제22회 한국정보처리학회 추계학술발표대회 논문집 제11권 제2호 (2004. 11)

VOD 전용 파일 시스템 개발 및 성능 분석

김병섭, 김홍연, 김영철, 원종호, 이미영
한국전자통신연구원 디지털홈연구단 인터넷서버그룹
e-mail : {powerkim, kimhy, kimyc, jhwon, mylee}@etri.re.kr

Implementation and Performance Evaluation of the VOD Multimedia File System

Byoung-Seob Kim Hong-Yeon Kim Young-Cheol Kim Jong-Ho Won Mi-Young Lee
Internet Server Technology Group, Digital Home Research Division, ETRI

요 약

차세대 인터넷 서버는 인터넷을 통하여 여러 사용자에게 HDTV 급 고품질 멀티미디어 서비스를 제공하고자 개발된 시스템이며, 이를 위하여 디스크 연결 및 네트워크 기능을 정합 시킨 특화된 NS 카드를 개발하였다. Contents Container 파일 시스템은 NS 카드 전용으로 개발된 멀티미디어 파일 시스템이며, 본 논문에서는 Contents Container 파일 시스템에 대한 개발 내용을 기술하고, 개발 시스템의 장단점을 파악하고자 EXT3 파일 시스템과 비교 분석하였다.

1. 서론

한국전자통신연구원에서 개발 중인 "차세대 인터넷 서버†(Next Generation Internet Server, NGIS)" 는 200 명의 동시 사용자에게 20Mbps 급의 고품질 HDTV(High-Definition Television) 급의 실시간 멀티미디어 서비스를 제공하는 것이 목표이며, 빌딩, 아파트, 학교 등의 지역망을 이용하여 스트리밍 서비스를 가능하도록 네트워킹 기능을 강화한 지역 서버와 데이터 센터용 광역 서버를 포함하는 계층적 구조를 갖는 서비스 시스템이다[1]. 광역 서버는 서비스하고자 하는 멀티미디어 파일 등의 컨텐츠를 저장하고 있으며, 각 지역 서버는 해당 지역망에서 필요한 컨텐츠들을 광역 서버로부터 필요 시 전송 받아 저장 관리하며 양질의 멀티미디어 서비스를 제공한다. 또한, 지역 서버가 보유하고 있는 컨텐츠들은 가까운 지역 서버들과 연계되어 상호 분배 서비스 된다. 이러한 지역 서버가 원활한 멀티미디어 서비스 제공을 위해서 기존의 파일 시스템이 지원하기 어려운 대용량 및 다수 사용자를 지원하는 전용 파일 시스템인 Contents Container 멀티미디어 파일 시스템(이하 CCFS)을 개발하였다.

본 논문에서는 CCFS 의 개발에 대한 주요 내용을 설명하고 CCFS 의 성능을 리눅스 기본 파일 시스템인 EXT3 와 비교 분석하였다.

본 논문은 다음과 같이 구성된다. 먼저 2 장에서 CCFS 에 대한 개발 내용을, 3 장에서 시험 환경, 시험 방법 및 시험 결과를 기술하고, 끝으로 4 장에서 결론을 맺는다.

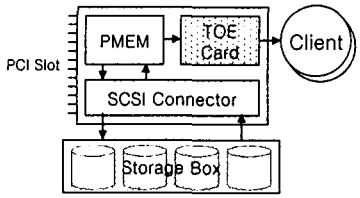
2. Contents Container 멀티미디어 파일 시스템 개발

2.1 NS 카드 개요

차세대 인터넷 서버 시스템은 HDTV 급 고품질 실시간 서비스를 제공하기 위하여 한 카드에 메모리와 스카시(SCSI:Small Computer System Interface) 디스크 연결 및 네트워크 기능을 집적한 NS(Network and Storage) 카드를 핵심 기술로 설계 및 개발하였다[2]. NS 카드는 <그림 1>과 같이 멀티미디어 스트리밍 서비스 시에 병목 현상이 발생할 수 있는 메인 메모리와 네트워크 카드의 데이터 흐름을 없애기 위하여 전송 데이터의 버퍼 위치를 NS 카드 내부의 PMEM(PCI Memory)으로 집적하였으며, TCP/IP 포로토콜 처리를 위한 소프트웨어적인 부하를 줄이고자 TOE 카드를 내장하였다. 결과적으로 NS 카드를 사용하는 멀티미디

† 본 연구는 정보통신부가 지원하는 정보통신연구개발사업 중 차세대 인터넷 서버 기술 개발 사업으로 진행 중에 있음.

