

Clustered MNOD(Multimedia News On Demand) 서버의 설계와 성능 평가

^{*)}강 연경, ^{*)}박 성호, ^{*)}김 영주, ^{*)}정 기동
^{*)}부산대학교 전자계산학과

Design and Performance Analysis for Clustered MNOD(Multimedia News On Demand) Server

^{*)}Youné-Kyoung Kang, ^{*)}Seong-Ho Park, ^{*)}Young-Ju Kim, ^{*)}Ki-Dong Chung
^{*)}Department of Computer Science, Pusan National University

요 약

단일 데이터적 특성을 가지는 VOD(Video On Demand)와 같은 멀티미디어 응용의 경우 복잡적이고 집합적인 클러스터링 서버 구조(composite clustered server structure)를 가정하는 경우가 대부분 이었다. 그러나, 혼합 데이터적 특성이 강한 MNOD 와 같은 멀티미디어 응용은 단일화된 클러스터링 서버 구조 보다는 각 데이터의 특성을 잘 반영할 수 있는 독립적이고 분산된 형태의 클러스터링 구조가 필수적이다. 이에 본 논문에서는 MNOD 서비스를 위한 새로운 형태의 클러스터링 서버 구조를 설계하고, 구조의 당위성을 큐잉 분석 및 스케줄링의 효율성 측면에서 증명하려고 시도하였다. 큐잉 분석 결과, 독립적 형태의 구조가 지니는 확장성, 실패회복의 용이성 등의 제한 장점 뿐 아니라 응답시간의 측면에서도 기존의 집합적인 클러스터링 구조보다 좋은 성능을 보였다.

1. 서론

현재 NOD 시스템이라고 불리는 거의 대부분이 텍스트 위주의 서비스를 기반으로 하여 제공되고 있지만, 향후 5년 내에 인터넷을 통하여 접근되는 50% 이상의 데이터가 멀티미디어 형태일 것이라는 점을 감안해 볼 때[1], 앞으로의 멀티미디어 NOD (MNOD) 데이터의 기본 구성을 "멀티미디어 데이터 + 텍스트(또는 이미지) 데이터"로 가정하는 것에 별 부리가 없다고 본다. 이 때, 멀티미디어 데이터 및 텍스트/이미지 데이터는 시간적 상관성을 가지고 사용자에게 보여지며, 멀티미디어 데이터 및 텍스트/이미지 데이터는 모두 마감시간(completion deadline)을 가지는 실시간 데이터로서, 각 데이터의 실시간 정도가 같지 않다는 점에 주목할 필요가 있다. 즉 멀티미디어 데이터 및 텍스트/이미지 데이터가 동일한 마감시간을 가진다고 가정할 때, 두 데이터의 play 시간이 다름에 따라 각 데이터의 slack-time(deadline - arrival-time - play-time)에 많은 차이가 발생하여 실시간 정도가 달라짐을 알 수 있다. 슬랙타임의 현저한 차이는 스케줄링 방법의 현저한 차이를 의미하므로, MNOD 데이터는 마감시간뿐 아니라 슬랙타임까지 고려한 실시간 스케줄링 기법이 필요하다.

본 논문의 2 장에서는 관련연구를 소개하고, MNOD 시스템 서버 모델을 3 장에서 설명한다. 4 장에서는 제안된 모델에 대한 분석 결과를 설명하고, 끝으로 5 장에서는 결론과 향후 연구과제를 언급한다.

2. 관련 연구

2.1 서버 관련 연구

상대적으로 작고 비싸지 않은 컴퓨터들을 조합하여 결과적으로 대규모 컴퓨터와 동등한 계산 능력을 산출하려는 노력의 일환으로 클러스터링 기법을 들 수 있으며, 이 기법은 확장성, 신뢰성, 비용 등의 측면에서 대규모 멀티미디어 서버에 적당한 것으로 판단된다. 이미 많은 연구가 진행된 VOD 응용의 경우 대부분 단일화된 클러스터링 서버 구조를 가정하지만[2, 3, 4], 복합 데이터적 특성을 가지는 MNOD 응용은 독립적인 MNOD 서버를 가정하는 것이 더욱 타당하다[5]. 본 논문에서는 독립적 서버의 필요성을 큐잉 이론과 스케줄링 측면에서 살펴 보고자 한다.

