

유무선 연동 웹 클러스터링 환경에서의 QoS 향상을 위한 연구

김 승 수

문 승 진

◆ 목 차 ◆

1. 서 론
2. 관련 연구
3. QoS 클러스터 서버그룹 설계 및 구현
4. 성능 평가
5. 결론 및 향후 연구과제

1. 서 론

인터넷의 급속한 성장은 인터넷 트래픽을 증가시키고 있으며, 이러한 트래픽의 증가로 인하여 웹 서버의 성능과 가용성을 높이는 방안이 더욱 중시되고 있다[2]. 대부분의 인터넷 시스템은 단일서버로 구성되어 있다. 그러나 특정한 서비스의 경우 접속량이 폭주하면 안정된 서비스를 제공하기 어려운 경향이 있다.

본 논문의 연구에서는 이러한 사용자의 증가로 인한 트래픽을 분산할 수 있는 QoS(Quality of Service) 클러스터 서버그룹을 제안한다. 여러 대의 저가형 일반 서버를 QoS 클러스터 서버그룹으로 만들어 서비스의 안정성과 서비스의 질을 높일 수 있고 필요에 따라 서버그룹을 증가시켜 안정된 서비스를 할 수 있는 향상된 QoS 클러스터 서버그룹 시스템을 제안한다.

제안된 시스템은 대학입학 원서접수시스템과 합격자발표 시스템에 적용시켜 보았다[6]. 본 논문의 기법을 적용할 경우 서버의 과부하를 효율적으로 처리하여 보다 신뢰적이고 안정적인 서비스가 가능하다.

2. 관련 연구

본 장에서는 서버의 부하를 분산하여 안정적인

서비스를 제공하는 부하분산의 개요 및 기본 동작에 대한 내용들을 살펴본다. 또한 제안한 QoS 클러스터 서버그룹 기법과도 비교하여 본다[2].

2.1 부하분산의 개요

부하분산이란 하나의 서버로 서비스 요청이 집중되는 것을 여러 대의 서버로 확장 분산하는 것을 말한다. 경쟁력 있는 인터넷 인프라 구축을 하기 위해서 중요한 고려 사항은 확장성(Scalability)이다. 이러한 부하분산 기법은 크게 사용자 측면과 서버 측면에서의 부하분산 기법으로 나누어 볼 수 있다[2,6].

사용자 측면에서의 부하분산 기법은 사용자가 부하분산을 위해 사용되는 웹서버들 중에서 부하가 상대적으로 적게 걸리는 서버를 동적으로 선택하는 방법으로 Bandwidth Probing과 Berkely Smart Client등이 있다. 이는 서버의 부하 정보를 얻을 수 있는 애플릿으로 사용자의 응용프로그램을 바꾸어야 하는 점과 네트워크 부하를 증가시킨다는 단점 때문에 대부분의 연구는 사용자 측면에서의 부하분산보다는 서버 측면에서의 부하 분산 문제에 대해 다루고 있다[2].

서버 측면에서의 부하분산을 사용하는 방법은 RR-DNS기법을 사용하는 NCSA 확장가능 웹서버가 대표적이며, RR-DNS기법과 응용계층 스케줄링 기법을 사용한 것으로는 SWEB, 그리고 IP계층 스케줄링 기법으로는 IBM사의 TCP Router와 Cisco사

* 수원대학교 IT대학 컴퓨터학과

의 Local Director등이 있다[2].

2.2 기존 분산기법과 QoS클러스터의 비교

위에서 설명한 부하분산 기법들의 문제점을 분석해 보면 RR-DNS의 문제점을 해결하기 위해선 응용계층 스케줄링 기법이나 IP계층 스케줄링 기법을 사용해야 하나 이들을 사용하면 분산 웹서버의 확장성이 떨어진다는 문제점이 있다.