어 서비스 시스템은 시스템 부하를 줄일 수 있게 되어 대용량 파일 스트리밍 및 동시에 많은 사용자에게 양질의 서비스를 제공할 수 있다.



• PCI : Peripheral Component Interconnect
• PMEM : PCI Memory
• TOE : TCP/IP Offload Engine
• SCSI : Small Computer System Interface

<그림 1> NS 카드 구성

이러한 NS 카드를 사용한 스트리밍 서비스의 효과적인 지원을 위해서는 NS 카드 내부의 PMEM 및 NS 카드 드라이버에 의해 보여지는 디스크(SDA:Stream Disk Array) 등을 효과적으로 사용할 수 있는 파일 시스템이 필요하며, 본 연구에서는 대용량 스트리밍 지원 및 다수 사용자의 지원이 가능한 전용 파일 시스템으로 Contents Container 멀티미디어 파일 시스템을 개발하였다.

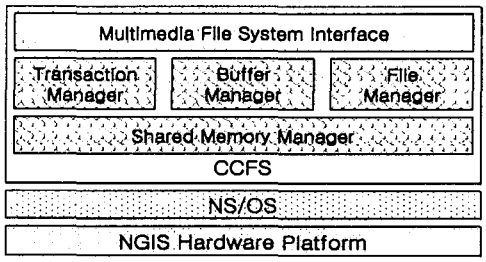
2.2 CCFS 특징

차세대 인터넷 서버에 특화된 Contents Container 멀티미디어 파일 시스템의 주요 특징은 다음과 같다.

- NS 카드에 최적화
  NS 카드가 최적으로 사용되도록 하며, NS 카드에서 제한하고 있는 고정된 크기의 디스크 I/O 로 인한 성능 저하가 발생하지 않도록 했으며, 보다 우수한 성능을 얻기 위하여 미처리 디스크 드라이버 (raw device driver)를 사용하여 디스크를 관리함으로써, 운영체제의 버퍼를 거치지 않고 디스크의 특정 영역을 직접 읽고 쓸 수 있다.
- 접근 시간 최소화
  고화질의 멀티미디어 데이터 서비스를 지원할 수 있도록 CCFS 가 관리하는 멀티미디어 데이터에 대한 접근 시간을 최소화 하고, HDTV 급의 대용량 멀티미디어 데이터에 대한 지속적으로 안정적인 접근 성능을 지원하기 위하여 최대 10 기가바이트 크기의 멀티미디어 데이터에 대하여 최대 한번의 인디렉션만으로 접근이 가능하다.
- 대용량 파일 지원
  일반 32 비트 프로세서 환경에서 운영되는 일반 파일 시스템에서 지원하지 못하는 대용량의 멀티미디어 파일 크기(최대 2TB 까지)을 지원한다.
- 동일 파일 동시 Read/Write
  차세대 인터넷 서버는 효과적인 디스크 사용을 위해서 파일의 앞부분 일부를 제공하다가 사용자의 요구 시에 파일 서비스와 동시에 컨텐츠 분배 시스템이 광역 서버 및 주위의 지역 서버로부터 나머지 파일에 대하여 실시간으로 다운로드 하는 Prefix 기능이 있으며, 이에 대한 기능 제공을 위해서 CCFS 는 동일한 파일에 대하여 기록과 읽기를 동시에 할 수 있는 동시성을 제공한다.
- 회복 및 빠른 재가동
  시스템 고장으로부터 CCFS 를 보다 빠르게 복구하고 재가동할 수 있는 회복 방법을 지원한다. 또한, 로그의 재사용과 효율적인 관리 방법을 제공한다 [3].
- 멀티 프로세스 및 멀티 쓰레드 환경 지원
  다양한 응용 환경에서 CCFS 가 활용될 수 있도록 멀티 프로세스 환경과 멀티 쓰레드 환경을 지원한다.

2.3 CCFS 블록 구조

CCFS 의 내부 블록 구조는 <그림 2>와 같이 공유 메모리 관리기(Shared Memory Manager), 트랜잭션 관리기(Transaction Manager), 버퍼 관리기(Buffer Manager), 파일 관리기(File Manager), 파일 시스템 인터페이스(File System Interface)로 구성되며 각 기능은 다음과 같다.



