

클라이언트/서버 기반의 네트워크 게임에 관한 연구

백현옥*, 이승혁*, 변영기*, 이한권*, 박병수**, 조태경*

*상명대학교 대학원 컴퓨터 정보 통신공학과

**상명대학교 컴퓨터시스템공학과

e-mail:roll23@nate.com

A Study on Network Game based on Client/Server

Hyun-Ok Baek*, Seung-Hyuk Lee*, Young-Ki Byun*, Han-Kwon Lee*,
Byoung-Soo Park**, Tae-Kyung Cho*

*Dept of Computer Information Communications, Sangmyung University

**Dept of Computer System, Sangmyung University

요 약

본 논문에서는 네트워크 게임을 통하여 Client/Server 환경의 전반에 대해 알아볼 것이며, 네트워크 게임의 한 예로 사다리 게임을 구현하였다. 사다리 게임은 JAVA를 이용하였으며 이를 통해 네트워크 게임의 실행 원리를 분석하고 이를 통하여 Client/Server의 분산 환경에 대해 고찰한다.

1. 서론

오늘날 인터넷의 빠른 보급과 네트워크의 발전에 따라 인터넷의 영향력이 매우 커져가고 있다. 거기에는 Client/Server의 분산환경이 기여하였으며 이에 따라 Client/Server환경은 컴퓨팅 환경의 주요요소라 할 수 있겠다. 또한 멀티미디어의 도입으로 네트워크게임의 시장도 크게 성장하고 있으며 이는 게임시장이 2003년 대비 15.8%의 높은 성장률을 보였고 그 중 네트워크게임이 66.3%를 차지하고 있다는 것에서 알 수 있다. 일반적으로 네트워크게임은 분산환경의 구조 방식을 유지하고 있다. 분산환경 기술은 지역적으로 떨어져 있는 사용자들이 실시간으로 상호작용 할 수 있도록 설계된 가상의 공간을 말한다. 이러한 분산환경 기술에 대한 연구는 궁극적으로 기존의 인터넷 환경과 운영체제의 모습을 변화시켜 나갈 것이다. 따라서 본 논문에서는 분산환경과 분산처리를 위한 Client/Server구조에 대해 알아보하고자 하며 플랫폼에 독립적인 JAVA로 개발한

사다리 게임을 통하여 이해를 돕고자 한다. 본 논문의 구성은 다음과 같다. 2장에서는 분산 네트워크 Client/Server 기능 알아보고 3장에서는 구현한 네트워크 사다리 게임 대해 알아볼 것이다. 끝으로 4장에서 결론을 맺는다.

2. 분산네트워크 Client/Server

클라이언트는 다른 프로그램에게 서비스를 요청하고, 서버는 그 요청에 대해 응답을 한다. Client/Server개념은 단일 컴퓨터 내에서도 적용될 수 있지만, 네트워크 환경에서 더 큰 의미를 가진다. 네트워크상에서 Client/Server모델은 여러 다른 지역에 걸쳐 분산되어 있는 프로그램들을 연결시켜주는 편리한 수단을 제공한다. 클라이언트/서버 모델은 네트워크 컴퓨팅의 주요 개념이 되었다. 오늘날 만 들어지고 있는 대부분의 업무용 프로그램들은 클라이언트/서버모델을 적용하고 있으며, 인터넷의 주요 프로그램인 TCP/IP 또한 마찬가지다. 인터넷의 경

우를 예를 들면 웹 브라우저는 인터넷상의 어디가에 위치한 웹서버에게 웹 페이지나 파일의 전송을 요구하는 클라이언트 프로그램이다. 일반적인 Client/Server 모델에서는, 보통 데몬(daemon)이라 불리는 서버프로그램이 먼저 활성화된 상태에서 클라이언트의 요구사항을 기다리는데, 대체로 다수의 클라이언트 프로그램이 하나의 서버 프로그램을 공유한다.

2.1 Client/Server환경

클라이언트/서버 환경은 프로세서(클라이언트)와 공유 호스트 프로세서(서버) 사이의 공동 멀티프로세싱 환경이다. 개방형을 지향하므로 상호 독립적으로 작업을 하면서도 정보가 고립되는 정보섬(Information Island)이 발생하지 않는다.

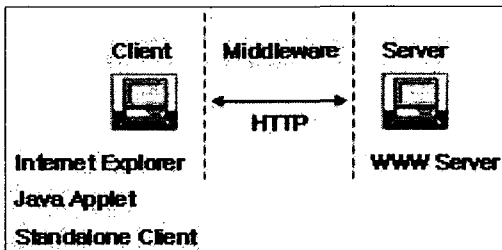


그림 1. Client/Server 환경

3. 네트워크 기반의사다리 게임

본 논문에서는 분산처리환경의 Client/Server구조로 네트워크 사다리 게임을 구현하였다. 개발언어로는 JAVA를 사용하였으며, JAVA는 인터넷 또는 네트워크를 통하여 효율적으로 수행될 수 있도록 http, TCP/IP 프로토콜 등과 연관된 라이브러리를 제공하기 때문에 분산환경에 적합하다고 할 수 있다.