2.2 큐잉 분석 관련 연구

제한된 MNOD 구조의 타당성을 검증하기 위한 방법으로 이미 잘 알려져 있는 큐잉 이론을 사용한다. 본 논문에서는 기본적인 큐잉 이론[6, 7]뿐만 아니라, 네트워크 상에서의 사용자 요구에 대한 분할, 병합, 대담 큐[8] 등을 고려한다. 특히, MNOD 서버와 매우 유사한 특징을 지니는 웹 서버[9] 및 대용량 저장 장치의 성능[10]에 대한 평가 결과를 이용하여 어떤 구조가 MNOD 시스템에 더욱 적합한지를 밝히고자 한다.

2.3 스케줄링 관련 연구

MNOD 데이터는 그 구성의 특성상 여러 가지 방식의 스케줄링이 가능하다. 대부분의 연구들은 텍스트/이미지 데이터를 비실시간 데이터로 가정하고 스케줄링 하지만[11], 텍스트/이미지 데이터를 멀티미디어 데이터와 동일한 실시간 데이터로 보고 스케줄링하는 방법도 생각해 볼 수 있다. 그러나, 데이터간의 실시간성이 다른 점을 감안해 볼 때 각 데이

11분 실시간 정도를 반영하는 스케줄링 기법이 필요하다[12].

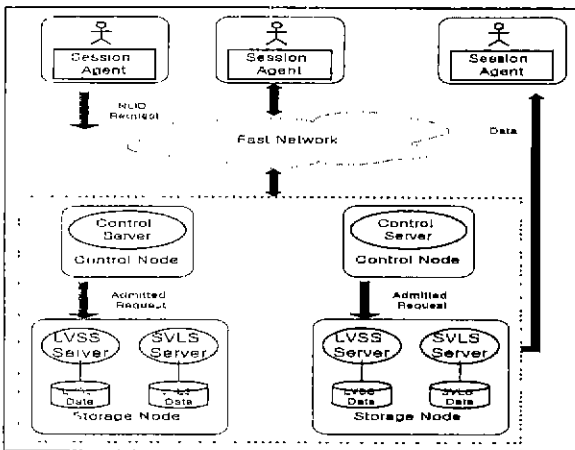
3. 서버 모델

3.1 MNOD 시스템 개관

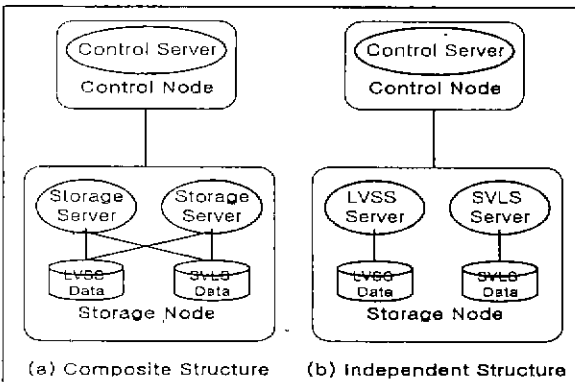
MNOD 데이터는 텍스트, 이미지, 오디오, 비디오 등 다양한 미디어로 구성되며, 각 미디어는 그 특성에 따라 <표 1>과 같이 크게 두 가지 형태로 나뉘어 질 수 있다. LVSS 데이터의 SVLS 데이터를 비교해 보면, 크기는 7826 배, 처리 시간은 1170 배, CPU 사용 비율은 95927 배, 슬랙타임은 3 배 정도 차이가 나며, LVSS 데이터가 실시간 정도가 더 크다.

	LVSS(Large Volume Short Slack) Data	SVLS(Small Volume Long Slack) Data
Example	Video, Audio Data	Image, Text Data
Size	21,915KB	2.8KB
Processing Time	117 sec	0.1 sec
CPU Usage Rate	95,927	1
Slack Time	58 sec	175 sec

<표 1> MNOD 데이터의 분류



<그림 1> 제안된 MNOD System 구조



<그림 2> Composite/Independent Clustered Server Structure의 Block Diagram 비교

MNOD 시스템에는 다른 특성의 데이터가 공존하기 때문에 기존의 단일화된 VOD 서버 형태 보다는 <그림 1>과 같이 각 데이터 형태별로 독립적인 서버 구조가 더욱 타당하다. <그림 2>는 기존의 클러스터링 서버구조와 제안하는 구조와의 차이를 블록 다이어그램 형태로 보여준다.

