

# 객체기반 IP 스토리지 상의 멀티미디어 콘텐츠 재생 성능 향상을 고려한 시스템 설계

최민석\*, 전영준\*, 남영진\*  
\*대구대학교 컴퓨터·IT공학부  
e-mail:mschoi@daegu.ac.kr

## System Design for Improvement of Multimedia Contents Playback Performance on Object-based IP Storage

Minseok Choi\*, Young Joon Jeon\*, Young Jin Nam\*  
\*School of Computer & Information Tech., Daegu University

### 요 약

본 연구에서는 모바일 기기의 중요한 성능 요소 중 하나인 스토리지에 초점을 맞추고 멀티미디어 콘텐츠 재생 성능 향상을 위한 시스템을 제안한다. 이를 위해 모바일 기기에서 사용 가능한 여러 스토리지 기술 중 최신 기술인 객체기반 IP 스토리지 시스템을 기반으로 모바일 단말 측에 캐시 및 선반입(prefetch) 기능을 적용하여 네트워크를 사용하는 입출력으로 인한 재생 시스템의 성능저하를 보완한다.

### 1. 서론

오늘날 모바일 기기에서 영화와 음악, 그리고 동영상 강좌 등과 같은 멀티미디어 콘텐츠의 사용은 일상이 되었다. 그리고 멀티미디어 콘텐츠의 품질은 고사양화 되고 있고 대용량화 되고 있다. 이러한 멀티미디어 콘텐츠들을 저장하기 위해 모바일 기기용 스토리지의 중요성이 커지고 있다. 모바일 기기에 일반적으로 사용되는 스토리지는 미니 하드디스크, 플래시 메모리 등이 있다. 그러나 하드디스크는 전력소모가 많고 소형화가 어렵다. 또한 충격에 약하며 용량대비 가격이 비싸다. 반면 플래시 메모리는 상대적으로 크기가 작고 안정성이 좋고 적은 전력을 소모하지만 고용량 구현이 힘들고 용량대비 가격이 비싸다. 최근 연구되는 스토리로는 객체기반 IP 스토리지가 있다 [1]. 객체기반 IP 스토리지는 위의 두개의 스토리지가 가진 단점을 극복할 수 있는 가능성을 가졌다.

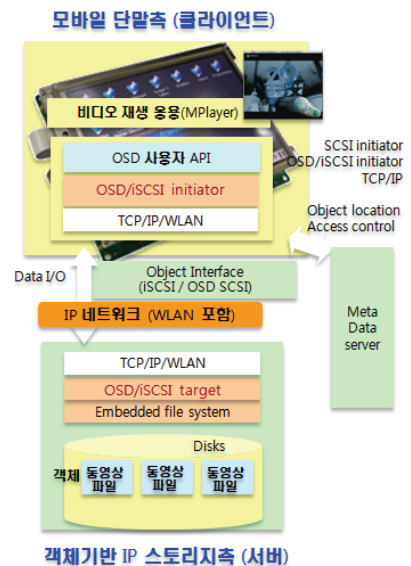
본 연구에서는 객체기반 IP 스토리지를 이용하여 모바일 기기에서 멀티미디어 콘텐츠 재생에 적합한 시스템 구조를 제안한다. 특히, 여러 멀티미디어 콘텐츠 중 동영상을 대상으로 하여 객체기반 IP 스토리지에 캐시를 적용하여 네트워크를 통한 입출력에 따른 성능저하를 보완한다.

### 2. 모바일 기기용 객체기반 IP 스토리지

객체기반 IP 스토리지는 IP 스토리지와 객체기반 스토리지[3,4]를 통합한 형태의 스토리지이다. 이 스토리지에서 데이터들은 다양한 크기와 속성을 가진 객체(Object)의 형태로 표현된다. 멀티미디어 콘텐츠, 데이터 베이스의 릴레이션 또는 필드 하나가 객체로 표현될 수 있다.

