

분산 환경에서의 NGS를 위한 성능평가 도구 설계

강윤희*, 정승국**

*백석대학교 정보통신학부

**한국전자통신연구원

e-mail:yhkang@bu.ac.kr

Design of Performance Evaluation Tool for NGS on distributed System

YunHee Kang*, SeungKook Cheong**

*Division of Information and Communication, BaekSeok

University

**ETRI

요약

NGS(Next Generation Storage) 시스템은 전형적인 분산파일 시스템 구조의 병목 현상을 제거하고 입출력 대역폭을 늘려 성능을 최대화 하기 위한 차세대 저장시스템으로 기존의 저장시스템과는 달리 DRAM을 기반으로 스토리지를 구성하고 있다. NGS 시스템의 대용량 지원 및 기업 내부에서의 활용을 위해서는 SAN 기반에서 활용할 수 있도록 설계되어야 하며, SAN 환경에서 성능 향상을 위한 연구가 필요하다. 본 논문에서는 NGS 시스템에 대한 성능평가 및 분산 환경에서 NGS 를 활용하기 위한 성능평가도구 개발을 기술한다. 성능 도구의 활용은 전형적인 전체 시스템 아키텍처 내의 병목 현상을 제거하고 입출력 대역폭을 늘려 성능을 최대화 할 수 있어야 한다.

1. 서론

하드웨어와 소프트웨어 기술의 발전에 따라 고속의 네트워크를 통해 대량의 데이터 요청을 처리하기 위한 분산 저장 시스템인 SAN(Storage Area Network)에 대한 연구가 진행되고 있다. SAN 기반의 저장장치는 화이버 채널(fiber channel) 인터페이스 및 iSCSI 등의 네트워크 부착형 저장장치 형태로 발전되고 있으며, NFS 기반의 분산 파일시스템은 SAN 기반의 공유파일 시스템으로 진화되고 있다[3]. SAN 기반 파일시스템은 NFS와 같은 데이터 공유 프로토콜을 채용한 네트워크 접속(network attached) 파일서버에 비해 다중, 이형 플랫폼 지원 및 높은 대역폭에 대한 요구 대응이 용이하고 관리 비용을 줄일 수 있는 장점이 있다.

NGS(Next Generation Storage) 시스템은 전형적인 분산파일 시스템 구조의 병목 현상을 제거하고 입출력 대역폭을 늘려 성능을 최대화 하기 위한 차

세대 저장시스템으로 기존의 저장시스템과는 달리 DRAM을 기반으로 스토리지를 구성하고 있다.

외국에서의 DRAM 기반 메모리 저장장치는 TMS(Texas Memory Systems), SD(Solid Data Systems) 등 일부 업체에서 개발이 진행 중이다. SD사에서는 DRAM 기반 메모리 저장장치를 SAN(Storage Area Network)에서 RAID와 결합하여 파일 캐싱에 활용하고자 하는 연구를 진행하고 있다[1,2]. 국내에서도 기업 및 정부 부처에서는 DRAM 기반 메모리저장장치 시스템에 대한 개발을 진행하고 있다. 주메모리는 휘발성(volatile)의 매체 특성으로 인해 저장된 정보의 유지를 위해서는 지속적인 전원의 공급이 필수적으로 이루어져야 하며 백업을 위한 시스템이 요구된다.

본 논문에서는 NGS 시스템에 대한 성능평가 및 분산 환경에서 NGS 를 활용하기 위한 환경개발을 기술한다. 성능평가 환경개발은 전형적인 전체 시스템 아키텍처 내의 병목 현상을 제거하고 입출력 대

역폭을 늘려 성능을 최대화 할 수 있어야 한다. 또한 저장장치로 부터 응용까지의 파일서버 스택의 경로 없이 직접적으로 데이터 전송을 할 수 있어야 하며, 병렬 데이터 접근을 보장하여 응용별로 입출력 대역폭을 증가할 수 있어야 한다.

본 논문의 구성은 다음과 같다. 2장에서는 NGS 성능분석을 기술하고 3장에서는 NGS 평가도구 설계를 기술한다.

