

모바일 소셜 네트워크 게임을 위한 가상화 기반 분산 게임서버의 설계 및 구현

이원진*, 이택균*, 김강석*, 홍만표*
*아주대학교 대학원 지식정보공학과
e-mail:dna03@ajou.ac.kr

Design and Implementation of Virtualization Based Distributed Game Server for Mobile Social Network Game

Wonjin Lee*, Taekkyun Lee*, Kangseok Kim*, Manpyo Hong*
*Dept. of Knowledge Information Engineering, Ajou University

요 약

스마트폰의 보급으로 모바일 소셜 네트워크 게임(SNG: Social Network Game)을 즐기는 사용자들이 증가하고 있다. 그러나, SNG의 특성에 맞는 자원 활용률을 고려한 효율적인 게임서버에 대한 연구는 매우 부족한 실정이다. 본 논문은 가상화 기반으로 SNG게임서버를 설계 및 구현한다. 또한 가상화 기반 분산 게임서버의 CPU 사용률과 Memory 사용량을 분석하여 게임서버의 자원 활용률을 보인다. 이를 토대로 SNG게임서버 환경구축의 기반지식을 제공한다.

1. 서론

최근 스마트폰의 발전에 따라 모바일 SNG(Social Network Game)는 MMORPG(Massively Multi-player Online Role Playing Game)와 달리 게임의 진행이 클라이언트의 모바일에서 주로 이루어진다. 모바일 SNG게임서버의 주요 역할은 클라이언트의 회원가입을 통한 인증처리와 게임을 진행한 결과 데이터를 전송받아 데이터베이스에 저장하고 그 데이터를 통해 순위 리스트를 구성하여 클라이언트에게 전달하는 것이다.

또한 온라인 게임서버는 대부분 분산서버이다[1]. 한 가지의 기능을 여러 서버로 나누어 부하를 분산하는 방식과 서로 다른 역할 별로 서버를 분산하는 방식으로 게임서버를 구성한다.

이에 덧붙여 하드웨어가 발전함에 따라 고성능의 서버를 게임에 활용하고 있다. 하지만 대표적인 x86 서버는 지속적인 성능 향상으로 CPU 성능에 비해 낮은 활용률을 보이고 있다[2]. 이에 대한 해결책으로 여러 대의 물리적 시스템을 하나로 통합해주는 서버 가상화(Server Virtualization) 기술을 활용한 시스템 구축이 보편화되고 있다[3,4,5].

따라서 본 논문에서는 게임서버의 가상화를 통해 게임서버의 자원낭비 문제점을 해결하고, 모바일 SNG의 특성에 맞는 분산 게임서버의 설계 및 구현을 통해 SNG게임서버 환경구축의 기반 지식을 제공한다. 본 논문의 구성은 다음과 같다. 2장에서는 서버 운영의 변화 및 기존 온라인 분산 게임서버에 대해 분석한다.

3장과 4장에서는 가상화 기반 분산 게임서버를 설계 및 구현한다. 또한 자원 활용률에 대한 실험을 통해 5장에서 결론을 맺도록 한다.

2. 관련 연구

이 장에서는 서버 운영의 변화, 온라인 게임 서버의 특징 및 온라인 게임에서의 분산처리를 위한 서버 구조에 대해 살펴본다.

2.1 서버 운영의 변화

서버 운영의 형태는 서버가 등장한 초기부터 현재까지 크게 4가지 형태로 변화했다고 볼 수 있다. 첫 번째는 공인 IP주소 기반 단일 서버 운영이다. 두 번째는 공인 IP주소 기반 다중 서버 운영이다. 세 번째는 사설 IP기반의 다중 서버 운영이다. 마지막 네 번째로는 가상화 기반 다중 서버 운영이다. 하나의 물리적 서버에 여러 대의 가상 서버로 구성하는 방법으로 물리적 서버로 인한 공간 문제를 줄일 수 있다. 또한 서버의 이용도에 따라 유연하게 자원을 할당할 수도 있고, 여러 대의 분산 서버 구입비용도 낮출 수 있다[2].

2.2 온라인 게임서버의 특징

온라인 게임서버 설계를 위한 특징은 다음과 같이 정리해 볼 수 있다[6,7].

