

# SAN 기반 클러스터 파일 시스템 SANique™의 성능평가 및 분석\*

이 규 응\*\*

## Performance Analysis of Cluster File System SANique™ based on Storage Area Network\*

Kyu Woong Lee\*\*

### ■ Abstract ■

As the dependency to network system and demands of efficient storage systems rapidly grows in every networking filed, the current trends initiated by explosive networked data grow due to the wide-spread of internet multimedia data and internet requires a paradigm shift from computing-centric to data-centric in storage systems. Furthermore, the new environment of file systems such as SAN(Storage Area Network) is adopted to the existing storage paradigm for providing high availability and efficient data access. We describe the design issues and system components of SANique™, which is the cluster file system based on SAN environment. We, especially, present the comparative results of performance analysis for the intensive I/O test by using the DBMSs that are operated at the top of cluster file system SANique™, EXT3 and NFS respectively.

Keyword : SAN, Cluster File System, DBMS, Performance, TPC-C

\* 본 논문은 2006년도 상지대학교 연구비 지원에 의한 것임.

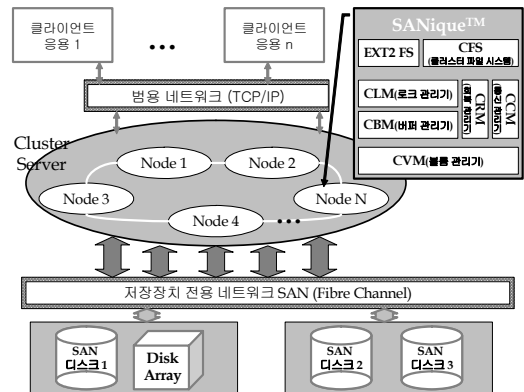
\*\* 상지대학교 컴퓨터정보공학부

## 1. 서 론

사용자 제공 콘텐츠와 같은 Web 2.0의 수요 증가와 함께 멀티미디어 데이터 서비스의 요구와 네트워크 기반 시스템의 발전으로 인해 인터넷 기반 응용 시스템들이 요구하는 데이터의 종류가 다양해지면서 대용량의 데이터 처리 요구를 수용할 수 있는 분산 파일 시스템의 요구가 급증하고 있다. 다수의 인터넷 사용자에게 대용량의 멀티미디어 데이터 서비스를 효율적으로 제공해주기 위해서 기존 서버 시스템내의 파일 시스템은 더 이상 적합한 서비스를 제공할 수 없게 되었다. 즉 동시 접근을 요구하는 사용자가 많아지고, 각각의 접근 요구마다 데이터의 크기가 커지면서 저장장치의 부하가 증가하여 대용량의 멀티미디어 데이터 서비스 해주는 데 어려움이 증가하고 있다. 기존의 클라이언트/서버 구조에 근간을 두고 있는 서버 지향적 파일 시스템은 서버 자체적인 용량의 한계와 서버와 클라이언트 간의 잦은 데이터 전송에 따른 메모리 복사가 요구되며, 서버의 오류 발생 시 모든 클라이언트들이 서비스 받지 못하는 치명적인 오류를 유발할 수 있게 된다.

지역적, 독립적으로 운영되던 저장장치들을 저장장치 전용 네트워크(Storage Area Network; SAN)이라는 광채널을 이용한 고속의 전용 네트워크에 직접 연결하여 특정 지역 서버의 간섭 및 제어 없이 네트워크를 통해 직접적인 접근이 가능한 데이터 중심적인 새로운 저장장치 환경으로 제공하는 것이 SAN 기반 클러스터 파일 시스템(Cluster File System; CFS)이다. 현재 SAN을 통한 저장장치의 클러스터링에 대한 연구가 증가하고 있으며, 이러한 환경에서의 클러스터 파일 시스템 및 시스템 소프트웨어들이 상용화되고 있다[1, 2, 3]. SAN 기반 클러스터 파일 시스템은 [그림 1]에 나타난 바와 같이 SAN에 직접 연결된 저장장치들을 특정한 서버의 도움 없이 전용 네트워크를 통해 접근할 수 있으므로 기존 분산 파일 시스템보다 확장

성이나 가용성이 우수하다[1]. 기존 분산 파일 시스템들이 하나의 중앙 집중적인 서버에 의한 전체 저장장치 관리라는 서버 시스템의 병목현상 단점을 가지고 있는 반면, SAN 기반 클러스터 파일 시스템은 특정한 서버와의 접근없이 SAN에 부착된 공유 저장장치들을 자유롭게 접근할 수 있다는 장점을 제공한다[12]. 클러스터내의 각 서버들은 파일 공유를 위해서 클러스터를 위한 전용 파일 시스템을 운영해야 한다. 즉 파일 공유를 위한 동시성 제어나 메타 데이터의 관리기능을 할 수 있는 클러스터 파일 시스템으로 대치되어야 한다. SANique™ 파일 시스템<sup>1)</sup>은 이와 같은 환경에서 사용할 수 있는 클러스터 파일 시스템이다.



