

분산 네트워크 게임 서버의 설계 및 구현

김기남, 김상국, 박정환, 신동일, 신동규, 노용덕

세종대학교 컴퓨터 공학과

A Design and Implementation of the Distributed Network Game Server

Kinam Kim, Sangkuk Kim, Jungwhan Park, Dongil Shin, Dongkyoo Shin, Yongdeok Noh

Department of Computer Engineering Sejong University

요 약

온라인 네트워크 게임은 인터넷을 이용하여 동시에 수백명 이상이 접속하여 게임을 즐길 수 있으므로 초고속 통신망의 실현이후 고부가가치를 창출하는 미래 핵심 산업으로 각광 받을 전망이다. 그러나, 서버에 접속하는 사용자의 수가 크게 증가함에 따라 나타나는 많은 문제점들이 발생하고 있다. 본 논문에서는 기존 네트워크 게임 서버의 구조적 단점을 보완한 3-tier 방식의 분산 네트워크 게임 서버를 설계하고 구현하였다. 이 시스템은 클라이언트/서버 모델의 2-tier방식에서 오는 클라이언트 수 증가에 따른 네트워크 부하 가중에 대한 처리 문제와, 분산된 2-tier방식을 적용한 서버에서 각 서버간의 통신, 데이터 공유에 대한 문제 등을 해결하기 위하여 3-tier방식을 적용하여 구현되었다. 또한 각 서버의 접속자나 이동되는 데이터를 모니터링하는 미들웨어 시스템을 구현하였다.

1. 서론

오늘날 컴퓨터의 빠른 보급과 하드웨어의 발달로 인해 게임은 급속도로 확산되고 있다. 또한 멀티미디어의 도입으로 컴퓨터 게임은 고급화의 계기를 맞이하였고 인터넷이 대중화되면서 네트워크 환경으로 게임을 즐기게 되었다. 네트워크 게임의 장점은 사람과 사람의 대결이다. 컴퓨터의 정형화된 패턴에 의한 게임이 아닌 다양한 사람들의 사고에 의해 긴장되고 예측이 불가능하기 때문이다. 또한 네트워크 게임은 여러 사람들이 모여서 하나의 사이버 공간을 창출할 수도 있다. 사이버 공간 내에서 클라이언트는 가상 사회를 형성할 수 있을 뿐만 아니라 공간

내에서 나름대로의 인간관계를 맺을 수도 있다. 이러한 네트워크 게임은 TCP/IP(Transmission Control Protocol/Internet Protocol)기반의 게임과 IPX(Internet Network Packet Exchange)기반의 게임으로 나눌 수 있다. 전자는 연결지향 프로토콜을 사용하며 하나 이상의 서버를 필요로 한다. 각각의 서버는 하나 이상의 클라이언트의 접속을 처리할 수 있어야 한다. 이러한 서버 구현은 다수의 클라이언트 동시 접속에 의한 네트워크 부하 가중에 대한 해결책을 제시하지 못하고 있다. 또한 하나 이상의 서버를 가지고 있다면 각각의 서버는 접속한 클라이언트를 각 서버에게 적절하게 분배하여 네트워크 부하를 최소화해야 한다. 또한 각 서버들간의 데이터 공유 문제도 해결되어야 한다. 후자는 LAN(Local Area

Network)기반에서 사용되며, 서버가 없는 방식이며, 디아블로, 스타크래프트[1] 등이 이에 속한다. 본 논문에서는 분산 네트워크 게임 서버의 설계를 다루고 있으며, 이는 인터넷 기반 TCP/IP를 이용하여, 네트워크 부하에 따른 사용자 수 제한에 중점을 두고 해결책을 모색하여, three-tier 방식을 적용한 분산 네트워크 게임 서버 엔진을 제안한다. 또한 제안한 서버의 구조, 기능, 장점과 기대효과에 대해 논의한다.

2. 인터넷을 이용한 분산 네트워크 게임 서버 구조

현재 일반적으로 많이 사용되고 있는 방식은 2-tier방식으로서, 클라이언트와 서버간에 인터넷망을 통해 일대 다 관계를 유지한다. 이러한 방식을 사용하는 구조방식을 중앙 집중형 구조(단일 서버 구조)라 한다. 또한 LAN에서와 같이 어느 특정한 서버도 존재하지 않을 뿐 아니라, 어느 특정한 역할만 하는 클라이언트도 존재하지 않는 구조 형태가 있는데 이를 분산 서버 구조라 한다. 이는 서버가 클라이언트가 될 수도 있고, 클라이언트가 서버가 될 수도 있다.