- 1) 온라인 게임 진행을 위해 로그인서버, 데이터베이스 서버, 게임서버 등으로 나뉘어 각각의 역할을 하게 된다.
- 2) 게임서버에서 동기화 매커니즘이 필요하다. 클라이언트에 의한 명령이 처리되는 과정에서 동기화가 필요하다.

본 연구는 지식경제부 및 정보통신산업진흥원의 “고용계약형 SW석사과정 지원사업”의 연구결과로 수행되었음.

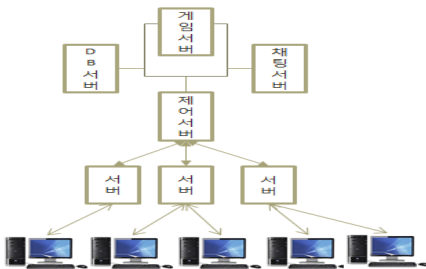
게임의 실시간성, 동기화 수준 정도 등을 파악해야 한다.

- 3) 게임서버는 프로세싱이 많은 실시간 시스템이기 때문에 데이터베이스 관리 시스템에 덜 의존적이어야 한다.
- 4) 게임과 연계해야 하는 웹 서비스를 지원해야 한다.

2.3 온라인 게임에서의 분산 처리를 위한 서버 구조

온라인 게임에서의 분산처리를 위한 서버 구조는 크게 부하분산형, 중앙집중형, 맵서버방식, 대칭서버방식 등이 있다. 이 중 부하분산형은 여러 대의 서버가 하나의 게임 세계를 이루는 방식으로 클라이언트를 여러 곳에 분산하여 접속시킨 다음 서버가 이를 제어하는 방식의 형태를 말한다[8,9].

본 논문에서는 기존에 제안된 그림 1의 부하분산형 방식을 SNG 특징에 맞게 그림 2와 같이 자원 활용률을 고려하여 1대의 가상화 서버에 기능별로 분산된 게임서버를 설계하였다. 제어서버가 인증과 분배처리를 하고, 게임접수와 순위리스트는 3대로 구성된 게임서버에서 처리함으로써 처리 속도를 향상시키고, 서버의 부하를 분산시킴으로써 시스템의 안정성을 확보할 수 있다.



(그림 1) 부하 분산형

3. 게임서버 설계

이 장에서는 모바일 SNG게임서버의 개발 환경, 전체 시스템 구조 및 메시지 처리 절차 순으로 살펴본다.

3.1 개발 환경

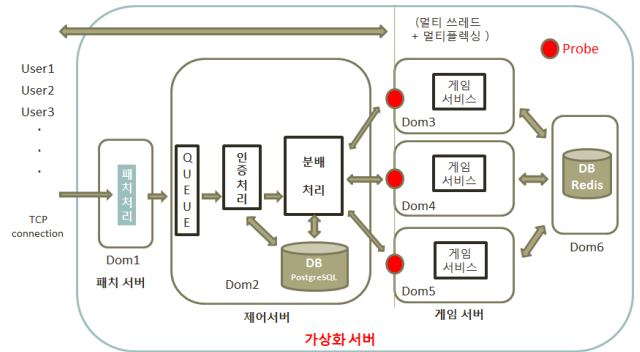
본 논문에서는 표1과 같은 환경에서 개발 및 구현하였다. SNG게임서버를 개발하기 위해 Linux Ubuntu 12.04 LTS를 기본으로 Xen[10]을 이용하여 반 가상화 환경을 구축하였다.

<표1> 개발 환경

Client	OS	Android 2.3.3
	Tool	eclipse
	Language	Java 1.7
Server (Xen)	OS	Linux Ubuntu 12.04 LTS
	Tool	vi, vim
	Language	C
	인증 DB	PostgreSQL
	게임 DB	(NoSQL) Redis

3.2 전체 시스템 구조

전체 시스템 구조는 다음과 같다.



(그림 2) 전체 시스템 구조

3.2.1 Client

클라이언트는 TCP(Transport Control Protocol) 프로토콜에 의해 패치서버를 거쳐 다음에서 설명하는 제어서버에 접속한다.

