집중적인 입출력 스트레스 테스트를 통한 클러스터 파일 시스템 SANique™의 성능평가

이규웅*

Performance Evaluation of I/O Intensive Stress Test in Cluster File System SANique™

Kyu-Woong Lee*

이 논문은 2009년도 상지대학교 교내연구비 지원에 의한 결과임

요 약

본 논문은 저장장치 전용 네트워크인 SAN 상에서 운영되는 공유 파일 시스템 SANique™의 성능평가에 대한 분석내용을 기술하고 이를 통해 독립적 파일 시스템인 EXT3와 비교 분석한다. 성능평가를 위해 클러스터 파일 시스템 위에 오라클 10g 데이터베이스 시스템을 설치하고 ESQL/C 데이터베이스 응용 프로그램을 제작하여 집중적인 입출력 스트레스 테스트를 수행하였다. 다양한 성능평가 결과 비교를 위해 클러스터 파일 시스템 구조, 독립 파일 시스템 구조, 클러스터 및 독립 파일 시스템의 공용으로 사용하는 구조에서 각각 성능평가를 수행하고 그 결과를 분석하였다. 본 논문의 다양한 성능평가 결과를 통해 집중적인 입출력 테스트에서 클러스터 파일 시스템 SANique™이 독립 파일 시스템에 비해 우수한 성능을 보임을 입증하였다.

ABSTRACT

This paper describes the design overview of shared file system SANique™ and analyzes the performance evaluation results of I/O intensive stress test based on various cluster file system architectures. Especially, we illustrate the performance analysis for the comparison results between the SANique™ and the Linux file system EXT3 system that is used to generally in Unix world. In order to perform our evaluation, Oracle 10g database system is operated on the top of cluster file system, and we developed the various kinds of testing tools which are compiled by ESQL/C from Oracle. Three types of architectures are used in this performance evaluation. Those are the cluster file system SANique™, EXT3 and the combined architecture of SANique™ and EXT3. In this paper, we present that the results of SANique™ outperforms other cluster file systems in the overhead for providing the true sharing over the connecting server nodes.

키워드

클러스터 시스템, 공유 파일 시스템, 성능평가, 데이터베이스 임베디드 SQL

Key word

Cluster System, Shared File System, Performance Evaluation, Database, Embedded SQL

* 상지대학교 컴퓨터정보공학부

접수일자 : 2009. 09. 09
심사완료일자 : 2009. 10. 01

I. 서 론

다수의 컴퓨터들을 네트워크로 결합하여 전체적인 성능 향상을 추구하는 흐름이 생기게 되었고, 이러한 과정을 통하여 클러스터 파일 시스템 기술이 대용량 스토리지 및 확장성을 제공하는 수단으로 인정받고 있다. 대표적인 클러스터 파일 시스템의 구조는 그림 1과 같이 상호 연결된 컴퓨팅 노드 그룹이 화이버 채널과 같은 저장장치 전용 네트워크인 SAN상에 연결된 저장 서브시스템을 공유 할 수 있도록 하는 공유 파일 시스템을 제공하는 구조이다[8].

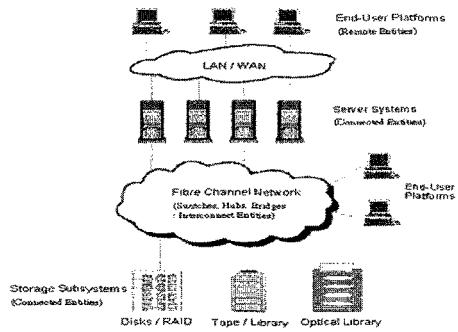

그림 1. SAN 기반 클러스터 파일 시스템 구조
Fig. 1 SAN Cluster File System Architecture

본 논문은 위와 같은 클러스터 공유 파일 시스템 상에서 집중적인 I/O 스트레스 테스트를 통하여 클러스터 파일 시스템의 성능평가를 수행한다. 또한 그 결과를 독립 파일 시스템인 EXT3 파일 시스템과 비교하므로써, 클러스터 간의 진정한 파일 공유를 제공하기 위한 오버헤드가 독립 파일 시스템에 비해 어느 정도 되는지를 관측해 보이고자 한다. 본 논문의 구성은 다음과 같다. 제 2장에서 클러스터 공유 파일 시스템의 특징 및 기존 관련 연구를 기술하고 제3장에서 성능평가를 위한 다양한 시스템 구조를 설명한다. 오라클 10g 데이터베이스 시스템을 각각 클러스터 시스템, 독립 파일 시스템 그리고 두 가지 파일 시스템을 공유하는 혼용 파일 시스템 위에 설치하고 성능평가를 운영하는 방법 및 환경에 대하여 설명한다. 또한 다양한 시스템 구조와 여러 유형의 성능평가 프로그램을 수행하여 산출된 결과들에 대한 분석을 제4장에서 기술하고 제5장에서 결론을 맺는다.