분산 서버 구조는 중앙 집중형 서버 구조와 달리 서버가 한 대 이상일 때 필요하다. 네트워크 게임을 즐기는 사용자들이 급증하면서, 단일 서버로만 운영되는 시스템이 한계에 달하게 되었다. 이러한 이유로 분산 서버구조가 제안되었다[4]. 분산 서버 구조는 2개 이상의 서버로 구성되어 있으며, 이 서버들은 서버들간에 데이터 공유가 되는 방식과 데이터 공유를 할 수 없는 방식으로 나뉜다. 즉 대칭 서버(Replicated Server)와 비대칭 서버(Non-Replicated Server)로 나뉜다.

이 서버 구조는 단지 동일한 서버구조를 가진 시스템을 2개 이상 연결하여, 클라이언트로 하여금 서버에 접속할 수 있는 사용자 수를 늘리는 방법중의 하나이다.

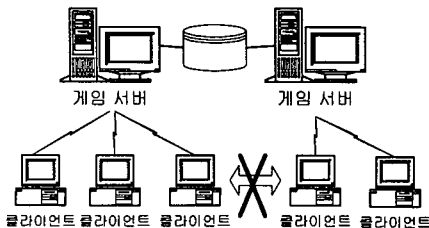


그림 1 대칭 서버 구조

그림 2의 구조 형태는 시간 동기화(Clock Synchronization)가 발생하며, 서버들간의 교착 상태(Deadlock), 게임 데이터의 무결성(integrity)와 동시성(Consistency)이 발생할 수 있다.

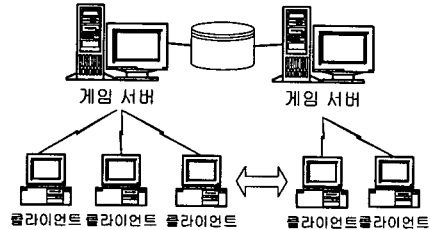


그림 2 비대칭 서버 구조

이 서버 구조는 대칭 서버와는 달리 서버들간의 데이터 공유가 가능하다. 하지만 클라이언트가 특정 서버에 많이 접속할 경우가 발생할 수 있으며, 대칭 서버와는 달리 서버간의 부하가 완전히 균등하게 분산되지는 않는다는 단점이 있다.

3. 분산 네트워크 게임 서버 구조

본 논문에서 구현한 서버 구조는 3-tier방식을 적용한 분산 네트워크 게임 서버 구조로서, 미들웨어를 통해 각각의 서버에 분배된 클라이언트들에 대한 데이터를 가지고 있다. 또한 각각의 서버들간에 데이터를 공유하기 때문에 동기화에 대한 문제를 해결할 수 있으며, 게임 미들웨어를 통해 클라이언트가 접속이 됨으로서 네트워크에 대한 부하나 서버 엔진의 부하를 조절할 수 있다. 즉 대칭 서버 구조방식의 장점인 서버들간의 부하 균등 분배와 비대칭 서버 방식이 가진 서버들간의 데이터 공유성을 가진 서버 시스템이다. 이 제안한 서버 구조는 4.1에서 설명할 분산 시스템을 응용한 게임 서버 시스템이다.

이 시스템의 전체적인 구조는 그림 3과 같다. 이 분산 네트워크 게임 서버는 Windows NT를 기반으로 Visual C++를 이용하여 제작되었다. 게임의 원활한 동작을 위하여 사용자 정보 및 기타 게임정보를 저장하는 데이터베이스는 리눅스 환경의 오라클 8.0을 이용하여 구현되었다. 기본적인 네트워크 게임 시스템은 분산 미들웨어를 탑재한 서버 1대, 데이터베이스를 위한 서버 1대, 게임 서비스 서버 1대로서 구성된다. 따라서 사용자 부하 증가에 따라서 게임 서비스 서버만의 증설로 문제 해결이 가능하도록 구현되었다.

