

동시 작업 가능한 파워포인트 플러그인 개발

신윤호, 이정호, 이정화, 손진현
한양대학교 컴퓨터공학부

e-mail: friendsl@hanmail.net, dsuri84@hotmail.com, jwlee@database.hanyang.ac.kr,
jhson@hanyang.ac.kr

Powerpoint plug-in editing at the same time

Yun-ho Shin, Jung-ho Lee, Jung-wha Lee, Jin Hyun Son
Department of Computer Science and Engineering, Hanyang University in Ansan

요 약

현재 Powerpoint는 팀 프로젝트에도 불구하고 PPT파일을 혼자서 만들고 수정해야 하면서 한 사람이 많은 시간을 소비해야한다. 또는 여러 명에서 각각의 페이지를 따로 만들고 한 사람이 각자가 만든 페이지를 받아 하나의 파일로 합쳐야하는데 이 방법은 비효율성과 불편함이 존재한다. 따라서 하나의 PPT파일을 여러 명에서 동시에 접속하여 각자의 페이지를 만들면서 시간도 절약하고 불편함과 비효율성을 제거하기 위하여 이 프로그램을 제안하였다.

키워드 : Powerpoint, Multi, 파워포인트, 멀티, 다중

I. 서론

WithPPT는 여러 명이 동시에 파워포인트 문서 파일에 접근하여 읽기, 쓰기가 가능한 프로그램으로 PPT제작이 많은 학생들을 타겟으로 하는 프로그램이다. 기존의 파워포인트 문서 작성은 여러 사람이 같이 작업을 할 수 없어서 만드는 사람을 제외하고는 작업 중간에 작업 내용을 확인 할 수 없으며, 분배 작업시 파워포인트의 디자인과 레이아웃 합병에 많은 시간이 소모되기 때문에 작업을 진행하기 어려운 형태로 되어있다. WithPPT는 이러한 파워포인트 작업의 불편함과 비효율성을 없애고 동시에 작업을 할 수 있는 환경을 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 WithPPT 프로그램에 대해서 알아본다. 3장에서는 프로그램의 중요 알고리즘에 대해서 알아보고, 4장에서는 상업적인 활용방안에 대해서 알아보고, 마지막으로 5장에서는 논문의 결론을 맺는다.

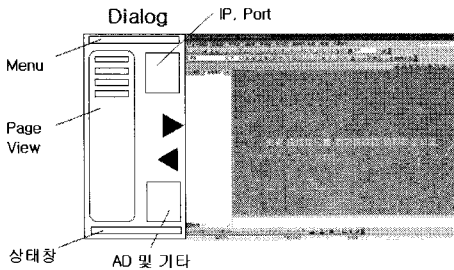
II. WithPPT 프로그램 구조

이 장에서는 WithPPT 프로그램의 외부 구조와 내부 구조에 대해서 알아본다. 2.1절에서는 WithPPT의 외부 Interface에 대해서 알아보고, 2.2절에서는 프로그램 내부의 데이터 전송 구조에 대해서 알아본다. 그리고 2.3절에서는 Server와 Client에서의 파일 관리에 대해서 알아보고 마지막으로 2.4절에서는 예상되는 문제점과 해결방안에 대해서 알아본다.

2.1 WithPPT 외부 Interface

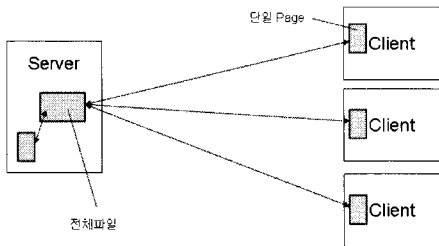
Powerpoint 프로그램을 왼쪽 편에 두고 사용 할 수 있는 연계 가능한 플러그인 형식으로 제작되었으며 기존 PPT 프로그램의 모든 기능을 그대로 사용가능하며 동시 작업을 위한 서버로의 전송과 받아오는 부분은 왼쪽의 다이얼로그 창을 통해서 이루어진다[1]. 왼쪽의 다이얼로그는 아래 <그림1>과 같이 구성되어 있다. 가장 위의 Menu부분을 통해 Dialog에서 새로운 PPT파일을 열거나 새로운 페이지를 추가하고 서버

