 있어야 한다는 것과 기존의 업무가 그대로 옮겨갈 수 있는 이식성이 보장되어야 하는 것을 의미 한다. 최근 클라우드 컴퓨팅 시장에 진출했거나 진출을 선언한 Google, IBM, Microsoft, Oracle 등과 같은 글로벌 IT 기업들은 그 동안의 하드웨어, 소프트웨어의 기술력을 바탕으로 클라우드 컴퓨팅을 제공하는 데 필요한 IT 인프라 및 소프트웨어의 서비스 형태를 지속적으로 확

충해 나가고 있으며 제반 기술들을 개발하고 향상시키기 위해 막대한 개발 투자를 쏟아 붓고 있다[6].

또한 클라우드 컴퓨팅을 위해 막대한 양의 데이터를 수많은 서버들에 분산 저장하고 관리하는 분산 파일 시스템 기술은 그러한 기술 중의 하나로 인식 될 수밖에 없다. 클라우드 컴퓨팅에서 요구하는 분산 파일 시스템은 단순히 데이터의 저장과 관리만을 하는 것이 아니라 하드웨어의 장애에 유연하게 대처하여 서비스가 중단되지 않도록 하는 고가용성의 시스템 아키텍처가 필요하며 또한 적절한 병렬 처리를 통해 서비스가 요구하는 성능도 클라이언트를 만족시킬 수 있어야 한다. 현재 수많은 분산 파일 시스템들이 존재하지만 최근까지 클라우드 컴퓨팅에 활용되고 있는 분산 파일 시스템은 그리 많지 않다. 대표적으로 Google의 파일 시스템[6]과 Apache의 Hadoop 분산 파일 시스템[5]이 클라우드 컴퓨팅에서 가장 잘 알려진 분산 파일 시스템이다[19].

클라우드 컴퓨팅을 위해 분산 파일 시스템이 갖추어야 할 사항들은 실로 다양하다. 이러한 사항들을 도출하기 위해 클라우드 컴퓨팅의 대표적인 특징 들을 분산 파일 시스템 측면에서 살펴보면 다음과 같이 몇 가지로 크게 요약해 볼 수 있다[6].

첫째, 클라우드 컴퓨팅을 위한 분산 파일 시스템들이 다루는 데이터와 서버의 규모는 기존의 분산 파일 시스템들과는 비교가 되지 않을 정도로 거대하다는 점이다. 대부분의 핵심적인 상황들은 여기에서 발생하며 비용적인 측면에서의 효율성, 지속적으로 증가하는 데이터의 수용, 빈번하게 발생하는 고장에 대한 대처, 관리의 편리성과 같은 사항들이 해당된다[5].

둘째, 점점 다양해지고 있는 사용자들의 요구를 충족시키기 위해 제공되는 클라우드 서비스들이 분산 파일 시스템에게 만족할 만한 데이터의 입출력과 처리 성능을 요구한다는 점이다. 여기에는 대용량 파일에 대한 신속한 입출력 성능, 네트워크 위상 구조 인식을 통한 데이터 최적 배치, 효과적인 캐시의 사용, 순간적으로 집중되는 부하의 유연한 대처와 같은 사항들이 해당된다.

셋째, 클라우드 컴퓨팅을 활용하고자 하는 기업들뿐만 아니라 일반 사용자들은 데이터에 대한 보안문제를 가장 불안해하고 있으며, 확실하고 안전한 보안 체계를

요구한다는 점이다. 여기에는 데이터에 대한 암호화, 데이터 영역에 대한 사용자간 엄격한 접근 제어, 사용자 데이터에 대한 관리자의 접근 제한 같은 사항들이 해당된다.

셋째, 사용자가 저장한 데이터에 오류가 발생하지 않도록 방지하고, 저장 공간을 최적으로 사용하기 위한 방법들을 요구한다는 점이다. 여기에는 데이터 오류 감지 및 복구, 데이터 중복 제거와 같은 사항들이 해당된다[5][6]. 본 논문의 목적은 최근 IT 메가트렌드로 이슈화가 되고 있는 클라우드 서비스 구축을 위한 기술적인 측면에서의 서비스 성능의 극대화를 위한 파일시스템과 고가용성 아키텍처를 제안 하였다. 또한 특징적으로 데이터베이스의 데이터 서비스의 확장 적으로 비정형 데이터의 원활한 서비스를 위한 오픈소스 기반의 하둡 시스템과 메모리 데이터베이스를 기반으로 클라우드 컴퓨팅의 핵심 기술로 고가용성 서비스를 가능케 하는 그리드 기술 구성들을 소프트웨어 아키텍처 측면에서 제시 하였다. 마지막 결론에서는 본 논문을 통해 정리된 고가용성 클라우드 서비스를 위한 기술을 제시 하였다.

II. 본론(클라우드 핵심 기술 및 기술 구조)

1. 클라우드 컴퓨팅의 핵심 기술

클라우드 컴퓨팅이란 인터넷 기술을 활용하여 '가상화된 IT 자원을 서비스'로 제공하는 컴퓨팅으로, 사용자는 IT 자원을 필요한 만큼 빌려서 사용하고, 서비스 부하에 따라서 실시간 확장성을 지원받으며, 사용한 만큼 비용을 지불하는 컴퓨팅을 말한다[5][15-17].

따라서 클라우드의 구현을 위한 핵심 기술들을 나열하면 첫째, 애플리케이션에서 하드웨어에 이르는 모든 솔루션 전반에서 엔지니어링 역량을 활용하여 가능한 최강의 클라우드 기술들이 정의 되어야 한다. 이는 단순히 버추얼머신을 이용하는 것만이 아니라, 모든 애플리케이션을 수정 없이 그대로 실행 할 수 있어야 한다. 그 이유는 미션크리티컬한 인프라 전체를 관리하고 보호하여 적절히 운영될 수 있도록 하여야 한다. 두 번째

핵심기술, 서버 및 스토리지에 투자하여 사람들이 원하는 경우 기존 방식대로의 소프트웨어를 이용하여 인프라를 개발 하거나, 기존 인프라에 추가 또는 보완 할 수 있어야 한다. 그 이후 가능한 최고의 성능, 보안 및 가용성을 기술에 적용 하여야 한다. 셋째, 소프트웨어 스택에 대한 지식을 활용하여 압축, 보안, 성능, 테스트 및 안정성 향상 등의 영역에서 하드웨어의 근본적인 변화를 단행하여 인프라의 개선을 하여야 한다. 넷째, 소프트웨어와 하드웨어를 결합하여 성능과 기능을 제공하는 서비스를 근본적으로 구현 하여야 한다.