[그림 1] SAN 기반 클러스터 파일 시스템의 구성도

본 논문은 이와 같은 클러스터 파일 시스템의 기본 구조와 각 구성요소의 설계 내용을 제안하며, 클러스터 파일 시스템의 독립적 지역 파일 시스템 및 기존 분산 파일 시스템에 대한 성능평가를 보인다. 본 성능 평가를 위해 각 파일 시스템 상에 상용 데이터베이스 시스템을 설치하고 표준 DBMS 벤치마크 테스트를 수행하여 집중적인 저장장치 접근에 대한 성능평가를 보인다.

1) SANique™은 (주)매크로임팩트 사의 클러스터 파일 시스템 등록상표입니다.

## 2. 클러스터 파일 시스템 관련 연구

클러스터 파일 시스템에 가장 필수적인 기능은 클러스터 노드간의 파일 공유 기능이다. 한 노드가 소유하고 있는 디스크의 파일 시스템 영역을 다른 노드에서도 읽기/쓰기 작업이 가능한 공유가 제공되어야 한다. 파일 시스템의 공유 기능에서 가장 신중하게 고려되어야 하는 문제점은 디스크를 소유한 노드의 병목현상이다. 즉, 특정 노드가 소유한 디스크를 여러 노드에서 집중적으로 접근하는 경우 그 노드에 부하가 걸려 효율적인 데이터 공유 서비스를 제공할 수 없다는 것이다. 분산파일 시스템 중에서 클라이언트/서버 유형의 대표적 예인 네트워크 파일 시스템(NFS)은 바로 이와 같은 병목 현상을 피할 수 없다[5, 6, 7, 11]. 따라서 이러한 특정 노드의 병목현상을 없애기 위해 저장장치를 위한 전용 네트워크인 SAN을 구축하고, 또한 네트워크에 직접 부착하여 사용하는 NAS(network attached storage)를 이용하여 특정 디스크가 특정 노드에 소유되지 않는 새로운 기법의 클러스터 공유파일 시스템들이 등장하게 되었다.

IBM의 Tivoli사에서 나온 SANergy라는 제품은 파일 공유 솔루션으로 가장 먼저 출시되었으며 많이 사용되고 있다. 이 제품은 메타 데이터 서비스에 기반하는 미들웨어 방식을 채택하고 있다. 또한 SAN을 통한 물리적인 공유를 전제로 하고 있으며, 클라이언트/서버 형식을 취하고 있어, 서버에 오버헤드가 발생 할 수 있으며 병목현상의 문제점을 안고 있다. 파일 공유의 특성상 대용량 멀티미디어 데이터를 취급하는 환경이 많은데 비해, 이 제품은 공유 데이터 전송에 많은 시간이 소요된다는 단점을 지니고 있다.

리눅스 시스템을 기반으로 배포되어온 Minnesota 대학의 GFS(Global File System)은 주로 개인적, 연구용 목적으로 사용되었으나 Sistina라는 기업으로 창업하여 현재 제품으로 출시되고 있다[2, 3, 4]. 이 공유 파일 시스템은 GFS 로크 서버 역할을

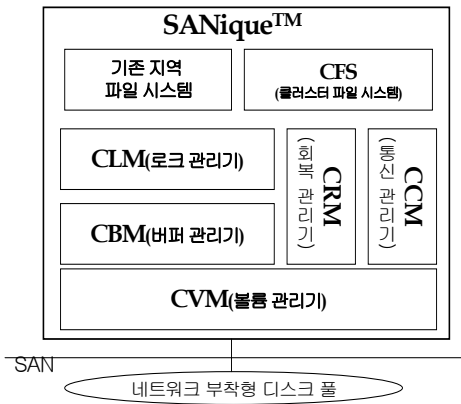
하는 노드와 클라이언트 노드들로 구성되는 비대칭형 구조이다. 각 클라이언트들은 GFS 로크 서버의 제어 하에 로크를 획득한 후, 공유 디스크를 접근하게 된다. 이 제품 역시 비대칭형 구조의 전형적인 문제인 병목현상으로 인한 성능 저하 현상을 갖게 되며, 로크 서버의 오류로 인하여 전체 시스템 서비스가 중단되는 저가용성 문제를 갖는다.

