

# 계층적 저장장치에 기반한 대용량 비디오 서버

오재학                      차호정  
광운대학교 컴퓨터학과  
{ojh, hojunc}@cs.kwangwoon.ac.kr

## A Large Scale Video Server Based On Tertiary Storage

Jaehak Oh                      Hojung Cha  
Dept. of Computer Science, Kwangwoon University

### 요 약

본 논문은 Tertiary 저장장치에 기반한 대용량 비디오 서버를 구현함에 있어 사용자 QoS의 보장과 효율적인 자원관리를 위한 계층적 저장장치의 구성, 계층적 마감시간 스케줄링을 제안하고 설계한다. 저장장치의 효율적인 사용을 위해 디스크 캐쉬 모델과 연속 미디어의 선호도에 따른 디스크 캐쉬 풀의 구성을 통하여 Tertiary 저장장치를 스트리밍 시스템에 활용할 수 있는 모델을 제시한다.

### 1. 서론

최근 네트워크 인프라의 향상과 다양한 사용자 서비스 요구로 주문형 비디오 서비스는 상용 서비스로 발전하고 있으며 시스템 구성에 있어 대형화 추세에 있다. 주문형 비디오 서비스는 사용자의 수요증가와 비디오 타이틀의 크기 및 수량 면에서 대용량의 저장장치의 사용이 필수적이다. 그러나 기존에 디스크 기반의 비디오 서버의 문제는 고품질의 서비스와 안전성을 보장하는 반면에 저장장치 구성에 있어 비싼 비용으로 대형 비디오 서버의 구성에 어려움이 있다[1].

저장장치 구성의 문제는 디스크 기반의 비디오 서버의 연구 성과와 MO Jukebox 혹은 Tape Library와 같은 Tertiary 저장장치를 통해 해결할 수 있으며 이러한 기술의 접목에는 몇 가지 문제점을 고려해야 한다. 첫째, 저장장치의 성능과 용량에 따른 계층적 저장시스템의 구축과 연속 미디어 제한조건을 만족하는 저장장치 사이에 데이터 전송기술을 개발해야한다. 즉 Tertiary 저장장치에 대한 디스크 캐쉬를 구성함으로써 고성능 디스크 저장장치를 효율적으로 사용할 수 있어야 한다. 둘째, 효율적인 디스크 캐쉬의 관리이다. 선호도에 따른 캐싱 블록의 라이프 사이클을 연장하기보다는 선호도에 비해 캐쉬량을 조절함으로써 Tertiary 저장장치의 저 내역폭의 단점을 보완해야 한다. 셋째, 대용량 비디오 서버는 연속 미디어에 따른 사용자 QoS를 보장하기 위한 프로세서 스케줄링, 디스크 입출력 스케줄링, Tertiary 입출력 스케줄링이 필요하다. [2]는 계층적 저장장치에 기반한 VOD 서버의 구성에 있어 유연한 수용제어와 효율적인 디스크 캐쉬 플레이스먼트 정책을 제안하였다. [3]은 멀티 미디어 환경에서 비디오 서버의 데이터 저장방식을 제안하였다. 시스템의 구성은 태입 기반의 저장장치, 디스크 캐쉬 풀, 입출력 제어기를 고속 네트워크 망으로 연결한 구조이다.

• 본 연구는 한국과학재단의 특정기초연구사업(97-01-00-12-01-5)에 의해 지원받았음.

기존 연구의 문제점은 지역적인 정책에 따라 실시간 시스템의 주요요건을 만족하지 못하는 상황이 발생할 수 있으며 자원 관리의 비효율성을 초래할 수 있다. 특히 디스크캐쉬의 설정이 운영체제의 캐쉬나 파일 시스템의 버퍼 캐쉬 정도로 가정되어 있어 사용자 관점에 연속 미디어의 조건을 잘 반영하지 못하고 있다.

본 논문에서는 Tertiary 저장장치를 기반한 대용량 비디오 서버의 구현에 있어 사용자 QoS의 보장과 효율적인 자원관리를 위한 계층적 저장장치의 구성, 계층적 마감시간 스케줄링을 제안하고 설계하였다. 논문의 구성은 2장에서 전체구성과 3장에서 계층적 마감시간 스케줄링 알고리즘을 제안하고 성능을 분석한다. 4장에서 결론과 향후 연구과제를 기술한다.

### 2. 시스템 모델

그림 1은 Tertiray 저장장치를 이용한 대용량 비디오 서버를 보여준다. 시스템은 프로세서 스케줄러, 수용제어기, 디스크 입출력 스케줄러, Tertiary 입출력 스케줄러, 네트워크 관리자, 디스크 입출력 제어기, Tertiary 입출력 제어기로 나누어진다. 하드웨어 특성으로 주 메모리의 버퍼, Hot, Partial, Cold 디스크 풀, Tertiary 저장장치 그리고 범용 운영체제를 통해 제어되는 워크 스테이션으로 구성되어 있다.

