

원격제어에 의한 강의 지원 시스템의 구현

문해룡, 김진현, 박정구, 김병철, 이호영, Ya-lin Wu, 권순각
동의대학교 컴퓨터소프트웨어공학과

Implementation of Lecture Support System by Remote Control

Haeryong Moon, Jinhyun Kim, Junggu Pak, Byungchul Kim, Hoyoung Lee, Ya-lin Wu, Soon-kak
Kwon

Department of Computer Software Eng., Dongeui University

E-mail : dragom23@nate.com

요 약

학교의 컴퓨터 실습실 환경에서 컴퓨터들의 교육용 관리, 제어 시스템으로 교육의 질적 성장욕구에 적극적으로 대응하고, 교육환경을 효과적으로 지원하여 보다 관리하기 편리한 실습실 환경을 구축하기 위한 강의지원 시스템을 구현한다. 시스템의 기능으로 학습자의 화면을 실시간으로 모니터링할 수 있어 수업상황을 관리, 감독할 수 있다. 그리고 원격제어를 이용하여 특정 학습자와 1:1 개별학습이 가능하다. 또한, 강사의 실습화면을 학습자 컴퓨터로 전송, 전원제어, 채팅, 실습 컴퓨터 Lock 기능 등이 있다. 이 시스템으로 인한 기대효과로 수업의 집중화, 인적자원의 낭비 최소화, 시간적 자원의 낭비 최소화, 편의성 증대 등을 들 수 있다.

키워드

강의지원, 원격제어, Lock

1. 서 론

강의지원 시스템은 컴퓨터 실습환경에서 강의를 보다 효율적으로 진행하기 위하여 필요하다. 기존 강의지원 시스템의 경우 학생 컴퓨터에서 교수 컴퓨터로 다중 접속되어 교수 컴퓨터에서 학생컴퓨터의 화면을 monitoring하고, 파일 전송, 채팅의 기능이 포함되어 있었다[3].

본 강의지원 시스템은 기존의 강의지원 시스템의 단순한 기능을 보다 다양하게 구현하여 더욱 효율적인 강의 지원이 되도록 구현하였다.

기존의 기능에 추가적으로 쪽지전송기능, 역 monitoring기능으로 학생 컴퓨터에서 교수 컴퓨터의 화면을 볼 수 있도록 구현하였다. 이는 좁은 배치된 실습실 환경에서 사각지대에 위치한 학생들은 빔 프로젝트로 표시되는 화면 중 일부분을 앞사람에 의해 가려지거나 다른 모니터에 의해 가려져 보지 못할 경우를 해결하기 위하여 개발하였다[4].

강의지원 시스템을 통하여 교수는 강의를 좀 더 원활하게 할 수 있고, 학생은 좀 더 수업에 집중할 수 있다.

2. 강의지원 시스템 구성

2.1 서버 시스템 구성

그림1은 서버시스템의 구조를 나타낸다.

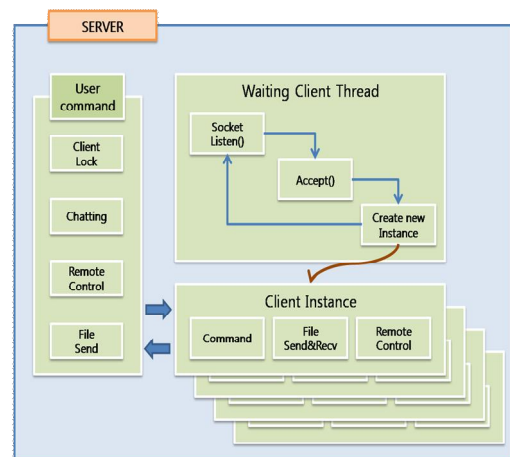


그림 1. Server System Architecture

서버시스템은 사용자의 명령을 전달하는 User Command와 Client의 접속을 처리하고 각 Instance의 생성을 담당하는 Waiting Client Thread, 그리고 Waiting Client Thread에서 생성된 인스턴스인 Client Instance로 구성된다.

User Command는 Client의 화면을 제어할 수 있는 Client Lock, Chatting, 원격제어, 파일송신의 명령을 Client Instance에게 전달할 수 있도록 통신을 담당한다.

Waiting Client Thread는 Client와의 통신을 담당한다. Client의 접속 요청이 들어오면 요청을 승인해주고, 각 Client를 담당하는 Client Instance를 생성하여Client의 요청을 처리한다.