2. NGS 성능분석

디스크는 매체 특징으로 인해 디스크 탐색 시간 (seek time)을 가지며, 이는 전체 I/O 시간의 80%를 차지하고 있다. 이러한 지연시간을 줄이기 위해 유닉스 및 리눅스 기반의 디스크 기반 파일 시스템은 I/O성능 향상을 위해 버퍼캐쉬(Buffer cache)를 활용하고 있다[3,5,6,7]

최근 대표적인 SSD 저장장치인 플래시 메모리는 비휘발성(non-volatile), 빠른 접근 속도, 저전력, 소형 및 경량의 특징으로 인해 다양한 분야에서 활용되고 있다. 게임기, 카메라, PDA, 휴대폰 및 셋탑박스 등 휴대용 전자기기에서 플래시 메모리는 저장매체로 활용되고 있으며, 최근에는 노트북 컴퓨터의 저장매체로 플래시 메모리가 디스크를 대체하고 있다. 그러나 플래시 메모리는 매체의 특성상 쓰기 위해 기존의 내용을 갖는 블록을 지우고 써야 하는 등의 제약으로 인해 기록연산 지연 등의 문제점이 있다.

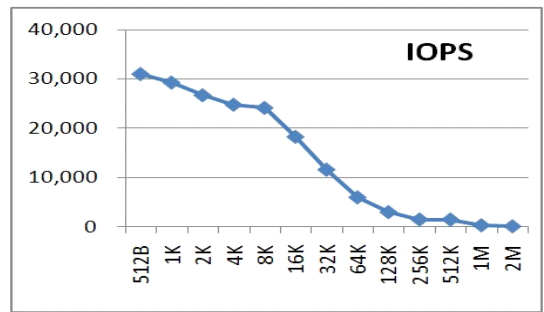
이의 대안으로 플래시 메모리 보다 성능 및 확장성이 뛰어난 DRAM 기반 메모리 저장시스템의 구성은 플래시 메모리의 연산지연을 해결할 뿐만 아니라 읽기와 쓰기연산 속도향상도 기대할 수 있다.

성능평가를 위한 대상 저장장치는 태진인포텍의 Jetspeed 으로 SCSI 기반의 전용 장치 드라이버를 통해 입출력이 이루어진다. 표 1은 성능 평가를 위한 대상시스템의 하드웨어 플랫폼 및 운영체제 구성을 기술한 것이다.

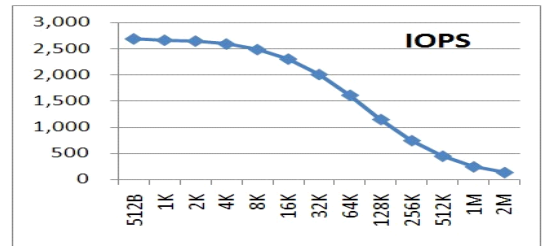
[표 1] 대상 시스템

구성요소	사양	특징
CPU	Intel(R) Xeon(R) CPU E5472@ 3.00GHz 64 bit Xeon 쿼드코어 X 2	L1 I cache: 32K L1 D cache: 32K L2 cache: 6144K X 2
메모리	FBDIMM DDR2 1GB(PC2-6400) X 8	
프로세서 버스	FSB 1600MHz	
OS 디스크	Seagate S-ATA2 500GB,	7200 rpm 16 MB disk buffer
메모리 저장장치 외부	Jetspeed 16G	
데이터 버스	PCI Express	2.5Gb/s:Width x4
운영체제	RHEL 4_update 2버전	Linux kernel 2.6.20

그림 2와 그림 3은 SAN 환경에서 디스크 저장장치와 NGS 선행시스템을 대상으로 한 성능실험 결과를 보인 것으로 디스크의 경우 임의읽기가 순차읽기에 비해 10%의 IOPS 성능을 보인 반면 DRAM 기반 메모리 장치인 NGS는 순차읽기와 임의읽기의 차이가 없음을 알 수 있다.