사용 기법에 따른 부하 분산 기법과 본 논문에서 제안한 QoS 클러스터 서버그룹의 개선 사항을 비교 정리하면 표 1과 같다[2]. 본 논문의 개선 사항은 QoS 클러스터 시스템을 쉽고 간단하게 Software적으로 구현한 것과 저가형 시스템을 이용한 비용절감을 들 수 있다.

3. QoS 클러스터 서버그룹 설계 및 구현

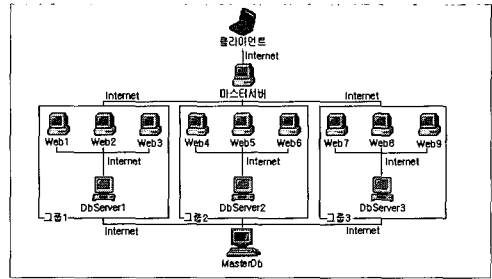
QoS 클러스터 서버그룹에서 부하분산을 담당하는 서버를 마스터서버라 한다.

3.1 QoS 클러스터 서버그룹의 모델구조

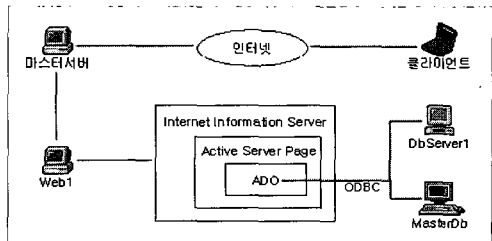
QoS 클러스터 서버그룹 모델에서 엔트리 포인트인 마스터서버는 부하분산을 책임지는 서버이다. 그룹 당 한 개의 데이터베이스가 있고, 모든 시스템에 공통적으로 사용하는 필수 데이터 요소만을 저장하는 마스터 데이터베이스를 가지고 있다. 하나의 서

(표 1) 분산 웹서버의 부하분산 기법 비교

사용기법	문제점	본 논문의 개선사항
RR-DNS	부하 불균형 문제	QoS 클러스터 서버 그룹으로 부하분산 및 저가형 시스템을 통한 비용 절감
응용계층 스케줄링	병목현상 네트워크 트래픽 증가	QoS 클러스터 서버그룹에서 부가 정보 유지 불필요
IP계층 스케줄링	가상서버의 병목현상	기존의 저가형 시스템의 이용을 통한 비용 절감



(그림 1) QoS 클러스터 서버그룹 모델구조



(그림 2) QoS 클러스터 서버그룹의 작동과정

버에 중앙집중방식은 사용자가 집중되었을 경우 그 효율성은 상당히 떨어지게 된다. 따라서 본 논문에서는 QoS 클러스터 서버그룹을 이용한 부하분산 시스템을 구성한다.

그림 1을 살펴보면 최초의 사용자는 알려진 URL을 통하여 마스터 서버에 접속하게 된다.

이후 사용자는 자신이 속한 그룹 중에서 하나의 그룹에서 서비스를 받는다. 그룹은 여러 개로 확장 가능하며, 모든 서버는 인터넷에 연결되어 있다. 데이터의 변경 시 소속 그룹의 서버에 저장하고 필수 요소만 마스터서버에 저장하여 관리한다[2]. 마스터 데이터베이스가 QoS 클러스터 서버그룹의 데이터의 일관성을 유지에 중요한 역할을 한다. 서버의 분산 계획에 따른 데이터베이스 연결모듈은 ASP(Active Server Page)로 구성이 하였으며, 순차적인 방법으로 구성하였다[5].

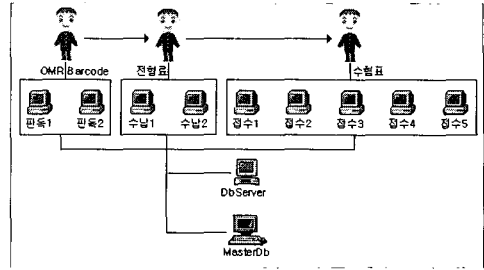
3.2 알고리즘(Algorithm)

QoS 클러스터 서버그룹의 부하분산을 책임지는 마스터서버의 부하분산 알고리즘에는 순차적 방식 알고리즘은 표 2에 보여주고 있다.