1.1 저비용분산파일 시스템

클라우드 컴퓨팅에서 제공하는 서비스는 제한적인 것은 아니지만, SaaS, PaaS, IaaS 세 가지를 가장 대표적인 서비스로 분류한다. 애플리케이션을 서비스 대상으로 하는 SaaS는 클라우드 컴퓨팅 서비스 사업자가 인터넷을 통해 소프트웨어를 제공하고, 사용자가 인터넷상에서 이에 원격 접속해 해당 소프트웨어를 활용하는 모델이다[11-13]. 클라우드 컴퓨팅에서 다루어지는 데이터는 꾸준히 증가하고 있으며 이러한 데이터를 저장하고 관리하기 위해 막대한 IT 인프라가 요구된다. 만일 고가의 서버와 고속의 네트워크를 활용하여 이러한 인프라를 구축하고 유지할 수 있다면 두말할 나위 없이 좋겠지만 비용을 고려했을 때 현실적으로 쉽지 않은 것이 사실이다. 따라서 클라우드 컴퓨팅 시장에 뛰어들어 기업에게 있어서 이러한 인프라를 구축하고 유지하는 데 드는 비용을 최소화하는 것은 가장 중요한 경쟁력 중의 하나이다. 이를 위해 저렴한 서버들과 네트워크를 활용하여 비용을 대폭적으로 줄이면 서도 비슷한 성능을 낼 수 있는 분산 파일 시스템 기술들을 개발하고 적용해야 한다.

1.2 고가용성 컴퓨팅

앞서 언급하였던 저비용 측면에서 저가의 장비들을 대규모로 활용하여 시스템을 구축함으로써 고장이 빈번하게 발생할 수밖에 없다. 물론 고가의 장비들은 저가의 장비들에 비해 고장이 좀 더 적게 발생 하기는 하지만 고장이 발생한다는 사실 자체는 변하지 않는다.

중요한 점은 이러한 고장으로 인해 시스템 전체 또는 서비스를 중단시키는 상황이 발생하지 않아야 한다는 것이다. 분산 파일 시스템은 모니터링을 통하여 고장이 발생한 상황을 인지하고 적절하게 대처함으로써 서비스가 중단되는 상황을 방지해야 한다. 또한 고장이 발생한 서버나 디스크로 인해 데이터가 유실되는 것을 방지하기 위해 데이터의 중복 저장과 같은 안정적인 데이터 보관 방법이 갖춰져야 한다.

1.3 고성능 컴퓨팅

기존의 PC 환경에서 사용하던 수많은 응용들이 클라우드 서비스를 통해 클라우드 컴퓨팅 속으로 들어오고 있다. 이러한 응용들의 접근 패턴은 분산 파일 시스템의 설계에 있어서 성능과 밀접한 관련을 갖는다. 예를 들면 사진, 동영상과 같은 대용량의 데이터들을 다루는 클라우드 서비스들이 최근 많은 관심을 끌고 있는데, 분산 파일 시스템의 입장에서 바라볼 때 이러한 응용들은 대용량의 데이터를 순차적으로 입출력하는 패턴을 가지며, 대부분 저장된 데이터를 읽어가는 연산을 수행한다. 이러한 패턴에 최적화된 분산 파일 시스템은 이러한 패턴에 있어서는 최상의 성능을 나타내겠지만, 저용량의 임의적인 입출력 패턴의 경우에는 현저하게 성능이 저하될 수도 있다. 결과적으로 분산 파일 시스템이 모든 접근패턴에 최고의 성능을 보장할 수 있으면 좋겠지만 사실상 그것은 상당히 어려우며 분산 파일 시스템 상위에서 접근 패턴 자체를 분산 파일 시스템이 지원하는 최적의 접근 패턴으로 변형하여 최상의 성능을 얻어내거나 적절한 다른 분산 파일 시스템을 활용하는 것이 보다 현실적인 대안이 될 수 있다. 분산 파일 시스템은 서버, 스위치, 랙 등의 네트워크 위상 구조를 인식하고 이를 이용하여 최적으로 데이터를 배치함으로써 클라이언트의 요청을 빠르게 처리할 수 있어야 한다. 또한 메모리 캐시를 효율적으로 활용하여 디스크 입출력을 최소화함으로써 성능을 향상시킬 수 있다. 앞서 예로 들었던 클라우드 서비스에서 수많은 사용자들이 관심을 가지고 짧은 기간 안에 접근하게 되는 hot 데이터라 불리는 특정 데이터는 과도한 접근으로 인해 서비스가 불가능한 상태에 빠질 수도 있는데, 이러한 것

을 방지하기 위한 방법이 갖춰져야 한다.

2. 클라우드 컴퓨팅 분산파일 시스템 기술

현재까지 무수히 많은 분산 파일 시스템들이 존재하고 있으며 여전히 진화를 거듭하고 있다. 최근 클라우드 컴퓨팅이 부각되면서 기존의 분산 파일 시스템 기술들이 다양한 형태로 활용되고 있으며, 적절한 변형을 통해 클라우드 컴퓨팅으로의 접근을 피하고 있다[14]. 이 장에서 살펴볼 클라우드 컴퓨팅에 활용되고 있거나 활용이 가능한 분산 파일 시스템들은 앞서 살펴보았던 요구 사항들을 반영하고 있다.