3.2.2 Control Server

- 1) QUEUE : 접속을 요청한 클라이언트는 개인 식별을 위하여 쓰레드를 형성한다. 그 후 인증처리를 위하여 접속한 순서대로 QUEUE에 저장되어 FIFO(First In First Out) 형태로 처리된다.
- 2) 인증처리 : PostgreSQL 데이터베이스에 저장된 아이디, 패스워드와 비교 연산을 처리한다.
- 3) 분배처리 : 클라이언트는 실시간으로 게임서버에 접속하고, 게임을 종료한다. 접속과 종료시점마다 그 정보를 게임서버에서 제어서버에 전송한다면 제어서버의 부하가 가중될 수 있다. 이에 본 논문에서는 각각의 게임서버로부터 5초 단위로 클라이언트 수를 전송 받아 인증된 클라이언트를 현재 클라이언트 수가 가장 적은 게임서버로 연결을 처리한다.

3.2.3 Game Server

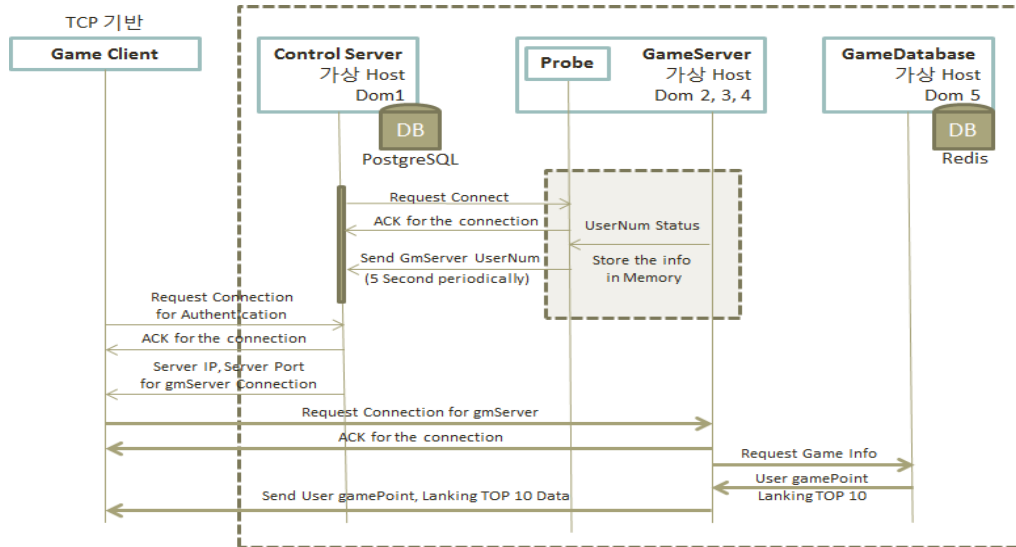
게임을 진행할 수 있도록 설계된 서버로 프로브(probe) 모듈을 구성하여 현재의 서버 상태(클라이언트 수)를 5초 단위의 주기로 제어서버에 값으로 전송한다. 또한 게임 진행시 필요한 게임로직을 처리하고, 게임데이터베이스와의 통신을 위한 메시지 및 Query를 전송한다.

3.2.4 GameDatabase Server

인증을 위한 관계형 데이터베이스로 PostgreSQL[11]을 사용한 반면, 게임 데이터베이스는 SNG게임의 특성에 맞게 순위 리스트와 포인트 점수 등을 빠르게 처리하기 위하여 key와 value로 구성된 Redis[12] 데이터베이스를 사용하였다.

3.3 메시지 처리 절차

클라이언트가 제어 서버에 접속을 요청하면 제어 서버에서 아이디와 패스워드를 비교하여 인증처리를 한다. 그 후 인증된 클라이언트에게 현재 가장 클라이언트수가 적은 게임서버의 IP와 PORT번호를 메시지로 전송한다. 그 결과로 클라이언트는 해당 게임서버로 접속을 요청하여 접속을 완료한다. 접속이 완료된 직후에는 기존에 제안된 게임서버의 처리절차[7]와 달리 게임데이터베이스(Redis)가 있는 가상호스트로 현재 접속한 클라이언트의 게임점수 및 순위 리스트를 요청하여 그 결과 값을 클라이언트에게 전송한다.



(그림 3) 메시지 처리 절차

4. Game Server 구현

4.1 클라이언트 인터페이스

SNG는 그림 4와 같이 실제 게임을 할 수 있도록 구현하였다. 클라이언트가 모바일에서 서버로 인증을 요청하여 서버에서 인증처리가 완료된 후 게임을 진행 할 수 있다. 랜덤으로 발생하는 숫자와 동일한 숫자를 인터페이스 상에서 선택하여 지우는 방식으로 진행시간에 반비례하여 게임점수가 반영된다. 게임이 끝나면 데이터베이스 서버에 게임점수를 전송하여 기존 게임점수와 비교하여 최고 점수일 경우에만 데이터베이스에 저장한다. 그 후 순위리스트를 생성하여 클라이언트에 전송하게 된다.