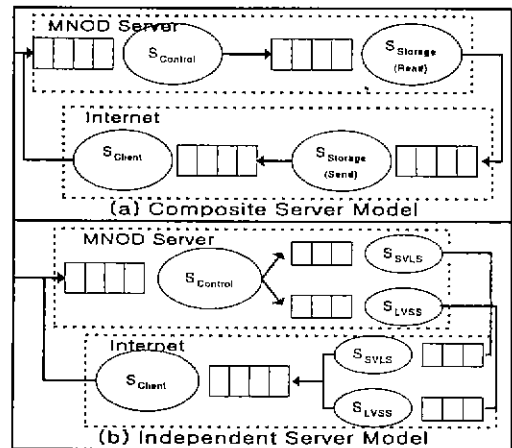
3.2 큐잉 분석을 위한 Parameters

기호	의미
W_r	큐의 응답시간(response time)
λ	하루의 평균 도착율
s, μ	평균 서비스 시간, 평균 서비스율, $\mu = 1/s$
ρ	서버 이용률, $\rho = u/c = \lambda s = \lambda/\mu c$
c	서버의 개수
u	트래픽 밀도; $u = \lambda/\mu$
F	평균 파일 크기
B	버퍼의 크기
C	클라이언트의 네트워크 대역폭

<표 2> 큐잉 모델에 사용되는 변수들

<표 2>는 MNOD 서버의 큐잉 분석에 사용되는 주요 인수들 나타내며, 이를 이용한 응답시간은 아래와 같다[9].

$$W_r = \frac{F}{C} + \frac{1}{1-\lambda} + \frac{F}{\lambda - \lambda F} + \frac{F(B+\rho)}{B\rho - \lambda F(B+\rho)}$$

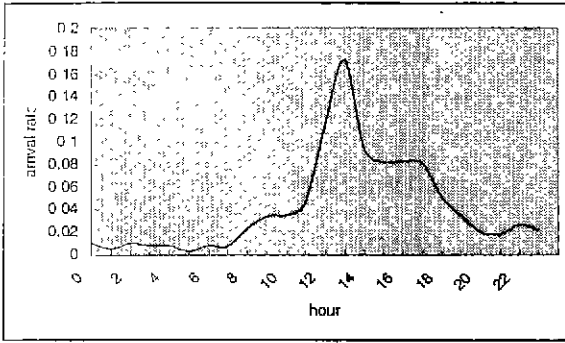


<그림 3> Independent/Composite Server의 큐잉 모델 블록도

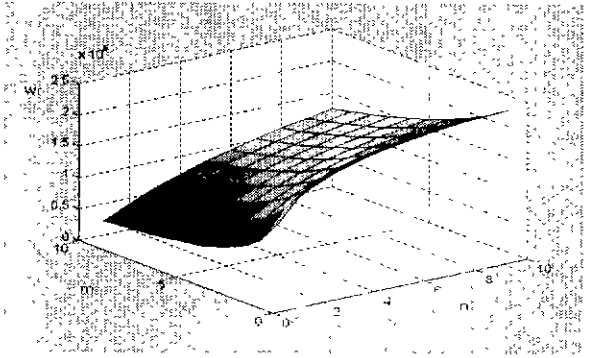
본 논문에서 언급되는 Independent 서버와 Composite 서버의 큐잉 모델은 <그림 3>과 같다.

4. 성능평가

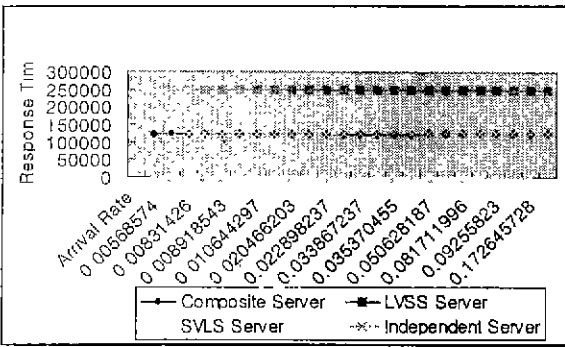
<그림 4>는 부산일보사가 제공하는 인터넷 신문에 대한 하루 동안의 평균 도착율에 대한 분석이다. 평균 도착율을 기반으로 하여 큐의 응답시간을 살펴보면 <그림 5>와 같다.