폴리서브 사의 Matrix 서버 제품은 본래 클러스터링 솔루션으로 개발되었으나, 최근 공유 파일 시스템의 수요가 증가함에 따라 공유 파일 기능이 추가된 경우이다. 최근 출시 제품이므로, 아직 시장 평가가 많이 이루어 지지 않은 상태로서 현재 10노드 클러스터 규모를 지원하며 플랫폼의 운영체제 또한 리눅스로 제한되고 있다.

썬 마이크로 시스템의 기술진들이 설립한 Veritas에서 출시한 SAN Point Foundation Suite는 클러스터 파일 시스템, 볼륨 관리기를 포함하는 제품으로 가장 최근 출시되었으며, 아직 오류로 인한 고가용성 제공의 미흡과 비용 및 번들 구성으로 인한 사용자 요구 만족 등의 문제점을 안고 있다[1].

## 3. 클러스터 파일 시스템 SANique™의 구조

SANique™은 SAN 기반의 클러스터 서버들을 위한 클러스터 공유 파일 시스템으로서, 전형적인 분산 파일 서버 시스템 구조의 병목 현상을 제거한 공유 파일 시스템이다[8]. SANique™ 파일 시스템은 [그림 1]의 각 노드에 기술된 클러스터 공유 파일 시스템의 역할을 하는 파일 시스템이며 디스크의 소유 개념이 없는 진정한 클러스터 공유 파일 시스템이다. 각 노드에 위치한 클러스터 공유 파일 시스템은 크게 클러스터 볼륨 관리기(CVM), 클러스터 로크 관리기(CLM), 클러스터 버퍼 관리기(CBM), 클러스터 회복 관리기(CBM), 클러스터 파일 시스템(CFS)로 구성되며 [그림 2]와 같다.



[그림 2] SANique™ 시스템 구성도

SANique™ CFS에 의해 제공되는 노드간 파일 공유는 클러스터의 모든 단일 노드들이 SAN에 직접 연결된 디스크 장치들에 대해 병행적 공유 접근이 가능하다는 점에서 기존 NFS 등의 파일 시스템의 공유와 구별된다. SANique™ CFS에 의해 클러스터의 모든 단일 노드들은 SAN에 직접 연결된 디스크 장치들에 대해 병행적 공유 접근이 가능하다. CFS는 수퍼블록 영역, 비트맵 영역, 데이터 블록 영역으로 구성된다. 수퍼블록 영역은 파일 시스템의 데이터를 관리하기 위한 메타 데이터를 저장하고, 비트맵 영역은 디스크 공간의 할당 영역과 가용 영역을 관리하기 위한 영역이다. 전형적인 분산 파일 시스템에서는 파일 서버가 비트맵 영역을 관리하지만, 공유 파일 시스템 상에서는 여러 노드에 의해 관리 및 접근되므로 상호 배타적인 접근 방법이 필요하고, 비트맵 수정 연산에 대한 일관성 유지 방법이 필요하게 된다.

SANique™ CLM은 클러스터의 여러 노드에 의해 동시 접근되는 같은 파일에 대해 직렬성을 보장하기 위한 기능을 제공한다. 공유 볼륨 상의 파일 접근을 직렬화하고 일관된 데이터 접근을 보장하기 위하여 파일 수준의 병행수행 제어 방법을 이용한다. 이에 따라 같은 파일에 접근하는 모든 연산은 파일 단위의 데이터 일관성을 보장 받는다. SANique™은 대칭형 클러스터 파일 시스템의 구

성이므로, 특정 서버가 로크 서버 역할을 하는 대신, 모든 노드들에게 지역 로크 서버 역할을 부여하고, 그 중 일부 몇 개의 노드들이 전역적 통제를 위한 전역 서버 역할을 한다.

