

u-러닝시스템 최적화를 위한 가상화 애플리케이션 연구

오정삼^o 김효곤

고려대학교 컴퓨터정보통신대학원 컴퓨터정보통신공학과
baro@korea.ac.kr hyogon@korea.ac.kr

The Study of Virtualized Application for the Optimization of u-Learning System

Jeongsam Oh^o Hyogon Kim

The Graduate School of Computer & Information Technology, Korea University

요 약

u-러닝시스템의 개발과 활용은 유비쿼터스 컴퓨팅 환경의 발달에 따른 미래교실에서의 교육 및 학습 환경을 실현하기 위한 기술적 노력을 의미한다. 본 연구에서 다루고자 하는 u-러닝시스템이 높은 수준의 유비쿼터스 학습을 구현하기 위해서는 학교에서 뿐만 아니라 가정에서도 학습할 수 있는 u-러닝 환경을 구축할 수 있어야 하는데, 현재 u-러닝 시스템 구축을 통한 미래교육 연구는 광범위한 관심과 보급에도 불구하고 u-러닝시스템 활성화의 한계를 도출하고 있다. 즉, 현재 다양한 플랫폼 하에서 운영되고 있는 모바일 기기 환경에서 기존의 콘텐츠를 활용하기 위해서는 여전히 u-러닝 지원을 위한 단말기의 안정성이 떨어질 뿐만 아니라 u-러닝을 안정적으로 지원하는 기반 시스템이 적절하게 개발·보급되지 못하고 있는 실정이며 사용자의 편의성이 떨어져 u-러닝 환경에 적합하지 못한 경우가 많다.

따라서 본 연구는 이와같은 u-러닝시스템의 기반기술 구축에 필수적인 기술적 요소를 보완하기 위하여 오픈소스를 포함한 다양한 운영체제의 환경에서 윈도우 운영체제 기반 애플리케이션 사용이 가능한 Remote Desktop Protocol을 통한 가상화 기술을 구현하여 u-러닝시스템의 컴퓨팅 성능을 보다 효율적으로 증대시킬 수 있는 네트워크 시스템을 구현하고 타 시스템과의 성능평가를 수행한다.

1. 서 론

u-러닝이란 ‘유비쿼터스 학습환경을 기반으로 시간, 장소, 환경 등에 구애 받지 않고 일상 생활 속에서 언제, 어디서나 원하는 학습을 할 수 있게 되는 교육형태’(u-러닝의 이해, KERIS, 2005)라고 정의하고 있다. 이러한 정의에 입각하여 우리나라 초·중등 교육에서의 u-러닝은 2004년 말 ‘u-러닝 연구학교 운영 기본 계획’에서부터 시작되었으며, 대학에서의 u-러닝은 주로 학사 행정 등 학교생활에 편의를 제공하는 목적성이 강하게 반영되어 있다.[1]

그러나 u-러닝 시스템을 통한 미래교육연구는 u-러닝에 대한 높은 관심과 보급에도 불구하고 u-러닝 활성화의 한계를 도출하고 있다. 이는 최근 다양하게 보급되고 있는 모바일 단말기에 대하여 u-러닝을 안정적으

로 지원하는 기반 시스템이 적절하게 개발·보급되지 못하고 기존의 학습 콘텐츠들이 사용자의 편의성이 떨어질 뿐만 아니라 u-러닝 환경에 적합하지 않은 경우가 많기 때문이다. 따라서 이러한 u-LSS(u-Learning Supporting System)의 기반기술의 확충을 위해서는 유·무선 기반의 근거리망의 안정적인 서비스 제공(Internet connectivity), 모든 콘텐츠의 무결성 유지를 위한 단일저장소 보관(Single content repository), 이종간 단말기 사이의 콘텐츠 최적화(Output variation & Filtering) 등의 기술적 보완이 이루어져야 할 것이다.[2]

본 연구는 최근 다양한 모바일기기의 보급과 운영체제 환경하에서 다양한 플랫폼간에 애플리케이션을 사용 가능하게 하는 시스템의 개발을 통해서 u-러닝지원시스템의 활성화를 꾀할 필요성이 시급하며, 이를 위해서는 특히 실질적으로 자주 쓰이는 윈도우즈용 애플리케이션

을 리눅스를 비롯한 모든 주요 운영체제(Windows, Linux, Mac)에서 사용할 수 있도록 하여 이종간 단말기 사이의 콘텐츠를 최적화하여 학습자의 편의성을 증대시키고자 크로스 플랫폼간의 애플리케이션 가상화를 통한 모델을 제시하고 구현하고자 한다.

