

스트림 서버에서 자원 사용 우선순위 결정을 위한 시뮬레이션 연구

박진원**

Simulation Study on the Stream Server for Deciding the Priority for Using Resources

Jin-Won Park

Abstract

Stream servers are for supplying multimedia stream data to users through the internet such as movies and music without discontinuation. A typical stream server is designed roughly by considering the characteristics of stream services and by employing processors, memory, PCI bus, Ethernet, TOE and disks.

This study focuses on deciding the priority for using resources such as PCI bus, buffer memory and TOE buffer, which have limited capacities in a typical stream server. When the priorities for using limited resources are not given properly, the stream servers may not even function as originally designed.

The simulation study shows that the top priority for using PCI bus for normal streaming services should be given to the operation that sends data from buffer memory to TOE buffer. Giving priority for using PCI bus to other operation such as sending data from disks to memory results in a deadlock phenomenon.

Key Words: Simulation, Stream server, PCI bus, TOE, Buffer memory, Stream services

* 본 연구는 2003년 한국과학재단 지역대학 우수과학자 지원연구사업의 일환으로 수행되었음.
(R05-2003-000-10212-0)

** 홍익대학교 과학기술대학 전자전기컴퓨터공학부

1. 서론

인터넷 스트림 서비스는 주로 영화나 음악과 같이 긴 시간동안 끊어짐이 없이 동영상 자료가 실시간으로 제공되는 것으로, 기존의 과학 계산이나 트랜잭션 서비스 등과 현저히 구분되는 특징을 갖고 있다. 이에 따라 스트림 서버에 대한 설계도 스트림 서비스를 효율적으로 지원할 수 있는 구조를 지향하게 된다. 스트림 서버의 구성 요소로 프로세서, 메모리, 하드디스크, PCI 버스, Ethernet, Buffer 메모리, TOE(TCP/IP Offload Engine) 등이 포함되며 이들을 이용하여 구성되는 스트림 서버의 설계 대안에 대한 기능, 성능 측면에서의 평가는 이산사건 기반 컴퓨터 시뮬레이션 실험을 통해 수행될 수 있으며, 궁극적으로 최적의 가격대비 성능비를 발휘할 수 있는 설계 대안을 선택하게 된다. 이를 위해 경우에 따라서는 통계적 분석과 최적화 기법을 사용할 필요성이 제기되기도 한다.

멀티미디어 데이터 처리용 서버가 연구되기 시작한 초기에 Shenoy et. al.[1]은 멀티미디어 서비스를 효율적으로 제공하기 위해 최적의 블록 크기와 디스크 개수를 결정하는 문제, 데이터 스트라이핑 수준과 미디어 스트림 데이터를 복사해서 저장하는 문제, 그리고 고장 감내(Fault Tolerance) 문제 등 3 가지 내용을 연구 과제로 제시하였다.

Tewari et. al.[2]은 다양한 시스템 구성에 따라 스트림당 전체 비용을 비교하는 방식으로 중복 스트림 데이터를 디스크와 노드에 분산 배치하여 성능을 분석하고, 무작위 분산 방식이 실시간 성능 향상에 좋은 영향을 미친다는 점을 발표했다. Tewari et. al. [3]는 또한 웹 서버가 취급하는 데이터가 향후(1998년 기준으로) 연속형 멀티미디어 형태가 주를 이룰 것으로(50% 이상) 예상하고 자원 기반 캐싱 알고리즘을 연구하여 다양한 형태의 데이터 타입을 요구하는 서비스를 관리하기 위해 활용하였다.

최근의 인터넷 서버에 관한 연구는 Shenoy와 Vin[4]의 예에서 보는 바와 같이 멀티미디어 저장 장치에서 사용되고 있는 미디어 스트림 자료 저장과 회수에 관한 내용과, Waldvogel[5]에서와 같이 확장성 있는 멀티미디어 서버의 구축을 위해 버퍼 메모리를 설정하고 이들의 운영을 통한 성능 향상을 도모하고 있다.