II. 연구 배경 및 관련 연구

가. 클러스터 공유 파일 시스템의 관련 연구

저장 장치 구조에 대한 클러스터링에 대한 연구가 급증하고 있으며 이러한 환경의 클러스터 파일 시스템 및 시스템 소프트웨어의 상용화가 증가하고 있다[1,2,3]. 최근 분산 클러스터 파일 시스템은 구글 파일 시스템 GFS(Google File System)[4,5,6], 한국전자통신연구원의 OASIS[7]와 같이, 다양한 사용자의 대규모 데이터 웹 서비스 요구에 맞추기 위해 각 서버들을 클러스터링 하여 하나의 대용량 파일 시스템으로 형성하는 객체 기반 파일 시스템 및 지능형 클러스터 파일 시스템으로 확장 개발되고 있다. 구글 클러스터 파일 시스템은 그림 2와 같이 구글의 동영상 검색 및 저장 서비스, 이미지 서비스, 구글 어스 등과 같은 구글의 모든 데이터 집중적인 웹 서비스를 제공하기 위한 구글 플랫폼으로 사용되는 대표적인 분산 클러스터 파일 시스템이다. 구글 파일 시스템은 GFS 서버 그룹들로 클러스터링 되어 있으며 하나의 응용을 서비스 하기 위하여 전담 서버 역할을 하는 마스터 서버로 구성된다[4,5]. 마스터 서버에 의해 파일 이름, 접근 오프셋 등이 결정되어 지면, 공유 파일 시스템내의 디스크 풀에 존재하는 공유 데이터들은 직접 응용으로 전달된다[6].

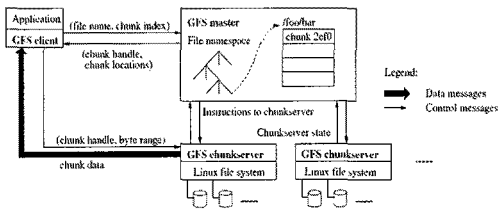

그림 2. 구글 파일 시스템 GFS의 아키텍쳐
Fig. 2. The Architecture of Google File System

나. SANique™ 클러스터 파일 시스템의 구조

SANique™ 시스템은 현재 국내 외의 많은 곳에서 활용되고 있는 매크로임팩트사의 SAN 기반의 지능형 범용 클러스터 공유 파일 시스템이다. 최근의 클러스터 파일 시스템은 특정 목적을 전제로 서비스되는 반면 SANique™은 범용적 목적의 완벽한 응용 서비스를 제공하기 위한 클러스터 공유 파일 시스템이다. SANique™

은 클러스터 파일 시스템의 각 컴퓨팅 서버간 진정한 공유를 제공하기 위하여 그림 3과 같은 시스템 구조를 갖는다. 각 서버들은 공유 파일 서비스를 제공하기 위하여 기존 파일 시스템위에 SANique™의 주요 구성 모듈인 CFS와 CVM을 탑재하여, 공유 디스크 영역에 대해 읽기/쓰기가 가능하게 되며, 그 연산의 결과는 즉각적으로 클러스터 내 모든 노드들에 의해 공유되어 질 수 있다.

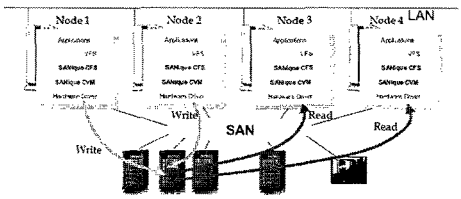

그림 3. SANique™의 시스템 구성도
Fig. 3. System Architecture of SANique™

각 서버 노드 마다 탑재되는 CFS는 64비트 주소 공간을 갖는 저널링 공유 파일 시스템이다. CFS의 주요 파일 시스템 정보는 특정 노드에 할당되지 않고 클러스터를 구성하는 모든 서버들에 분할 저장되므로 공유 디스크를 접근할 때 마다 각 서버들의 정보 공유를 통해 디스크 접근을 수행하게 된다. CVM은 클러스터 볼륨 관리기로서 SAN에 부착된 모든 디스크들을 공유 볼륨으로 구성하는 볼륨 관리기이며 볼륨은 물리적 디스크의 구성 방식에 따라 스트라이핑, 미러링, 패러티 스트라이핑 등 다양한 형식으로 지원될 수 있다.