따라서 본 논문에서는 가상화 기술을 통하여 u-LSS 애플리케이션을 서버에만 설치하고 이를 클라이언트가 공유하게 함으로써 학습자가 어느 플랫폼에도 종속되지 않고 안정적으로 애플리케이션을 사용할 수 있게 한다. 또한 기존의 가상화 시스템의 단점으로 지적되었던 서버로부터 클라이언트로의 이미지 전송 때문에 발생하는 네트워크상의 트래픽 과부하현상을 RDP(Remote Desktop Protocol)를 사용하여 해결하였다.

2. 관련연구의 동향과 과제

2.1 u-러닝시스템활성화를 위한 연구과제 진행

유비쿼터스 컴퓨팅 환경은 교육 및 학습 환경에 새로운 가능성을 제시하고 있다. 이에 따라 교육과학기술부에서는 2006년 3월부터 미래학습환경에 대비하기 위하여 전국에 u-러닝 연구학교를 22개 지정·운영하여 u-러닝 시범 사업을 진행하고, 인텔, 마이크로소프트, KT 등의 IT기업의 지원을 받아 u-러닝 연구학교의 시설 및 기술이 지원하고 있으며 한국교육학술정보원(이하 KERIS)을 통해서 u-러닝 연구에 박차를 가하고 있다.

현재 u-러닝 연구학교에서 사용하는 단말기는 TPC, UMPC, PDA, MAC 등이며, 교수학습에서는 EBSi, 사이버가정학습, 에듀넷 및 시·도 교육청 교수학습지원센터 등의 다양한 콘텐츠로 이루어져 있다. 또한 KERIS는 2006년에 u-Class 모델 교실을 완공하여 u-러닝 현장 실험 및 u-러닝 연구의 가속화를 추진하고 있다.[2]

따라서 u-러닝 시스템은 새로운 사용자 환경의 변화에 따라서 기존의 e-러닝 시스템인 LMS/LCMS의 한계를 극복하기 위해 필터링 지원, 렌더링 지원, 원격지원, 모바일 단말기 지원 등의 시스템 차원의 노력을 요구하고 있는 데 이는 궁극적으로 다기종간의 호환성, 이식성의 문제를 해결해야 하는 과제에 직면하게 된다.

2.2 가상화 기술을 통한 문제 해결

이때 '하나의 통신망으로 연결된 모든 자원 및 애플리케이션들을 플랫폼의 변경없이 자유로이 작동하고, 안전하게 관리할 수 있는 미들웨어를 개발하여 보안성과 안전성을 갖춘 상태에서 시스템이 가동될 수 있게 하는 것이 소프트웨어 가상화 기법의 특징'이다.[3]

이와 같이 근래에 가상화 기술의 발전을 통해서 많은 이미지 전송 기술이 소개되고 있는데, 그 가운데 하나는 일반 메신저에서 지원하는 원격제어 프로그램이 있

다. 이 경우 서버측은 사용자가 메시지를 전달할 때 후킹(Hooking)을 수행하고, 네트워크 상에 이미지를 전송하기 때문에 네트워크 트래픽 상의 많은 부하가 발생한다.

본 논문에서 제안하는 시스템은 RDP(Remote Desktop Protocol)를 사용하여 서버측에만 모든 학습자 애플리케이션과 그에 따른 스토리지 및 지원 모듈을 설치하고, 학습자는 단순히 애플리케이션 서비스를 받기 위한 에이전트만을 설치할 수 있도록 한다. 즉 학습자는 자신의 컴퓨터에 별도의 애플리케이션을 설치하지 않아도 되기 때문에 어느 플랫폼에도 종속되지 않고 이기종 플랫폼 간의 u-러닝시스템 최적화를 보장받을 수 있다.

그리고 앞서 소개한 기존의 원격제어 프로그램에서 발생할 수 있는 네트워크 트래픽 과부하를 해결하기 위하여 사용자에게 서버측에서 구동되고 있는 애플리케이션의 이미지를 전송하는 방식 대신, RDP를 통해서 전송되는 패킷에 좌표 및 함수만을 전송하기 때문에 네트워크의 트래픽을 줄일 수 있는 방법을 모색하였다.