한편, 국내에서 멀티미디어 서버에 대한 성능 연구는 박기진과 김성수[6]의 멀티미디어 서버에 대한 시뮬레이션 모델링이 시초로써, 서버 성능 및 서비스 질에 중요한 영향을 미치는 서버 설계 요소 파악, 고객 요구에 대한 비용 효율적인 서비스 수준과 작업 부하의 변동을 수용할 수 있는 서버 용량에 대한 개략적인 정보 획득을 목표로 하였다. 박기진과 김성수[7]는 또한 대규모 주문형 멀티미디어 서비스 시스템에 대해 고객 도착률과 자원 고장을 변동에 대한 서버 자원의 이용률과 큐의 길이를 구하여 효율적인 시스템 구성 자원의 부분별 용량을 산정하였다.

박성식과 이호석[8]은 다중 쓰레드를 이용한 개인용 주문형 비디오 시스템을 Windows NT 환경에서 구현하였는데, 연속 매체를 서비스하기 위한 비디오 객체를 병렬 디스크에 저장하고 이를 실시간으로 서비스하며 사용자 요구로 인한 VCR 연산, 사용자 프로그램과 서버와의 프로토콜 개발, 서버 자원에 대한 사용을 실시간으로 파악할 수 있게 하였다. 정진국 등[9]은 다중처리기 기반 웹 서버 구조를 제안하고 이에 대한 성능 분석을 시도하였다. 최근 널리 보급되고 있는 다중 처리기 상에서 요구기반 병행 구조, 작업기반 병행 구조, 쓰레드 풀 구조 등을 대상으로 어떤 병행 웹 서버 구조가 가장 높은 성능을 낼 수 있는지 실험을 통해 분석하였으며, 결론으로는 쓰레드 기반 풀 구조가 다른 웹 서버들에 비해 우수한 성능을 보인 것으로 밝혔다.

지금까지 인터넷 서버에 관한 연구는 멀티미디어 서버를 중심으로 제한된 종류의 시스

템 구성요소, 예를 들면 프로세서, 메모리, 디스크, 네트워크 등을 대상으로 단순한 시스템 구성을 설정한 후 이를 대상으로 큐잉 네트워크 모델이나 시뮬레이션을 통해 성능을 분석하는 데에 초점을 맞추어 왔다. 또한, 일부 연구에서[7],[10] 결합이 발생한 경우를 고려한 성능 분석 모델을 설정하고 전체 시스템에 대한 성능을 분석하였다. 그러나 TOE(TCP/IP Offload Engine)나 SAN(Storage Area Network), NAS(Network Attached Storage) 등을 비롯하여 새로운 하드웨어 요소들이 개발되고 인터넷 서버의 운영 방식도 단순히 하나의 서버가 모든 서비스를 담당하지 않고 특화된 여러 종류의 서버가 존재하며, 어떤 경우에는 계층적 구조를 갖는 서버 시스템을 구축하는 경우도 발생하고 있다.