<그림 4> 하루동안의 평균 도착율(Average arrival rate)



<그림 6> m, n 값에 따른 전체 응답 시간의 변화



<그림 5> 도착율 변화에 따른 서버의 응답시간

5. 결론 및 향후 연구 과제

단일 데이터적 특성을 가지는 VOD와는 달리 혼합 데이터적 특성이 강한 MNOD와 같은 멀티미디어 응용은 단일화된 클러스터링 서버 구조 보다는 각 데이터의 특성을 잘 반영할 수 있는 독립적이고 분산된 형태의 클러스터링 구조가 설득력 있다. 본 논문에서 제시하는 '데이터 특성별로 독립된 서버 구조'는 독립적 특성으로 인하여 확장성, 대응량성, 비용, 실패 회복 측면에서 유리하며 클러스터링 구조에 잘 부합된다. 구조의 당위성은 직관적 측면 뿐만 아니라 큐잉 분석 결과 및 스케줄링의 용이성 등에서도 찾아 볼 수 있다. 특히, MNOD 서버를 구축할 때 각 서버간 구성 비율 설정에 도움이 될 수 있도록 독립 서버간의 구성비율에 대한 연구를 첨가하였다. 큐잉 분석 결과를 바탕으로 하여 향후에는 MNOD 시스템 구조에 어떤 영향을 미치는가를 연구할 계획이다.

참고 문헌

- [1] *Resource Based Caching for Web Servers*, Renu Tewari, Harrick M. Vin, Asit Dan and Dinkar Sitaram, 1997
- [2] *Providing A Scalable, Interactive Video-On-Demand Service Using Multicast Communication*, Kevin C. Almeroth, Mostafa H. Ammar, 1994
- [3] *Real-Time Issues for Clustered Multimedia Servers*, Renu Tewari, Jajat Mukherjee, Daniel M. Dias, Harrick M. Vin, 1995
- [4] *An Implicitly Scalable, Fully Interactive Multimedia Storage Server*, Frank Fabbrocino, Jose Renato Santos and Richard Muntz, 1997
- [5] *Synchronization Architecture and Protocols for a Multimedia News Service Application*, L. Lamont and N. D. Georganas, 1996
- [6] *Operating Systems*, William Stallings, Prentice-Hall International Inc. 1995, p633-p657
- [7] *Operating Systems*, H. M. Deitel, Addison Wesley, 1994, p521-p552
- [8] <http://networks.ecse.rpi.edu/~vastola/pslunks/perf/>
- [9] *A Model of Web Server Performance*, Louis P. Stouffer, 1996
- [10] *An Approximate Performance Model of a Unitree Mass Storage System*, Odysseas I. Pentakalos, 1998
- [11] *Designing a Distributed Multimedia Synchronization Scheduler*, Jerzy P. Jarmasz and Nicolas D. Georganas, 1997 IEEE, p451-p457
- [12] *기존의 운영체제를 이용한 멀티미디어 데이터의 실시간 스케줄링*, 강연경, 박성호, 김영주, 정기동, 1998 정보과학회

Open Queueing Network에 기반한 평가에 따르면, 도착율이 0.008 이하인 경우에는 Independent Server가 더 좋은 응답시간을 보인다. 그리고 도착율이 0.008 이상인 경우에는 동일한 응답시간을 보이는데, 같은 응답시간일 경우 확장성, 대응량성, 비용, 실패 회복 측면에서 독립적 서버 구조가 더 좋다.

서비스 시간에 따른 평가에서는 평균 도착율 $\lambda = 0.0417$ 로 하여 Composite 서버를 M/G/1 큐잉 이론에 적용하고, LVSS 및 SVLS 서버를 M/D/1 공식에 각각 적용하여 Independent 서버의 성능을 조사한 결과 서버 이용율과 큐 대기 시간 등의 측면에서 Independent 서버가 좋음을 알 수 있었다.

Workload 및 Cluster에 기반한 평가에서는 모든 도착율 λ_i 에 대하여 Independent 서버와 Composite 서버가 동일한 성능을 내는 것으로 조사되었다(이는 동일하다고 가정).

Independent 서버가 Composite 서버보다 성능이 좋다고 가정할 때, Independent 서버 내의 LVSS 서버와 SVLS 서버간의 구성 비율을 알아보기 위해 각 서버간 능력의 차이 비율을 $LVSS : SVLS = m : n$ 으로 가정하면, 최적의 m, n 값은 W_i 을 최소화 하는 양수로서 $m = 8, n = 2$ 이다.

$$W_i = \frac{mW_1 + nW_2}{m+n} \quad (\text{단, } W_1 = \frac{F_1}{C} + \frac{1}{1-\lambda} + \frac{F_1}{\lambda - \lambda F_1} + \frac{F_1(B+\rho)}{B\rho - \lambda F_1(B+\rho)})$$