3.1. 네트워크 통신

소켓은 두 개의 프로그램이 하부 네트워크에 신경 쓰지 않고 통신하기 위한 방법론을 제공해 주는 도구이다. 즉, 소켓은 네트워크의 통신 개념을 전부 내부에 포함하고 있는 간단한 메커니즘을 제공해 주고, 이 소켓이 연결되면 소켓을 통해서 양방향 통신이 이루어진다. 데이터를 주고받기 위해 통신하려는 두 개의 프로그램 사이에 채널이 형성되고, 이 채널을 통해서 양방향 통신이 이루어진다. 이러한 채널이 존재하기 위해, 서버 프로그램은 자신의 포트 번호를 통해 들어오는 요청을 계속 듣게 되며, 포트 번호로 요청이 들어왔을 때 이를 받아 해당서비스를 수행한다. 그림 2에다 TCP 소켓을 이용한 네트워킹에 대해 나타내었다. 먼

저, 서버는 4444번 포트에 대해 요청이 들어오나 계속 듣게 되며, 이때 클라이언트는 4444번 포트를 통해 서버에게 서비스를 요청할 것이다. 마찬가지로, 서버도 서비스를 수행한 후, 그 결과를 클라이언트가 사용하는 포트 1234를 통해 전달하게 된다. 이렇게 서버 프로그램과 클라이언트 프로그램은 각각 사용하는 포트를 연결함으로써 하나의 논리적인 채널을 생성하게 된다.

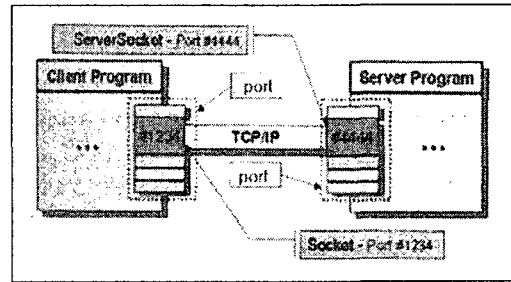


그림 2. TCP 소켓을 이용한 네트워킹

3.2. 사다리게임 Client/Server

네트워크 사다리 게임은 기본적으로 크게 서버프로그램과 클라이언트 프로그램으로 구성되어 있다. 게임을 실행시키면 소켓통신을 위하여, 주어진 포트번호의 서버소켓이 생성된다. 이 서버소켓은 단순히 기다리고 있다가 클라이언트 소켓과의 통신을 위해 서버측의 소켓을 생성해 주는 역할을 한다. 클라이언트에서는 서버의 IP, 포트번호, 그리고 ID를 가지고 서버에 접속한다. 포트번호는 0부터 1023번까지는 Well Known Port라고 해서 시스템에 예약된 Port이며 1024부터 65535까지의 포트는 사용자가 임의로 사용할 수 있다. 본 사다리 게임에서는 포트번호로 4001을 부여하였다. 클라이언트에서 서버소켓으로의 접속을 시도하며, 클라이언트 소켓의 접속이 서버에서 수락되면, 서버는 클라이언트와의 통신을 하기 위해서 서버측의 소켓을 생성한다. 서버에서는 클라이언트에서 보내오는 메시지를 받아, 이것을 다시 모든 클라이언트로 보내주기 위한 스레드를 생성하고, 클라이언트도 역시 서버에서 보내오는 메시지를 계속 받아 화면에 출력하기 위한 스레드를 생성한다. 서버는 클라이언트에게 같은 모양의 사다리를 뿌려주며 클라이언트들이 사다리를 타기 시작하면 그 결과를 클라이언트로 전송하여, 클라이언트의 스레드를 통해서 프레임에 출력하게 한다. 사다리게임의 구조를 그림 3번에 개략적으로 나타내어 보았다.

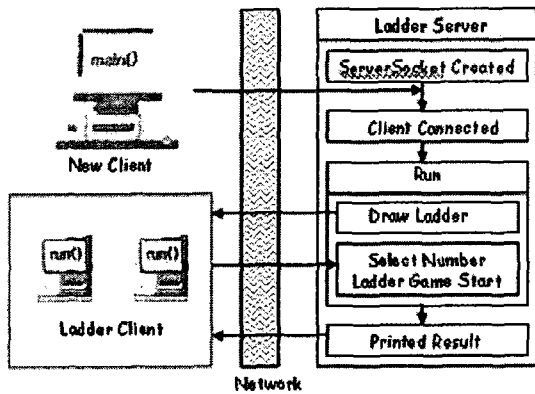


그림 3. 사다리게임의 구조

3.3 사다리 게임의 실행

클라이언트 프로그램을 실행시키면 서버 컴퓨터와 연결이 된다. 접속된 순서대로 사다리의 순번이 정해지고 서버는 접속된 IP각각에 같이 사다리를 뿌려주게 된다. 클라이언트 프로그램에는 참여자 정보와 IP, ID, 결과 등이 출력되며 참여자는 Start버튼을 눌러 게임을 실행하게 된다.