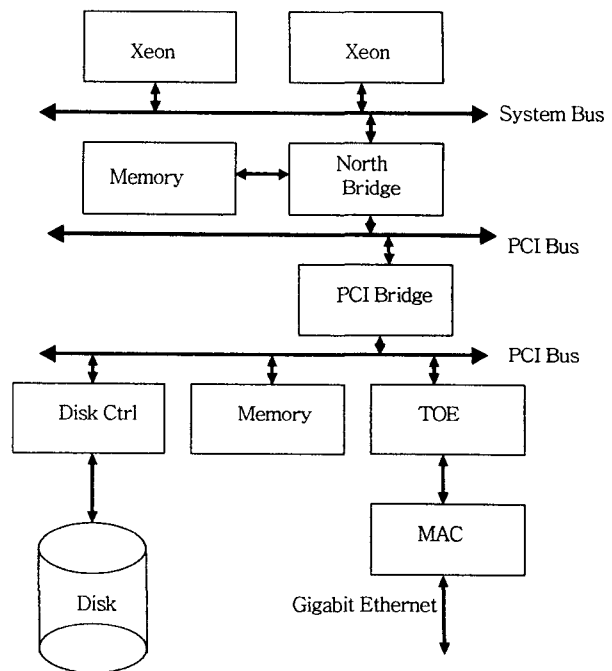
스트림 서버는 기존의 인터넷 서버가 갖고 있는 기능에 고속 스트림 서비스를 처리하는 기능을 포함하고 있다. 구조적으로는 작업 처리 기능을 담당하는 프로세싱 유닛과 고속 스트림 데이터 처리를 담당하는 데이터 저장장치 및 네트워크 장치로 구성할 수 있다. 본 논문은 스트림 서버의 시스템 구조를 스트림 서버가 제공할 서비스의 특성을 고려하여 개략적으로 설계한 내용을 소개한다. 그리고, 이를 AweSim을 이용한 이산사건 기반 시뮬레이션 실험을 통해 시스템 구성 자원들의 사용 우선순위를 결정함으로써 스트림 서버의 원활한 동작을 보장하는 것을 목적으로 한다. 이를 위해 스트림 서버의 개념도를 설정하고 스트림 서버의 동작을 모형화함으로써 시뮬레이션 모델이 정의되도록 하였다.

본 논문은 제2장에서 스트림 서버에 대한 개략적인 요구 규격과 개념적인 시스템 구조를 살펴보고, 제3장에서는 제2장에서 서술한 서비스와 구조를 바탕으로 스트림 서비스에 포함되는 구체적인 동작을 시뮬레이션 모델로 표현한다. 제4장에서는 시뮬레이션 실험 수행 결과를 분석하여 정상적인 스트림 서버의 동작을 보장하기 위해 PCI 버스 사용에 대한 우

선권을 어떤 동작에 부여해야 하는가를 결정하고, 제5장에서 결론을 맺는다.

2. 스트림 서버에 대한 요구 규격과 개념적 구조

스트림 서버는 우선 대략 1,000명까지의 사용자에게 서비스를 제공하고 이중에서 동시에 200명에게 멀티미디어 서비스를 제공할 수 있도록 설정했다. 그러나 대부분의 스트림 서버는 클러스터 형태로 구성되어 있어 시뮬레이션을 통한 분석의 대상이 되는 시스템은 대략 50명의 사용자에게 총 1Gbps 수준의 스트림 데이터를 제공할 수 있는 단위로 설정했다. 이밖에 스트림 서버가 제공하는 주요 기능으로는, 대용량 멀티미디어 자료를 저장할 수 있는 스토리지 시스템 제공, 시스템의 선형적 확장성 및 고가용성 보장, 성능을 극대화할 수 있는 효율적인 시스템 자원 관리 기능 등을 들 수 있다.



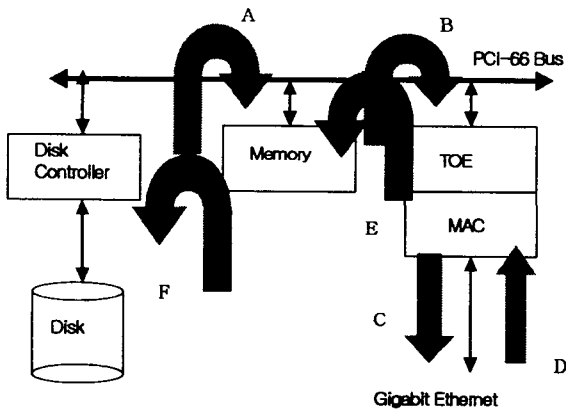
<그림 1> 스트림 서버 개념도

본 논문에서 설정한 스트림 서버는 <그림 1>에서 보는 바와 같이 프로세싱 유닛과 네트워크 및 스토리지 유닛이 결합된 시스템으로 Gigabit Ethernet을 통해 클러스터 형태로 구성된다. 스트림 서버는 기본적으로 멀티미디어 관련 데이터를 사용자에게 실시간으로 제공하는 것을 목표로 설정했기 때문에 이와 같은 구조를 선택하게 되었다.