<그림 2> Contents Container 블록 구조

- 공유 메모리 관리기
  멀티 프로세스 또는 멀티쓰레드 환경에서 CCFS 가 사용하는 시스템 메타 데이터(open file table, lock table 등)와 같은 공유 정보를 포함하는 공유 영역을 관리한다.
- 트랜잭션 관리기
  CCFS 의 동시 사용에 대한 안정성 및 시스템 오류 시 시스템 메타 데이터에 대한 일관성을 제공한다. 트랜잭션 관리기는 각 기능에 따라 트랜잭션, 잠금, 회복 모듈로 구성되며, 세부 내용은 다음과 같다.
  - 트랜잭션 모듈은 시스템의 ACID(Atomicity, Consistency, Isolation, Durability) 특성을 제공하기 위한 BOT(Begin Of Transaction), EOT(End of Transaction)등을 지원한다.
  - 잠금 모듈은 다중 트랜잭션에 의해 공유되는 데이터 접근을 제어하고, 공유 데이터의 일관성을 보장하기 위하여 래치(Latch)를 제공하며, 래치를 사용한 공유 자원의 순차 접근으로 교착 상태의 발생을 방지한다.
  - 회복 모듈은 시스템 고장 시 메타 데이터에 대한 회복 기능을 제공함으로써 시스템을 일관된

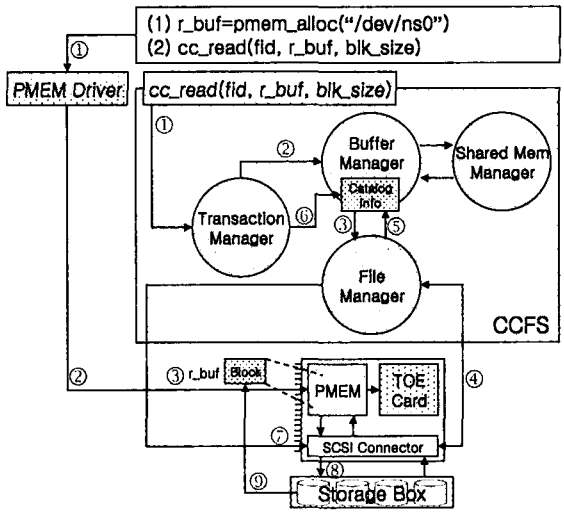
상태로 복구한다. 이를 위하여 로그를 기록하며, 로그는 고정된 크기의 공간을 반복적으로 재이용하는 원형 로그(Circular Log)를 사용한다[3].

- 버퍼 관리기
  자주 사용되는 카탈로그 및 inode 정보 등을 버퍼를 사용하여 접근함으로써 디스크 입출력을 줄이고 시스템 성능을 향상시킬 수 있도록 버퍼를 제공 및 관리한다. 버퍼 교체 알고리즘은 가장 오래 동안 사용되지 않은 버퍼를 교체하는 LRU(Least Recently Used) 정책을 따른다.
- 파일 관리기
  NS 카드가 제공하는 디스크 공간을 관리하며, 응용 프로그램에게 논리적인 파일 구조 및 디렉토리 구조를 제공한다. 파일 관리기는 기능에 따라 디스크 관리 모듈, 파일 관리 모듈, 디렉토리 관리 모듈로 구성되며, 세부 내용은 다음과 같다.
  - 디스크 관리 모듈은 NS 장치의 디스크를 블록(block), 블록 그룹(block group), 익스텐트(extent), 파일(file), 할당 그룹(allocation group), 디스크(disk) 등의 객체를 이용하여 관리하며, 주요 내용은 다음과 같다.
    - 블록 : 입출력(메타 데이터) 기본 단위
    - 블록 그룹 : 인접 블록을 PMEM 블록 단위로 묶은 것으로 스트리밍 데이터 입출력 단위
    - 익스텐트 : 인접 블록 그룹을 묶은 것
    - 할당 그룹 : 디스크를 여러 개의 논리적인 영역으로 분할하여 메타 데이터를 할당 그룹별 관리
  - 파일 관리 모듈은 CCFS 파일들을 관리하기 위해 각 파일에 하나의 inode 구조체를 유지하며, inode 는 현재 파일에 할당된 모든 익스텐트 식별자를 유지하기 위해 B+ 트리를 변형한 색인 구조를 사용한다.
  - 디렉토리 관리 모듈은 CCFS 에서 파일 이름과 이에 대응되는 inode 식별자로 구성된 디렉토리 항목을 관리한다. 디렉토리는 디렉토리 자체를 위한 inode 와 필요에 따라 다수의 블록들로 구성된다. 특정 디렉토리에 저장해야 할 디렉토리 항목수가 적을 경우 디렉토리 inode 내의 여유 공간에 직접 이들 자료를 관리하는 Stuffed 디렉토리 구조를 가지며 이를 초과할 경우 B+ 트리 구조를 가진다.
- 인터페이스
  사용자가 응용개발 시에 사용할 수 있는 POSIX(Portable Operating System Interface) 인터페이스 및 전용 인터페이스를 제공한다.