에서 사용되는 주요 기능들을 포함하고 있다. IP, Port부분은 클라이언트에서 서버에 접속하기 위해 IP 주소와 Port번호를 써서 해당되는 서버로 접속하는 부분이며 서버의 경우는 접속 되는 Port만이 구성되어 있다. 또한 이곳에는 버튼을 두어 서버에 있는 전체파일을 다운받아 전체의 모양을 확인 할 수 있도록 한다. 중심의 화살표는 Save 및 Load기능의 버튼으로 PPT 프로그램을 향한 화살표 그림을 통해 Server에서 Client쪽으로 PageView에서 선택한 Page를 전송해오는 load기능을 수행하며 Dialog 방향의 화살표는 현재 PPT에서 작업한 부분을 서버로 전송하는 Save기능을 수행 한다. PageView는 현재 PPT의 존재하는 Page를 출력하며 이를 선택하여 화살표를 통해 선택한 페이지로 Save와 Load를 할 수 있다. 제일 아래의 상태 창은 현재 서버와 클라이언트와의 작업하고 있는 상황을 실시간으로 보여준다. 마지막으로 옆에 AD및 기타 위치를 두었는데 그곳은 현재 화면에서는 따로 구성되어 있지 않지만 차후에 광고를 도입 하거나 특수한 기능이 필요할 때 사용할 수 있는 여유 공간으로 되어있다.



〈그림 1〉 WithPPT의 외부 interface

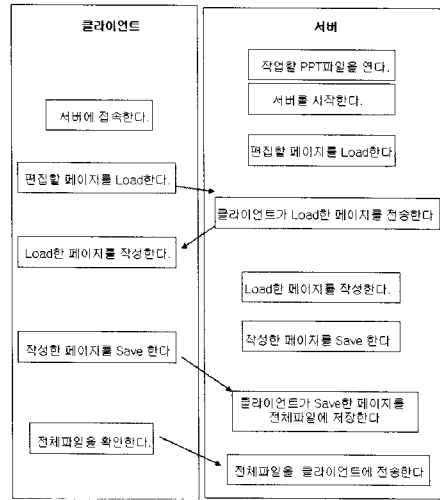
2.2 내부의 전송 구조



〈그림 2〉 Server와 Client의 Page 주고받기

〈그림2〉에서 보면 서버에서 처음에 작업할 전체 파일을 열고 서버를 시작한다. 그 후에 클라이언트는 서버의 IP 및 Port를 입력하여 서버에 접속한다. 클라이언트가 서버에 접속하면 서버에서 연 파일의 페이지가 옆 페이지 뷰에 출력된다. 클라이언트는 작업할 페이지를 선택하고 Load를 누르면 전체파일에서 선택한 단일페이지지만 클라이언트로 전송된다. 이 때 다른 클라이언트가 작업 중인 페이지를 선택해서 Load 하면 읽기 전용으로만 Load가 된다. 각자의 클라이언트는 전송받은 페이지를 보고 그 페이지의 PT작업을 시행한다. 각자의 클라이언트는 작업이 끝나면 Save버튼을 눌러 서버에게

단일페이지를 전송하고 서버는 전송받은 단일 페이지를 전체 파일의 해당 페이지에 붙여넣기한다. 클라이언트가 전송받은 페이지를 수정하지 않을 시에는 NoSave버튼을 눌러 처음의 단계로 돌아갈 수 있다. 서버 또한 하나의 클라이언트로서 작업하고 그 작업 내용을 전체파일에 적용시킬 수 있다. 클라이언트는 작업을 수행하고 나서 전체 페이지의 상태를 확인하기 위해 서버에게 전체 페이지를 요청하여 전체파일을 전송 받고 이를 확인할 수 있다. 아래의 〈그림3〉은 이와 관련된 서버와 클라이언트의 순서이다.