(표 2) 순차적(Sequence)방식 알고리즘

```

<%
Set IpsiLogin = Server.CreateObject("ADODB.connection")
Set RsLogin = Server.CreateObject("ADODB.RecordSet")
IpsiLogin.open "IpsiLogin","said","sapasswd"
Sql = "UPDATE counter SET counter = counter + 1"
IpsiLogin.Execute Sql
Sql = "SELECT counter FROM counter"
RsLogin.Open Sql, IpsiLogin, 3
buffer = RsLogin("counter")
Set IpsiLogin = Nothing
Set RsLogin = Nothing
%>
<Script Language="JavaScript">
var rand1 = <%=buffer%>;
var quotes = new Array;
quotes[0]="http://ipsi01.suwon.ac.kr"; //서비스가능서버
quotes[1]="http://ipsi02.suwon.ac.kr";
quotes[2]="http://ipsi03.suwon.ac.kr";
quotes[3]="http://ipsi04.suwon.ac.kr";
quotes[4]="http://ipsi05.suwon.ac.kr";
quotes[5]="http://ipsi06.suwon.ac.kr";
quotes[6]="http://ipsi07.suwon.ac.kr";
quotes[7]="http://ipsi08.suwon.ac.kr";
quotes[8]="http://ipsi09.suwon.ac.kr"; //서비스가능서버
rand1 = rand1 % quotes.length; //서버의숫자로 나눔.
var quote = quotes[rand1];
location.href = quote; //해당서버에서 서비스하도록함.
</Script>
    
```

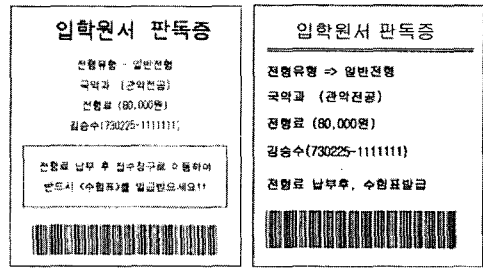


(그림 4) 대학 교내원서접수 시스템의 구조

4.1.2 대학 교내 로컬 원서접수 시스템

교내 원서접수 절차는 아래 그림 4와 같다[4,7].

편독후 수험생을 구별할 수 있는 아래 그림 5와 같은 BARCODE인쇄 용지를 발급 받는다.



(그림 5) OMR 편독 후 발급 받은 BARCODE

수납에서 전형료를 지불하고 접수 단계로 이동한다. 접수 단계에서는 기재내용 확인 후 접수하고 수험표를 발급 받는다.

4.1.3 대학 인터넷 원서접수 시스템

인터넷 원서접수는 현재 가장 많이 사용되며, 계속적으로 늘어나고 있는 추세이다[3,4,7]. 아래 그림 6은 원서접수 화면이다.

4.1.4 대학 모바일 원서접수 시스템

휴대통신과 웹의 통합을 위하여 모바일 원서접수 시스템을 구현하였다[1,8].

모바일 언어인 WML을 사용하였고[1], ASP를 데이터베이스 접속 모듈로 사용하였다[5,8]. 휴대전화와 PDA등의 발달로 인하여 빠른 시일 내에 모바일 원서접수도 보편화되리라 생각된다.

4. 성능 평가

본 실험은 동시 연결된 사용자를 늘려가면서 같은 트랜잭션의 CPU사용률 및 응답시간을 산출해내는 프로그램을 사용하였다. 본 실험에 사용된 서버는 xSeries, Win2000, SQL 2000 이다[7].