2.4 CCFS 동작 흐름

본 절에서는 CCFS 의 인터페이스 중 read 의 동작 흐름을 살펴 봄으로써 시스템 흐름의 이해 및 각 블록의 연동 관계를 기술한다.

<그림 3>은 CCFS 를 사용하여 특정 파일로부터 blk_size 만큼 데이터를 읽는 동작 흐름도이다.



<그림 3> Contents Container 동작 흐름도

(1) PMEM 드라이버를 사용하여 시스템 기동시 정의한 블록 사이즈 크기만큼 PMEM 버퍼를 r_buf 에 할당
  ① PMEM 드라이버의 pmem_alloc 인터페이스를 사용하여 1 블록 크기만큼의 PMEM 메모리 할당
  ② PMEM 드라이버는 지정한 NS 카드의 PMEM 1 블록을 할당하여,
  ③ 사용자 메모리 주소로 반환

(2) (1)에서 할당 받은 버퍼에 CCFS 의 파일의 내용을 blk_size 만큼 읽어 온다.
  ① 트랜잭션 관리기의 BOT 를 사용하여 트랜잭션의 시작점 기록
  ② 해당 파일의 카탈로그 정보가 존재하는 지 버퍼 탐색
  ③ 카탈로그 정보가 존재하지 않으면, 파일 관리기를 통하여 해당 정보를 요구
  ④ 파일 관리기는 NS 카드가 관리하는 디스크에서 해당 파일의 메타 데이터를 탐색하여,
  ⑤ 버퍼 관리기에 로드
  ⑥ 트랜잭션 관리기의 잠금 모듈을 사용하여 해당 메타 데이터에 공유 잠금을 설정한 후,
  ⑦ 파일 관리기로부터 해당 파일의 현재 offset 의 위치를 찾아서, ⑧ NS 카드의 디스크에서 해당 영역을 읽어서,
  ⑨ r_buf 에 복사

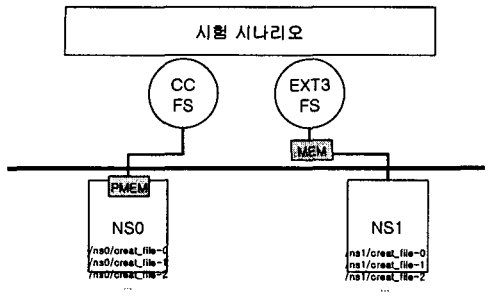
3. CCFS 성능 분석

본 연구에서는 CCFS 의 대용량 다수 사용자에 대한 서비스 가능성 및 취약점을 분석하고자 성능 시험을 수행하였다. 기존 시스템과의 비교를 위해서 리눅스 기본 파일 시스템인 EXT3 와 비교 시험하였다.

3.1 시험 환경

CCFS 와 EXT3 에 대한 시험 환경을 가급적 물리적으로

동일한 환경에서 제공할 수 있도록 하기 위해서 <그림 4>와 같은 시험 환경을 구성하였다.



<그림 4> 시험 환경

차세대 인터넷 서버(지역 서버)에 2 장의 NS 카드를 설치하고, 첫 번째 카드에는 CCFS 를 마운트하였으며, 두 번째 카드에는 EXT3 파일 시스템을 마운트 하였다. CCFS 의 경우 read/write 인터페이스에 사용되는 버퍼로 NS 카드에 내장되어 있는 PMEM 을 사용하였다. 그러나, EXT3 의 경우 PMEM 을 사용하면 커널 버퍼(캐시)로의 복사가 이루어지며, 시험 수행 현재 PMEM 과 일반 사용자 메모리(MEM)사이의 복사에 대한 성능 저하가 있어 일반 사용자 메모리를 사용한 버퍼로 시험하였다(이는 가급적 EXT3 를 최상의 시험 환경으로 구성하기 위해서 임).