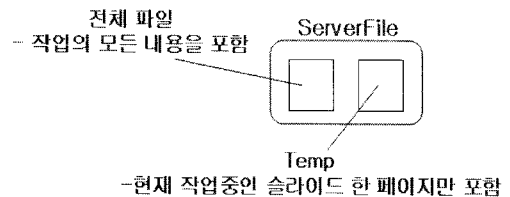


〈그림 3〉 서버와 클라이언트의 순서

2.3 Server와 Client의 파일 관리

WithPPT는 'C:\WithPPT\' 폴더를 기본으로 하고 있으면 폴더 안에 서버의 경우는 WithPPTServer폴더, 클라이언트의 경우는 WithPPTClient폴더로 나누어져 있다.

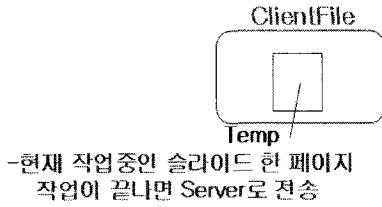
서버는 WithPPTServer폴더 안에 〈그림4〉와 같이 ServerFile폴더를 가지고 있으며 이 폴더 내에서 전체파일과 Temp파일을 가지게 된다. 전체파일은 프로젝트 내에서 작업할 전체 PPT파일이 되며 클라이언트에서 전송된 파일을 저장하는 공간이기도 하다. Temp파일은 서버가 클라이언트가 되어 작업한 내용을 일시적으로 저장하고 Temp의 내용을 전체파일로 통합시킨다.



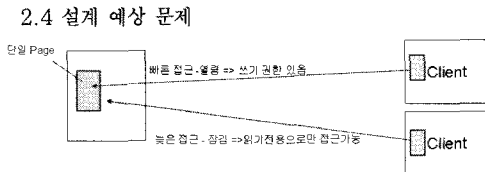
〈그림 4〉 Server 파일 관리

클라이언트는 WithPPTClient폴더 안에 〈그림5〉와 같이 ClientFile폴더를 가지며 이 폴더 내에는 Temp파일이 존재하고 클라이언트가 서버로부터 전송 받은 단일페이지의 임시

적인 저장공간이 된다. 클라이언트는 이러한 Temp파일을 수정하여 이 파일을 서버로 전송하여 자신의 작업 내용을 서버에게 전송하게 된다.



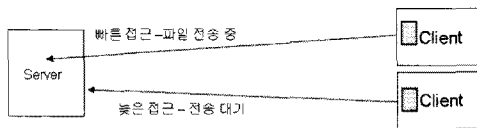
〈그림 5〉 Client 파일 관리



〈그림 6〉 서로 다른 Client의 같은 Page 접근 경우

해당 시스템에서 가장 중요하면서 문제시 되는 부분을 찾는다면 클라이언트가 동시에 같은 파일을 선택한 〈그림6〉과 같은 때의 문제이다. 이 문제를 해결하기 위해 WithPPT는 다른 클라이언트가 이미 load한 페이지의 경우 페이지뷰에서 '(수정중)'으로 표시되며, 더 늦게 수정하기 위해 파일에 접근한 클라이언트에 있어서는 해당 페이지를 읽기전용으로만 읽을 수 있도록 하여 페이지의 중복 수정을 방지 하였다.

위의 문제와 비슷하게 일어날 수 있는 문제가 접속한 클라이언트로부터 동시에 전송되는 파일의 〈그림7〉의 문제이다. 서버를 향해 다수의 클라이언트에서 동시의 파일을 전송하게 되면 파일이 내용이 겹쳐지게 되어 하나의 내용만 저장되거나 시스템의 오류를 일으킬 수 있다. 이에 대한 대처방안으로 한 클라이언트가 서버에 파일을 전송하고 있는 순간에는 다른 클라이언트로부터의 파일 전송을 사전에 차단하는 방법이 있다(2). 이러한 세마포를 이용하여 서버로 동시에 데이터를 전송함으로써 생기는 오류를 차단할 수 있다.