2.1 해외 클라우드 서비스 업체의 기술 동향

아마존은 1851년부터 1980년까지 1,100만 건에 달하는 뉴욕타임스 기사를 클라우드 컴퓨팅을 통해 단 하루 만에 서비스를 완료함으로써 세계적인 클라우드 컴퓨팅 성공사례로 주목을 받았다[3]. Amazon S3 파일 시스템은 Amazon 웹 서비스에 의해 제공되는 온라인 스토리지 웹 서비스인 Amazon S3을 위한 분산 파일 시스템이다. 현재 Amazon S3 파일 시스템의 구조는 Amazon의 특허를 통해 살펴볼 수 있다[5]. S3 파일 시스템은 5 기가바이트 크기의 객체를 저장할 수 있으며 각 객체 당 최대 2 킬로바이트 크기의 메타데이터를 수반한다[5][10]. [그림 1]에서 처럼 각 객체는 버킷(bucket)들로 구성되며 각 버킷과 객체는 ACL을 통해 접근이 관리된다. 아마존 분산파일 시스템의 스토리지 모델은 사용자의 데이터를 저장하는 객체들과 객체들의 집합인 버킷들로 구성된다.

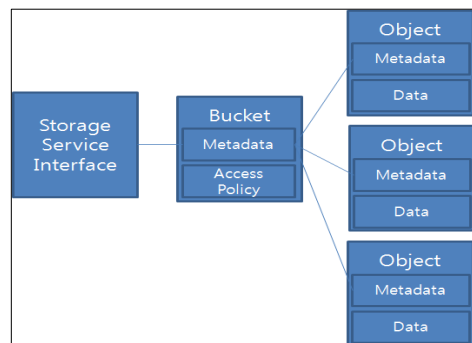


그림 1. 아마존의 스토리지 모델

사용자는 스토리지 서비스를 위해 제공되는 스토리지 서비스 인터페이스를 사용하여 버킷에 접근 할 수 있다. 스토리지 서비스 인터페이스는 우리가 일반적으로 사용하는 웹 주소 표현 방식을 그대로 사용한다. 버킷은 메타데이터와 접근 정책으로 이루어진다. 버킷에 포함된 메타데이터는 버킷의 생성 시간, 생성자의 ID, 버킷이 객체들을 포함하고 있는지에 대한 정보, 버킷에 포함된 객체들의 총 크기, 버킷에 대한 사용자의 접근 기록, 버킷과 관련된 청구 기록 등이다. 객체는 메타데이터와 데이터로 이루어진다. 객체에 포함된 메타데이터는 객체의 생성 시간, 객체의 크기, 객체에 저장된 데이터의 타입, 객체의 사용과 이력 정보, 접근 정책 등이다. 각 객체들은 스토리지 서비스 시스템에서 키(key)와 로케이터(locator)에 의해 구별된다. 키는 버킷 단위로 유일한 값을 가지는 반면 로케이터는 모든 버킷에 대해 유일한 값을 갖는다[1].

Google은 클라우드 서비스를 제공하기 위해 자체 개발한 분산 파일 시스템인 Google 파일 시스템(이하 GFS)을 사용하고 있다[6][7]. GFS는 여러 클라이언트에 의해 접근되는 단일 마스터와 여러 청크(chunk) 서버들로 구성되며, 저가의 리눅스 서버 상에서 사용자 수준 프로세스로 구동된다. 마스터는 모든 파일 시스템 메타데이터를 관리하고 시스템 전반적인 동작을 제어하는 역할을 수행하며, 청크 서버들은 실제 데이터를 보관하고 클라이언트의 입출력 요청을 처리하는 역할을 수행한다. 이처럼 GFS는 메타데이터와 데이터의 흐름을 분리한 비대칭형 구조를 통해 확장성과 고성능을 보장하고 빈번하게 발생하는 장애에 적절히 대처함으로써 안정성을 확보한다. GFS에서 파일들은 청크라 불리는 64 메가바이트 크기의 단위로 나뉘어 관리되며 각 청크들은 여러 청크 서버에 분산되어 저장된다[8][9]. 또한 각 청크에 대한 다수의 사본을 여러 청크 서버에 분산하여 저장함으로써 장애로 인한 데이터의 손실을 방지한다. GFS는 이렇게 청크라 불리는 큰 단위로 데이터를 관리함으로써 마스터가 관리해야 하는 메타데이터의 수를 줄이고 메타데이터를 교환하기 위해 필요로 하는 네트워크 부하를 감소시킨다[2][8].

MS는 클라우드 컴퓨팅을 최근 10대 공략 분야로 선

정하여 클라우드 컴퓨팅 시장을 공략할 계획이다. 공용(Public)클라우드 시장 공략을 위해 자사의 "클라우드 운영체제"인 윈도우 애저(Azure)를 상용버전으로 제공하고 있으며, 이는 조직들이 자체적으로 클라우드 기반 애플리케이션을 구축해서 구동할 수 있게 해 준다. 또한 사설(Private)클라우드 시장공략을 위하여 윈도우 서버 2008 R2에 포함된 하이퍼-V를 통해 기업 데이터센터를 가상화 하고, 시스템 관리는 자사의 '시스템 센터'로 하여 추진할 예정이다. 또한 모바일 클라우드 서비스로 My Phone 서비스를 제공 중이다. My Phone 서비스는 윈도우 모바일 기반 휴대폰과 웹 간에 정보를 동기화 할 수 있도록 해주는 서비스로 사진이나 비디오, 텍스트 메시지, 일정표 등의 데이터를 저장할 수 있는 공간을 제공하며, 사용자는 이 서비스를 휴대폰의 데이터를 백업하는 용도 외에도 다른 사용자와의 데이터 공유에도 사용 할 수 있다[11][17].

IBM은 클라우드 컴퓨팅 사업을 유전공학, 멀티미디어, 금융 등 다양한 분야로 확산시키는데 적극 나서고 있다. 최근 IBM은 미국 미주리대학교와 대규모 유전 연구를 위한 클라우드 컴퓨팅 환경 개발 계획을 발표하였다. 또한 말레이시아 멀티미디어개발공사(MDeC)와 함께 말레이시아에 클라우드 컴퓨팅 애니메이션센터를 구축하였다. 최근 유럽연합(EU) 및 유럽 대학들과 인터넷 기반 서비스의 호스팅과 유지보수 비용을 낮추는 새로운 클라우드 컴퓨팅 모델 개발을 위해 컨소시엄을 구성 하여 유연한 클라우드 환경에서의 다양한 하드웨어(HW)와 소프트웨어(SW)를 지원하는 인터넷 기반 서비스 관리와 새로운 컴퓨터 과학 모델을 연구할 예정이다[11].