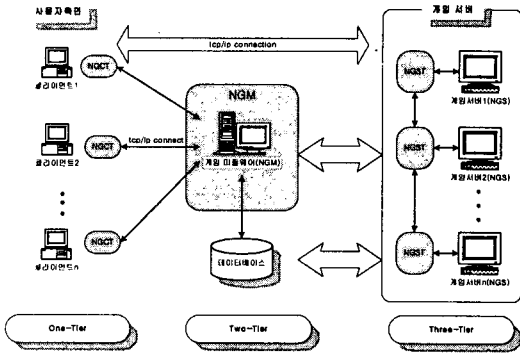


그림 3 분산 네트워크 게임 서버의 전체 구조

3.1 분산 게임 엔진을 위한 미들웨어 분산 미들웨어 구조는 그림 4와 같다.

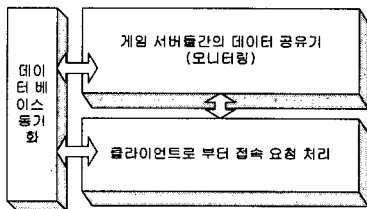


그림 4 분산 미들웨어 구조

클라이언트는 서버와 접속되기 전에 미들웨어를 통해 자신의 ID로 접속을 하며, 미들웨어는 접속한 클라이언트에 대해 사용자 인증을 거친 다음 접속할 게임 서버의 IP(Internet Protocol)를 가진 토큰을 보낸다. 이 IP는 게임 서버 시스템의 인터넷 주소가 되며, 이 시스템의 IP부여는 미들웨어에 의해 네트워크 부하나 서버 엔진의 부하에 따라 달라지게 된다. 그리고 나서 클라이언트는 그 IP를 통해 서버와 연결을 요청하고, 서버는 그 요청으로 공유된 데이터베이스에서 접속 ID를 가진 클라이언트의 Map 자료를 가지고 오며, 그에 대한 데이터를 클라이언트로 보낸다. 또한 접속한 클라이언트에 대해 다른 서버들에게 시스템 동기화를 위해 토큰을 보내게 된다. 이 미들웨어는 다수의 클라이언트의 요청을 받아 서버와 연결작업을 할 수 있으며, 서버들간의 데이터 공유에 대한 감시 기능과 적절한 부하 조절 기능을 가진다.

구현된 분산 미들웨어는 대칭 서버의 장점과 비대칭 서버의 장점을 모두 가지고 있다. 접속하는 클라

이언트의 수에 따라 게임 서버의 수가 조절 가능하며, 서버 모니터링 기능에 의해 서버의 Idle상태를 감시 할 수 있다[7]. 또한 데이터 베이스를 게임서버와 공유함으로써 서버들간의 클라이언트 동기화 문제를 해결 할 수 있다.

또한, 게임 클라이언트 프로그램의 손쉬운 작성을 위하여 네트워크 관련 부분을 DLL로 묶어서 제공한다. 이 DLL은 서버소켓을 생성하여 클라이언트로부터 받은 메시지를 처리하는 기능을 포함하며, 각종 메시지 및 데이터를 클라이언트로 전송해주는 역할과 동기화의 기능도 포함한다.

4. 결론 및 향후 발전 방향

본 논문에서는 사용자 수 증가에 의한 네트워크 게임 서버의 부하에 대한 해결책을 제시하였으며, three-tier방식을 적용한 네트워크 게임 서버와 이를 위한 미들웨어를 제안했다. 이 미들웨어는 클라이언트와 서버간의 중간단계이므로 이 둘의 네트워크 연결을 원활하게 하고 클라이언트와의 메시지 처리 문제의 공유와 서버와의 메시지 처리 문제가 있어서 네트워크 속도에 따라 클라이언트간의 동기화에 대한 해결책이 필요하다. 또한 사용자 ID와 패스워드의 네트워크 전송에 관한 보안 문제에 대한 처리가 개선되어야 한다.

참고 문헌

- [1] BattleNet, <http://www.battle.net>
- [2] 울티마온라인, <http://www.owo.com>
- [3] 바람의 나라, <http://www.nexon.co.kr>
- [4] 신 동원, "첨단기술 네트워크 게임을 위한 운영체제 연구", pp1-13.
- [5] 정 상원, "Game on the Net : 네트워크 게임의 세계로 가자", 마이크로소프트웨어, 1997.12.
- [6] 고 옥, "첨단 게임 기술 동향", pp7-11, 정보과학회(A), 제 15권 제 8호(통권 제 99호), 1997.8.
- [7] 김형도, 문지영, "멀티미디어 네트워크 게임을 위한 그룹통신 플랫폼", 정보과학회논문지(A), 제 24권 제 2호, 1997.2.