

Clustered Storage Server 환경에서 뉴스 데이터에 적합한 분산 저장방법

정귀옥, 박성호, 김영주, 정기동
부산대학교 전자계산학과

Efficient striping policy of NOD data on clustered storage server

*Kwy-Ok Jung, Seong-Ho Park, Young-Ju Kim, Ki-Dong Chung
Department of Computer Science, Pusan National University

요 약

현대 사회의 정보 요구 증가와 편리함의 추구는 정보통신 기술의 발달과 함께 멀티미디어 데이터 서비스를 급증 시켰다. NOD 데이터의 경우 이러한 요구에 부합하므로, 많은 사용자를 가지게 될 것이며, 그에 따르는 제반 요건으로 서버 구현에서 scalability, availability, reliability 등이 중요한 요건이다. 따라서 이러한 요건을 멀티미디어 데이터 특성을 이용한 저장 방법으로 만족 시키려는 많은 연구가 있다. 그러나, NOD 시스템에 대한 연구는 미흡한 실정이며, clustered 환경에서의 News 데이터에 대한 연구는 거의 없다. VOD 데이터에 적합한 것으로 알려진 일반적인 저장 방법이 NOD 데이터에 반드시 적합한 것이 아니며, 본 논문에서는 기존에 연구된 데이터 저장 방법 중에서 NOD 데이터의 small volume, skewed popularity distribution 등의 특성을 고려하여, clustered storage server 환경에 맞는 striping 정책을 찾는다.

1. 서론

정보화 사회로 접어들면서 일반 사용자의 정보에 대한 욕구 증가는 컴퓨터 및 그에 기반한 기술의 발달과 함께 정보통신 분야를 활성화 시키게 되었다. 뿐만 아니라, 사용자들은 좀 더 편리한 인터페이스를 요구함에 따라 문자 위주의 정보에서 멀티미디어를 이용하게 되었고, 이에 상응 하는 여러 서비스가 제공 되고 있다. 이처럼 정보와 편리함의 두 가지를 모두 갖춘 대표적인 예로 NOD(News On Demand) 서비스를 들 수 있다. 그러므로 현재 NOD 서버 설계에 관한 많은 연구가 이루어지고 있다. 특히, NOD 서비스의 경우, 멀티미디어 데이터를 다룬다는 점에서 VOD 등의 서비스와 유사한 특성을 가지는 것으로 여겨져 왔다. 그러나, NOD 서비스는 다음과 같은 면에서 다른 특성을 가진다. 하나의 뉴스 비디오 클립의 길이가 짧아 상대적으로 적은 용량이지만, 하루에 생성되는 데이터의 수는 많고, 각 데이터의 생명 주기가 짧다는 특성은 NOD 서버를 구현하는데 많은 영향을 준다.

멀티미디어 서버를 구현하는 방법에는 여러 가지가 있지만, 현재 제반 사실을 이용하여 구현하고자 할 경우, 전체 구성 요소 중에서 저장 서버를 구현하는 비용이 가장 많이 든다. 이를 극복하기 위한 대용량 데이터의 여러 저장 방법이 제시되어왔지만, 대부분의 정책이 VOD 즉 비디오 데이터에 대한 방법이 주류를 이루며, NOD 데이터에 대한 연구는 드물다.

본 논문에서는 clustered 환경에서 NOD 시스템을 구현하고자 할 경우 어떠한 저장 정책을 사용하는 것이 가장 효율적인가를 알아보기

위해, 기존의 여러 저장 방법 중에서 clustered 환경에 적용하여 효율적인 방법을 찾고자 한다.