사다리 게임을 실행한 화면을 그림 4에 나타내었으며 이것은 총 5명이 접속해 실행한 것이다.

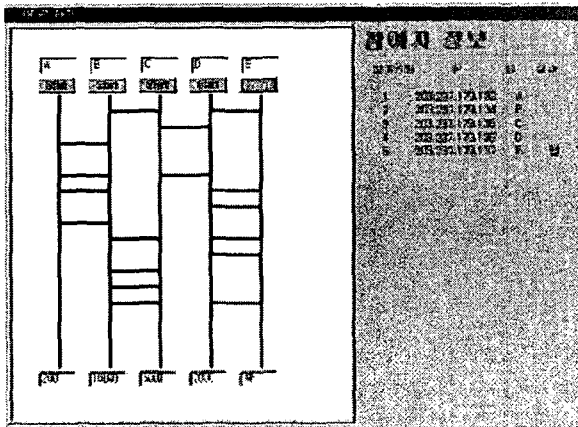


그림 4. 사다리게임 실행 화면

3.4 사다리게임의 중요 알고리즘

사다리 게임의 가장 중요한 부분은 왼쪽 오른쪽에 길이 있는지 여부를 확인하고 가까운 쪽으로 이동하는 것과, 좌우에 길이 없으면 아래로 이동하는 것을 해결하는 부분이라 할 수 있다. 사다리 게임은 가로선과 세로선이 존재한다. 세로선은 선택하게 되는 선이고 가로선은 좌우로 이동하는 선이 되겠다. 그러므로 세로선과 가로선을 구분해주어야 하며 세로선은 짝수로 가

로선은 홀수로 구분을 하였다. 또한 가로선은 게임이 실행될 때마다 랜덤으로 그려주어야 한다. 가로선이 그려진 곳을 $Dim[][] = 1$; 이라고 정의를 해두고 $Dim[][]=1$ 이 되는 값을 Random클래스와 $abs(a)$ 함수 등을 사용하여 임의로 정의 해준다. 그림 5에 가로선을 랜덤 하게 그려주는 code를 나타내었다.

```

Random randomNum = new Random();
int a,b;
int c=0;

for(int k = 1 ; k < end ; k += 2 ){
    a = randomNum.nextInt() * 5;
    b = Math.abs(a)+1;

    int d = 0;

    while(c <= b) {
        d = random.nextInt() * 15;
        d = Math.abs(d)+1;

        if(k>1) {
            if(Dim[k-2][d] != 1) {
                if(Dim[k][d] != 1) { Dim[k][d] = 1; c++; }
            }
        } else { if(Dim[k][d] != 1) { Dim[k][d] = 1; c++; } }
    }
    a=0; b=0; c=0;
}
    
```

그림 5. 랜덤하게 그려주는 가로선

4. 결론

본 논문에서는 분산환경에 대해서 고찰하였고 Client/Server기반의 네트워크 게임을 JAVA를 사용하여 구현하였다. Client/Server을 통하여 자료 및 자원을 공유할 수 있으며 이기종간의 분산형의 확대, 객체지향적 프로그래밍의 사용이 용이해질 것이다. 분산환경시스템은 네트워크에 의해서 서로 연결되는 여러 개의 프로세서들과 네트워크 자체를 말하는 것으로 이 시스템은 특별한 목적을 가지고 서로 결합된 형태로 설계 되었다. 향후 인터넷을 기반으로 분산처리 기술과 객체 지향 기술을 통해 인터넷 환경 또는 분산컴퓨팅 환경의 규모가 더욱 증가할 것이고, 다양한 응용 서비스가 요구 될 것이다.

참고문헌

[1] 김형도, 문지형, 멀티미디어 네트워크 게임을 위한 그룹통신 플랫폼, 정보과학논문지(A), 제24권 제2호, 1997.2
 [2] 최우영, 정정민, 김도원, 최성, "네트워크 게임의 다중 시스템 운영체제에 관한 연구", 한국 게임학회 동계 학술발표 대회 논문집, pp401-406, 2002

- [3] Elliotte Rusty Harold, Java Network Programming, 한빛출판사, 1999.
- [4] 만호기, 자바 데이터베이스 Programming, SamGakHyung Press, 1998
- [5] Park, J. and Park, C. Development of a Multiser & Multimedia Game Engine Based on TCP/IP. Proceedings of the 1999. IEEE Pacific Rim Computers and Signal Processing. August, 1997.
- [6] Berger, J.E.; Dinh, L.T.; Masiello, M.F.; Schell, J.N., "NVR: a system for networked virtual reality," Proceedings of the International Conference on Multimedia Computing and Systems, p. 37-44, Boston, MA, May 1994.
- [7] Macedonia, M.R.; Zyda, M.J.; Pratt, D.R.; Brutzman, D.P.; Barham, P.T., "Exploiting reality with multicast groups: a network architecture for largescale virtual environments." Proceedings. Virtual Reality Annual International Symposium. p. 2-10, Research Triangle Park, NC, March 1995.