Ⅲ. 입출력 성능평가를 위한 시스템 환경 및 성능평가 도구

본 절에서는 SANique™ 시스템의 집중적인 입출력 성능평가를 위해 테스트베드로 사용될 다양한 클러스터 파일 시스템 구조에 대하여 설명하고, 이와 같은 구조에서 수행될 입출력 테스트 시험도구에 대하여 설명한다.

가. 성능평가 플랫폼 구조도

클러스터 파일 시스템의 다양한 성능평가를 위해서 각 컴퓨팅 노드는 유니와이드사의 익스트림 X1322 서버로 구축하였으며,그림 4와 같은 시스템 사양을 갖는다.

	type	4 X Intel(R) Xeon(TM) CPU
CPU	Clock Pulse	3.0 GHz
	Cache Size	2 MB
Memory	Total Size	8 GB
Disk	Volume 1 /	180 GB
	Volume 2 /oracle	200 GB
	Volume 3 /sanique	100GB

그림 4. 각 노드의 시스템 사양
Fig. 4. System Node Specification

그림 5와 같이 두 개의 노드로 구성되는 클러스터 파일 시스템은 SAN에 부착된 디스크를 공유 볼륨으로 구성하여 공유 볼륨 SANique™에 설치되는 데이터베이스 엔진 및 파일들은 노드 1과 2에서 모두 접근가능하다.

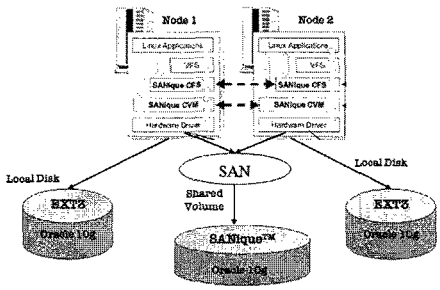

그림 5. 성능평가를 위한 시스템 구조
Fig. 5. System Architecture for Performance Evaluation

	DBMS Engine	Database File
TESTBED A	EXT3	EXT3
TESTBED B	EXT3	SANique™
TESTBED C	SANique™	SANique™

그림 6. 성능평가 테스트베드 조합
Fig. 6. Testbed Comninations

SANique™의 입출력 성능을 독립 파일 시스템인 리눅스 파일 시스템 EXT3와 비교하기 위하여 그림 6과 같이 실험한다. 실험 구조 A는 오라클 데이터베이스 엔진과 데이터베이스 파일 모두를 독립 파일 시스템인 EXT3에 설치하는 구조이고, 실험 구조 B는 데이터베이스 엔

진은 공유 볼륨인 SANique™에 설치하고 데이터베이스 파일은 EXT3에 설치하는 구조이며, 실험 구조 C는 오라클 데이터베이스 엔진 및 파일 모두를 공유 볼륨 SANique™에 설치하는 구조이다. EXT3 파일 시스템은 그림 5에 표현한 것처럼 노드 1의 지역적 파일 시스템이며 SANique™ 파일 시스템은 노드 1과 노드 2에서 공유하는 공유 볼륨이다. 실험 구조 A와 C를 통해서 지역 파일 시스템이며 독립 파일 시스템의 성능과 공유 파일 시스템의 성능을 비교할 수 있으며, 실험 구조 B를 통해 지역 파일 시스템과 공유 파일 시스템을 공용으로 사용하는 경우의 성능을 평가할 수 있다.

나. 성능평가 도구

집중적인 입출력 성능평가를 위해 오라클 10g의 임베디드 SQL 컴파일러 Pro*c를 활용하여 4가지의 트랜잭션 프로그램을 개발하였다. 각 트랜잭션은 데이터베이스 내의 데이터 삽입, 검색, 삭제 및 갱신 트랜잭션이다. 오라클 데이터베이스의 구성은 기본 구성 환경으로 구축하였다. 1G 바이트 크기의 물리적 데이터 파일 하나를 갖는 테이블 스페이스를 생성하여 성능평가 전용 데이터베이스를 새로 구축하고, 각 트랜잭션이 수행될 수 있도록 그림 6과 같은 대상 테이블을 구축하였다.