3. 제안 시스템의 아키텍처와 성능

3.1 제안시스템의 아키텍처

본 논문에서 제안하는 시스템의 핵심은 다양한 플랫폼간에 애플리케이션을 사용 가능하게 하는 시스템의 구현을 통해서 u-러닝지원시스템에서 사용하는 이종간 단말기 사이의 콘텐츠를 최적화하여 CPU 자원과 네트워크 트래픽을 최소화하는 것이다. 따라서 학습자가 사용할 수 있는 모든 애플리케이션을 서버에 설치하고 이를 학습자의 단말기인 클라이언트에서 제공받을 수 있는 다음과 같은 시스템을 제안한다.

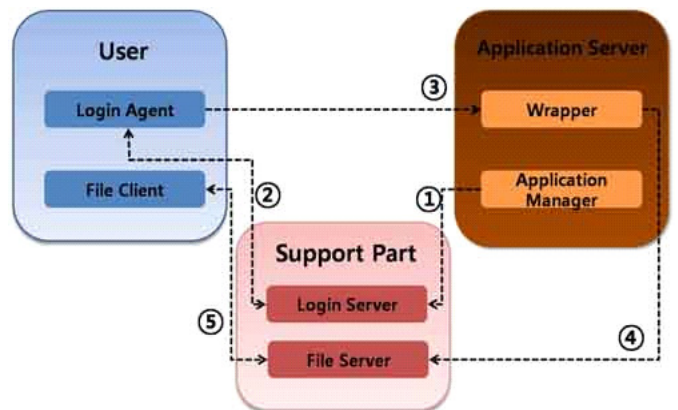


그림 1 제안 시스템의 아키텍처

[그림 1]에서 [Application Server]의 Application Manager는 설치된 애플리케이션에 대한 모든 정보를 조사한 뒤 이를 [Support Part]의 Login Server에게 전달한

다. 그 이후 Login Server는 수신한 데이터를 알맞게 파싱한 후에 데이터베이스에 저장한다. 이러한 일련의 작업을 통해서 [Application Server]와 [Support Part]는 학습자에게 서비스 할 모든 준비를 마친다.

학습자는 Login Agent를 통해 Login Server에 접속하고 인증절차를 걸쳐 회원인증시 [Support Part]는 [Application Server]에게 서비스 받을 수 있는 모든 정보를 수신하여 [User]에게 GUI로 제공할 수 있도록 전달한다. 그 후 사용자가 특정 애플리케이션을 선택함에 따라 [Application Server]에 접속을 하게 되면 애플리케이션 서비스 요청을 받은 [Application Server]는 해당 애플리케이션을 실행하기 위해, Wrapper를 구동하여 [Support Part]에 있는 사용자별 스토리지를 네트워크 드라이브로 연결한다.

| | | |
|----------|--------------------|---------------------------------|
| O/S | Application Server | MS Windows Server 2003 |
| | Login Server | Ubuntu |
| | Client | MS Windows XP, Ubuntu, Mac OS X |
| H/W | Server | Intel Pentium 2Ghz, RAM 1GB |
| | Client | Intel Pentium 1.2Ghz, RAM 1GB |
| Language | | ISO C++ , Trolltech Qt4 |
| Library | | Open SSL, STL |

표 1. 제안 시스템 개발환경

3.2 User System

[User]는 어느 플랫폼에서도 사용 가능해야 하기 때문에 C++기반의 언어이며, 운영체제의 제약이 없이 어떠한 플랫폼에서도 컴파일 가능한 언어인 Qt4(Trolltech)를 사용하여 구현하였는데 [그림 2]와같이 [Login Agent]와 [File Client] 두 가지로 나뉘게 된다.

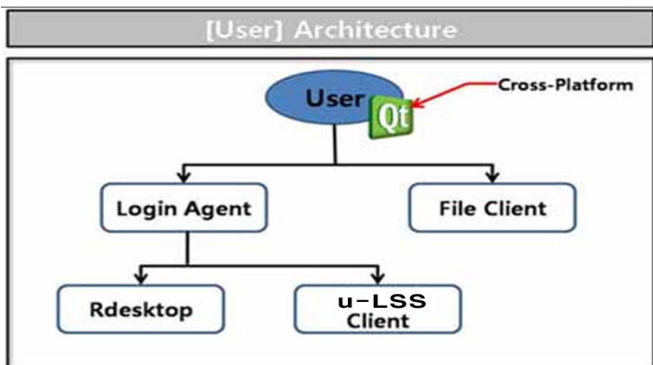


그림 2. User 시스템의 아키텍처

여기서 [File Client]는 사용자가 [Application Server]를 통해 스토리지를 접근하는 방식 외에 임의로 파일 관리를 할 수 있도록 파일을 송·수신하는 기능을 수행한다.

