

# 스마트폰 기반 가상 데스크탑과 주변 레거시 디바이스 사이의 연결 중계 소프트웨어

박세진<sup>○</sup> 박찬익

포항공과대학교

{baksejin<sup>○</sup>, cipark}@postech.ac.kr

## A Smartphone-based Connection Broker between Virtual Desktop and Local Legacy Devices

Sejin Park<sup>○</sup> Chanik Park

POSTECH

### 요 약

하나의 물리 머신만으로 복수개의 가상 머신을 지원할 수 있는 가상화 기술[5,6]이 발달함에 따라, 기업에서는 비용 절감을 위해 가상 데스크탑 기술 도입이 많이 이루어지고 있다. 또한, 일반 사용자들을 위한 가상 데스크탑 기반의 클라우드 서비스 역시 많이 소개되고 있는데. 이러한 가상 데스크탑의 효과적인 사용을 위해서는 원격 접속 기술이 필수적이지만, 이는 키보드, 마우스 및 디스플레이가 갖추어져 있는 PC와 같은 전형적인 컴퓨팅 환경에서만 접속이 가능하다[3,4]. 본 논문에서는 스마트폰을 이용하여, 사용자 주변에서 원격 컴퓨팅 장치로 사용할 수 있는 LCD TV, 키보드 등의 레거시 디바이스를 검색하고 해당 디바이스를 원격지에 있는 가상 데스크탑과 연계시켜[1,2], 사용자 주변의 디바이스를 통해 원격 가상 데스크탑 제어를 연계 할 수 있는 중계 소프트웨어에 대한 설계 및 구현 내용을 기술하고 있다.

### 1. 서 론

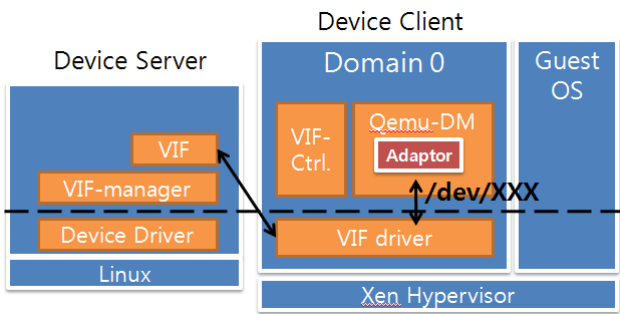
가상화 기술이 보편화 됨에 따라, 기업들에서는 기업 총 소유 비용 절감을 위해 가상 데스크탑 기술을 도입하고 있으며[9], 또한, 클라우드 서비스가 보편화 되면서, 클라우드 기반의 가상 데스크탑 서비스 [7,8]들이 등장하고 있다. 이러한 가상 데스크탑에 접속하기 위해서는 원격 접속 프로토콜 [3,4] 기반의 클라이언트가 있어야 한다. 일반적으로 원격 데스크탑 클라이언트들은 키보드, 마우스, 디스플레이 장치를 모두 가지고 있는 PC 또는 태블릿 디바이스 형태를 갖추고 있다.

그러나 현재 주변에 있는 LCD TV, 오디오 등의 장치들 또한, 원격지에 있는 가상 데스크탑과 연계되어 사용하기에 충분한 성능을 가지고 있으며, 오히려 대형 LCD TV 화면은 PC나 태블릿의 모니터 보다 더 좋은 성능을 가지고 있다. 따라서, 본 논문에서는 사용자가 언제나 휴대하고 있는 스마트폰을 이용하여, 가상 데스크탑의 자원으로 활용 가능한 주변의 디바이스를 탐지 하고, 실제 원격지의 가상 데스크탑 서버와 연계 시켜주는 중계 소프트웨어를 제안한다.

### 2. 기초 지식 및 관련 연구

가상 데스크탑 환경은 Xen[6] 이나, VMWare[5] 와 같은 가상머신 관리 소프트웨어상에서 구동되는 가상머신에 Microsoft Windows 나 Linux Gnome Desktop Environment 등과 같은 데스크탑 환경을 설치하여 사용자가 쓸 수 있도록 구성한 것을 말한다. 일반적으로 이러한 가상 데스크탑 환경의 접근은 RDP[4], VNC[3] 등의 원격접속 프로토콜을 통해 이루어지며, 이를 구동하는 클라이언트에는 원격 접속에 필요한 자원들(키보드, 마우스, 모니터등)이 모두 갖추어져 있어야 하며, 주변 장치와의 연계는 불가능하다는 한계점을 가진다.