2.2 Monitoring System

Monitoring System은 Client로부터 현재 학생이 사용하고 있는 화면을 전송 받아 실시간으로 Server에서 확인할 수 있도록 해주는 역할을 수행한다. Client는 컴퓨터 실습실을 기준으로 하여 40여개의 Client시스템이 동시접속을 할 것이라 예상하여 Server System을 설계하였다. 40여개의 Client System으로부터 동시에 화면 캡처 데이터를 전송받으면 시스템 부하가 발생할 것으로 예상하여 데이터의 크기를 최소화할 필요성이 있다. Server System에서 Client의 화면은 default 값으로350 * 300(mm) 사이즈에 표시된다. default 사이즈가 작으므로 기본 전송데이터는 이 사이즈에 맞도록 데이터의 크기를 줄여서 전송한다.

Server System을 사용하는 User가 Client의 화면을 확대하여 보기를 원할 경우 해당Client의 데이터를 FullScreen 사이즈에 맞도록 크기를 늘려 전송한다.



그림 2. Client 모니터 화면 확인

2.3 1:1 원격제어

1:1 원격제어 기능은 교수와 학생 사이에서 실습장 장애사항 해결과 학생의 질의, 요청에 대한 지원을 제공함으로써 강의 지원에 대한 효율적인 인터페이스 역할을 수행한다.

1:1 원격제어 기능의 구현에 있어서는 .NET Remoting 즉, 원격지의 컴퓨터에 접속하여 해당 컴퓨터의 자원을 이용하는 기술인 RPC (Remote Procedure Call)를 이용한다. 이는 클라이언트에서 원격서버 (Remote Server)에 존재하는 원격 객체(Remote Object)의 함수를 호출하여 그 결과를 얻어내는 기술이다. 결과적으로 .NET Remoting은 원격서비스에 등록되어 있는 원격 클래스의 함수를 호출하기 위한 아키텍처 (Architecture)이다.

클라이언트에서 원격 객체 (Remote Object)의 함수를 호출했을 때 함수 자체는 원격 서버에서 호출되기 때문에 원격 서버의 CPU를 사용한다. 만약 매개변수가 필요하다면 클라이언트에서 매개변수를 네트워크로 넘겨주게 되며, 이것을 넘겨받은 원격 객체는 함수를 호출하여 작업을 처리한 후, 리턴 값이 존재한다면 이것을 다시 클라이언트에게 넘겨주게 된다. 실제적인 일은 원격 서버에서 하기 때문에 클라이언트는 원격 서버에게 필요한 정보만을 전송한 후 서비스를 요청한다. 그리고 원격 서버는 이를 처리하여 클라이언트에게 그 결과만을 전송하는 서비스 (Service) 개념을 포함하고 있다.

.NET Remoting에서 사용되는 기술에는 다음과 같은 기술이 있다.

1. 원격객체 (Remote Object)
2. 원격 객체를 대신하는 클라이언트의 프록시 객체 (Proxy Object)
3. 마샬링 (Marshaling), 언마샬링 (Unmarshaling)
4. 직렬화 (Serialization), 역직렬화 (Deserialization)
5. 네트워크 통신을 하기 위한 채널 (Channel)
6. 네트워크로 전송되는 데이터를 인코딩하는 포맷터 (Fomatter)

원격 서비스를 하기 위해서 서버에는 원격 클래스와 원격 서버가 존재해야 한다. 그리고 원격 서버는 클라이언트와 통신을 하기 위해서 서버 채널 (Channel)을 사용한다. 채널에서 전송되는 메시지들은 적절한 인코딩을 거쳐 통신상에서 전송되어질 데이터로 변환하게 된다. 이 때 인코딩을 담당하는 것이 포맷터 (Fomatter) 이다.

클라이언트는 원격 객체를 원격으로 핸들하기

위해서 원격 객체를 참조할 수 있는 프록시 (Proxy) 객체가 있어야 한다. 일반적인 프로그램의 경우 객체의 참조 값으로 모든 작업을 할 수 있다. 하지만 원격 시스템에서는 클라이언트와 서버 사이를 넘나드는 참조 값을 이용하게 되는데, 클라이언트에 존재하는 이 원격 참조 값을 프록시 (Proxy)라고 한다. 클라이언트에서도 원격 서버와 통신하기 위해서 클라이언트 채널 (Channel)을 가지고 있으며, 채널로 전송되는 메시지 또한 적절한 포맷터 (Formatter)를 통해서 인코딩 된다.