〈그림 7〉 다수의 Client가 서버에게 동시에 데이터를 전송하는 경우

III. 제품의 구현 내용 및 테스트 방법

이 장에서는 WithPPT의 세부 알고리즘 및 테스트 환경에 대해서 알아본다. 3.1절에서는 Powerpoint Control에 대해서 알아보고, 3.2절에서는 Socket Programming 에

대해서 알아본다. 3.3절에서는 테스트 환경 및 최소 환경에 대해서 알아본다.

3.1 Visual C++을 이용한 Powerpoint Control

이 프로그램을 설계시 가장 고민되었던 내용이 PowerPoint의 기능을 그대로 유지한 채로 작업한 내용을 전송하고 여는 프로세싱이었다. 처음에는 PowerPoint에서 지원하는 VSTO 즉 'Visual Studio Tool for Office'라는 Microsoft에서 지원된 오피스 플러그인 Toolkit을 사용할 예정이었으나 위 프로그램은 최신의 기술로서 이에 대한 자료가 거의 전무하고 제대로 된 도움말조차 찾을 수 없어 내용을 수정해야 했다. 그러던 중에 발견된 Microsoft의 고객지원 내용이 이와 관련된 내용이어서 이 내용에서 착안하여 본 작업의 내용이 진행되었다.

〈표 1〉의 알고리즘은 Visual C++에서의 PowerPoint 프로그램의 열기 및 실행에서부터 페이지 생성, 페이지 저장, 페이지 로드, 페이지 레이아웃 관리, 페이지 디자인등 Visual C++에서 Powerpoint를 관리하는 다양한 기본 내용이 되었으며 이를 통해서 Powerpoint 프로그램과 관련된 WithPPT의 기초가 마련되었다(3).

〈표 1〉 Powerpoint control Algorithm

```

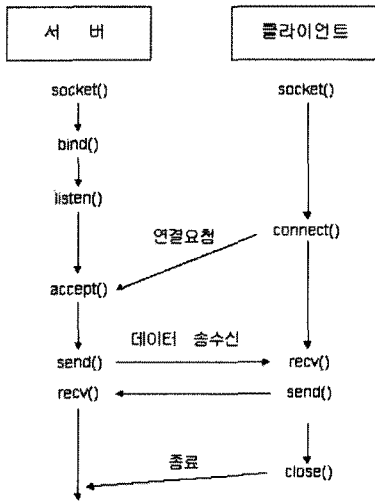
Powerpoint(){
// Get Presentations collection and add a new
presentation.
Presentations presSet(app.GetPresentations());
_Presentation pres(presSet.Add(TRUE));

// Get Slides collection and add a new slide.
Slides slideSet(pres.GetSlides());
_Slide slide(slideSet.Add(page));
}
    
```

3.2 Socket Programming

WithPPT는 여러 명의 클라이언트와 서버와의 긴밀한 접속을 위해서 TCP/IP방식의 Socket Programming을 사용하며 이를 위해 MFCSocket을 이용하여 구현하였다. MFCSocket의 기본적인 정보들을 구해서 구성하였는데 주 구현 내용을 서버에서의 소켓 생성과 소켓의 대기의 내용과 클라이언트의 소켓 생성, 특정 IP와 포트에 접속하여 데이터의 송/수신, 그리고 소켓의 종료의 관련된 내용이다.

소켓의 기본적인 순서도 〈그림8〉의 순서로 함수가 구현되어 이 순서에 맞도록 서버와 클라이언트간의 소켓 데이터 전송이 이루어진다(4)(5).



<그림 8> Socket 순서도

3.3 테스트 환경

주요 테스트는 두 제작자의 집에서 이루어졌으며 양쪽의 컴퓨터와 추가적으로 실험된 학교의 컴퓨터 사양은 <표 2>와 같다.