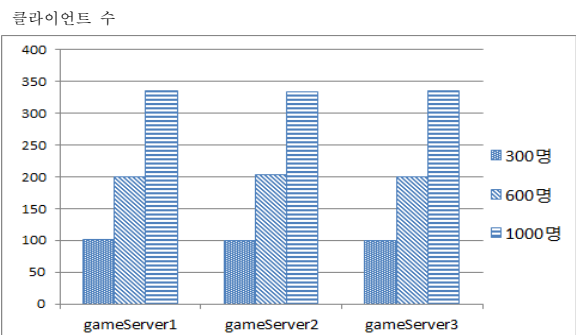
(그림 4) 클라이언트 인터페이스 구현 화면

4.2 Control Server

클라이언트 300명, 600명, 1000명을 가정하여 게임서버에 연속적으로 접속하여 SNG게임을 하는 시뮬레이션을 50회 반복하여 구현하였다.

기존에 제안된 라운드로빈(Round Robin) 방식으로 클라이언트를 분배할 경우에는 분배된 게임서버에서 게임을 진행하던 클라이언트가 게임을 종료할 경우를 반영하지 않고 클라이언트를 분배하기 때문에 특정 게임서버에 클라이언트가 집중되어 특정 게임서버에 과부하가 발생할 수 있다.

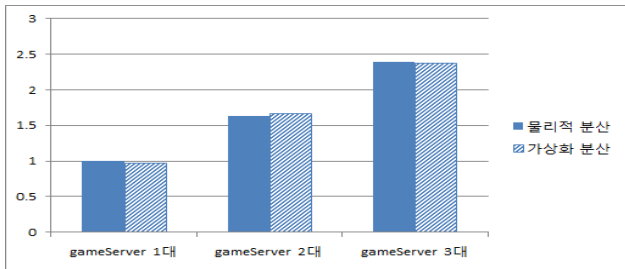
이에 본 논문에서 제안한 분배방식은 게임서버로부터 주기적으로 5초 단위로 전송된 클라이언트 수에 따른 분배방식으로 그림 5에서 보듯이 접속하는 클라이언트수가 300명, 600명, 1000명일 경우 각각 게임서버에 고르게 분산됨을 알 수 있다.



(그림 5) 클라이언트 수에 따른 분산 그래프

4.3 Game Server

게임서버에 클라이언트 300명을 가정하여 게임서버 1대에 300명일 경우, 2대에 각각 150명씩 분산되었을 경우, 3대에 각각 100명씩 분산되었을 경우를 가정한 시뮬레이션을 50회 반복하여 구현하였다. 그림 6은 게임서버 1대에서 300명이 동시에 게임을 진행할 경우의 처리속도를 1로 두었을 때 게임서버 2대일 경우와 3대일 경우의 상대적 처리속도를 비교하였다. 또한 가상화 기반 분산 게임서버와 물리적 게임서버 3대로 구성된 분산 게임서버를 비교하여 실험한 결과 처리속도에는 큰 차이가 없음을 알 수 있다.



(그림 6) 처리속도

4.4 서버의 자원 활용률에 따른 가상 머신의 할당

Xen 기반으로 내부에 생성하는 가상 머신의 개수는 CPU 사용률과 부하량을 검사하여 할당하였다[13].

4.5 자원 활용률 분석

본 논문에서는 물리적으로 분산된 여러 대의 게임서버가 아닌 한 대의 서버에 Xen 반가상화 기술을 이용하여 총 5대의 리눅스 기반 호스트를 각각 기능에 맞게 분산서버로 활용하였다.