본 논문에서 설정한 시스템 구조 시뮬레이션 모델은 <그림 1>의 스트림 서버 개념도를 기반으로 정의한 것이다.

3. 스트림 서버에 대한 시뮬레이션 모델

스트림 서버에 대한 시스템 구조와 동작을 AweSim을 이용하여 모델링했다 이 모델은 <그림 1>에서 제시한 스트림 서버 개념 모델을 대상으로 크게 두 종류의 서비스를 수행하는 것으로 가정했다. 즉, 스트림 서버가 외부 사용자들에게 스트림 서비스를 제공하기 위한 작업(서비스)과 스트림 서버가 외부의 글로벌 서버로부터 스트림 데이터를 다운받기 위한 작업(다운로딩)이 동시에 진행되는 것으로 설정했다. 이를 구체적으로 살펴보면 <그림 2>와 같다.



<그림 2> 스트림 서버 동작 모형

본 논문에서 정의한 시뮬레이션 모델에서는 사용자의 스트림 서비스를 프로세서가 명령을

내림으로써 시작된다고 설정하였다. 스트림 데이터를 사용자에게 전달해 주라는 명령은 우선 디스크 컨트롤러에게 전달되며, 이에 따라 디스크에서 디스크 컨트롤러와 PCI 버스를 거쳐 메모리에 일시적으로 저장된다(A 동작).

A 동작을 수행하기 위해 PCI 버스와 메모리 1개 유닛(2MB)가 필요한데 메모리는 AweSim에서 Resource로 정의되었다. 메모리에 도착한 스트림 데이터는 TOE에 의해 다시 PCI 버스를 거쳐 TOE에 내장된 TOE 버퍼로 이동되었다가(B 동작) 최종적으로 Gigabit Ethernet을 통해 사용자에게 전달된다(C 동작). B 동작에는 메모리 1개 유닛, PCI 버스, TOE 버퍼 1개 유닛이 필요하고 C 동작에는 Gigabit Ethernet, TOE 버퍼 1개 유닛, 메모리 1개 유닛이 필요한 것으로 정의되었다.

A 동작이 수행되고 나면 PCI 버스를 일시적으로 내놓았다가 다시 PCI 버스와 TOE 버퍼 1개 유닛을 요구하여 B 동작을 수행하며, B 동작이 끝나면 PCI 버스를 내놓는다. C 동작은 TOE 주관으로 수행되는데 Gigabit Ethernet이 동원된다. C 동작이 수행되고 나면 TOE 버퍼 1개 유닛, Gigabit Ethernet, 메모리 1개 유닛이 사용 가능하게 되어 다른 동작에 활용된다.

한편 글로벌 서버에서 스트림 데이터를 다운받는 작업은 우선, Gigabit Ethernet을 통해 TOE 버퍼에 스트림 데이터가 전달되고(D 동작), PCI 버스를 통해 메모리에 전달되며(E 동작), 다시 PCI 버스를 거쳐 디스크에 저장된다(F 동작). D 동작에는 Gigabit Ethernet, TOE 버퍼 등이 사용되고, E 동작에는 TOE 버퍼, PCI 버스 및 메모리 1개 유닛이 사용되며, F 동작에는 PCI 버스와 메모리 1개 유닛이 사용된다.

본 시뮬레이션 모델에서 적용한 시스템 파라미터는 다음과 같다.

- 데이터 블록 크기: 2MB
- PCI-66 Bus : 66MHz, 64 bit

- Memory : 256MB
- TOE Buffer : 16MB
- Gigabit Ethernet : 20 Mbps/user
- 스트림 데이터 크기: 영화 1편의 평균 길이로서 10GB

4. 시뮬레이션 실험 결과 분석

4.1 서로 다른 자원사용 우선순위 설정

스트림에 대한 시뮬레이션을 수행하는데 초점이 되는 것은 PCI 버스에 대한 사용 우선권을 어느 동작에 부여하는가 하는 문제이다. 앞에서 살펴본 바와 같이 PCI 버스를 사용하는 동작은 A, B 동작과 E, F 동작이다. 그런데 이들 동작은 PCI 버스 뿐만 아니라 메모리, TOE 버퍼 등의 자원을 동시에 요구하고 있다. 따라서 이들 동작에 어떤 방식으로 PCI 버스에 대한 사용 우선순위를 부여하는가에 따라 시스템이 정상적으로 동작할 수도 있고, 교착 상태(Dead Lock)에 빠질 수도 있다.