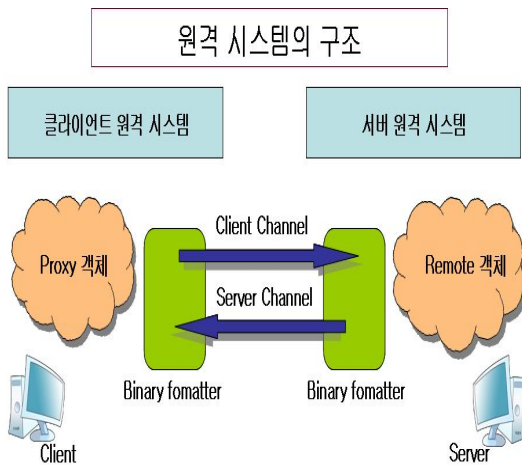


그림 3. 원격시스템의 구조

채널 (Channel)은 원격 시스템의 구조에서 데이터의 전송을 담당하고 있으며, 채널에 포맷터가 등록되며, 등록된 포맷터에 의해서 채널로 전송되어지는 모든 데이터들은 인코딩되고 디코딩된다. 1:1 원격제어에서는 TcpChannel을 사용하며 TcpChannel은 내부에서 TCP프로토콜을 이용해서 통신을 하고 있으며 Binary포맷터를 이용하게 된다.

원격 서버에서 원격 서비스를 하기 위해서는 먼저 채널을 생성한 후 채널을 원격 서비스에 등록한다. TcpChannel을 사용하며 서비스할 포트가 9009일때 채널생성은 다음과 같다.

```
TcpChannel channel = new TcpChannel(9009);
```

생성된 채널을 원격 서비스에 등록할 때에는 ChannelServices의 스테틱 함수인 RegisterChannel()을 이용한다.

```
ChannelServices.RegisterChannel(channel, false);
```

원격 서비스와 채널을 등록한 후 원격 객체를 생성하기 위한 클래스를 등록한다. 클래스를 등

록하기 위해서 RemotingConfiguration의 스테틱 메서드인 RegisterWellKnownServiceType()을 이용한다. 이 때 함수의 매개변수로 원격 서비스에 필요한 여러 가지 정보들을 지정하면 된다.

클라이언트 프로그램에서는 메타데이터 (원격 객체를 dll 형태로 컴파일)를 필요로 하는데 이는 클라이언트에서 프록시 (Proxy)를 만들기 위해서 원격 클래스의 메타데이터가 필요하다. 클라이언트 프로그램에서는 먼저 원격 서비스에 채널을 등록한 뒤, 프록시 객체를 생성하기 위해서 Activator.GetObject()를 호출한다. 이때의 매개변수는 원격 객체의 타입정보와 서버의 URL을 준다.

닷넷 리모팅에서 기술은 위에서 언급했으며, 아래는 닷넷 리모팅을 바탕으로 1:1 원격제어를 위한 방법에 대해 언급한다.

1:1원격제어를 위해서는 원격 서버의 화면을 클라이언트에게 실시간으로 전송해야 하며 또한 클라이언트는 전송되어 보여지는 원격 서버의 화면상에 원격 서버에 Keyboard, Mouse와 같은 제어 장치를 통해서 Input명령을 내려서 원격서버가 동작되게 한다. 이러한 작업들을 원격 클래스로 구현하여 원격객체로 등록하게 되면 클라이언트는 이 원격객체를 통해 원격 서버에 Mouse, Keyboard를 통한 입출력 명령이 가능해진다.

원격 클래스 상에는 화면을 캡처를 담당하는 메서드, 그리고 캡처한 이미지의 전송을 위해서 직렬화 시키는 메서드, 그리고 Keyboard, Mouse Key값을 매개변수로 받아 WINAPI 이벤트를 사용자 호출해주는 메서드가 Interface로 구현되어 원격 서버가 이를 상속하고 구현해서 클라이언트에게 서비스 해준다.