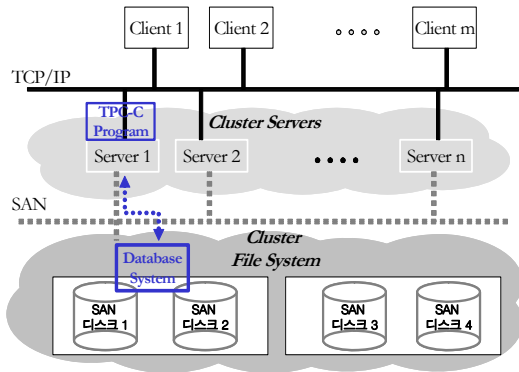
SANique™ CVM은 주어진 클러스터 내에 사용할 수 있는 모든 디스크 공간을 사용 가능한 가상 볼륨으로 구성하여 저장 풀(pool)을 형성하고 파일 시스템으로부터 사용될 수 있게 한다. 물리적 디스크 집합으로부터 논리적 디스크 영역을 구성하고 이를 파일 시스템의 주소 공간으로 전환할 수 있는 매핑 테이블을 유지한다. 또한 논리적 디스크는 물리적 디스크의 구성 방식에 따라, 스트라이핑, 미러링(RAID-1), 패러티 스트라이핑(RAID-5) 등을 지원할 수 있다.

SANique™ CRM은 한 노드의 오류로 인하여 서비스 장애가 발생하는 경우 다른 노드에 영향을 주지 않도록 온라인상에서 오류 노드를 복구하는 회복 관리기이다. 특히 두 개 이상의 노드에서 서로 상대방 노드를 오류로 인식하여 발생하는 분할 그룹(split brain) 문제는 전형적인 분산 시스템의 문제이다. 공유 파일 시스템 상에서는 이러한 분할 그룹 문제가 파일의 일관성에 치명적인 문제를 발생시킨다. CRM은 이러한 분할 그룹 문제를 해결하기 위하여 공유 디스크 보드(shared disk board)를 활용하여 양분 또는 다분화된 클러스터내의 노드들을 장애 복구 하는 회복 관리기를 설계하였다[10].

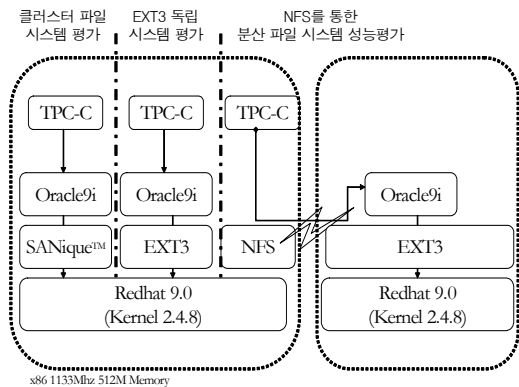
SANique™ CBM은 클러스터 노드들이 보유하고 있는 데이터 캐쉬 블록들이 다른 노드에서 필요할 때 전달될 수 있는 전역 버퍼 관리 기법(Global Buffer Management)을 이용한다. 기존 버퍼 관리 기법과는 달리, 전역 버퍼 관리 기법인 상호협조 캐싱 기법을 활용한 전역 버퍼 관리 기법을 제공한다. 캐싱된 메타 데이터를 요구하는 서로 다른 노드에게 전달하게 함으로서 공유 파일 시스템 관리를 위한 제어 정보가 효율적으로 전달될 수 있게 한다.

## 4. 클러스터 파일 시스템의 성능 평가

### 4.1 성능평가 개요 및 시험환경



[그림 3] 클러스터 파일 시스템 성능평가 환경



[그림 4] 성능평가 비교를 위한 논리적 모형

클러스터 파일 시스템의 SAN에 부착되어 있는 저장장치에 대한 집중적인 접근 시험을 수행하기 위해 클러스터 공유 파일 시스템 위에 상용 데이터베이스 시스템인 오라클 9i 범용 데이터베이스 시스템을 설치하고, 표준 성능평가 도구인 TPC-C[9] 성능 시험 프로그램을 수행하여 그 결과를 각 시스템 별로 비교하는 것을 목적으로 한다. 클러스터 공유 파일 시스템의 성능평가 환경에 대한 물리적 구조도는 [그림 3]과 같다. [그림 3]에서 각 클라이언트들은 네트워크에 부착된 클러스터 공유 파일 시스템을 중앙 집중적인 제어 없이 상시 공