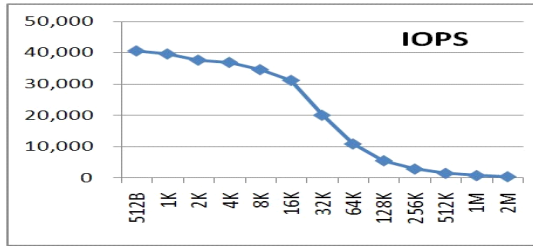


a) 디스크 순차읽기

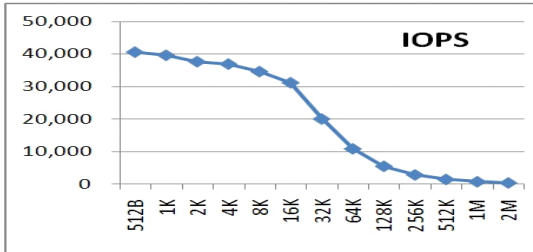


b) 디스크 임의읽기

[그림 2] 디스크 장치의 읽기성능



a) NGS 실행시스템 순차읽기



b) NGS 실행시스템 임의읽기
[그림 3] NGS의 읽기성능

3. NGS 평가도구 설계

3.1 성능평가도구 요구사항 정의

분산환경에서 저장시스템의 설계 및 구성을 위해서는 다양한 SAN 아키텍처에서의 정량적인 성능평가를 필요로 한다. 성능분석을 위한 부하의 생성을 위한 주요한 파라미터로는 다음과 같다.

- 블록크기 : 단일 테스트 런에서 서로 다른 블록크기를 설정한다. 이를 통해 실제적인 응용에서의 데이터 블록요청에 대한 처리의 결과를 예상할 수 있다. 또는 이는 NGS 저장장치의 활용목적에 따라 값을 변화시킨다.
- 트랜잭션 수 : 테스트 런에서 생성하는 트랜잭션의 수를 제어한다. 트랜잭션은 단일의 블록 수준 읽기/쓰기 접근으로 정의한다.
- 안정 주기 : 실제적으로 읽기/쓰기 성능의 안정화가 이루어질 때까지의 임시적인 시간으로 시작 성능측정 전까지의 시점을 정의한다.
- 요청 주기 : 연속적인 입출력 요청 간의 주기를 정의한다.
- 읽기/쓰기 비율 및 순차/임의 비율 : 응용의 형태에 따라 I/O의 비율을 조정한다.

기본적 저장시스템의 목적은 높은 대역폭(high bandwidth)과 데이터접근의 높은 신뢰성(high reliability)을 제공하는 것이다. 추가적으로 분산 환경에서 요구하는 위치 투명성(location transparency), 확장성(scalability)을 요구하고 있다. 이러한 저장시스템은 멀티미디어 서버, 데이터웨어 하우스, 실시간 데이터 저장 등 다양한 목적으로 사용한다. 또한 다양한 형태의 파일 크기 및 다양한 유형의 사용자의 입출력 요청을 처리하여야 한다.

성능평가 도구 설계 시에는 단일 지역파일 시스템과 달리 분산 저장 시스템 네트워크에 대한 고려가 필요하다. 즉, NGS는 공통, 공유 저장소로서 정의될 수 있다. 이는 이형의 클라이언트 컴퓨터 간의 데이터 공유를 가능하도록 지원해야 한다. 또한 단일 파일에 대한 동시 읽기 및 쓰기에 대한 고려도 필요하다.

올바른 성능평가를 위해서는 성능을 결정하는 행태에 대한 정확한 측정 및 분석이 필요하다. 일례로 I/O 요청에서 여러 서브 시스템이 활용될 수 있다. 또한 문맥 교환 수 및 시스템 호출의 오버헤드에 대한 고려가 필요하다. NGS 시스템의 병렬 I/O 및 실시간 백업을 위해 RAID 기반의 저장 시스템에 대한 성능평가는 필수적이다. NGS 전용 파일시스템의 설계를 위한 성능평가는 대용량의 저장 시스템을 지원하며, 빠른 회복이 가능한 리눅스 기반의 저널링 파일시스템으로 한정한다.

3.2 NGS 성능평가도구 구성

NGS 성능평가도구의 주요구성요소는 벤치마크도구 응용, 파일시스템, 장치드라이버 및 트레이스 생성 및 분석로 구성된다.

3.2.1 벤치마크도구

파일시스템의 벤치마크 도구는 마이크로 벤치마크(micro-benchmark)와 매크로 벤치마크(macro-benchmark)로 구분할 수 있다. 마이크로 벤치마크는 순차적인 읽기/쓰기, 다중스트림의 읽기/쓰기, 임의 읽기/쓰기, 파일 생성/삭제, 파일 메타데이터 연산 등의 성능측정을 목적으로 한다. 마이크로 벤치마크 프로그램의 설계는 파일시스템 실행의 단일측면의 값을 명확하게 측정하여야 한다[8,9]. 매크로 벤치마크는 소형 데이터베이스, 대형 데이터베이스, 다중쓰레드 기반의 웹서버, 메일 서버, 비디오 서버 등과 같은 실제 응용환경에서 발생하는 작업부하의 특징에 따른 성능측정을 목적으로 한다[7]. 벤치마크 프로그램의 목적은 실제로 작업부하를 주었을 때 파일시스템이 어떻게 실행되는지에 대한 평가를 위해 사용된다. 마이크로 벤치마크는 이를 위한 평가하고자 하는 환경에 따라 특별한 파라미터를 설정한 후 수행한다.