3.2 시험 방법

시험 방법은 멀티 프로세스를 사용하여 사용자 가상 접근을 시뮬레이션 하였다. 하나의 프로세스는 한 사용자를 의미하며 각 사용자는 각자의 파일을 접근한다. 전송 속도는 파일을 버퍼에 읽는 시간까지의 속도이며, TOE 를 통한 전송은 파일 시스템의 범위를 벗어나므로 생략하였다. 성능은 CCFS/EXT3 순차(Sequential) 읽기, 랜덤(Random) 읽기에 대하여 측정하였으며, 시험 시나리오는 다음과 같다.

(1) 50 명의 사용자가 동시에 각자 서로 다른 50MB 파일을 접근 하는 경우
(2) 100 명의 사용자가 동시에 각자 서로 다른 50MB 파일을 접근 하는 경우
(3) 50 명의 사용자가 동시에 각자 서로 다른 1GB 파일을 접근 하는 경우

3.3 시험 결과

각 시나리오별 모든 사용자의 평균 전송 속도는 <표 1>과 같다.

<표 1> 시나리오별 평균 전송 속도(Mbps)

구 분	순차 읽기		랜덤 읽기	
	CCFS	EXT3	CCFS	EXT3
(1) 50 명/50MB	21.910	14.482	21.084	14.358
(2) 100 명/50MB	10.907	6.247	10.410	5.630
(3) 50 명/1GB	20.269	8.680	19.549	7.407

<표 1>을 기반으로 사용자 증가 및 파일 크기 증가에 따른 전송 속도 감소율을 얻을 수 있으며 그 결과는 <표 2>와 같다. <표 1>로부터 (1)과 (2)의 결과를 분석하면 사용자 증가에 따른 서비스 변화를 알 수 있으며, 50 명에서 100 명의 사용자로 증가하면서 CCFS 의 순차 읽기 평균 전송 속도는 21.910Mbps 에서 10.970MBps 로 감소하여 100 명일 때의 전송 속도가 50 명 사용자일 경우의 약 49% 속도이며, EXT3 의 경우 평균 14.482Mbps 에서 6.247Mbps 로 43%로 감소하였다. 또한, (1)과 (3)의 결과를 분석하면 파일 크기 증가에 따른 서비스 변화를 알 수 있으며, 50MB 파일에서 1GB 파일로 증가하면서 CCFS 의 순차 읽기 평균 전송 속도는 21.910Mbps 에서 20.269MBps 로 감소하여 1GB 파일의 경우 전송 속도가 50MB 일 경우의 92% 속도이며, EXT3 의 경우 평균 14.482Mbps 에서 8.680Mbps 로 약 59%로 감소하였다. 결과에 따르면 사용자가 2 배로 증가한 경우 CCFS 는 이론적인 50%에 가까우며, EXT3 의 경우 CCFS 에 비해 감소 비율이 크다. 또한, 파일의 크기가 증가하면 CCFS 의 경우 EXT3 에 비하여 파일 크기 증가에 따른 성능 감소가 현저히 작다는 것을 알 수 있다.

<표 2> 사용자 및 파일 증가에 따른 성능 비율(%)

구 분	순차 읽기		랜덤 읽기	
	CCFS	EXT3	CCFS	EXT3
사용자 증가	49.784	43.140	49.373	39.213
파일 크기 증가	92.509	59.939	92.722	51.589

4. 결론

본 논문에서는 차세대 인터넷 서버 운영 환경에서 NS 카드를 기반으로 개발된 CCFS 에 대한 소개와 성능 분석을 수행하였다. EXT3 와 비교 했을 때 사용자가 증가 할 수록 CCFS 의 성능 감소가 적었으며, 파일의 크기가 증가 할 때 더욱 성능의 차이를 보였다.

참고문헌

[1] 김명준, 임기욱, "차세대 인터넷 서버(SMART 서버) 기술 개발", 한국콘텐츠학회지, 제 1 권, 제 1 호, 2003.
[2] 김성운, 김명준, 김보관, "차세대 인터넷 서버를 위한 스트리밍 가속장치", 전자공학회 추계학술대회, 2003.
[3] 김영철 외, "멀티미디어 파일 시스템을 위한 회복 기법의 설계 및 구현", 한국정보과학회 추계학술대회, 제 30 권 제 2 호, 2003.