[Login Agent]는 Rdesktop과 u-LSS Client를 총칭하는 것으로 두가지로 분류된다. u-LSS Client는 [Support Part]의 Login Server와 통신을 하는데 사용자의 아이디 및 패스워드 확인완료 시 Login Server에서 전송하는 [Application Server]의 정보를 정해진 프로토콜을 통해 파싱하여 처리하며, 사용자에게 [Application Server]가 가지고 있는 애플리케이션 리스트(아이콘 이미지, 애플리케이션 절대경로 등)를 GUI로 표현한다.

[User]에서 핵심에 해당하는 Rdesktop은 터미널서비스를 기반으로 통신을 한다.

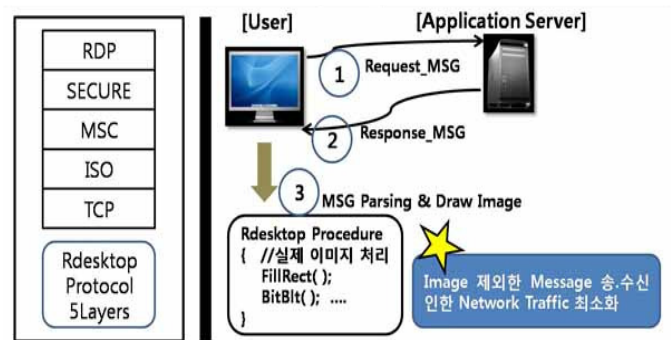


그림 3. Rdesktop 프로토콜의 계층과 처리과정

[그림 3]에서 Rdesktop에서 사용되는 RDP(Remote Desktop Protocol)는 터미널서비스 기반에서 사용되는 프로토콜로서 위 그림과 같이 5계층으로 구성되어 있다. Rdesktop에서는 기존에 이미지를 전송하는 방식과 다르게 [Application Server]에서 수행되는 애플리케이션에 이미지를 직접 전송하지 않고 메시지(좌표 및 함수 등) 송·수신만을 함으로써 네트워크 트래픽의 최소화라는 장점을 수반한다.

즉 Rdesktop은 [Application Server]에게 이미지 정보를 받지 않고, 오직 메시지 형태의 데이터만을 수신 함으로서 수신한 정보를 기반으로 자체적으로 이미지를 생성한다는 의미이다. 따라서 Rdesktop은 메시지를 단 한번만 수신하여 한번에 모든 이미지를 그려내는 것이 아니라 여러번의 메시지 수신을 통해, 여러 번의 그리기를 수행한다.

[그림 4]는 이런 일련의 과정을 순차적으로 보이는 그림이다. 예컨대, [Application Server]에게 수신된 메시지가 있다면 여기에 포함된 정보를 통해 사각형을 자체적으로 그린다. Rdesktop에서 그리기의 단위는 대개 사각형, 선 단위가 되며, 메시지에 포함된 색상정보를 통해 색상들을 표현한다. 이러한 과정의 반복을 통

해 Login Window가 생성된다.

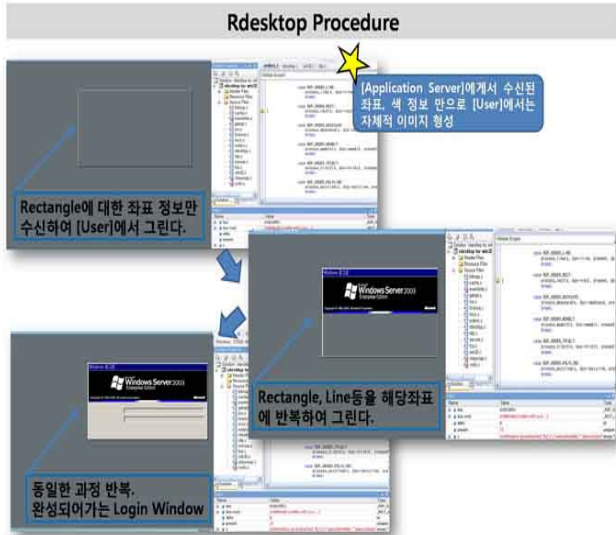


그림 4. Rdesktop의 메시지 처리과정

또한 Rdesktop은 무효화 영역을 제외한 필요 영역만의 메시지를 전송함으로써 데이터 전송량을 현저히 감소시킬 수 있다. [그림 5]는 기존의 원격제어 프로그램과 Rdesktop의 데이터 전송방식을 비교한 것이다.