사용자 요구는 수용제어기를 통해 허용여부를 결정한다. 수용제어기는 자원표와 비디오 블록표, 미디어 블록 캐쉬표를 통해 수용여부를 결정한다. 자원표는 현재의 자원량을 기록하고 유동적으로 변하는 자원상황에 따라 갱신되고 비디오 블록표는 Tertiary 저장장치에 등록되어 있는 비디오 블록을 기록하고 연속 미디어의 요구율, 크기, 상영시간 등을 기록한다. 비디오 블록 캐쉬표는 각각의 디스크 풀에 캐쉬된 블록을 기록하고 캐쉬량 등을 기록한다. 프로세서 스케줄러는 쓰레드 표를 통해 디스크 입출력 스케줄러와 Tertiary 입출력 스케줄러, 네트워크 관리자를 주기적으로 관리한다. 디스크 입출력 및 Tertiary 입출

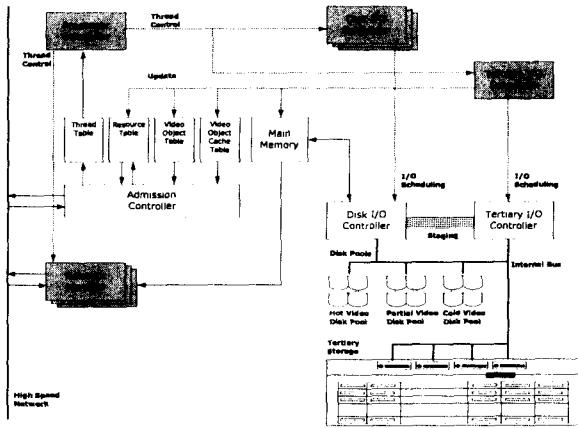


그림 1 시스템 구성도

력 스케줄러는 비디오 서버 초기화 때에 등록되며, 네트워크 관리자는 수용제어기를 통해 등록된다. 쓰레드 표에 기록된 네트워크 관리자는 클라이언트의 요구율에 따라 위급도를 동적으로 정하고 프로세서 스케줄러의 주기에 배정된다. 디스크 입출력 스케줄러는 네트워크 관리자의 요구를 큐에 받아들이고 비디오 블럭의 위치에 따른 Hot, Partial, Cold 비디오의 특징으로 각각의 독립된 스케줄러에 의해서 입출력 요구를 수행한다. Hot 비디오가 아닌 Partial 비디오나 Cold 비디오의 요구는 Tertiary 저장장치에 저장된 미디어를 디스크 풀로 캐스팅함으로써 마치 비디오 객체 전체가 디스크 풀에 위치한 것과 같은 효과를 얻는다. Tertiary 저장장치의 캐스팅은 클라이언트의 요구율과 디스크 캐쉬 풀의 캐쉬량을 통해 결정되고, 네트워크 관리자의 요구에 따른 디스크 입출력의 풀(Pull) 모델과는 다르게 캐스팅에 따라 디스크 캐쉬에 푸시(Push)하는 방식이다. 이러한 입출력 요구는 디스크 입출력 제어기와 Tertiary 입출력 제어기를 통해 수행된다.

### 3 계층적 마감시간 스케줄링

Tertiary 저장장치 기반에 비디오 서버의 스케줄링 모델은 프로세서 스케줄링, 디스크 입출력 스케줄링, Tertiary 입출력 스케줄링으로 구분한다. 각각의 계층의 마감시간은 연속미디어의 제한조건을 만족해야하고 상호연관구조로 이루어져 있다. 이러한 스케줄링구조를 계층적 마감시간 스케줄링이라 한다. 그림 2(a)는 스케줄러의 계층구조를 도식이며 연관관계를 나타내고 있다. 그림 2(b)는 디스크 입출력과 Tertiary 입출력의 관계를 나타내며, 프로세서 스케줄링 주기를 기준으로 디스크 입출력과 Tertiary 입출력을 구분한다.