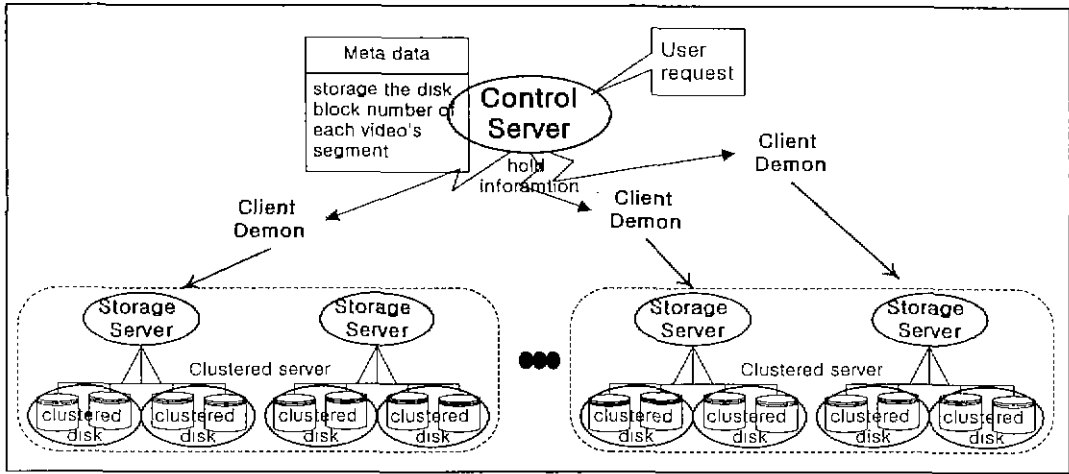
본 논문의 구성은 다음과 같다. 2 장에서는 관련 연구를 살펴 보도록 하고, 3 장에서는 서버 구조 및 배치 방법을 제시한다, 4 장에서는 이를 증명하는 시뮬레이션과 결과를 살펴보고, 마지막 5 장에서 결론 및 향후 과제에 대해 기술하고자 한다.

2. 관련 연구

본 논문에서는 인기도뿐 가지면서 실시간으로 처리되어야 하는 대용량 데이터의 처리를 위해서 이루어지는 많은 연구 중, 두 가지 분야에 관해 알아본다. 하나는 인기도와 실시간성을 고려한 서버의 구조에 관한 연구이고, 또 다른 하나는 대용량성과 실시간성을 고려한 데이터 배치 방법에 관한 연구이다.

서버 구조에 관한 연구로는 하나의 서버 환경에서 벗어나, 분산환경에 관한 연구가 많이 이루어 지고 있으며, 특히 계층구조에 많은 연구[5]가 이루어지고 있다. 또 계층구조의 변형으로 중복 계층구조[2]가 있으며, 여러 서버를 묶어서 사용하는 clustered 서버 구조[6]가 있다.

데이터 배치 방법에 관한 연구로는 분산 배치 방법에 복사본, 인기도 등을 함께 고려한 방법들이 많이 사용되고 있다. 기본적인 분산 배치 방법으로 round robin 방법[7]이 있으며, phased-based 분산 배치 방법[4], NOD 데이터용 위한 random 배치 방법[3] 등이 있다. 그리고, 분산환경에서 비디오 데이터를 clustered overlap 배치를 사용하



[그림 1] 시스템 구조 4

는 연구[6], 전체 저장하는 방법 평가에 관한 연구[1] 등이 있다. 그러나, clustered NOD server 환경에서의 적용에 관한 연구는 이루어지지 못한 실정이다 따라서, 본 논문에서는 기존에 나와있는 여러 방법들을 조합하여, clustered server 환경에서 가장 적합한 배치 방법을 찾는다

3. 시스템 구조 와 배치방법

3.1 시스템 구조

현재 정보 요구자 수의 증가 추세를 보면, web 서버의 경우 매우 많은 사용자 접근을 보이고 있다[8]. NOD 서비스의 경우도 비슷한 증가 추세를 보여 사용자 수가 급증 할 것임을 알 수 있다. 이러한 현실에서는 scalability, reliability 가 중요하게 되고, 따라서 clustered 구조를 이용할 경우 많은 장점을 가진다. 본 논문에서는 설정하는 clustered NOD 저장서버의 구조는 [그림 1]과 같다

3.2 배치방법

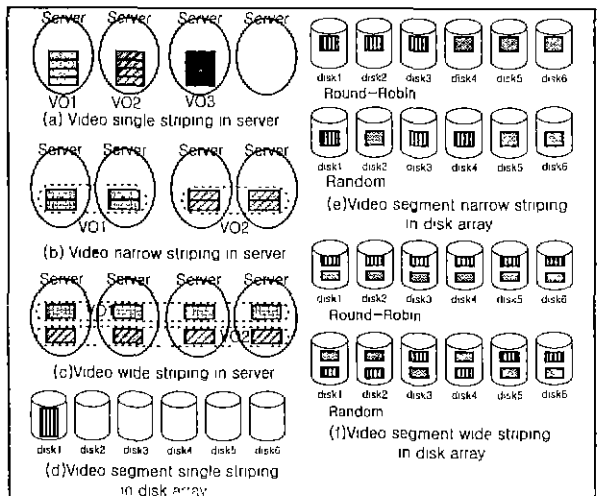
NOD data의 특성에 맞는 striping 정책을 알아보기 위해, 기존에 나와있는 여러 striping 정책을 조합하여 적당한 배치 기법을 찾는다. 기존의 striping 정책은 다음과 같다. 먼저 배치 알고리즘 기준에 따라 random 과 round-robin으로 나누어지며, 데이터의 분산 정도에 따라 single striping, narrow striping, wide striping으로 나누어진다.