<표 2> 실험 컴퓨터 사양

	컴퓨터	OS	CPU	RAM
신윤호	노트북	Windows XP Home edition	Intel 1.83GHz z Coredu o	1GB
이정호	데스크탑	Windows XP professional	AMD 2.0GHz	1GB
학교	데스크탑	Windows XP professional	Inter 3.0GHz	448MB

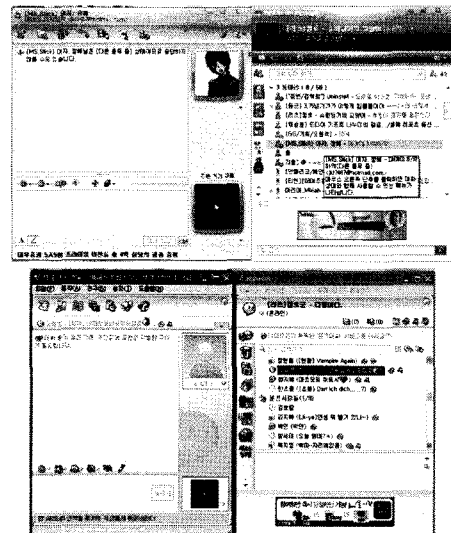
또한 실험 중에 찾아진 내용으로는 필수적인 요구 사항들이 존재 하였는데, Powerpoint2007이 필요하며 MFC 프로그램을 실행시키기 위한 Visual C++의 설치가 필요로 하였다. 그리고 무선인터넷의 불안정 때문인지 무선 인터넷 상황에서는 소켓의 프로그램이 제대로 실행되지 않아서 되다 안되다 하는 불안정한 모습이 Test되기도 하였다. 이를 위해 본 프로그램이 실행되기 위한 최소한의 요구사항을 만들자면 다음과 같다.

- WindowsXP에 최적화
- 유선인터넷에 최적화
- Microsoft Office2007 - Powerpoint 2007 필요
- Visual Studio 6.0 Or Visual C++ 설치환경

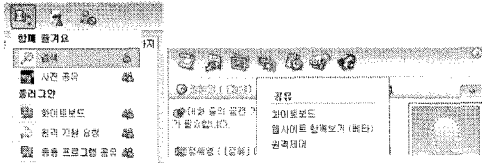
IV. 상업적 활용 방안

WithPPT는 Team Project 를 통해 Powerpoint 제작이 잦은 학생을 핵심 Target으로 한다. 그리고 실질적으로 이러한 학생들이 팀 프로젝트에 있어서 가장 많이 이용하게 되는 Messenger의 추가기능에 주목하게 되었다. Messenger의 경우에도 핵심 Target은 학생이었으며 이러한 Target의 다양한 요구에 맞추기 위해 Messenger내에서도 다양한 추가기능을 지원하고 있었다. WithPPT는 이러한 Target의 일치성과 사용자 확보 및 경쟁력 강화를 위해 Messenger의 추가기능에 WithPPT를 지원하고 WithPPT에서 부족한 팀원들과의 대화를 Messenger를 통해 지원하여 서로가 상업적으로 이익이 되는 형태의 상업적 활용 방안을 세웠다.

<그림9>에서 보이는 Messenger가 사용자가 가장 많다고 알려져 있는 두 가지 Messenger이다[6][7]. 이러한 메신저들을 많은 사용자 확보를 위해 서로 여러 가지 추가기능을 지원하고 있고 이는 <그림10>과 같다. 이에 따라서 많은 사용자가 해당 기능을 사용하기 위해 메신저를 사용한다. 따라서 메신저의 추가기능이 중요하게 여겨지고 있다. 이러한 경쟁이 되고 있는 추가기능에 핵심 타깃과 일치하는 WithPPT를 추가하여 상업적으로 이용하게 되면 메신저 사업의 사용자 증가 및 이용률에 많은 이익이 예상된다.