연구 [1]에서는 레거시 디바이스를 네트워크 상에서 접근 가능하게 하는 프레임워크를 제안하였다. 이 방식은 디바이스 서버와 디바이스 클라이언트가 직접 네트워크를 통하여 연결되는 방식으로, 레거시 디바이스는 디바이스 서버에 연결되어, 네트워크 디바이스로서 동작을 할 수 있게 된다. 이 경우, 디바이스 클라이언트는 디바이스 서버가 네트워크로 브로드 캐스팅하는 정보를 확인하여 디바이스 검색을 하며, 직접 디바이스 서버에 접속을 하게 된다.



<그림 1> VIF-VIFdrv Architecture : Xen Hypervisor 기반으로 구성되며, Guest OS 에 가상 디바이스를 제공하는 Qemu-dm에 Adaptor라는 모듈이 원격 디바이스를 서비스하고 있는 VIF driver를 접근

연구 [2]에서는 VIF driver 라는 가상 디바이스 드라이버가 디바이스 서버가 표출하고 있는 VIF라는 가상 디바이스와 연결되어, 가상머신이 원격 디바이스를 사용할 수 있는 프레임워크를 제안하였다.

<그림 1>은 연구 [2]의 구조를 나타내고 있다. VIF는 디바이스 클라이언트의 VIF-driver와 TCP/IP 통신을 통해 실제 디바이스 입출력 및 제어를 처리해 주는 모듈로, 클라이언트의 요청에 의해 동적으로 생성 / 삭제가 된다. <그림 1>에 나타나 있는 것처럼 VIF는 User-level로 작성되어 있는데, 이는 커널 드라이버 수준에서 제공하는 디바이스 드라이버의 Multiplexing 기능을 그대로 승계 받을 수 있으며, 또한 User-level로 구현되어 있기 때문에 네트워크 에러 등의 문제가 발생하여도 디바이스 서버 시스템에 영향을 미치지 않는다.

Xen 기반의 전가상화 가상 머신은 Qemu[10] 에서 제공하는 가상 디바이스를 사용하고 있다. 원격 디바이스를 제공하는 VIF Driver는 디바이스 드라이버 수준으로 개발되어 있으며 Linux의 VFS Layer의 file\_operation 인터페이스를 구현되어 있어, 사용하고자 하는 디바이스 파일을 open() 시스템 콜을 통해 접근할 수 있다. 따라서 가상머신과 원격 디바이스의 연결은 Qemu 에서 접속하고자 하는 VIF driver를 open()하여 이루어 지게 된다

이 두 연구의 경우[1,2], 로컬 머신이나, 가상 머신이 직접, 동일한 네트워크 서브넷 내의 디바이스를 탐지하여 연결 시키는 방법을 사용하고 있다. 하지만, 클라우드 기반의 원격 가상 데스크탑 환경의 경우, 주변 디바이스와 원격 데스크탑을 연계 시켜줄 수 있는 수단이 없기 때문에, 이를 중계해줄 수 있는 소프트웨어가 필요하다.

### 3. 설계

본 연구에서는 기존 연구[1,2]를 기반으로 주변 디바이스를 탐지하고, 동시에 원격지에 위치한 가상

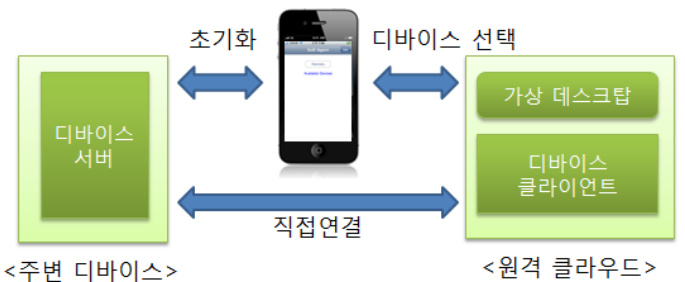
데스크탑과 연계할 수 있는 스마트폰 기반 중계 소프트웨어를 제안한다. 본 중계 소프트웨어는 디바이스 서버와 가상머신의 디바이스 클라이언트를 중계하며, 각 개체는 다음과 같이 동작한다.