2.2 국내 클라우드 서비스 업체 기술 동향

국내 초기 클라우드 시장은 통신사업자들과 IDC(Internet Data Center) 서비스 사업자들이 주도하는 중소기업 및 개인사용자 대상의 퍼블릭 클라우드 중심이었다. 이에 따라 한국IDC는 클라우드 시장 기회는 현 단계에서는 프라이빗 클라우드에 있는 것으로 분석했다. 퍼블릭 클라우드 서비스의 경우에는 성공적인 유료 비즈니스 모델이 자리를 잡고 업계가 사업성을 확신하

기까지에는 시간이 더 소요될 것으로 전망됐다. 실제로 한국IDC가 실시한 조사에 따르면 국내 대기업을 대상으로 클라우드 도입 계획에 대한 질의 결과 응답 기업의 22%가 이미 도입했거나 도입할 계획이 있었으며 이들 중 76%는 프라이빗 클라우드에 보다 관심이 있는 것으로 나타났다. 이 같은 수치는 전년과 비교해 실제 도입과 도입 예정 비중이 약 10%포인트 가량 증가한 것이다. 하지만 퍼블릭 클라우드 서비스의 주 수요층으로 기대되는 중소기업들을 대상으로 한 조사에서는 도입 관심도나 의향이 현저히 낮은 것으로 나타나 퍼블릭 클라우드 서비스 시장의 성장은 다소 더딜 것으로 분석됐다. 그러나 통신업계가 모바일 클라우드나 개인용 클라우드 서비스를 위한 투자를 확대하고 있어 퍼블릭 클라우드 서비스 환경 구축을 위한 인프라와 솔루션 시장은 성장세를 보이고 있다. 국내에서는 개인이나 불특정 소기업 보다는 계열사 대상의 클라우드 서비스가 보다 지속 가능한 수익 모델로 인식됨에 따라 전통적인 스토리지 솔루션 업체들은 금융 지주사와 대형 그룹사 기반의 시스템통합(SI) 기업들과 연계해 프라이빗 클라우드 시장을 확대하는데 주력하고 있다. 스토리지 업체가 가상 환경에서 네트워크 스토리지 관리를 간소화하고 가변적인 클라우드 환경에서 유연성과 확장성, 고가용성을 확보할 수 있는 통합 플랫폼으로 시장 선점 경쟁이 치열할 것으로 예상된다. 한편, 클라우드 스토리지 솔루션은 퍼블릭 클라우드 서비스와 프라이빗 클라우드 환경을 지원하기 위해 도입된 디스크 스토리지 시스템과 하드디스크드라이브(HDD), 스토리지 소프트웨어, 관련 서비스를 모두 포함하는 개념이다. 이 같은 성장세는 데이터 중복제거와 원거리통신망(WAN) 최적화, 가상화 스케일아웃 파일시스템, 그리드 아키텍처 등과 같은 클라우드 스토리지 구현을 가능하게 하는 기술들에 대한 관심이 높아지고 있기 때문으로 분석됐다[20].

국내 클라우드 컴퓨팅 시장은 아직 미성숙 단계이지만 최근 국내 대기업을 중심으로 대형 IT서비스 기업들과 통신사 등을 중심으로 클라우드 컴퓨팅 사업을 본격화 되어 클라우드 컴퓨팅에 대한 투자가 확대되고, 서비스 도입을 위한 준비가 진행되고 있다. 그러나 원천 핵심 기술 보유 부재에 따른 기술 격차, 국내 생태계 부

재에 따른 외국 기업의 국내 시장 잠식위협, 표준화 및 서비스 도입에 친화적인 법/제도 기반 미비에 따른 사회적 신뢰기반 취약 및 국가 차원의 통합된 대응체계 및 구심점 부재 등이 문제점으로 지적되고 있다. 중소기업 및 벤처기업들도 클라우드 열기에 동승하기 위해 활발히 움직이고 있다. [그림 2]은 국내 클라우드 컴퓨팅 시장 현황 및 전망치를 보여 주고 있다[20].

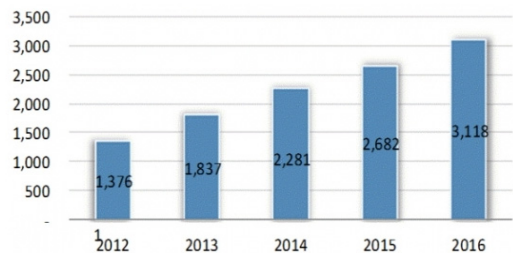


그림 2. 국내 클라우드 컴퓨팅 시장 현황 (단위: 억원)

3. 클라우드 컴퓨팅 기술 구성

3.1 클라우드 서비스 데이터 분산처리 기술 활용

안정성과 성능향상을 위해 HW Vendor에서 제시하고 있는 솔루션은 단적으로 볼 때 MPP(Massively parallel Processing)와 SMP(Symmetric multi processing) 머신으로 이어져 오면서, 업무에 따라 Scale-up 과 Scale-out 형태로 확장성이 진화해 왔다고 볼 수 있다. Scale-up이란 사용하고 있는 머신을 보다 큰 상위 기종으로 업그레이드를 하거나 기존 운영서버에 각종 리소스, 즉 CPU나 Memory 확장 및 빠른 접근 속도를 주장하는 하드디스크를 추가로 장착하여 확장하는 방식이다[4]. 이러한 Scale-up방식을 채택함으로써 얻을 수 있는 장점으로는 시스템 구조나 개발 및 운영에 변화가 없으며, 현재 사용하고 있는 Application이나 Data Source, Middleware등에 대한 수정이 전혀 필요 없다. 가장 손쉽게 당면하는 성능의 문제점을 해결할 수 있는 방법으로 자원이 추가되는 동안 서버만 잠시 다운시켜 빠른 시간 내에 확장을 할 수 있다는 장점이 있다. 그러나 단점으로는 확장 시 요구되는 비용이 상대적으로 크며, 지속적으로 확장을 하는데 한계가