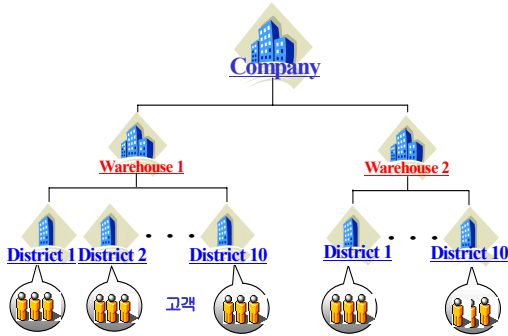
유할 수 있는 환경이다. 이러한 클러스터 공유 파일 시스템 환경에서 상용 DBMS를 설치하여 운영한다. DBMS의 각 엔진 및 데이터베이스 파일들은 클러스터 내의 어느 저장장치에 설치될 지 클러스터 서버 입장에서는 알 수 없으며 오직 클러스터 파일 시스템의 제어 하에 설치되게 된다. 클러스터 서버 1은 SAN 상에 연결된 저장장치들의 그룹으로 형성된 공유 파일 시스템 위에 설치된 데이터베이스 시스템에 접근하여 TPC-C 성능평가 프로그램을 수행하고 그 결과를 측정하게 된다. 클러스터 파일 시스템 환경과 성능평가 결과를 비교하기 위하여 [그림 4]와 같이 전형적인 기존 분산 파일 시스템의 한 유형인 NFS상에 데이터베이스 시스템을 설치한 후 같은 TPC-C 성능평가 프로그램을 수행하여 그 결과를 비교한다. 또한 분산 환경이 아닌 독립적 시스템의 결과와 비교하기 위해 일반적인 리눅스 시스템의 EXT3 파일 시스템 상에 데이터베이스 시스템을 설치하고 같은 평가 프로그램을 수행하여 그 결과를 비교한다.

### 4.2 성능평가 도구 및 방법

클러스터 파일 시스템 위에 설치된 상용 DBMS 오라클 9i의 성능평가를 통해 클러스터 파일 시스템의 집중적인 저장장치 접근 성능을 측정하고, 이 결과를 기존 독립 파일 시스템인 리눅스 시스템의 EXT3 파일 시스템과 비교하고 또한 NFS로 마운트 된 저장장치에 탑재된 데이터베이스 시스템의 성능과도 비교하여 기존 분산 파일 시스템과의 성능 차이를 측정한다.

성능평가 도구로는 표준 벤치마크 테스트로 인정한 OLTP 성능평가 도구인 TPC-C 도구를 활용한다. TPC-C 성능평가 도구는 온라인 물품판매 회사의 일련의 작업을 시뮬레이션 하는 성능평가 도구로서, 회사에서 고객으로부터 주문받고, 주문 상황 파악, 물품배송, 재고관리 등의 행위를 시뮬레이션하기 위한 모든 작업을 데이터베이스로 구축한 도구이다[9]. [그림 5]에서와 같이 구축된 데

이터베이스에서 고객의 요청처리와 물품, 회사관리  
에 필요한 5개의 트랜잭션을 수행하도록 프로그램  
되어진 성능평가 도구이다.



[그림 5] 성능평가 도구 TPC-C의 논리적 스키마

구축된 스키마를 기반으로 수행될 5개의 트랜잭  
션은 신규 주문, 지불, 주문확인, 배송, 재고관리  
트랜잭션이며, 각 트랜잭션이 포함하고 있는 SQL  
기본 연산의 구성은 다음 <표 1>과 같다.

<표 1> TPC-C의 트랜잭션 연산 내용 분석

	New Order	Payment	Order Status	Delivery	Stock Level
Select	22	4.2	3.6	30	2
Insert	12	1	0	0	0
Update	11	3	0	30	0
Delete	0	0	0	10	0
Nonunique Select	0	0.6	0.6	0	0
Join	1	0	0	0	0

이와 같은 트랜잭션을 수행하기 위해 다양한 데  
이터베이스 물리적 구축방법을 적용한다. [그림 5]  
의 창고(warehouse) 한 개당 약 60M바이트의 저  
장공간을 차지하도록 프로그램되어 있으므로, 창  
고 10개를 갖는 실험과 20개를 갖는 실험을 구축  
하도록 데이터를 발생시키고 <표 2>와 같이 세가