또한, TOE 버퍼에 대한 사용 우선순위를 스트림 서버에서 사용자에게 스트림 데이터를 전달해 주는 서비스 방향과, 글로벌 서버에서 스트림 서버에게 스트림 데이터를 전달해 주는 다운로드 방향 중에서 어디에 우선권을 주는가에 따라 시스템이 정상적으로 동작할 수도 있고 교착상태에 빠질 수도 있다.

메모리 사용에 대한 우선순위 결정도 시스템의 정상 동작 여부에 영향을 미칠 수 있을 것으로 판단하여 이에 대한 결정도 고려하였다. 즉, 메모리가 사용되는 동작은 A, B, C 동작과 E, F 동작으로, 일단 A 동작에서 메모리가 점유되면 C 동작이 끝날 때까지 점유되는 것으로 설정했고, E 동작에서 점유되면 역시 F 동작이 끝날 때까지 메모리가 점유되는 것으로 설정했다.

<표 1>은 각 자원별 사용 우선순위를 다르게 설정한데 대한 시뮬레이션을 실행한 결과이다.

<표 1>에서 보는 바와 같이 서비스 방향은 경우에 따라서 정상적인 동작을 보이거나 다운로드 방향은 모두 교착 상태에 빠지는 것을 알 수 있다. <표 1>은 서로 다른 자원사용 우선순위를 부여하는 모든 경우에 대한 실험 결과는 아니나 표시되지 않은 경우에도 서비스 방향과 다운로드 방향 모두에서 정상적인 동작을 보이는 경우는 없었다.

<표 1> 각 동작에 서로 다른 자원사용 우선 순위 부여 결과

PCI 버스	TOE 버퍼	메모리	서비스	다운로딩
A,B,E,F	B,D	A,E	정상	교착
A,B,E,F	D,B	E,A	교착	교착
A,E,B,F	B,D	E,A	교착	교착
A,E,B,F	D,B	E,A	교착	교착
F,A,B,E	B,D	A,E	정상	교착
F,A,B,E	D,B	A,E	교착	교착
F,B,A,E	B,D	A,E	정상	교착
F,B,A,E	D,B	E,A	교착	교착

4.2 PCI 버스 사용에 동일한 우선순위 부여

다음에는 A, B 동작에 PCI 버스 사용에 대한 우선권을 동일한 순위로 부여하고 E, F 동작에도 동일한 우선순위를 부여하되, (A,B) 동작 우선의 경우와 (E,F) 동작 우선의 경우를 나누어 살펴보고, 각 경우마다 각각 다른 TOE 버퍼 및 메모리 사용 우선순위를 설정할 경우의 시뮬레이션 실험 결과이다. 이를 정리해 보면 <표 2>와 같다. <표 2>에서 괄호 안에 포함된 동작은 해당 자원에 대해 동일한 우선순위를 부여하는 것을 의미한다.

<표 2>에서 보는 바와 같이 이 경우에도 교착 상태가 5 가지 경우에서 발생하고 3 가지 경우에 단방향에서 정상적인 동작이 이루어진다. 그러나 정상적인 동작의 경우에도 서비스 방향이나 다운로드 방향의 한 방향에서만 정상 동작이 수행되어, 이들 경우도 사실상 정상적인 동작이 이루어진다고 보기 어렵다.