있다는 점이다. 또한 SOPF(Single Point of Failure)에 대한 높은 장애 비용을 감안해야 한다는 점 또한 무시할 수 없다. 데이터의 트랜잭션의 성능을 높이기 위해 고려가 되고 있는 최근의 동향으로, 데이터베이스 서버를 미들티어, 즉 메모리상에 올려놓고 트랜잭션을 처리함으로써 Disk I/O에 대한 비용을 줄이고 메모리 I/O를 통해 처리토록 하는 방안으로 고려가 되고 있는 영역이 메모리 DBMS이다. 메모리 데이터베이스는 Disk 기반으로 트랜잭션을 처리할 때 발생하는 Disk I/O 부분을 메모리상에 데이터베이스를 둬서 I/O극대화로서 성능 문제를 해결하고자 하는 것이다. 주로 Scale-up을 위해 고려되는 솔루션이며, Disk기반의 DBMS와 비교를 해 볼 때 나름대로의 장단점이 있다고 볼 수 있다. 따라서 클라우드 컴퓨팅 분산 파일시스템의 극대화를 위해 메모리 데이터베이스의 활용에 대한 기술을 구성하고자 한다.

3.2 메모리 분산 그리드 아키텍처

클라우드 서비스의 구축하기 위해서는 다양한 기술 요건들이 필요하다. 본 논문의 본문을 통해 설명된 클라우드 기술 중 무중단 클라우드 서비스의 측면에서 가장 기본이 되는 기술이 그리드의 기술이다. 또한 시스템, 스토리지, 어플리케이션, 데이터베이스의 수직적, 수평적 노드의 확장에 있어서 가장 유연한 기술 구조가 그리드 기술 이다. 따라서 본 논문에서는 그리드 기술 구조를 기반으로 한 고가용성 아키텍처를 최상의 클라우드 서비스를 위한 소프트웨어 기술요건으로 제시하고자 한다. 데이터 소스로부터 읽혀온 데이터를 여러 미들티어 서버의 메모리상에 캐시 형태로 상주 시키면서, 어떤 노드를 통해 데이터를 요구하더라도 그리드로 연결된 노드상의 메모리에서 데이터를 찾아 빠르게 서비스하는 구조로 제언 될 수 있다. 따라서 메모리 구조의 특징을 살펴보면, 첫째 데이터 분산이 투명하게 처리가 되며, 각 노드의 메모리를 로드 분산을 자동 처리하는 장점이 있다. 또한 병렬처리 방식으로 데이터를 로딩 할 수 있으며, 데이터를 조회, 데이터 처리가 가능하며, 수정이나 입력 등 트랜잭션을 위한 처리일 경우 Queue에 보관하여 원하는 시기에 데이터베이스에 저

장시킬 수 있다. 클러스터링 되어 있는 각 노드에 검색 되어질 데이터를 Primary 데이터로 정의하며, 각 노드마다 Primary 데이터를 메모리에 저장하고 이 Primary 데이터에 대한 백업을 클러스터내의 다른 노드들에게 분산되어 백업이 된다[그림 3]. 그러므로 사용자는 어떤 노드를 통해 접속하더라도 동일한 논리적인 데이터 뷰를 볼 수가 있게 된다. 두 번째는 클러스터 상의 여러 미들티어 노드들에 대해 가용상태를 서로 확인하여, 가용 상태가 불안정할 때는 나머지 노드들이 자체 상태를 진단하여, 불안정한 노드가 발생되었을 경우 클러스터 상에서 해당 노드를 제외시키며 남은 노드들이 불안정한 노드 상의 캐시에 있었던 데이터를 남은 노드들로 자동 재 분산시킴으로 Fault tolerance를 할 수 있는 아키텍처를 가지고 있다.

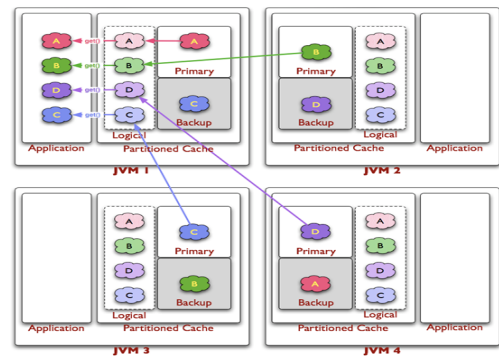


그림 3. 메모리 분산 파일 가용성 구조

3.3 데이터베이스 분산 처리 아키텍처

데이터베이스 그리드 컴퓨팅 기술은 차세대 비즈니스 모델을 위한 핵심인프라 기술로 인식되고 있다. 데이터베이스의 그리드 환경의 고 가용성을 구축해야 하는 크리티컬한 비즈니스의 IT환경의 요구에 따라 O사는 과거 OPS(Oracle Parallel Server)를 지원하였으며 현재 버전에서는 각 노드간 캐시의 일치성을 보장하기 위한 서버간의 통신 방식을 디스크를 이용한 방식에서 초고속 인터커넥트를 이용한 캐시 퓨전(Cache Fusion)으로 변경하면서 고가용성 구현의 완성도를 높였다. 현재 O사의 10g에서부터 그리드 컴퓨팅을 지원하는 더욱 발전된 RAC(real application cluster)구조를 상용화 하

었다. O사의 RAC는 다중 노드를 지원하는 공유 디스크 구조를 사용하여 한층 강화된 HA 솔루션을 데이터 베이스에 제공 한다.