지 방법(A, B, C)으로 물리적 데이터베이스를 구  
축한다. 각 방법의 차이점은 어느 파일 시스템에  
서 대규모 크기의 파일을 접근, 관리하는데 더 효  
율성을 갖는지를 측정하기 위해서이다. 데이터베  
이스 시스템이 갖는 데이터 파일은 파일 시스템  
입장에서 하나의 커다란 크기의 물리적 파일에 불  
과하므로, 파일 시스템 차원에서 대규모 크기의  
파일 관리 성능 평가를 하기 위함이다. 예를 들어  
실험 1A는 창고의 개수 10개를 위한 데이터 600M  
바이트가 발생되는 실험이며 실험 데이터 저장을  
위해 오라클 9i의 1개의 테이블 스페이스를 생성  
하며 이때 테이블 스페이스의 300M 바이트 규모  
의 물리적 데이터베이스 파일 2개를 갖도록 구축  
된 실험환경이다.

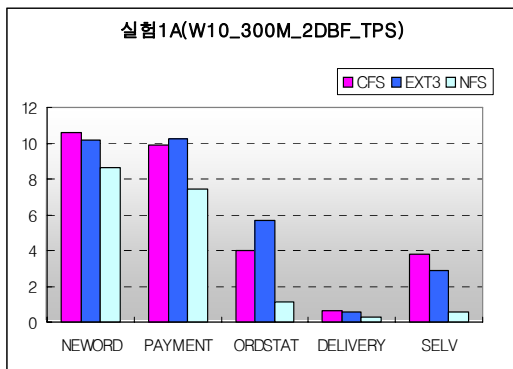
<표 2> 성능평가를 위한 물리적 구성방법

실험환경	실험 1	실험 2
Warehouse 개수	10개	20개
발생 데이터 크기	약 600M bytes	약 1.2G bytes
물리적 구성방법	A 1 Tablespace = 600M bytes 	1 Tablespace = 1.2G bytes 
	B 1 Tablespace = 600M bytes 	1 Tablespace = 1.2G bytes 
	C 1 Tablespace = 600M bytes 	1 Tablespace = 1.2G bytes 

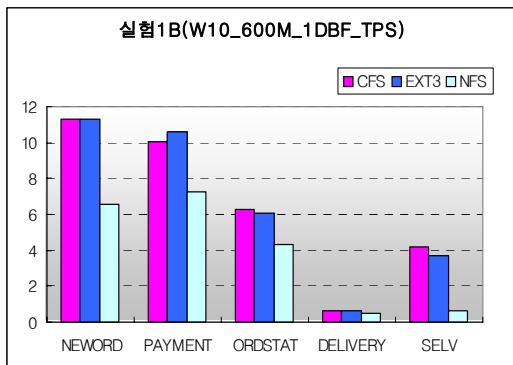
<표 2>의 6가지 다양한 물리적 구성방법에 따  
른 TPC-C 성능평가 시험을 클러스터 파일 시스  
템, EXT3, 및 NFS 시스템에 대하여 수행하여 총  
18 세트의 성능평가 시험을 수행하고 그 결과를 측  
정하여 각 시스템의 저장장치 접근 성능을 보인다.

### 4.3 성능평가 결과 분석

[그림 6]은 실험 1A, 즉 300M 바이트 크기의 데이터베이스 파일 2개를 갖는 600M 바이트 크기의 테이블 스페이스에 실험 데이터 약 600M 바이트를 구축하고 TPC-C 프로그램의 5개 트랜잭션을 CFS, EXT3, NFS 파일 시스템 상에서 각각 수행하여 얻은 성능 평가 결과이다. 분산 파일 시스템 NFS 보다 CFS 환경에서 우수한 성능평가 결과를 보이며, 또한 다른 여러 노드와 공유할 수 있는 기능을 제공하면서도 독립적으로 운영되는 EXT3 시스템과 유사한 성능을 가짐을 알 수 있다.