<표 2> A,B 동작과 E,F 동작에 PCI 버스 사용에 동일한 우선권을 부여하는 경우

PCI 버스	TOE 버퍼	메모리	서비스	다운로딩
(A,B),(E,F)	D,B	A,E	교착	교착
(A,B),(E,F)	D,B	E,A	교착	교착
(A,B),(E,F)	B,D	A,E	교착	교착
(A,B),(E,F)	B,D	E,A	교착	교착
(E,F),(A,B)	B,D	A,E	교착	교착
(E,F),(A,B)	B,D	E,A	정상	교착
(E,F),(A,B)	D,B	A,E	교착	정상
(E,F),(A,B)	D,B	E,A	교착	정상

다음에는 A, F 동작에 PCI 버스 사용에 대한 우선권을 동일한 순위로 부여하고 B, E 동작에도 동일한 우선권을 부여하되, (A, F) 동작 우선의 경우와 (B, E) 동작 우선의 경우를 나누어 살펴보고, 각 경우마다 각각 다른 TOE 버퍼 및 메모리 사용 우선순위를 설정할 경우의 시뮬레이션 실험 결과를 <표 3>에 보이고 있다. <표 3>에서 괄호 안에 포함된 동작은 해당 자원에 대해 동일한 우선순위를 갖는 것을 의미한다.

다음의 <표 3>에서 보는 바와 같이, 이 경우에도 교착 상태가 5 가지 경우에 발생하고 3 가지 경우에서 서비스 방향에서만 정상적인 동작이 이루어진다.

<표 3> A,F 동작과 B,E 동작에 PCI 버스 사용에 동일한 우선권을 부여하는 경우

PCI 버스	TOE 버퍼	메모리	서비스	다운로딩
(A,F),(B,E)	B,D	A,E	정상	교착
(A,F),(B,E)	B,D	E,A	정상	교착
(A,F),(B,E)	D,B	A,E	교착	교착
(A,F),(B,E)	D,B	E,A	교착	교착
(B,E),(A,F)	B,D	A,E	교착	교착
(B,E),(A,F)	B,D	E,A	정상	교착
(B,E),(A,F)	D,B	A,E	교착	교착
(B,E),(A,F)	D,B	E,A	교착	교착

<표 3>을 살펴보면 서비스 방향의 동작은

정상적으로 수행되는 3가지 경우 모두 B, D 동작의 순으로 TOE 버퍼 사용에 대한 우선순위가 부여될 때임을 알 수 있다. 이는 서비스 방향과 다운로드 방향 모두에서 정상적인 동작이 수행되는 자원사용 우선순위를 찾는 단서가 되는 것으로 보인다.

다음에는 동작 A, B, E, F에 PCI 버스 사용에 대한 우선권을 동일하게 부여하고 PCI 버스를 사용하지 않는 C 동작과 D 동작만 분리하는 경우에 대한 실험을 수행하였다. 이 실험에 대한 결과를 정리해 보면 <표 4>와 같다. <표 4>에서도 괄호 안에 포함된 동작은 해당 자원에 대해 동일한 우선순위를 부여하는 것을 의미한다.

<표 4> A, B, E, F 동작에 동일한 우선권을 부여하는 경우

PCI 버스	TOE 버퍼	메모리	서비스	다운로딩
(A,B,E,F)	D,B	(A,E)	교착	교착
(A,B,E,F)	B,D	(A,E)	정상	교착

<표 4>에서 보는 바와 같이 이 경우에도 <표 3>에서 나타난 현상과 유사하게 D 동작에 TOE 버퍼 사용에 관한 우선권을 부여하면 교착 상태에 빠지고 그렇지 않은 경우에는 서비스 방향에서만 정상적으로 동작함을 알 수 있다.