프로세서 스케줄러의 실행 주기인  $t_p$ 와 각 클라이언트의 데이터 요구 주기  $t_i$ 가 존재한다.  $t_i$ 는 클라이언트의 데이터율 ( $r_i$ )에 따라 결정되며 데이터율이 높은 클라이언트는 데이터율

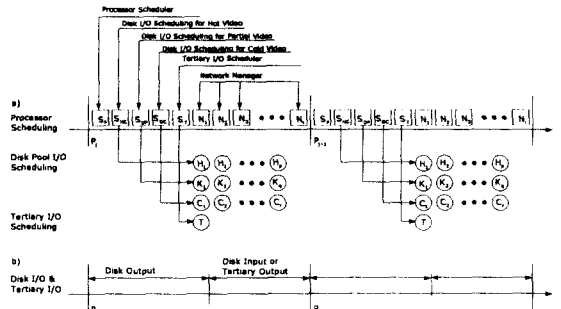


그림 2 계층적 마감시간 스케줄링의 개념

$$t_p \leq \text{MIN}(t_1, t_2, \dots, t_n) \quad (\text{식 1})$$

$$\text{MIN}(b_{HD}, b_{PD}, b_{CD}) < b_N: t_p = k \times t_{disk} \quad (\text{식 2})$$

$$\text{MIN}(b_{HD}, b_{PD}, b_{CD}) > b_N: t_p = k \times t_{net} \quad (\text{식 3})$$

이 낮은 클라이언트에 비해 짧은 주기를 갖는다.  $t_p$ 는 식 1과 식 2, 식 3에 의해 결정된다. 식 1은  $t_p$  주기 내에서  $t_i$ 가 한번의 요구 주기를 갖음을 의미하며 각각  $t_i$ 는 데드라인에 따라 우선순위를 정하게 된다.  $t_p$ 내에 스케줄링된 각각의 요구는 다른 어떤 요구에 의해서 선점되지 않으며 자신의 수행이 끝났을 때만 다음 요구를 수행한다. 프로세서 스케줄러의 주기  $t_p$ 에서 실행할 수 있는 요구의 갯수는 식 2와 식 3에 의해 결정된다. 디스크 풀 입출력 대역폭이 작은 것과 네트워크 대역폭 간에 비교를 통해 작은 대역폭을 기준으로 수용 스트림에 수를 정함으로써 자원량 초과를 방지한다.

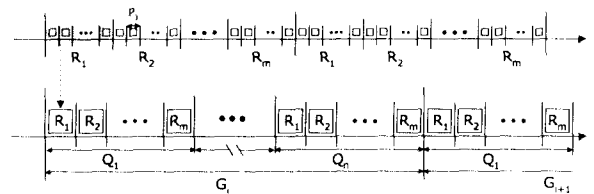


그림 3 주기 모델

그림 3은 주기 모델로 프로세서 주기에서부터 Tertiary 주기까지 계층적 관계를 나타내고 있다.  $P_i$  프로세서 주기  $t_p$ 내에 작업이며  $R_m$ 은 캐싱 주기 내에  $m$ 번째 작업으로 지역주기  $t_S$  동안 연속 미디어를 디스크 풀로 전송한다. 그리고 전역주기  $G_i$ 는 각각의 캐싱 스트림의  $t_S$ 를 만족하는 동기화의 단위작업이다. 캐싱 스트림  $S_i$ 의 주기는 지역주기  $t_S$ 와 전역주기  $t_C$ 로 구분한다. 지역주기  $t_S$ 는  $S_i$ 의 주기를 의미하고 전역주기  $t_C$ 는  $S_i$ 들의 공통 주기로 Tertiary 저장장치의 대역폭 및 새로운 스

캐시의 적용단위이다. 계층적 마감시간 스케줄링에서 캐싱 스케줄링은 Tertiary 저장장치에서 디스크 캐쉬 풀로 효율적인 데이터 전송 정책이고 Tertiary 저장장치의 기본 구조에 따라 대용량 저장공간, 저 대역폭의 입출력과 디스크 교체시간을 고려한 주기모형을 형성한다. 주기모형은 사용자 선호도와 디스크 캐쉬의 용량, 캐싱 블록 크기를 통해 일반화할 수 있다.

```
Cache_Scheduling ()
{
    /* 새로운  $G_k$ 의 시작, 새 스트림 허용, 스트림 종료 */
    if (change캐싱Rate(  $S_i$  ) || NEW(  $S_i$  ) || END(  $S_i$  ))
        /* Deadline Driven Scheduling */
        if (init)
            Initialize_Caching_Scheduling_Queue ()
            Set_Priority(  $\forall_i : S_i$  )

    /* Apply saving exchange time policy*/
    Global_Period_Scheduling ()
    Reconstruct_Cache_Queue_Scheduling ()

    Estimate_Free_Bandwidth_In_Tertiary_입출력 ()
    Update_Resource_Table ()
    Update_Cache_Table ()

    /*  $Q_i$  스케줄에 대한 Replacement 정책적용 */
    if (IsFull (Partial Disk Pool))
        Partial_Video_Replacement ()
    if (IsFull (Cold Video Pool))
        Cold_Video_Replacement ()

    Process_Caching ()
}
```