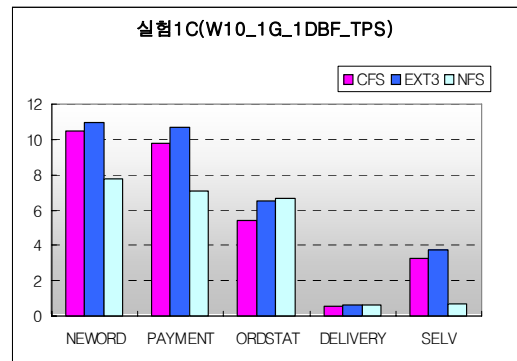
[그림 6] 실험 1A의 성능평가 결과



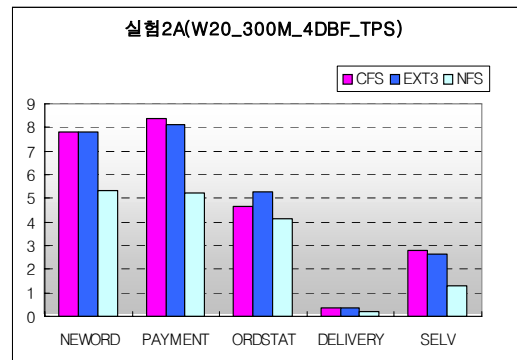
[그림 7] 실험 1B의 성능평가 결과

실험 1B, 1C 또한 실험 데이터의 크기가 600M 바이트인 성능평가로서 물리적 데이터 파일의 크기를 600M 바이트와 1G 바이트로 각각 적용하여

그 결과를 측정된 결과이다. 물리적 파일의 크기가 커질수록 CFS의 성능은 EXT3 보다 약간 느리게 측정되고 있으나, 기존 분산 파일 시스템인 NFS 보다는 2배 이상의 효율적 성능 우세를 보이고 있음을 [그림 7], [그림 8]에서 알 수 있다. 또한 물리적 데이터 파일의 크기가 커질수록 모든 시스템에서 다소 감소된 성능을 보이고 있어, 파일 시스템 입장에서 대규모 크기의 파일을 다루는데 오버헤드를 보이고 있을 수 있다.



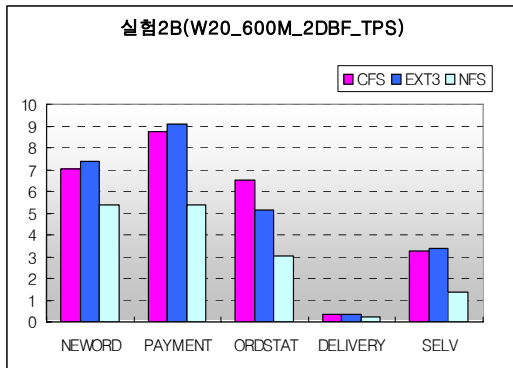
[그림 8] 실험 1C의 성능평가 결과



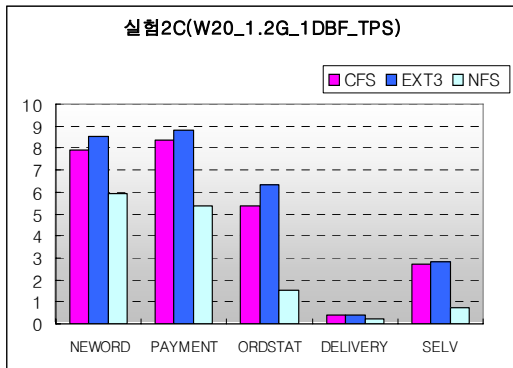
[그림 9] 실험 2A의 성능평가 결과

실험 2A, 2B, 및 2C는 실험 1보다 창고의 개수를 증가시켜 총 발생하는 실험 데이터의 크기를 약 1.2G 바이트가 되도록 한 후 성능평가를 측정된 실험이다. 따라서 실험 데이터를 저장하기 위한 테이블 스페이스 역시 최소 1.2G 바이트 크기의 테이블 스페이스를 가져야 한다. 테이블 스페

이스는 한 개로 고정 시킨 상황에서 테이블 스페이스가 갖는 물리적 데이터 파일의 크기를 각각 300M 바이트, 600M 바이트, 1.2G 바이트로 다양하게 변화시키고, 세 가지의 다양한 파일 시스템 상황에서 성능평가를 측정하여 그 결과를 [그림 9~11]에 나타내고 있다.



[그림 10] 실험 2B의 성능평가 결과



[그림 11] 실험 2C의 성능평가 결과

성능평가 결과에서 알 수 있듯이, 실험 데이터의 증가로 물리적 데이터 파일의 크기 및 테이블 스페이스가 증가하여 각 트랜잭션의 성능평가 결과는 실험 1 보다 다소 감소되었음을 알 수 있으며, 대규모 크기의 물리적 파일로 구성된 데이터베이스 시스템에서 보다 적당한 크기의 파일로 구성된 데이터베이스에서 더 좋은 성능 평가를 나타내고 있음을 알 수 있다. 또한 SANique™ 시스템은 클러스터 파일 시스템으로서 중앙 집중 제어

역학을 하는 서버 없이 진정한 공유를 제공하는 공유 파일 시스템이면서 기존의 독립적으로 운영되는 EXT3 파일 시스템과 유사한 수준의 성능을 보이고 있음을 성능평가를 통해 알 수 있었으며, 기존 분산 파일 시스템보다 우월한 성능을 보임을 알 수 있다.