상기와 같은 실험 결과를 바탕으로 PCI 버스 사용에 관한 여러 가지 조합의 우선권 부여 방안을 모색한 결과 최종적으로 다음과 같은 2가지 방안이 제시되었다. 즉, 동작 B와 F 동작은 분리하여 B, F 혹은 F, B 순으로 PCI 버스 사용에 대한 우선권을 부여하고 A와 E 동작에는 동일한 순위의 우선권을 부여하는 것으로 정했다. 그리고 TOE 버퍼 사용에 관한 우선권은 B 동작에 우선권을 부여하는 경우 정상적인 동작을 보일것으로 예상하면서, 만약의 경우를 대비하여 B, D 동작 순의 우선권 부여와 D, B 동작 순의 우선권 부여 조합을 검토하였다. 이에 대한 실험 결과를 살펴보면 <표 5>와 같다. <표 5>에서 괄호

내에 포함된 동작은 해당 자원에 대해 동일한 우선순위를 부여하는 것을 의미한다

<표 5> A, E 동작에 동일한 우선권을 부여하는 경우

PCI 버스	TOE 버퍼	메모리	서비스	다운로딩
F,B,(A,E)	D,B	(A,E)	교착	교착
F,B,(A,E)	B,D	(A,E)	정상	정상
B,F,(A,E)	B,D	(A,E)	정상	정상
B,F,(A,E)	D,B	(A,E)	교착	교착
(A,E),B,F	B,D	(A,E)	정상	교착
(A,E),F,B	B,D	(A,E)	정상	비확성

<표 5>에서 보는 바와 같이 이 경우에는 D 동작에 TOE 버퍼 사용에 관한 우선권을 부여할 경우 교착 상태에 빠지고, 반대로 B 동작에 TOE 버퍼 사용에 대한 우선권을 부여하는 경우 예상한 대로 정상적인 동작을 보여 주었다. 이 경우의 정상 동작은 스트림 서버에서 사용자 방향으로 스트림 데이터가 전달되는 것(서비스)과 글로벌 서버에서 스트림 서버로 스트림 데이터가 전달되는(다운로딩) 양방향에서 비교적 비슷한 수준으로 데이터가 전달됨을 의미한다. 한가지 특이한 사항은 (A,E), F, B 동작 순으로 PCI 버스 사용 우선권을 부여하고 B, D 동작순으로 TOE 버퍼 사용 우선권을 부여하는 경우, 서비스 방향은 정상적으로 동작하는데 다운로드 방향은 비교적 정상적으로 동작하나 데이터 전달량이 원활하지 못한 상황이 발생한다. 이 경우에는 전체적으로 자원사용에 대한 우선순위가 서비스 방향에 부여되기 때문에 발생하는 현상으로 보이며, 다운로드 방향의 작업량을 다소 감소시키면 다운로드 방향도 정상적인 동작이 이루어질 것으로 기대한다.

5. 결론

지금까지 살펴본 바와 같이 차세대 인터넷 서버가 스트림 데이터를 스트림 서버에서 사

용자에게 제공해 주거나 글로벌 서버에서 스트림 서버로 필요한 데이터를 전달해 주는 작업이 동시에 수행되어야 하는 경우 시스템 자원에 대한 우선 분배 원칙을 정하는 것이 매우 중요한 일로 나타나고 있다.

현재까지 살펴본 바와 같이 스트림 데이터가 블록 단위(2MB로 설정)로 전송된다고 가정하면 디스크에서 메모리, 메모리에서 TOE 버퍼를 거쳐 Gigabit Ethernet을 통해 사용자에게 전달되는 경로와, 글로벌 서버에서 Gigabit Ethernet을 거쳐 TOE 버퍼, PCI 버스 및 메모리를 경유한 후 다시 PCI 버스를 통해 디스크에 저장되는 경로 간에 PCI 버스 사용에 관한 우선권 부여 방법이 시스템 동작에 교착 상태를 가져올 상황도 발생하게 된다.