클라이언트는 타이머를 두어 실시간으로 원격 서버의 화면을 캡처하고 직렬화 시켜 TcpChannel을 통해 이미지를 전송받는다. 전송받은 이미지는 PictureBox등을 통해 화면에 뿌려지고 사용자는 그 위에서 Keyboard, Mouse의 제어명령을 내린다. 사용자로부터 입력받은 입력명령은 각 이벤트 메서드를 통해 각 Key 값과, 좌클릭, 우클릭, 더블클릭 등을 판별하게 되고, 미리 정의된 상수 값을 매개변수로 하여 원격 객체의 이벤트 메서드를 호출한다. 매개변수를 넘겨받아 호출된 이벤트 메서드는 WINAPI함수를 호출하여 원격 서버를 동작시키게 된다[1].

2.4 Lock module

Lock 기능은 Client 컴퓨터에서 사용자가 컴퓨터를 사용하지 못하도록 막는 역할을 수행한다. 이는 수업시간 수업외적인 행동을 제약하는 경우 사용하는 기능으로 Client 컴퓨터에선

화면이 교수컴퓨터의 수업화면으로 전환되고, 이 화면을 강제로 빠져나가지 못하도록 윈도우의 특수키 (윈도우키, Ctrl+Alt+Del, Ctrl+Esc, Alt+F4 등)기능을 키보드 후킹기능을 통하여 막는다.

Keyboard Hooking은 사용자가 발생시킨 message event가 Message Queue에서 윈도우가 기전에 이를 가로채기해서 event를 발생하지 않게 된다.

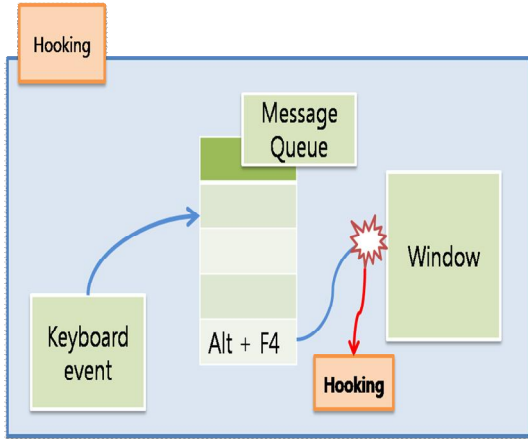


그림 4. Hooking 알고리즘

3. 구현 결과

강의지원 시스템을 구현한 결과는 아래 그림과 같이 각각의 Client화면을 실시간으로 Monitoring 할 수 있도록 구현되었다.

Client 가 접속시 해당 position의 화면이 Client의 화면으로 표시되고, 사용자는 각각의 Client화면을 확대하여 볼 수 있다.

파일전송 버튼을 이용하여 전 Client 에게 동시 파일전송이 가능하다. 원격제어, 채팅, Client Lock 기능을 사용할 수 있는 버튼이 구현되어있다[2].

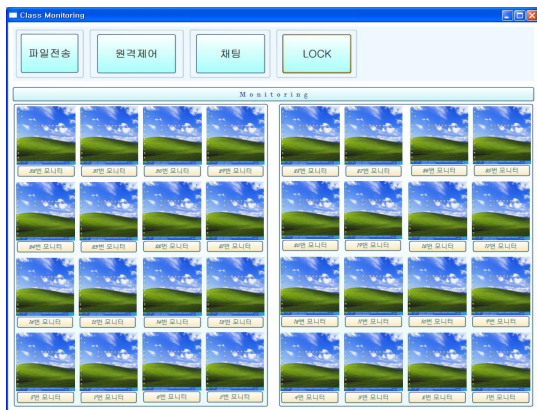


그림 5. 실행 화면

4. 결 론

강의지원 시스템을 통해 학생들의 화면 실시간 모니터링으로 수업상황을 관리, 감독할 수 있다. 그리고 원격제어를 이용하여 특정 학생과 1:1 개별학습이 가능하다.

이 시스템으로 통해 수업의 집중화, 인적자원의 낭비 최소화, 시간적 자원의 낭비 최소화, 편의성 증대 등의 기대효과를 얻을 수 있다.

참고문헌

- [1] 최영관저, 소설같은 C# Second Edition, Jabook, 2008.
- [2] 애덤 네어선, WPF 엔리쉬드, ITC, 2008.
- [3] 김혁구, "수준별 교육을 위한 멀티미디어 강의지원 시스템 개발", 멀티미디어학회지, Vol 11, No4, 2007.
- [4] 엄우용, "대학교수자의 웹기반 수업지원시스템 활용도 사례 분석", 한국교육정보미디어학회, Vol. 14, No. 2, 2008.