## 5. 결 론

본 논문은 SAN 기반의 대용량 저장장치들을 군집화하여 파일 서버 없이 직접 데이터 전송이 가능한 클러스터 공유 파일 시스템인 SANique™의 설계방법 및 각 구성요소들의 기능을 기술하였으며, 특히 클러스터 파일 시스템과 기존 파일 시스템과의 성능 평가 비교를 위해 상용 데이터베이스 시스템을 활용한 TPC-C 표준 벤치마크 성능 평가를 수행하여 그 결과를 비교하였다. 클러스터 파일 시스템의 기본 목적인 여러 서버의 공유 기능을 제공하면서, 클러스터링 된 저장장치에 대한 접근 성능을 표준 벤치마크 평가를 통해 우수성을 보였다. 또한 다양한 성능평가 비교를 위해 물리적 파일의 크기를 다양하게 변화하여 파일 시스템의 대규모 물리적 파일의 처리 성능에 대해서도 평가 하였다.

현재 클러스터 파일 시스템의 간의 비교를 진행 중이며, 개발된 SANique™ 클러스터 파일 시스템의 확장성에 대한 성능 평가를 계획중이다. 군집화된 저장장치의 크기 변화 및 공유 서버 대수의 증감에 따른 성능평가 결과를 측정 계획 중이다.

## 참 고 문 헌

- [1] VERITAS Software Corp., Veritas Volume Manager, <http://www.veritas.com>.
- [2] D. Teigland, "The Pool Driver : A volume Driver for SANs", *Master's Degree Thesis, University of Minnesota, Dept. of Electrical and Computer Engineering*, 1999.



- [3] H. Maulshagen, "Logical Volume Manager for Linux", Sistina Technical Memo, <http://www.sistina.com>.
- [4] S. R. Soltis, T. M. Ruwart, and M. T. O'keefe, "The Global File Systems", *Proc. Of the 5th NASA Goddard Conference on Mass Storage Systems and Technologies*, 1996.
- [5] R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, and B. Lyon, "Design and Implementation of the Sun Network File Systems", *Proc. Of the Summer USENIX Conf*, 1985.
- [6] M. Satyanarayanan, "Scalable, Secure, and Highly Available Distributed File Access", *IEEE Computer*, 1999.
- [7] U. Vahalia, *Unix Internals : The New Frontiers*, Prentice-Hall, NJ, 1999.
- [8] K. Lee, "Failure Recovery in the Linux Cluster File System SANique™", *KIPS Trans. Part A*, Vol.8, No.4, 2001.
- [9] Transaction Processing Performance Council, <http://www.tpc.org>.
- [10] 이규웅, "고가용성 클러스터 파일 시스템 SANique™의 분할 그룹 탐지 및 회복 기법", 『한국멀티미디어 학회논문지』, 제7권, 제4호, 2004.
- [11] R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, and B. Lyon, "Design and Implementation of the Sun Network File Systems", *Proc. Of the Summer USENIX Conf*, 1985.
- [12] M. D. Dahlin, "Severless Network File Systems", *Ph. D. Thesis at Computer Science Graduate Division of University of California at Berkely*, 1995.

## ◆ 저 자 소 개 ◆

**이 규 웅 (leekw@sangji.ac.kr)**

서강대학교 대학원에서 공학석사를 하고 동 대학원에서 데이터베이스 시스템 전공으로 공학박사를 받았다. 한국전자통신연구원 선임연구원을 지냈고, 현재 상지대학교 컴퓨터정보공학부 부교수로 재직중이며 주요 연구 관심분야로 클러스터 데이터베이스 시스템, 클러스터 기반 데이터 분산 트랜잭션 관리, 주기억장치 데이터베이스 시스템 등을 연구하고 있으며, 정보통신부 및 중소기업청 주관의 클러스터 시스템 데이터 관리 분야의 연구를 수행하고 있다.