<그림 1>은 모바일 기기용 객체기반 IP 스토리지 구조를 나타낸 것이다. 모바일 단말 측은 OSD 이니시에이터

(Initiator)이며, 객체기반 IP 스토리지 측은 실제 스토리지 장치가 있는 곳으로서 OSD 타겟(Target)이다.



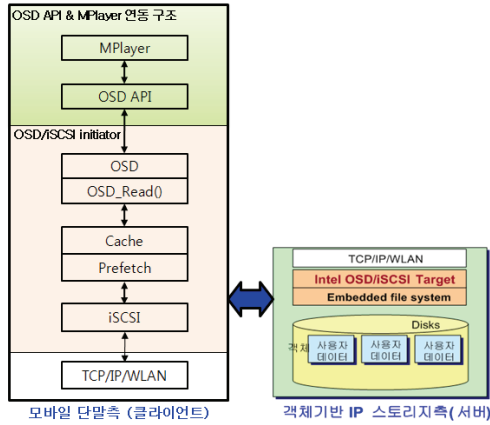
<그림 1> 모바일 기기용 객체기반 IP 스토리지

OSD 이니시에이터(Initiator)와 OSD 타겟(Target) 사이의 통신을 위해서는 iSCSI 프로토콜이 사용된다[5,6].

OSD 타겟(Target)에 저장된 데이터를 사용하기 위해서는 OSD 이니시에이터(Initiator)는 CDB라고 하는 OSD 명령어 구조 명령을 만들고 이 CDB는 iSCSI를 통해 OSD 타겟(Target)으로 보내져서 처리된다.

### 3. 멀티미디어 콘텐츠 재생 시스템 설계

본 연구에서는 객체기반 IP 스토리지를 이용하여 모바일 기기에서 멀티미디어 콘텐츠 재생에 적합한 시스템 구조를 제안한다. MPlayer에는 OSD API가 연동되어 있고 OSD Initiator 레이어와 iSCSI 사이에는 캐시(cache)가 존재하여 입출력의 성능저하를 보완한다. <그림 2>는 동영상 재생 시스템의 전체 구조이다.

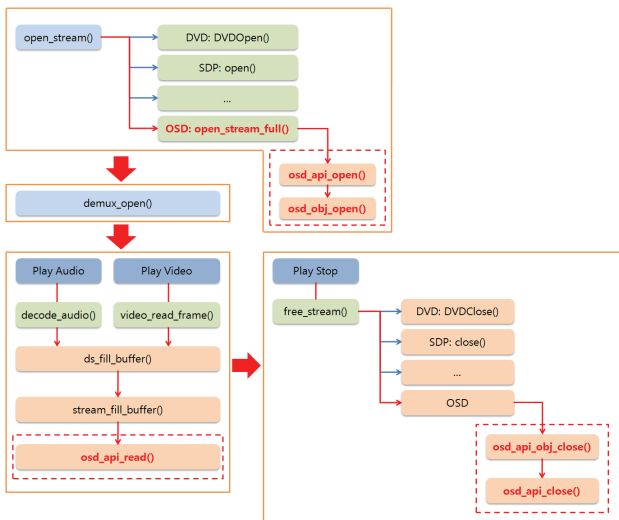


<그림 2> 동영상 재생 시스템 전체 구조

#### 3.1 멀티미디어 콘텐츠 재생 구조

본 연구에서 멀티미디어 콘텐츠 재생을 위해 사용하는 프로그램은 리눅스에서 동영상 재생 시 많이 사용되는 MPlayer이다. MPlayer는 객체 I/O를 위해 OSD API와 연동이 되어야 한다. <그림 3>는 OSD API와 MPlayer 간의 연동 구조이다.