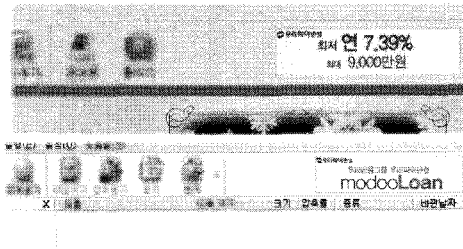
<그림 9> 위쪽 MSN Messenger, 아래쪽 NateOn Messenger



〈그림 10〉 각 Messenger에서 지원하는 추가 기능

위의 Messenger를 통한 협력 사업이 아닌 다른 수익 방안으로는 광고프리웨어 형식이 있다. 이는 본 프로그램의 여유 공간에 광고를 삽입하여 광고의 수익을 내는 방법이다. 본 프로그램은 Powerpoint 작업이 끝날 때까지 항상 화면 안에서 보이고 PPT의 긴 작업시간으로 인해 사용자에게 광고 노출력이 무척이나 크다고 볼 수 있다. 이와 비슷한 방법을 활용하여 수익을 창출하는 알소프트가 있으며 알 소프트웨어의 광고형식은 〈그림11〉과 같다[8].

이러한 광고 형식은 압축을 풀거나 그림을 보는 시간보다도 훨씬 많은 PPT제작시간에 노출되는 광고로써 광고의 효과가 크다고 예상된다.



〈그림 11〉 알 소프트웨어 광고형식

V. 결론

WithPPT는 기존에 학생들에게 필요한 사용자 Needs에 요구된 내용으로 제작 되었으며 현재는 아직 프로토타입의 모델이지만 이를 발전시켜 이용하면 실제로 PPT작업에 필수적으로 사용되는 프로그램으로 정착될 여지가 많다. 우선 이와 관련된 경쟁 제품은 유료로 사용되거나 회사에서 사용되는 버전관리용 툴로 일반적인 개인사용자가 사용하기 어려우며 구해서 설치하기에도 많은 불편함이 존재한다. 그리고 이와 가장 비슷한 동시 작업 툴과 편리한 UI로 이용되는 SpringNote라는 서비스가 존재하나 이 서비스는 PPT보다는 워드보고서에 한정되며 워드 작업을 동시에 진행하는 경우는 거의 없기 때문에 실제로는 많이 사용되지 않아 보인다 [9]. 그러나 WithPPT의 경우는 적은 용량과 간편한 인터페이스로 인해 많은 사용자들에게 좀 더 간편하게 사용 될 수 있다. 그러나 상업적인 활용도에 맞추어서 추가적인 개발이 필요하며 이에 따라 여러 가지 변화가 요구된다. 현재의 IP 입력 방식에서 자동적인 IP확인을 통한 접속이 우선되어야 하며 메시지의 기능으로 이용할 경우 메시지 프로그램의 특징에 맞는 변화가 요구된다.

또한 현재의 소켓프로그래밍의 불안정에 따라 예외 상황에 따른 완벽한 오류처리가 필요로 된다.

참고문헌

- [1] 김용성, Visual C++ 6 2nd Edition ,영진닷컴
- [2] 서덕록 이원석, 멀티미디어 시나리오를 이용한 효율적인 데이터 전송 기법 연구, 2003
- [3] Microsoft 고객지원-How to automate PowerPoint by using Visual C++ 5.0 or Visual C++ 6.0 with The Microsoft Foundation Classes : <http://support.microsoft.com/kb/222960>
- [4] 윤성우, TCP/IP 소켓 프로그래밍, 프리렉
- [5] 오영덕, 다중사용자 통신 환경을 지원하는 멀티미디어 머드 엔진 구조의 설계 및 구현, 1999
- [6] MSN 메신저 : <http://windowlive.msn.co.kr/wlm/messenger/>
- [7] NateOn 메신저 : <http://nateonweb.nate.com/>
- [8] 알소프트 : <http://www.altools.co.kr/>
- [9] Springnote : <http://www.springnote.com/ko>