4.1 대학 원서접수 및 합격자발표 적용

본 논문에서는 대학 교내 로컬 원서접수 와 인터넷 원서접수 그리고 모바일 원서접수를 하나의 QoS 클러스터 서버그룹내의 데이터베이스 시스템에 통합하여 구현하였다[5,6].

출해 내는 프로그램을 사용하였다. 서버에 요청 응답시간 측정과, 서버 측에서는 Windows 2000 작업 관리자를 이용하여 해당 서버의 CPU사용률 변화를 측정하였다. 웹서버의 요청 건수는 10, 50, 100명에 대하여 측정하였다. QoS 클러스터 서버그룹 기법의 성능을 살펴보기 위하여 일정한 단위시간동안 반응속도와 서버 측에서 사용한 CPU사용률 등을 서로 비교하였으며, 그룹내의 노드의 수와 그룹의 수를 증가시키면서 반응속도를 측정하였고, 측정된 결과의 비교를 통하여 제안한 기법이 효율적인 것으로 나타났다.

4.2.1 접속자의 수를 증가시키면서 측정

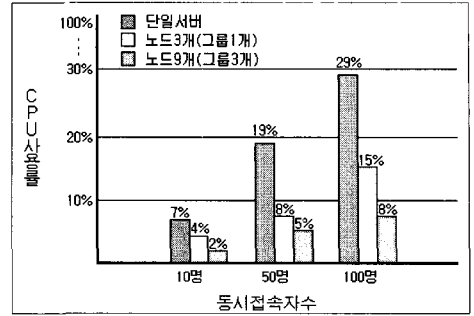
본 논문의 실험은 하나의 데이터베이스에 하나의 웹서버로 구성된 중앙집중방식 서버환경과 QoS 클러스터 서버그룹에서 비교 실험 하였다.

중앙집중방식에서 10명 동시접속 응답속도는 27ms, CPU사용률 7%, 50명은 140ms, CPU사용률 19%, 100명은 279ms, CPU사용은 29% 이었다. 노드3개의 QoS 클러스터에서 10명 동시접속 응답속도는 7ms, CPU사용률 4%, 50명은 43ms, CPU사용률 8%, 100명은 92ms, CPU사용은 15% 이었다. 노드9개의 QoS 클러스터에서 10명 동시접속 응답속도는 3ms, CPU사용률 2%, 50명은 15ms, CPU사용률 5%, 100명은 30ms, CPU사용은 8% 이었다.

위의 그림에서 알 수 있는 바와 같이 100명의 동시 사용자 환경에서의 수행결과는 그 차이가 10명이나 50명의 동시 사용자보다 그 차이가 더 커진다. 이러한 CPU사용률 변화는 동시 접속자 수의 차이에 따라 수행한 결과를 보면 동시접속 사용자의 수가 늘어날수록 해당 서버의 CPU사용률이 증가함을 알 수 있다. QoS 클러스터 서버그룹에서 CPU 사용률이 21%(29%→8%)가량 감소함을 보인다. 응답

(표 5) 100개의 세션 수행결과 (ms)

	min	mean	S. E	median	max
중앙집중방식	20	149	+77.4	149	279
노드3개시	7	53	+22.8	55	92
노드9개시	10	19	+6.8	18	30



(그림 8) 동시 접속자 수에 따른 평균 CPU사용률

속도는 QoS 클러스터 서버에서는 30ms로 249ms (279ms→30ms)가 단축됨을 알 수 있다. 본 논문에서는 QoS 클러스터 서버그룹을 사용하여 성능을 향상 시켰다.

4.2.2 QoS 클러스터 서버그룹의 장점

중앙집중방식 웹서버는 사용자가 늘어남에 따라 서버의 확장을 하는데 문제가 있다. 본 논문에서 제시한 QoS 클러스터 서버그룹은 그러한 단점을 보완하였으며, 요청이 늘어남에 따라 서버를 확장하여 안정적인 서비스를 하리라 기대된다. QoS 클러스터 서버그룹의 장점을 정리해 보면, 시스템을 쉽고 간단하게 Software적으로 구현한 것과 저가형 시스템을 이용한 비용절감을 들 수 있다.