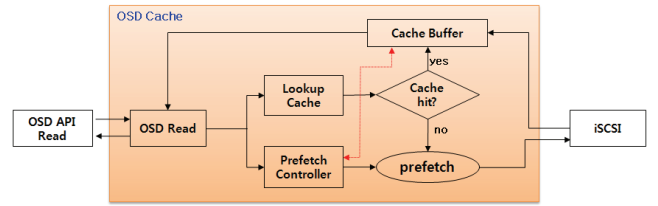
우선, 스트림을 오픈할 때 `osd_api_open()`를 호출하여 OSD 장치를 오픈하고 `osd_api_obj_open()`를 호출하여 읽어올 객체의 ID 정보 및 속성 정보를 읽어온다. 그 읽어온 정보를 이용하여 `osd_api_read()`를 반복적으로 호출하여 객체 내의 데이터를 읽어와서 MPlayer의 버퍼를 지속적으로 채운다. 재생 종료 후에는 `osd_api_obj_close()`와 `osd_api_close()`를 호출하여 객체와 장치를 닫는다.



<그림 3> OSD API & MPlayer 연동 구조

#### 3.2 OSD 캐시(cache)

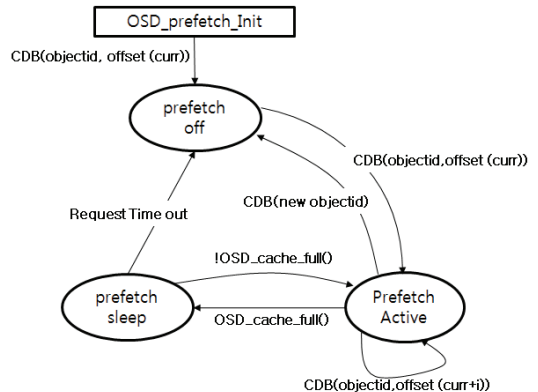
본 연구에서는 멀티미디어 콘텐츠의 재생에 대한 효율성을 높이기 위해 OSD에 캐시(cache)를 적용한다. 기본적인 OSD의 입출력 구조는 동기I/O로서 하나의 읽기(read) 명령이 종료되어야 다음 읽기(read) 명령을 보낼 수 있다. 따라서 네트워크를 사용하는 객체기반 IP 스토리지의 경우 동영상 재생 시 성능이 현저히 낮아진다. 이에 현재 클라이언트의 OSD 레이어에 캐시(cache) 및 선반입(prefetch)기능을 추가함으로써 동영상 재생 시 성능을 향상시킬 수 있다. 캐시 및 선반입 기능을 가진 OSD의 기본 구조는 <그림 4>와 같다.



<그림 4> OSD 캐시 기본 구조

OSD API에서 데이터의 읽기(read) 요청이 시작되면 OSD 이니시에이터(Initiator)는 읽기(read) CDB를 생성하고, 이 CDB를 통해 캐시에 선반입(prefetch)된 데이터를 읽어온다. 이 구조의 세부 작동은 다음과 같다. 읽기 요청이 시작되면 캐시를 검사함과 동시에 Prefetch Controller에 읽기 요청에 따른 CDB 정보를 전달한다. 캐시 히트(cache hit)가 발생하면 CDB에 해당하는 데이터를 캐시 버퍼에서 읽어와 반환한다. 캐시 미스(cache miss)가 발생하면 CDB를 iSCSI 레이어로 보내 데이터를 요청하고, 캐시 미스 발생 정보는 Prefetch Controller에 전달된다.