Clustered 환경에서 위의 정책을 조합하여 보면 [표 1]과 같이 25 가지의 경우를 만들 수 있다. 그러나, cluster group size의 크기 중 전체 서버로 하여 wide striping시키는 경우, NOD 데이터는 volume 작아 striping unit이 너무 작아지게 되어 디스크 효율을 저하시키고, 또 scalable한 서버의 변화에 따른 디스크 재배치가 요구되므로 재배치 overhead가 크다. 하나의 서버에서 하나의 디스크에 저장하는 경우, 인기있는 데이터를 저장한 서버가 hot spot이 되어 서버 이용률이 저하되고, 결과적으로 전체 성능을 높일 수 없다 Clustered 서버 전체에 random 배치 할 경우, 저장 데이터 정보인 metadata가 필요하고, 서비스를 위해서 network 통신 overhead를 수반한다 결과적으로, NOD 시스템 구현에 실현 가능성이 없는 배치 방법을 제외하면, [표 1]에서 보여지듯이 9가지 경우로 실용 가능성을 평가 할 것이다

[표 1] 적용 가능한 배치방법

Video obj striping	Server group	Segment striping				
		One disk	disk group		All disks	
			Round robin	Random	Round robin	Random
One server	One server	*	o	o	o	o
	Round robin	o	o	o	o	o
	Random	*	*	*	*	*
	All servers	*	*	*	*	*

[그림 2]는 각 저장 방식에 대한 데이터 배치 형식을 그림으로 보여준다 (a)-(e)는 각각 server level 에서 video object를 segment 단위로 배치하는 방법들, (d)-(f)는 disk level 에서 video object의 segment를 다시 striping unit으로 배치하는 방법이다.



[그림 2] 데이터 배치도

3.3 서버 구성

3.2에서 제시된 데이터 배치방법을 clustered 서버 환경에서 적용할 때는 먼저, 디스크 수, 서버 수, clustered group size 등의 결정이 필요하다. 이는 [표 2]의 변수를 이용하여 계산하면 하연 아래와 같다

먼저 디스크 수를 결정하기 위해서는 서버 환경이 bandwidth-bound 인지, capacity-bound 인지 결정이 필요하며, 이는 사용자 요구 Bandwidth/Capacity 와 디스크 Bandwidth/Capacity 로 결정한다.

사용자 요구와 디스크 서비스 비율을 계산하는 [식 1]로

$$U_{ser_cluster} = \frac{B_s}{C_{disk}}, \quad Disk_cluster = \frac{B_{disk}}{C_{disk}}$$

$$\frac{U_{ser_cluster}}{Disk_cluster} = \frac{B_s \times C_{disk}}{B_{disk} \times C_{server}} \quad [식 1]$$

동시 사용자 수와 전체 News 데이터의 수와의 상관계를 계산한다.

$$U_t \times \frac{U_{ser_cluster}}{Disk_cluster} \geq V_{all} \quad [식 2]$$

[식 2]를 만족시키는 경우 이 환경은 bandwidth-bound 를 가지게 된다 NOD 데이터의 경우, 상대적으로 작은 volume size 를 가지고, 많은 동시사용자를 가짐으로 bandwidth-bound 가 된다.

요구되어 지는 disk 의 개수는 bandwidth-bound 가 되는 경우는 [식 3.a]가, capacity-bound 의 경우는 [식 3.b]가 된다.

$$N_{Min_disk} = \frac{U_t \times B_s}{B_{disk}} \quad [식 3.a] \quad N_{Max_disk} = \frac{V_{all} \times C_{disk}}{C_{server}} \quad [식 3.b]$$

요구되어지는 clustered server group size 는 비디오 물림의 평균 인기도를 이용하여, 각 비디오에 대한 평균 사용자를 계산하여 구한다 [식 4]로 나타낼 수 있다.

$$S_{server_cluster} = \frac{U_t \times P \times B_s}{B_{disk} \times N_{disk}} \quad [식 4]$$