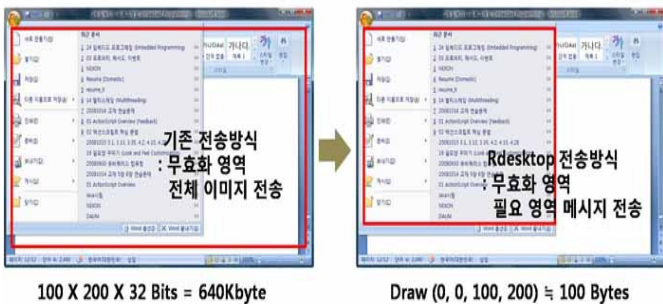


그림 5. 기존 원격제어 프로그램과 Rdesktop의 전송방식 비교

File Client는 사용자가 Support Part의 스토리지에 접근 시 Application Server를 통해 접근하는 방식을 대체하기 위해 만들어졌다. 유사시 접근성을 높이기 위해 임의로 직접 Support Part의 File Server에서 파일 관리가 가능하도록 설계되었다.

3.3 Support Part

[Support Part]는 [User]와 [Application Server]의 중간 매개체로써 Login시 [User]에게 필요한 내용을 전달하거나 스토리지 관리 등을 수행하는데 회원 인증을 담당하는 Login Server와 파일을 관리하는 File Server

로 나뉜다.

Login Server는 사용자의 회원인증을 관리하며, 회원 인증 시 데이터베이스에 저장되어 있는 Application Server의 정보를 학습자에게 전달한다. File Server는 File Client의 요청에 의해 수행된다. 서버의 파일을 저장하고 관리하는 것이 주된 역할이다.

3.4 Application server

[Application Server]는 Microsoft Windows Server 2003 플랫폼으로 수행된다. 이는 많은 사용자들이 터미널 서비스를 기반한 접속 시, 사용자 계정별로 효율적인 관리를 할 수 있는 가상 채널정책을 지원하기 때문인데 Wrapper와 Application Manager로 구성되어 있다. 그 과정은 학습자가 [Application Server]에게 서비스 요청 시 가장 먼저 Wrapper를 수행하게 된다. Wrapper는 특정 사용자가 접속 시 사용자에게 저장 공간을 제공하기 위해 Support Part의 스토리지와 연결하고, 임의의 사용자가 Application Server의 시스템 자원에 접근할 수 없도록 레지스트리를 수정하여 시스템의 접근을 제어하는 기능을 하며, 사용자가 요청한 특정 애플리케이션을 실행하는 기능도 한다.

Application Manager는 사용자에게 서비스 할 애플리케이션에 대한 절대경로 및 아이콘 이미지 등의 정보를 데이터베이스에 추가시키기 위해 Support Part의 Login Server에게 전달하여 하는 기능을 수행한다.

4. 제안시스템의 성능 평가

이상 본 논문에서 제시한 제안시스템은 실제 구현시 기존의 가상화 기술을 사용한 원격제어 프로그램과 비교해 볼 때 많은 성능향상을 보이는 것으로 나타났다. 이를 보여주기 위해서 최근 가상화 시스템의 성능 지표로 널리 쓰이는 프로세스의 상태에 따른 시스템 자원 점유율 및 네트워크의 데이터 전송량을 분석하여 시스템간의 성능을 비교해보도록 하겠다.

첫째, [표 2]는 제안 시스템과 일반 원격제어 프로그램의 CPU 자원을 비교한 것이다. 제안 시스템이 원격제어 프로그램에 비하여 초기 연결 시, 애플리케이션 실행 시, 유휴 상태 시 모두에서 CPU 자원의 점유율이 크게 낮아지는 것으로 나타났다.

둘째, [표 3]은 제안된 시스템과 일반 원격제어 프로그램의 수신 데이터 양에 대한 성능비교를 수행하였다. 이 성능측정에서도 역시 제안시스템의 경우 초기 연결 시, 애플리케이션 실행 시, 유휴 상태 시 모두에서 기존의 원격제어 프로그램에 비해서 현저한 데이터양의 감소를 보여주고 있으며, 특히 애플리케이션의 실행시 60%이상의 감소를 보여주고 있다.