Attribute	Type	Index
attr_A	INTEGER NOT NULL	Primary Key
attr_B	CHARACTER(8)	Index
attr_C	CHARACTER VARYING(8)	
attr_D	CHARACTER(1)	
attr_E	CHARACTER(1)	
attr_F	CHARACTER(14)	
attr_G	CHARACTER VARYING(22)	Index
attr_H	CHARACTER VARYING(162) NOT NULL	
attr_H	CHARACTER VARYING(22)	
attr_H	CHARACTER VARYING(4000)	

그림 7. 데이터베이스 스키마 기본 구조
Fig. 7. Database Schema Structure

또한 입출력 성능평가를 위해 그림 8과 같은 4가지의 ESQL/C 트랜잭션 프로그램을 수행한다. 약 1G 바이트 크기의 데이터를 삽입하는 트랜잭션과 기본 키 및 인덱스 키에 의한 검색 트랜잭션, 기본 키와 인덱스 키 속성에 대한 조건을 갖는 삭제 및 갱신 트랜잭션으로 구성된다.

트랜잭션	수행 내용
Insert	• 1,000,000 튜플 삽입 • 약 1G 바이트 데이터 삽입
Select	• 기본 키 속성에 대한 조건 검색 • 인덱스 키 속성에 대한 조건 검색 • 10,000건 ~ 90,000 건 수행
Update	• 기본 키 속성 조건 검색 후 갱신 • 인덱스 키 속성 조건 검색 후 갱신 • 10,000건 ~ 90,000 건 수행
Delete	• 기본 키 속성 조건 검색 후 삭제 • 인덱스 키 속성 조건 검색 후 삭제 • 기본 키 속성에 대한 범위 연산 검색 후 삭제 • 10,000건 ~ 90,000 건 수행

그림 8. 트랜잭션 수행 개요
Fig. 8. Transactions for Evaluation

Ⅳ. 입출력 성능평가 결과 분석

집중적인 입출력 성능평가를 위해서 약 1G 바이트 용량의 데이터를 삽입하는 대용량 삽입 트랜잭션을 먼저 실행하였다. 삽입 트랜잭션은 약 1K정도 크기의 한 튜플을 백만개를 일시적으로 삽입하도록 구성하였다. 백만건의 SQL 문장을 수행 후 트랜잭션이 완료될 수 있도록 구성하였으며, 검색과 갱신 및 삭제 트랜잭션 실행 전에 수행되도록 하였다. 동일한 삽입 트랜잭션을 그림 6의 세가지 테스트베드 구조에 각각 4회 수행하여 그림 9와 같은 성능 평가 결과를 얻을 수 있었다. 삽입 트랜잭션은 클러스터 파일 시스템에서 수행되는 경우 독립 파일 시스템인 EXT3에서의 결과보다 우세한 성능 결과를 보였다. 그러나 DBMS 엔진은EXT3설치하고 데이터베이스 화일을 클러스터 파일 시스템인 SANique™에 설치하는 경우 두 가지 파일 시스템을 동시 접근하는 오버헤드로 인하여 낮은 성능결과를 보이고 있다. 그러나 전체적인 성능 평가 결과를 보면, 공유 파일 시스템을 유지하기 위한 통신 오버헤드가 독립 파일 시스템과 비교해 볼 때 다소 낮음을 확인할 수 있었다. 그림 10은 1G 바이트의 데이터 삽입 후, 기본 키에 의한 검색과 인덱스 키에 의한 검색 트랜잭션을 수행한 성능평가 결과이다. 그림 10의 가로축에 표기된 EXT3는 해당 파일 시스템 구조에서 수행되었음을 나타내며, 각 파일 시스템위의 2개의 막대 그래프는 각각 기본키 검색 수행결과와 인덱스 속성 검색결과를 나타낸다. EXT3 구조와 SANique 구조, 그리고 혼용 구조에서 각각 검색 트랜잭션을 10,000건부터 90,000건까지 수행하여 그 결과를 분석한 결과 SANique 파일 시스템이 EXT3 시스템 대비 다소 낮은 결과를 보였으며 혼용 구조보다는 다소 좋은 결과를 보였다. 검색 트랜잭