4. 시뮬레이션

4.1 시뮬레이션 환경

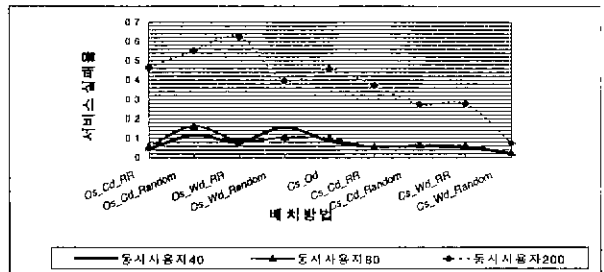
Sun Ultra 1 환경에서 News Video 개수를 200 개로 하여, 동시 사용자수에 따른 서비스 실패율을 비교하였다. 인기도 분포는 부산일보 인터넷 신문의 log-file 을 이용하였으며, video clip 크기는 120MB, 요구 대역폭은 1Mbps, 디스크 대역폭은 80Mbps 이다.

[표 2] 서버 평가를 위한 변수

Values	Parameter
U _M	전체 사용자수
U _t	최대 동시 사용자수
B _s	사용자 요구 대역폭
V _M	총 News video clip 의 수
C _{disk}	각 news video clip 의 용량
N _{server}	Server 의 개수
N _{disk}	전체 disk 의 개수
N _{disk}	각 server 의 disk 개수
C _{disk}	각 disk 용량
B _{disk}	각 disk 전송 대역폭
U _{server/c}	사용자 요구 bandwidth 대 capacity 비율
D _{strip/c}	디스크 지원 bandwidth 대 capacity 비율
S _{server}	Server cluster size
P	News video clip 인기도의 평균

4.2 결과 분석

[그림 3]은 동시 사용자 수가 적을 때에는 서비스 제공에 있어, 정책에 따른 성능의 차이가 두드러지지 않지만, 사용자의 수가 늘어 남에 따라 성능 차이가 존재함을 보여준다. Single 서버를 사용하는 방식의 경우, 사용자가 수가 증가 하면 hot spot 서버가 발생하여 성능이 급격히 떨어진다. 하지만, cluster 구조를 사용할 경우 hot spot 을 줄일 수 있고, 디스크의 cluster group 이 커질수록 load 분산이 가능하여 성능 변화에 영향을 적게 받는다. [1]에서 VOD data 의 경우 bandwidth-bound 에서는 비디오를 wide striping 하고, segment 는 single 로 저장하는 방식에 mirroring 을 하는 것이 효율이 높음을 보였다. 그러나 NOD 시스템에서는 VOD 와의 다른 NOD 데이터의 특성에 의해 Cs_Wd_Random 방법이 효율적이다



[그림 3] 각 배치방법에서 동시사용자수에 따른 서비스 실패율

5. 결론 및 향후과제

이번 실험에서 분산 clustered 서버구조에서 적당한 배치 방법을 알아 보았다. 인기도, 사용자 access pattern 을 배치 정책에 반영하지 않은 순수한 striping 배치 정책의 경우, clustered server striping_wide disk striping(random) 방식이 가장 좋은 것을 알았다. 앞으로는 이 정책을 바탕으로 전체 clustered group size 를 계산하는 방법, 인기도를 고려한 striping, replica 정책에 대하여 연구 할 것이다.

참고 문헌

- [1] Jamel Gafsi, Ernst W. Biersack, "Data Striping and Reliability Aspects in Distributed Video Servers", Institut EURECOM, 1998.6.
- [2] Cyrus Shahabi, Mohammad H. Alshayegi, and Shumeng Wang, "A Redundant Hierarchical Structure for a Distributed Continuous Media Server", IDMS, 1997. 9.
- [3] Juan Alemany, Jayram S.Thathacher, "Random Striping for News on Demand Servers" UW-CSE, 1997.2
- [4] Tat Seng Chua, Jiandong Li, Beng Chm Ooi, Kuan-Lee Tan, "Disk Striping Strategies for Large Video-on-Demand Servers". ACM Multimedia '96, 1996.11.
- [5] C. Busdikian and B.Patel, "Cost-Based Program Allocation for Distributed Multimedia-on-Demand Systems", IEEE MultiMedia, 1996 Fall.
- [6] Richard Muntz, Jose Renato Santo, Frank Fabbrocino, "Design of a Fault Tolerant Realtime Storage System for Multimedia Applications", cs.UCLA, technical report.
- [7] B. Ozden, R. Rastogi and A. Siberschatz, "Disk striping in video server environments", Data Engineering, 1995
- [8] Louis P. Slothouber, "A Model of Web Server Performance" <http://www.starmne.com/webstar/overnew.html>