5. 결론 및 향후 연구과제

본 연구에서는 QoS 클러스터 서버그룹을 이용하여 인터넷 서비스의 안정성과 서비스의 질을 향상 시켰다. 이는 대부분의 웹 서비스 형태가 하나의 단일서버에 단일 데이터베이스를 사용하는 중앙집중방식 서버 형태로 이루어져 사용자의 증가에 따라 디스크 입력 및 출력을 발생하는 트랜잭션 상태에서 서버의 효율성이 현저히 저하되는 것을 실험을 통하여 증명하였으며, 이러한 문제점을 해결하기 위하여 하나의 서버로 집중되는 서비스를 다른 서버로 분산하는 분산 데이터베이스 환경의 QoS 클러스터 서버그룹 형태를 제안하였다. QoS 클러스터

서버그룹 환경에서 사용자는 안정된 서비스를 공급 받을 수가 있는 것이다.

특히 대학 원서접수나 합격자 발표의 경우에는 순간에 접속하는 동시 접속자의 수가 많다. 이 논문의 연구에서 부하분산 서버를 설계하여 이러한 폭발적인 요구를 안정적으로 수행할 수 있는 시스템을 만든 것은 접속이 폭주하는 서비스를 효과적으로 대처할 수 있는 시스템으로서 활용 가능성이 높으리라 기대된다.

향후 연구로는 QoS 클러스터 서버그룹 내의 마스터서버를 고장 허용 시스템으로 구축해 마스터 서버의 고장 시 마스터 서버를 대신해서 백업 서버가 서비스를 제공할 수 있도록 하는 것이다. QoS 클러스터 서버그룹 내의 각 노드를 실시간 적으로 모니터링 해 더욱더 효과적인 부하분산 알고리즘을 개발하여 현재의 QoS 클러스터 서버그룹에 적용시키는 것이다.

참고 문헌

- [1] Charles Arehart (방대인, 박성준공역), "Professional WAP", 정보문화사, 2001
- [2] 한광록, 이승원, "데이터 복제 서버를 이용한 학사 관리시스템의 부하 분산에 관한 연구", 정보처리학회논문지 D 제8-D권 제5호(2001.10), pp.1-8
- [3] (주)유웨이, 인터넷 원서접수, <http://uway.com>
- [4] (주)이노에스티, 원서접수, <http://innost.co.kr>
- [5] Francis, "Active Server Page", 정보문화사
- [6] 이상엽, "Visual Basic 6.0 Bible", 영진, 1998
- [7] 고일석, "MS SQL Server 2000", 가남사, 2001
- [8] Open Wave, Mobile Site, <http://openwave.com>

◎ 저자 소개 ◎



김 승 수

1999년 수원대학교 전자계산학과 학사 졸업
 2003년 수원대학교 컴퓨터학과 석사 졸업
 1999년~현재 수원대학교 정보전산원 근무
 관심분야 : 데이터베이스, 부하분산, 로드밸런스, 원서접수, 학사행정, 모바일, Wap



문 승 진

1986년 텍사스 주립대학교(오스틴) 컴퓨터학과 졸업 (학사)
 1991년 플로리다 주립대학교 컴퓨터학과 졸업 (석사)
 1997년 플로리다 주립대학교 컴퓨터학과 졸업 (박사)
 1997년 6월~1997년 8월 (주) 맥시스템 멀티미디어 및 통신 연구소 소프트웨어 실장
 1997년 9월~현재 수원대학교 IT대학 컴퓨터학과 부교수
 관심분야 : 실시간 데이터베이스, 웹기반 데이터베이스, 모바일 데이터베이스, 실시간 스케줄링 등