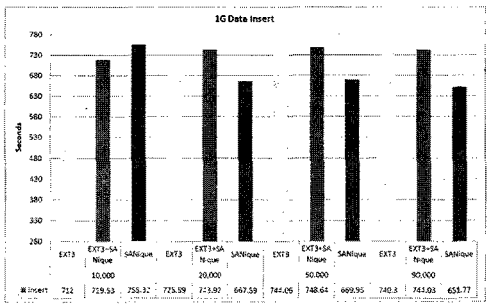

그림 9. 삽입 트랜잭션의 성능평가 결과 비교
Fi.g 9. Performance Evaluation of Insert Transaction

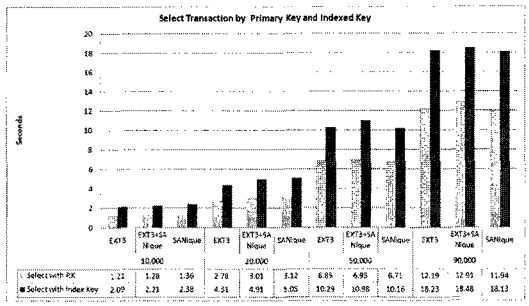

그림 10. 검색 트랜잭션의 성능평가 결과비교
Fig. 10. Performance Evaluation of Select Transaction

션의 수행 결과를 분석하여 볼 때, 클러스터 파일 시스템을 운영하기 위한 오버헤드가 다소 높지 않음을 알 수 있었다. 세가지 테스트 베드 구조에서 10,000건부터 90,000건 까지 갱신 트랜잭션을 수행한 성능평가 결과는 그림 11이다. 갱신 트랜잭션 역시 기본 키 속성에 대한 조건 검색 후 갱신 연산을 수행하는 부분과 인덱스 속성에 대한 검색 후 갱신 연산을 수행하는 부분으로 나뉘어 수행된다. EXT3 수행 결과와 비교하여 볼 때, SANique™ 시스템의 결과는 약 15%에서 20%의 성능 오버헤드를 보였다. 삽입과 검색트랜잭션 성능평가에서 EXT3와 동등한 수준의 성능평가를 보인것과는 달리 갱신 연산을 수행하는 경우 클러스터 파일 시스템의 통신 오버헤드가 EXT3보다 다소 낮은 성능 평가 결과를 유발한 것으로 보인다. 그림 12는 삭제 트랜잭션에 대한 성능 평가 결과 그래프이다. 삭제 트랜잭션은 세가지 부분으로 구성된다. 기본 키 검색 조건을 포함하는 삭제 연산과 인덱스 속성 검색조건을 포함하는 삭제 연산 그리고 기본 키에 대한 범위 검색을 포함하는 삭제 연산으로 나뉘어진다. 범위 검색은 조건 검색 범위를 10,000건, 20,000건, 50,000건, 90,000건으로 구별하여 수행하였다. 기본 키 검색 및 인덱스 키 검색 조건을 포함하는 삭제연산은 앞에서 수행된 검색, 갱신 트랜잭션과 마찬가지로 하나의 검색 조건에서 한 개의 튜플이 검색되는 조건으로 구성하였다. 따라서 10,000건의 튜플 삭제를 위해서는 10,000개의 SQL 문장이 하나의 트랜잭션에서 수행된다. 데이터베이스 시스템의 삭제연산은 삽입 연산보다 더 많은 디스크 접근을 시도하므로 갱신 및 검색 트랜잭션 보다 다소 지연된 결과를 보이고 있다. SANique™시스템의 성능 평가 결과는 EXT3 시스템 대비 약 14%에서 20%의 성능 이익을 보이고 있다. 앞에서 보인 삽입 트랜잭션의