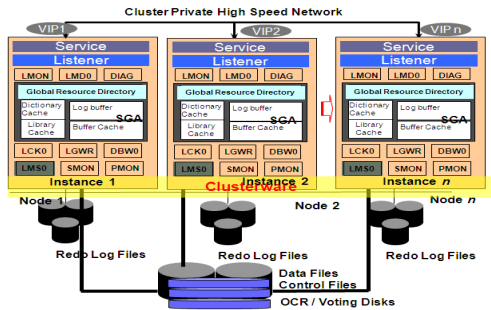


그림 4. 데이터베이스 Grid 구현 아키텍처

RAC는 동일 데이터베이스 또는 스토리지를 여러 인스턴스에서 동시에 액세스할 수 있는 장점을 가지고 있으며, 시스템 확장 즉 유기적으로 인스턴스 노드의 추가가 가능하기 때문에 탁월한 로드밸런싱 및 향상된 성능을 구현 할 수 있다. 또한 RAC 구조는 모든 노드가 동일한 데이터베이스를 액세스하기 때문에 한 인스턴스에서 장애가 발생해도 데이터베이스에 대한 액세스가 손실되지 않는 장점을 가지고 있다. 또한 M사는 비공유 구조(shared nothing architecture)를 하여 데이터 분할을 사용하는 클러스터링을 제공한다. 이 방법은 특히 데이터웨어하우징 애플리케이션의 확장성에 유용하다. 하지만 이 경우 하나의 노드가 전체 애플리케이션에 영향을 미칠 수 있으므로 HA를 위해서는 모든 노드에 데이터가 균일하게 분산되도록 해야 한다. 또한 각각의 노드의 장애시 데이터의 사용이 불가능하며, 새로운 노드가 추가시 데이터의 재분배(repartition)를 반드시 수행하여야 한다. 따라서 독립서버구조(federated)의 구성은 자동적인 장애극복을 위해서는 추가적인 시스템(witness)이 필요하며, 단하나의 미러서버(mirror server)만을 허용한다. 또한 I사 pureScale는 최근 애플리케이션과 데이터베이스의 고가용성 및 확장성을 지원하는 클러스터링 기술인 DB2 퓨터스케일(pure scale)을 구현한다. DB2의 퓨어스케일은 기존 메인프레임의 디스크 클러스터링 기술인 시스플렉스를 유닉스

환경에서도 이용할 수 있도록 구현한 것으로 중요하고 민감함 업무를 진행시 시스템 확장을 통해 비즈니스 요건을 충족시킬 수 있는 솔루션이다[11].

3.4 허둠(Hadoop) 분산처리 파일 시스템

Hadoop 분산 파일 시스템(이하 HDFS)은 신뢰성과 확장성을 갖춘 분산 컴퓨팅을 위한 오픈 소스 소프트웨어 개발 프로젝트인 Hadoop에서 분산 컴퓨팅 프레임워크를 지원하기 위해 개발된 분산 파일 시스템이다. HDFS는 현재 Amazon, IBM, Yahoo등과 같은 글로벌 IT 기업들의 클라우드 컴퓨팅 플랫폼의 기반이 되는 분산 파일 시스템으로 가장 널리 활용이 되고 있다[6]. HDFS는 GFS를 본보기로 삼아 개발된 분산 파일 시스템으로 플랫폼간의 이식성을 보장하기 위해 자바를 사용하여 구현되었으며 공개 소프트웨어이다. HDFS는 이름만 다를 뿐 구글의 파일 시스템과 동일한 구조와 기능을 제공한다. 네임노드는 구글 파일시스템의 마스터와 동일하며 데이터 노드는 구글 파일시스템의 청크 서버와 동일하다. 또한 HDFS 에서 블록이라 불리는 파일 관리 단위는 구글 파일시스템의 청크와 동일하다.

(1) 클라이언트

HDFS 클라이언트는 HDFS 파일을 생성할 때 우선 자신의 로컬 저장 영역에 한 블록 크기의 임시 파일을 생성하고 그 파일에 먼저 데이터를 기록한다. HDFS 클라이언트에서 HDFS 파일이 생성되는 과정을 살펴보면 다음과 같다.

- 한 블록 크기의 임시 파일이 데이터로 모두 채워지거나 더 이상 기록될 데이터가 없으면 클라이언트는 네임노드에게 메타데이터에 반영해 줄 것을 요청하고 사본이 위치할 데이터노드들의 목록을 얻어온다.
- 얻어온 데이터노드들의 목록에서 자신과 가장 근접한 첫 번째 데이터노드로 임시 파일과 나머지 데이터 노드들의 리스트를 전송한다.
- 첫 번째 데이터노드는 4 킬로바이트 단위로 데이터를 받아 자신의 로컬 파일 시스템에 파일로 저장하여 곧바로 다음 데이터노드로 데이터를 전송한다.

- 이러한 파이프라인 형태로 임시 파일의 모든 데이터를 목록에 포함된 데이터노드들로 전송하고 모든 데이터노드들이 정상적으로 그 데이터를 저장했는지 확인한다.
- 클라이언트는 모든 데이터노드들에 데이터가 정상적으로 저장된 것이 확인되면 네임노드에게 생성 완료를 알린다.

(2) 네임노드

간결하게 설계된 메타데이터 구조를 기반으로 모든 메타데이터를 메모리상에 유지하는 네임노드는 메타데이터를 자신의 로컬 파일 시스템을 사용하여 두 개의 파일로 저장한다. 첫 번째 파일은 네임스페이스, 접근 제어 정보, 파일과 블록들 간의 사상 정보, 블록들의 위치 정보와 같은 파일 시스템 메타데이터를 저장하는 FsImage라 불리는 파일이며, 두 번째 파일은 FsImage 파일이 기록된 시점 이후에 발생하는 모든 변경 사항을 기록하는 Edit-Log라 불리는 트랜잭션 로그 파일이다. 네임노드는 기동 시에 메타데이터를 메모리상에 구축하기 위해 FsImage 와 Edit-Log 파일을 사용하며, 구축이 완료된 후에는 현재 구축된 메타데이터를 FsImage에 저장하고 Edit-Log 파일에 체크포인트를 기록한다. 사본을 어떤 데이터노드에 배치할 것인가와 클라이언트에게 어떤 데이터노드로부터 데이터를 읽도록 추천할 것인가를 결정하는 정책은 안정성과 성능을 높이는 데 매우 밀접한 관련을 가진다. HDFS는 랙 인지(rack awareness)라 불리는 네트워크 위상 인지를 활용하여 기본적으로 3개의 사본을 유지하는 상황에서 동일한 랙의 서로 다른 서버에 한 개씩, 다른 랙에 있는 서버에 나머지 한 개를 배치하며, 이러한 배치 정책과 더불어 읽기 요청을 한 클라이언트에게 가장 가까운 데이터노드를 추천한다. 네임노드는 주기적으로 모든 데이터노드들로부터 자신의 상태를 나타내는 heartbeat 과 자신이 관리하고 있는 블록들의 목록인 block report 를 받는다. 네임노드는 heartbeat을 통해 장애가 발생한 데이터노드를 감지하고 그 데이터노드가 포함하고 있는 블록들에 대해 다른 데이터노드에 있는 사본을 활용하여 또 다른 데이터노드로 사본을 복제하는 방법을 통