그림 4 캐싱 입출력 스케줄링 알고리즘

그림 4은 캐싱 입출력 스케줄링 알고리즘이다. 본 알고리즘에서는 Tertiary 저장장치의 미디어 교체를 줄이고 그 여유시간을 활용방안을 모색한다. 알고리즘은 우선 순위 정책, 비선점 스케줄링, 전역 스케줄링, 지역 스케줄링, 자원관리 정책으로 분류한다. 알고리즘을 순차적으로 기술하면, 4줄은 전역 스케줄링의 시점으로 새로운 캐싱 스트림의 수용과 기존 스트림의 종료, 그리고 스케줄링 정책에 따른 디스크 캐쉬의 캐쉬량의 증가로 캐싱 주기율이 감소했을 때 즉 Tertiary 입출력 캐싱 량의 변화가 있을 때 전역 스케줄링을 수행한다. 6~8줄의 초기화 단계를 거쳐 9줄에서 비선점 EDF 스케줄링을 위한 캐싱 스트림의 우선순위를 정한다. 11줄의 전역 주기 스케줄링은 지역 주기로 스케줄된 캐싱 스트림에 대해서 각각의 지역주기의 대역폭 여유량을 확보하여 전역주기에서 고역폭의 새 스트림을 최대한 수용한다. 14줄에 Tertiary 입출력의 대역폭은 비선점 EDF 스케줄링에서 지역주기와 데이터량을 등록하고 전역 스케줄링에서 얻어지는 데이터량을 통해 대역폭을 평가한다. 그림 5은 시뮬레이션에 의한 캐싱 입출력 스케줄링 효과를 보이고 있

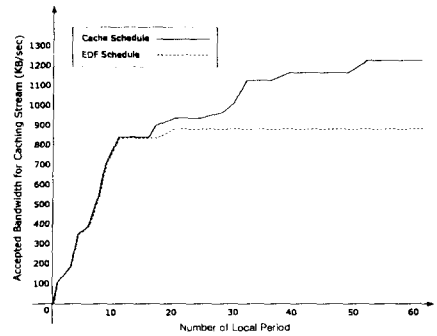


그림 5 캐싱 스트림 수용에 따른 대역폭 변화

다. 스트림 수의 증가에 따른 Tertiary 저장장치의 활용 대역폭은 스케줄링과 여유 대역폭의 평가로 일반적인 EDF를 적용했을 때보다 높은 대역폭 활용도를 보인다.

#### 4. 결론

본 연구에서는 연속 미디어 서비스를 위한 Tertiary 저장장치 기반의 대용량 비디오 서버를 설계 및 제안하였다. 저장장치의 현실성 있는 구성과 효율적인 사용을 위해 디스크 캐쉬 모델과 연속 미디어의 선호도에 따른 디스크 캐쉬 풀의 구성을 통해 Tertiary 저장장치를 스트리밍 시스템에 활용할 수 있는 이론적인 모델을 제시하였다. 계층적 데드라인 스케줄링은 프로세서 스케줄링, 디스크 입출력 스케줄링, Tertiary 입출력 스케줄링으로 구성하여 저장장치 입출력 지연시간에 따른 연속 미디어의 제한조건을 만족하였다.

향후연구과제는 몇 가지 점에서 확장을 고려해야 한다. 첫째, 디스크 캐쉬의 성능을 향상시키기 위한 효율적인 교체 정책과 동적인 자원량, 디스크 캐쉬량, 스케줄링 예측으로 유연한 수용 제어 정책을 개발해야 한다. 둘째 Tertiary 저장장치의 다중 드라이브 모델과 Tertiary 저장장치 풀 모델을 통한 효율적인 입출력 관리정책과 스케줄링 정책이 필요하다. 마지막으로 진정한 비디오 서비스를 위한 VCR 기술의 개발이 요구된다.

#### 참고문헌

- [1] Shueng-Han Gary Chan, Fouad A. Tobagi, 'Hierarchical Storage Systems for On-Demand Video Servers', *Proc of the 1995 SPIE High-Density Data Recording and Retrieval Technologies*, October 23-24 1995, Philadelphia, Vol.2604 pp.103-120.
- [2] Kian-Lee Tan, Beng Chin Ooi, Tat-Seng Chua, 'Or Video-On-Demand Servers with Hierarchical Storage', *Proc of the 5th Conf on Database Systems for Advance Applications*, 1997, Melbourne, pp.491-500.
- [3] Jihad Boulos, Kinji Ono, 'VOD Data Storage in Multimedia Environments', *IEICE TRANS. COMMUN* August 1998, Vol. E81-B No.8, pp.1656-1665.