본 연구에서 보인 바와 같이 디스크에서 메모리로 스트림 데이터가 전달되는 동작(A 동작)과 글로벌 서버에서 스트림 서버로 스트림 데이터가 전달되는 작업 중에서 TOE 버퍼에 있는 데이터를 메모리로 옮기는 동작(E 동작)은 같은 우선순위를 부여하되, 메모리에서 다시 PCI 버스를 거쳐 디스크로 최종 전달되는 동작(F 동작)과 메모리에서 TOE 버퍼로 데이터를 전달하는 동작(B 동작)보다는 낮은 우선순위를 부여하여 PCI 버스를 사용하게 하고, TOE 버퍼의 경우는 반드시 메모리에서 TOE 버퍼로 데이터를 전달하는 동작(B 동작)에 TOE 버퍼에 있는 글로벌 서버의 데이터를 TOE 버퍼에 전달하는 동작(D 동작)에 비해 높은 사용 우선순위를 부여해야 한다는 점이 밝혀졌다.

본 논문은 스트림 서버의 기능적인 측면을 강조하여 스트림 서버가 정상적으로 운용되기 위한 방안을 시뮬레이션 실험을 통해 제시하였다. 그러나 향후에는 스트림 서버에 대한 구성 요소의 효율적인 배치와 적절한 작업 부하 분배를 통한 성능 최적화 연구가 추가로 수행되어야 할 것이다.

참고 문헌

[1] Shenoy Prashant J., Goyal P., Vin H.M., "Issues in Multimedia Server Design," ACM Computing Surveys, Vol.27, No. 4, December 1995.

[2] Tewari Renu, Dias D.M., Mukherjee R., Vin H.M., "High Availability in Clustered Multimedia Servers," Proceedings of the IEEE International Conference on Data Engineering, New Orleans, pp345-354, February 1996.

[3] Tewari Renu, Vin H.M., Dan A., Sitaram D., "Resource-based Caching for Web Servers," Proceedings of ACM/SPIE multimedia Computing and Networking 1998(MMCN '98), San Jose, pp191-204, January 1998.

[4] Shenoy Prashant J., Vin H.M., "Multimedia Storage Servers," Readings in Multimedia Computing, Kevin Jeffay et. al. Editors, Morgan Kaufmann Publishers, 2001.

[5] Waldvogel Marcel, Deng W., Janakiraman R., "Efficient Buffer Management for Scalable Media-on-Demand," IBM Research Report, 2002.6.

[6] 박기진, 김성수, "시뮬레이션 기법을 이용한 주문형 멀티미디어 서버의 성능 평가," 한국시뮬레이션학회 논문지, 제7권 제2호, pp33-43, 1998.12.

[7] 박기진, 김성수, "결합 발생을 고려한 대규모 주문형 멀티미디어 서비스 시스템의 큐잉 네트워크 모델," 정보과학회논문지, 제26권 제3호, pp291-300, 1999.3.

[8] 박성식, 이호석, "다중 스트림을 이용한 개인용 VOD 시스템의 설계와 구현," 정보과학회논문지, 컴퓨팅의 실제, 제6권 제3호, pp299-310, 2000.6.

[9] 정진국, 남중호, 박성용, "다중처리기 기반 웹 서버 구조의 실험적 성능 분석," 정보과학회논문지, 정보통신 제28권 제1호, pp22-36, 2001.3.

[10] 정지영, 김성수, "결합 주입 시뮬레이션을 이용한 병렬 비디오 서버의 의존도 분석," 한국시뮬레이션학회 논문지, 제9권 제2호, pp51-60, 2000.6.

주 작 성 자 : 박진원
 논문투고일 : 2003. 12. 19
 논문심사일 : 2003. 12. 29(1차), 2003. 12. 30(2차), 2003. 12. 31(3차)
 심사판정일 : 2003. 12. 31

● 저자소개 ●



박진원

1975 서울대학교 공과대학 산업공학과 학사
 1982 미국 오하이오주립대학교 산업시스템공학과 석사
 1987 미국 오하이오주립대학교 산업시스템공학과 박사
 1987~1988 미국 남콜로라도대학교 산업공학과 교수
 1988~1999 한국전자통신연구원 책임연구원
 1999~2000 영산대학교 멀티미디어공학부 교수
 2000~현재 홍익대학교 전자전기컴퓨터공학부 교수
 관심분야 시스템 시뮬레이션, 컴퓨터 구조, 성능평가