이는 제안 시스템이 같은 서비스를 제공하면서도 RD

P를 사용함으로써 실제 사용자측에 무효화 영역 발생시에만 이미지를 전송하기 때문이다.

| CPU 점유율 동작상태 | 원격제어 프로그램 (국내 S사) | 제안시스템 | 감소율 |
|-----------------|----------------------|--------|--------|
| 초기연결시 | 30~35% | 10~13% | 20~22% |
| 애플리케이션 실행 | 25~28% | 6~12% | 16~19% |
| 유휴상태 | 2~4% | 1%미만 | 1~3% |

표 2. 제안시스템과 일반 원격제어 프로그램의 CPU 자원 점유율 비교

| 수신데이터 양 동작상태 | 원격제어 프로그램 (국내 S사) | 제안시스템 | 감소율 |
|-----------------|----------------------|------------|--------|
| 초기연결시 | 450 Kbyte | 230 Kbyte | 45~50% |
| 애플리케이션 실행 | 350 Kbyte | 130 Kbyte | 60~63% |
| 유휴상태 | 1~2 Kbyte | 1 Kbyte 미만 | 50%이상 |

표 3. 제안시스템과 일반 원격제어 프로그램 사용 시 수신 데이터의 양

5. 결론 및 향후과제

본 논문에서는 미래 학습공간 및 학습환경에서 이루어지는 u-러닝을 지원하는 시스템인 u-LSS의 효과적인 설계 및 구축을 위해서 근래에 이슈가 되고 있는 가상화에 의한 시스템 최적화 방안을 제시하고 있다.

즉 본 논문의 제안 시스템은 근래에 부상하고 있는 가상화 기술을 RDP를 사용하여 좀더 개선시켜 이미지 전송에 대한 부담을 줄이는 것을 통해서 기존의 원격제어 프로그램 대비 CPU 자원 점유율에서 약 15% 이상, 수신데이터 양에서 약 50% 이상의 뛰어난 성능 향상을 보이고 있다.

또한 학습자가 현재 널리 사용되고 있는 모바일 단말기를 통해서 운영체제의 제약 없이 어느 플랫폼에서도 학습 애플리케이션을 사용할 수 있도록 하기 위하여 자신의 단말기에 학습 애플리케이션을 설치하지 않고 에이전트만을 설치하여 이종간 플랫폼에서도 항상 최상의 컴퓨팅 성능을 유지할 수 있도록 하였다.

그러나 앞으로 미래 학습공간에서 교수학습의 동향에 관련된 다양한 기능을 지원하기 위해서는 본 연구에서 구현한 학습자지원 시스템 중심의 향상된 가상화 기술

이 u-LSS의 필수적인 제 범주인 교사지원, 평가지원, 관리지원, 정보제공지원 등의 각 범주에서도 향상된 기능이 충족되어야 할 것이다.

또한 이러한 u-LSS의 세부기능은 미래 사회의 학습 패러다임 및 교수학습 상황을 예측하여 설계된 기능들이 다수이기 때문에 향후 미래교실에서의 u-러닝지원 시스템의 활성화를 위해서는 좀더 세부적인 u-러닝시스템의 모형에 맞는 연구를 통해 미래교실의 유비쿼터스 컴퓨팅 환경 변화에 걸맞는 시스템의 구축을 이루어야 할 것이다.

참고 문헌

- [1] 김규년 외, “u-러닝 지원시스템(u-LSS) 연구 및 프로토타입 개발”, 한국교육학술정보원(KERIS), 2007
- [2] 김규년 외, “u-러닝 지원시스템 『u-LSS』 - u-Learning Supporting System”, 한국교육학술정보원(KERIS), 2007
- [3] 신동필, 여운동, “IT자원 가상화 기술”, 한국과학기술정보연구원, 2005
- [4] 김진미 외, “차세대 컴퓨팅을 위한 가상화 기술”, 전자통신동향분석 제23권 제4호, 2008
- [5] 장은정 외 역, “유비쿼터스 사회와 모바일 러닝”, jinhnan M&B, 2007
- [6] 한국교육학술정보원, “2008 교육정보화백서”, 교육기술과학부, 2008
- [7] (RDP 프로토콜) <http://www.rdesktop.org/#status>
- [8] (윈도우즈 서버 2003) <http://technet.microsoft.com/ko-kr/windowsserver/bb512919.aspx>