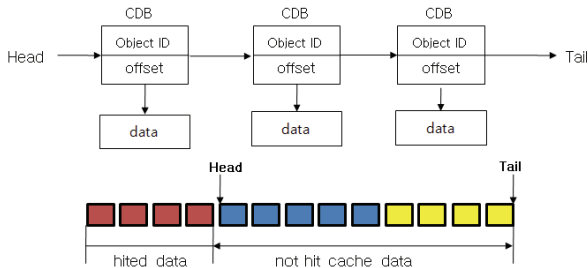
Prefetch Controller는 선반입 상태와 캐시의 상태를 확인하여 선반입의 수행여부를 결정한다. 선반입이 수행되지 않고 있을 때에는 선반입을 시키고, 선반입 중인 경우, 객체 ID를 확인하여 동일한 객체 ID의 데이터가 선반입 중일 때는 현재의 오프셋(offset) 다음부터 선반입을 다시 수행하게 한다. 만약 객체 ID가 다른 데이터의 요청이 오면 기존의 선반입 상태를 초기화 하여 새롭게 선반입을 수행하도록 한다. 선반입이 대기(sleep) 상태인 경우에는 캐시의 버퍼가 가득 찬 경우로서 이때는 캐시의 버퍼를 체크하여 버퍼가 비워지면 다시 선반입을 수행한다. 만약 선반입이 대기 상태에서 일정시간동안 지속되면 선반입은 off상태로 된다. <그림 5>은 선반입의 상태를 나타낸 것이다.



<그림 5> 선반입 상태도

참고문헌

캐시의 버퍼에 저장되는 데이터의 구조는 <그림 6>과 같이 리스트 구조로 구성된다. 그리고 캐시의 버퍼는 먼저 들어온 데이터부터 차례로 인출된다. 이미 히트 된 데이터는 바로 캐시에서 삭제 하지 않고 일정량을 캐시버퍼에 보관하여 지나간 부분을 재생하는 것을 가능 하도록 하였다.



<그림 6>캐시의 데이터 구조

[1] Y. Nam, "Prototyping object-based ubiquitous multimedia contents storage for mobile device," Lecture Notes in Computer Science (UIC2006), Springer-Verlag, Sept. 2006.

[2] Erik Riedel(Seagate Research), Object-based storage device(OSD) basics: <http://www.snia.org/education/tutorials/spr2005/storage/>, 2005.

[3] OSD Standard version 1.0 (rev.10): [www.t10.org/ftp/t10/drafts/osd](http://www.t10.org/ftp/t10/drafts/osd).

[4] J. Hufferd, iSCSI: The Universal Storage Connection. Addison-Wesley, 2003.

[5] K. Meth and J. Satran, "Design of the iSCSI protocol," Proc. of the Mass Storage Systems & Technologies/20th IEEE/11th NASA Goddard Conference, April 2003.

4. 결론 및 향후계획

본 연구에서는 객체기반 IP 스토리지는 차세대 스토리지 기술을 이용하여 최근 증가한 모바일 환경에서의 멀티미디어 콘텐츠 재생에 효율적인 시스템 구조를 설계하였다. 객체기반 IP 스토리지의 멀티미디어 콘텐츠 재생을 위해서 OSD 이니시에이터 레이어와 iSCSI 사이에 캐시를 두어 입출력 성능을 향상시켰으며 차후 일반적인 환경에서의 객체기반 IP 스토리지 성능 향상을 위한 기법 개발의 기반이 된다.

<표 1> 구현 환경

항목	세부사양	
CPU	Intel Bulverde PXA270(520MHz)	
메모리	SDRAM	64MB
	NOR Flash	64MB
	NAND Flash	64MB
디스플레이	6.4" TFT LCD	
Network	PCMCIA Wireless LAN(AeroNet 350)	
OS	Embedded Linux 2.6.11	

향후 진행 사항은 설계된 내용의 구현이다. <표 1>은 구현에 사용되는 환경이다. 하드웨어는 일반적인 모바일 기기의 환경과 유사하며 소프트웨어 플랫폼은 리눅스를 사용한다. 그리고 네트워크는 모바일 기기와 동일한 환경을 갖추기 위해 무선랜을 사용한다. 다음으로 동영상 테스트를 진행 할 것이다. 각 샘플은 mpeg2, mpeg4, xvid, h.264 등 다양한 종류의 샘플을 사용하여 테스트 할 것이다. 그리고 구현된 시스템을 기반으로 모바일 기기의 최대 관심 사항인 저전력 기법에 대한 연구를 진행할 것이다. 특히, 저전력 대상으로 Wireless LAN에 초점을 맞춰 진행할 계획이다.