해 안정성을 확보한다. 또한 block report와 메타데이터로 유지하고 있는 블록들의 정보를 비교하여 일치하지 않는 블록들을 삭제하도록 데이터 노드에게 알려준다. 네임노드는 파일이나 디렉터리에 대한 삭제 요청에 대해 곧바로 삭제하지 않고 임시 디렉터리로 옮겨서 삭제된 것처럼 보이도록 하고 설정된 일정 시간이 지나면 실제로 삭제를 수행한다. 이러한 지연 삭제는 실제로 삭제되지 않은 데이터의 복구를 빠르게 할 수 있으며, 곧바로 데이터를 삭제할 때 발생하는 부하를 줄일 수 있는 장점을 지닌다.

(3) 데이터노드

HDFS 파일들은 블록의 단위로 서로 다른 여러 데이터 노드들에 분산되어 저장되며, 해당 데이터노드들은 로컬 파일 시스템의 파일로 블록들을 저장하고 관리한다. 데이터노드는 디렉터리 당 최적의 파일수를 고려하여 서브디렉토리들을 구성하고 블록에 해당되는 파일들을 적절한 위치에 저장한다. 또한 데이터의 무결성을 보장하기 위해 블록에 해당되는 파일을 저장할 때 그 파일에 대한 체크섬(checksum)을 별도의 숨겨진 파일로 저장한다. 클라이언트가 어떤 블록에 대한 읽기를 요청하면 데이터노드는 블록에 해당되는 파일과 숨겨진 체크섬 파일을 함께 전달한다.

3.5 대용량 데이터 처리 아키텍처

앞서 살펴본 메모리데이터베이스의 기술, OLTP의 성능을 극대화 하기위한 클러스터링 기술, 비정형데이터의 RDBMS의 한계를 극복하기 위한 형태로의 수평적 확장성을 가지고 있고, RDBMS와는 다른 파일간의 릴레이션이 없는 Hadoop의 파일 시스템의 구조를 이용하여 [그림 5]와 같은 클라우드 데이터 서비스를 위한 대용량 파일 처리 시스템 아키텍처를 구현하였다. 클라우드를 데이터 서비스를 위한 가장 중요한 가용성에 대한 부분의 일정부분의 요구량과 트래픽에 대한 I/O를 적절하게 대응해야 하는 클라우드의 서비스의 특성상 위 아키텍처는 고가용성 서비스를 위한 OLTP 데이터베이스 서비스는 clustering 구조를 통해 데이터베이스 서버간의 상호 로드밸런싱을 할 수 있는 구조로 설계되

었으며, OLTP 및 어플리케이션의 데이터 서비스의 성능 극대화를 위한 메모리 데이터베이스 기술을 활용하였다. 또한 비정형화된, 데이터의 스키마가 필요 없는 대용량의 데이터를 처리하기 위해 오픈소스 기반으로 구글의 클라우드 서비스에서 대용량 데이터 처리로 검증된 대용량 분산 파일 처리 시스템인 Hadoop파일 시스템을 채택하여 아키텍처를 구현하였다.

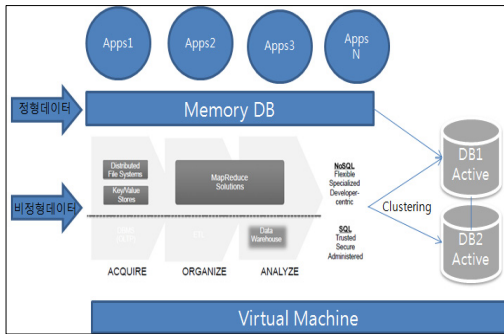


그림 5. 대용량 분산 파일 처리 시스템 아키텍처

III. 결론

클라우드 컴퓨팅은 그리드 컴퓨팅 기술을 모태로 하여 발전적인 서비스의 다양한 모델로 발전되고 있으나, 아직까지는 클라우드를 위한 다양한 기본 기술들을 조건들이 충족되어야만 하는 것이 당면 과제이기도 하다. 예를 들면 시큐어 플랫폼 기술, 고가용성 인프라 기술, 네트워크 및 스토리지 가상화 기술, 모바일 클라우드 분야에서는 아직도 가능한 여러 가지 클라우드 서비스 분야들이 존재하고 있다. 본 논문에서는 클라우드 컴퓨팅과 관련된 파일시스템들의 동향을 통해 최근 급격하게 성장하고 있는 대용량 데이터의 관리 기술 및 저장의 메커니즘을 이해하고 불특정 다수의 클라우드 서비스의 모델에 따른 데이터의 성능과 안정성을 보장해 주기 위한 분산파일 시스템이 갖추어야 할 사항들을 살펴 보았다. 클라우드의 가장 중요한 고가용성의 부분들을 고려하여, 데이터베이스의 고가용성 솔루션 및 또한 특징적으로 비정형 데이터의 처리를 위한 하둡 분산파일 시스템과, 메모리 데이터를 통한 시스템 퍼포먼스 향상