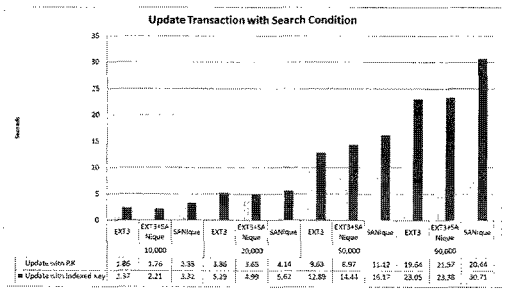

그림 11. 갱신 트랜잭션 성능평가
Fig. 11 Performace Evaluation of Update Transaction

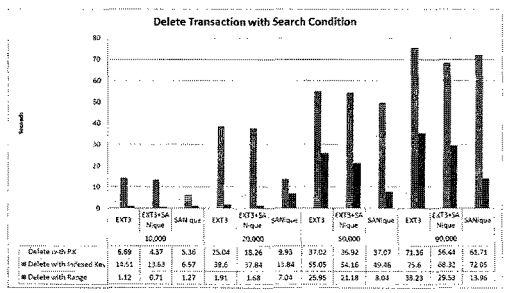

그림 12. 삭제 트랜잭션 성능평가
Fig. 12 Performace Evaluation of Delete Transaction

수행 성능 평가 결과에서도 보였듯이 연속적이고 집중적인 디스크 쓰기 연산의 성능은 클러스터 파일 시스템임에도 불구하고 훨신 우수한 성능을 보이고 있음을 알 수 있다. 그러나 두 개의 파일 시스템을 혼용하는 구조에서는 역시 독자적인 파일 시스템 구조 하나만을 사용하는 경우보다 다소 낮은 성능평가 결과를 보이고 있다. 본 성능평가 결과를 통해서 대량의 집중적인 쓰기 연산을 포함하는 트랜잭션의 경우 SANique™ 시스템이 EXT3 보다 우세한 성능을 보임을 알 수 있었다. 그러나 검색 트랜잭션과 갱신 트랜잭션의 경우 클러스터 파일 시스템의 노드 간 통신 오버헤드로 인하여 EXT3 결과의 90~95%에 해당하는 다소 낮은 성능 결과를 보였으나, 클러스터 파일 시스템이 공유 디스크를 제공한다는 장점을 고려해 볼 때 인정할 만한 성능 오버헤드로 판단할 수 있다.

V. 결론

본 논문에서는 SAN 기반의 클러스터 파일 시스템인 SANique™의 성능평가를 수행한 결과에 대하여 분석하였다. 다양한 성능 평가 분석을 위하여 독립 파일 시스템인 EXT3 시스템과 SANique™ 시스템을 비교하였고 또한 EXT3와 SANique™을 혼용하여 사용하는 구조와도 성능 결과를 비교하였다. 대량의 집중적인 입출력 성능 평가를 위해 각 시스템 구조마다 오라클 10g 데이터베이스 시스템을 구축하였고 삽입, 검색, 갱신 및 삭제를 위한 ESQL/C 성능평가 도구를 작성하여 수행하였다. 성능 평가를 분석한 결과 디스크에 대한 대량의 집중적인 쓰기 연산을 포함하는 성능 평가에서 SANqiue™ 시스템이 클러스터 파일 시스템의 통신 오버헤드에도 불구하고 우수한 성능을 보임을 입증하였다. 그러나 갱신 및 검색 연산에서는 이러한 오버헤드로 인하여 다소 낮은 성능을 유발하였으나 클러스터 파일 시스템의 장점을 고려할 때 인정될만한 낮은 비율이다. 추후 클러스터의 범위를 확장하여 노드의 개수를 추가한 성능평가와 표준 성능 평가 도구를 이용한 분석 및 타 상용 시스템과의 비교 분석을 계속 연구중이다.

참고문헌

[1] Symantc Corp.. "Veritas File System and Volume Manager", http://www. symantec.com

[2] "The Red Hat Global File System", Red Hat Technical Document, http://www. redhat.com/gfs

[3] MacroImpact, Inc., "SANique Cluster Volume Manager Functional Specification", MacroImpact Technical Memo, 2008.

[4] Ghemawat, S., Gobioff, H., and Leung, S. -T. The Google File System, In 19th SOSP, Dec. 2003. pp29-43.

[5] Burrows, M. The Chubby Lock Service for Loosely-Coupled Distributed Systems, In Proc. of the 7th OSDI, 2006. 11

[6] Jeffrey Dean and Sanjay Ghemawat, "MapReduce : Simplified Data Processing on large Clusters", In Proc. of the 5th OSDI, 2004. 11

[7] 김명준 외, 클러스터 기반 통합 멀티미디어 DBMS 개발, 정보통신연구진흥원, 연구결과보고서, 2002. 12.

[8] T.E.Anderson, M.D.Dahlin, J.M.Neefe et al., "Serverless Network File Systems," ACM Transactions on Computer Systems, Vol.14, No.1, February 1996, pp.41-79.

저자소개

이규웅(Kyu Woong Lee)

1986년 한국외국어대학교 이학사
1990년 서강대학교대학원 공학석사
1998년 서강대학교대학원 공학박사
1998년~2000년 한국전자통신연구원 선임연구원
2000년~ 상지대학교 컴퓨터정보공학부 부교수
※관심분야 : 데이터베이스, 클러스터 시스템 등