3.2.2 파일시스템

설계된 NGS 성능평가 시스템에서 벤치마크는 성

능평가를 위한 응용으로서 파일시스템의 요청은 가상파일시스템(virtual file system) 과 실제 파일시스템(practical file system)을 통해 장치드라이버에 요청을 생성하게 된다. 파일시스템의 설계를 위한 성능평가는 대용량의 저장 시스템을 지원하며, 빠른 회복이 가능한 리눅스 기반의 저널링 파일시스템으로 한정한다[4]. 성능평가를 위한 파일시스템은 선형 시스템 상에 개별 파티션으로 구성한다.

3.3.3 장치드라이버

파일시스템의 시스템호출은 장치드라이버의 수행 후 블록단위 입출력을 통해 인터럽트가 발생시키고 장치드라이버의 입출력결과는 DMA 를 통해 커널의 버퍼캐쉬로 복사가 수행되게 되며 실제파일시스템과 가상파일시스템으로 응답결과가 전달된다.

3.4.4 트레이스 관련 모듈

NGS 의 평가대상 시스템의 다차원적인 평가를 위해 응용 수준에서 파일시스템 벤치마크를 사용한다. 또한 파일시스템의 호출을 후킹(hooking) 하기 위한 랩핑(wrapping) 기능을 커널모듈로 구성한다. 모든 시스템 호출은 후킹랩핑모듈에 의해 가로채게 되며, 해당 시스템 호출은 트레이스로 생성되게 된다. 생성되어진 트레이스는 데이터 I/O 패턴의 분석을 위한 트레이스 분석모듈에 의해 이루어진다.

5. 결론

본 논문에서는 고속의 입출력을 지원하는 DRAM 기반의 저장매체로 사용하는 NGS 시스템의 성능평가를 위한 도구의 설계를 기술하였다. 성능평가를 위한 도구는 실험결과 분석을 통해시스템 장애시 발생하는 결함 처리를 처리하는 기법에 따른 성능영향을 평가 할수 있으며, 최적의 I/O 성능 및 안정성을 보장하기 위해 필수적이다.

성능평가도구의 상세설계에는 분산환경에서의 방대한 확장가능 파일 시스템을 구성할 수 있는 평가환경을 제공할 수 있도록 구성되어야 한다. 향후 벤치마크 도구 개발시 인터넷 기반의 멀티미디어 스트리밍 서비스를 위한 IPTV, 데이터웨어하우스 등의 기업 내의 저장 시스템으로 활용될 수 있도록 하기 위해서는 응용에 대한 데이터 접근패턴에 대해 추가적으로 고려되어야 한다.

참고문헌

- [1] Solid Data systems, "Impact of Solid-state disk on high-transaction rate databases", Solid data systems, Inc. White paper, 2005 Feb
- [2] TMS, "Increase Application Performance with Solid State Disks", TMS white paper, 2008.Feb
- [3] Nava Aizikowitz, Alex Glikson, COMPONENT-BASED PERFORMANCE MODELING OF A STORAGE AREA NETWORK, Proceedings of the 2005 Winter Simulation Conference
- [4] <http://tldp.org/LDP/tlk/fs/filesystem.html>
- [5] Dominic Giampaolo, Practical File System Design with the Be File System, Morgan Kaufmann Publishers, 1999
- [6] Elizabeth Shriver , Christopher Small and Keith A. Smith, Why does file system prefetching work?, Proceedings of the USENIX Annual Technical Conference Monterey, California, USA, June 6-11, 1999
- [7] J. Katcher, "PostMark: A New File System Benchmark," Technical Report 3022, Network Appliance, 1997.
- [8] D. Capps and W.D. Norcott, "Iozone Filesystem Benchmark," <http://www.iozone.org/>, 2004.
- [9] R. Coker, "Bonnie++ Benchmark Tool," <http://www.coker.com.au/bonnie++/>, 2004.