<표2> 물리적 서버 1대의 환경

CPU	Intel 2.3GHz
Memory	4GB
HDD	500GB

SNG의 특성에 맞게 가상의 클라이언트 1000명이 연속적으로 접속하여 자신의 게임점수를 전달 받고, 순위 리스트를 게임 데이터베이스에서 전달 받는 시뮬레이션을 가상화 기반 게임서버와 물리적인 5대의 서버로 구성된 환경에서 100회 구현하여 표3, 표4와 같은 결과를 얻었다. MMORPG와 달리 SNG는 게임의 진행이 클라이언트의 모바일 상에서 이루어지기 때문에 표3에서 보듯이 클라이언트 수에 따른 메모리 사용량은 큰 변화가 없음을 알 수 있다. 따라서 SNG에 있어서는 CPU사용률과 Memory사용량 중 CPU의 사용률에 따른 분산게임서버의 구성이 중요함을 알 수 있다. 또한 표4에서 보듯이 클라이언트의 접속빈도가 높은 SNG게임서버의 구축에 있어서 제어서버의 CPU 활용률이 가장 높기 때문에 가상화 기반 SNG게임서버의 안정적 운영을 위해서는 제어서버에 CPU 자원 할당을 높이고, Memory는 게임서버 1,2,3에 접속시킬 최대 클라이언트의 수에 따라 할당을 높이는 것이 필요하다.

<표3> 가상화된 게임서버의 자원 활용률

접속자 수	0명		1000명	
	CPU 사용률(%)	Memory 사용량	CPU 사용률(%)	Memory 사용량
자원 활용률	6.31	3.2G	60.39	3.3G

<표4> 물리적으로 분산된 게임서버의 자원 활용률

접속자 수	0명		1000명	
	CPU 사용률(%)	Memory 사용량	CPU 사용률(%)	Memory 사용량
제어서버	0.61	667M	56.39	669M
게임서버 1, 2, 3	0.61	667M	4.05	708M
게임DB	0.63	667M	1.52	667M

5. 결론

스마트폰의 보급으로 모바일 소셜 네트워크 게임을 즐기는 사용자들이 증가하고 있다. 그러나, 현재 SNG의 특성에 맞는 자원 활용률을 고려한 효율적인 게임서버에 대한 연구는 매우 부족하다. 이에 본 논문에서는 SNG게임서버 환경구축의 기반지식을 제공하였다. 또한 SNG게임서버 구성에 있어서 기능 별로 게임서버를 물리적으로 여러대로 구성하기 보다는 서버의 자원을 분석하여 가상화 기반으로 구성한다면 자원의 낭비를 막고 SNG에 보다 적합한 게임서버를 구축하는데 많은 도움이 되리라 여겨진다.

향후에는 가상화 기반 SNG게임서버의 안정적 운영을 위해 메모리 및 CPU의 자원활용률 뿐만 아니라 Disk I/O, Network 대역폭, Load Average 등의 게임서버의 부하를 고려한 SNG게임서버에 대한 연구를 진행할 것이다.

참고문헌

- [1] 장수민 외, “효율적인 MMORPG 분산 게임서버”, 한국콘텐츠학회논문지 Vol.7 No.1, 2006.
- [2] 정성재 외, “보안성이 강화된 리눅스 가상화 기반 효율적인 웹 시스템 연구”, 보안공학연구지, 제7권 제4호, pp. 335-339, 2010.
- [3] 이효, “서버 가상화 개요 및 활용방안”, 정보과학회지 제 26권 제10호, 2008.
- [4] 이효, “가상화 기술 어디까지 와 있나”, micro software, pp. 162-165, 2006.
- [5] A white paper from IBM, “가상화 기술 백서-IBM Systems Agenda와 가상화 엔진을 중심으로”, 2006.
- [6] 임정열 외, “온라인 게임서버 기술의 분석 및 전망”, 정보처리학회지 제12권 제6호, pp.60-68, 2005.
- [7] 엄남경 외, “게임 서버간 부하의 균일성에 기반한 부하 분산기 설계 및 구현”, 한국통신학회 논문집, '07-3 Vol.32 No.3, 2006.
- [8] 신동일 외, “온라인 게임 서버를 위한 부하 분산 미들웨어의 구현에 대한 연구”, 한국정보과학회 추계 학술발표 논문집, Vol.27, No.2, 2000.
- [9] 노홍식, “온라인 게임 환경에서 네트워크 분산처리기술”, 전자공학회지 제28권 제7호, pp. 807-811, 2001.
- [10] Xen, <http://en.wikipedia.org/wiki/xen/>
- [11] PostgreSQL, <http://www.postgresql.org/>
- [12] Redis, <http://redis.io/>
- [13] 정성재 외, “서버의 가용성 분석에 관한 연구”, 한국 지식정보기술학회 논문집 제6권 제4호, 2011.