등을 조합한 새로운 데이터 아키텍처를 제시 하였다.또한 클라우드 서비스 플랫폼을 위한 그리드(Grid)기술에 대하여 좀 더 세부적인 기술을 연구 하였다. 그리드(Grid)는 단일 제품도 아니고 또한 새로운 표준도 아니다. Grid의 궁극적인 목적은 고품질의 클라우드 플랫폼을 위한 서비스, 유연한 확장성, 효율적인 아키텍처를 위한 새로운 접근법이자 비전이다. 따라서 그리드 기술을 접목시킬 수 있는 대상이 정해져 있는 것이 아니며, 애플리케이션이든 데이터베이스 이든 제한적이지 않다. 어떤 업무 시스템이든 고가용성과 예측 가능한 성능, 확장성, 관리 성이 주요한 환경이 라면 더욱이 그리드 기술의 접목을 검토해 볼 수 있을 것이다. 또한 최근 이슈가 되고 있는 Green IT의 측면에서 IT 자원을 최적화함으로써 기업들이 환경에 미치는 영향을 최소화할 수 있도록 기여할 것이다. 해외 클라우드 서비스의 선두 주자인 구글과, 아마존 역시 클라우드 서비스의 활용되고 있는 분산 파일 시스템들은 거의 유사한 구조와 비슷한 기능들을 갖추고 있는 것을 알 수 있고, 이는 클라우드 컴퓨팅에서 분산 파일 시스템은 비용적인 측면에서의 효율성, 지속적으로 증가하는 데이터의 안정적인 처리, 가용성에 대한 대처, 관리의 용이성, 데이터의 최적 배치 및 효과적인 캐시의 사용, 부하 집중에 대한 로드 밸런싱, 데이터 보안등과 같이 상호 보완 되어야 할 사항들이 무수히 많다. 최근 다양한 해외 벤더들은 이러한 클라우드 서비스의 안정적인 서비스 및 대용량의 데이터들을 처리 할 수 있는 오픈소스 기반의 데이터 분산처리 기술들을 선택하여 메모리데이터베이스 기반의 어플라이언스 제품들을 출시하기 시작 하였다. 본 논문은 국내외의 안정적인 클라우드 서비스를 준비하는 많은 기업들이 갖추어야 할 주요 파일 시스템의 메커니즘을 이해하고 향후 클라우드 컴퓨팅을 위한 대용량 분산 파일 시스템으로 활용하기를 기대 한다.

참고 문헌

- [1] 김창수, 김학영, 남궁한, *클라우드 서비스를 위한 대규모 클러스터 관리 기술 개발*, 전자통신 동향

분석, 2009.

[2] 민영수, 진기성, 김홍연, 김영관, *클라우드 컴퓨팅을 위한 분산 파일 시스템 기술 동향, 전자통신 동향* 분석, 2009.

[3] 메릴린치, *The Clouds War: \$100 + Billion at Stake* 보고서, 2008(5).

[4] 민옥기, 김학영, 남궁환, *클라우드 컴퓨팅 기술 동향, 전자통신동향 분석*, 2009.

[5] 이정아, *모바일클라우드 서비스 국내의 정책 추진 현황*, KT경제경영 연구소, 2010.

[6] 정제호, *클라우드 컴퓨팅의 현재와 미래, 그리고 시장 전략*, 2008(10).

[7] 신현석, *마이크로소프트와 클라우드 컴퓨팅 MS 클라우드 전략의 '코어', 윈도우 애저, 마이크로소프트웨어*, pp.160-163, 2009(1),

[8] 세일즈포스닷컴, *Salesforce 마케팅*, <http://salesforce.com>

[9] IBM 클라우드 컴퓨팅 전략, '블루 클라우드' 컴퓨팅 패러다임 주도, 마이크로소프트웨어, 2009(1).

[10] Amazon, "Amazon Web Services: Overview of Security Process," <http://aws.amazon.com/whitepaper>, Sept. 2008.

[11] Amazon, *Amazon Elastic Compute Cloud (Amazon EC2)*, <http://aws.amazon.com/ec2>

[12] Amazon, "Amazon Simple Storage Service (Amazon S3)," <http://aws.amazon.com/s3>

[13] Chris & Suchitra Narayan, "클라우드 서비스 도입의 기폭제가 된 경기 침체", IDC Analyze the Future, 2009(2).

[14] Dave Thomas, *Enabling Application Agility - Software as a Service, Cloud Computing and Dynamic Languages*, Journal of Object Technology, Vol.7, No.4, 2008(5).

[15] George Lawton, *Developing Software Online with Platform-as-a-Service Technology Computer*, 2008(6).

[16] HDFS, <http://hadoop.apache.org/core/docs>

[17] KIPA, *SaaS 대표주자, Salesforce.com의 성장*

세 분석, 2007(11).

[18] M. Armbrust, *Above the Clouds: A Berkeley View of Cloud Computing*, <http://radlab.cs.berkeley.edu>, 2009(2).

[19] "Vision, Hype, and Reality for Delivering IT Services as Computing Utilities," HPC 2008 Keynote, 2008.

[20] http://news.inews24.com/php/news_view.php?g_serial=707924&g_menu=020200

저 자 소 개

이 병 엽(Byoung-Yup Lee)

중신회원



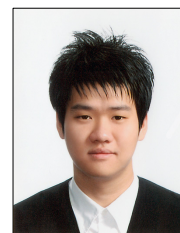
- 1991년 2월 : 한국과학기술원 전산학과(공학사)
- 1993년 2월 : 한국과학기술원 전산학과(공학석사)
- 1997년 2월 : 한국과학기술원 경영정보공학(공학박사)

- 1993년 1월 ~ 2003년 2월 : 대우정보시스템 차장
- 2003년 3월 ~ 현재 : 배재대학교 전자상거래학과 부교수

<관심분야> : XML, 지능정보시스템, 데이터베이스 시스템, 전자상거래학

박 준 호(Junho Park)

정회원



- 2008년 2월 : 충북대학교 정보통신공학과(공학사)
- 2010년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2010년 3월 ~ 현재 : 충북대학교 정보통신공학부 박사과정

<관심분야> : 무선 센서 네트워크, 데이터베이스 시스템, RFID, 차세대 웹, LMS/LCMS, 바이오인포매틱스 등

유 재 수(Jaesoo Yoo)

중신회원



- 1989년 2월 : 전북대학교 컴퓨터 공학과(공학사)
 - 1991년 2월 : KAIST 전산학과 (공학석사)
 - 1995년 2월 : KAIST 전산학과 (공학박사)
 - 1995년 3월 ~ 1996년 8월 : 목포대학교 전산통계학과 (전임강사)
 - 1996년 8월 ~ 현재 : 충북대학교 정보통신공학부 및 컴퓨터정보통신연구소 교수
 - 2009년 3월 ~ 2010년 2월 : 캘리포니아주립대학교 방문교수
- <관심분야> : 데이터베이스시스템, 빅데이터, 센서네트워크 및 RFID, 소셜네트워크서비스, 분산객체컴퓨팅, 바이오인포매틱스 등