#### 3.1 스마트폰 기반 중계 소프트웨어

중계 소프트웨어를 설계할 때 가장 중요한 것은, 기존 시스템을 얼마나 유지할 수 있는가 하는 점이다. 본 중계 소프트웨어는 크게 두 가지 모델로 구성될 수 있다.



<그림 2> 가상 데스크탑과 디바이스서버 간 모든 동작을 중계하는 모델



<그림 3> 가상 데스크탑과 디바이스 서버가 직접 연결될 수 있도록 중계소프트웨어는 일부 초기화 부분만 관여 하는 모델

<그림 2>와 같이 모든 동작을 중계하는 모델의 경우, 기존 디바이스 서버와 가상머신의 디바이스 클라이언트간의 모든 프로토콜을 중계 소프트웨어가 처리해 주어야 하며, 이로 인해 성능적인 오버헤드도 발생한다. <그림 3>의 경우, 중계소프트웨어의 수행이 시작되면, 마치 디바이스 서버에게 사용자의 가상 머신 정보가 마치 로컬 네트워크 내에 있는 것처럼 advertisement 하여, 실제 디바이스 서버와 클라이언트가 직접 연결되어 기존 모델과 동일하게 동작하게 되어, 성능적 오버헤드 및 구현의 복잡도가 낮아진다. 따라서, 본 연구에서는 <그림 3>의 모델을 기반으로 중계 소프트웨어를 구현하였다.

스마트폰 기반 중계 소프트웨어는 디바이스 서버로 Advertisement할 가상 데스크탑의 정보 (IP Address, Port 번호)를 유지하고 있으며, 사용자가 사용할 디바이스를 선택할 수 있는 인터페이스로 구성된다. 이 사용 가능한 디바이스 리스트는 디바이스 서버와 연결된 디바이스 클라이언트에서 받아오게 되며, 사용자의 디바이스 연결/해제 명령은 디바이스

클라이언트로 연결되어 처리된다.

### 3.2 디바이스 서버.

본 연구에서 디바이스 서버와 가상 데스크탑의 디바이스 클라이언트는 [2]에서 연구한 VIF-VIFdriver을 기반으로 연결된다. 즉, 각 레거시 디바이스들이 디바이스 서버에 물리적으로 연결이 되면, 해당 물리 기능들은 원격지에서 사용할 수 있는 VIF로 구성된다. 또한, 시스템의 디바이스 서버는 직접 디바이스 정보를 advertisement 하지 않고, 중계 소프트웨어의 advertisement를 항상 대기하고 있다. 이는 위치가 이동하는 주체가 스마트폰 기반 중계 소프트웨어이기 때문이다.

중계 소프트웨어로부터 가상 데스크탑 디바이스 클라이언트의 IP Address 와 연결할 Port 를 받으면 서버는 즉시 디바이스 클라이언트로 연결을 수행하며, 클라이언트에게 현재 가용한 디바이스 리스트를 보내 준다.

### 3.3 가상 데스크탑 디바이스 클라이언트

디바이스 클라이언트는 중계 소프트웨어 및 디바이스 서버의 연결을 대기한다. 중계 소프트웨어가 연결이 되면, 가상 머신 정보를 중계 소프트웨어로 전달한다. 디바이스 서버와 연결이 이루어지면, 우선 디바이스 서버로부터 사용 가능한 디바이스 리스트를 받아오며, 이 정보를 중계 소프트웨어로 보내주게 된다. 중계 소프트웨어로부터 디바이스 연결 요청이 들어오면, 디바이스 클라이언트는 VIF driver를 생성하고, 디바이스 서버측에 VIF driver의 Port 를 보내어 VIF 와 VIF driver의 연결을 요청한다.

<그림 4>는 실제 중계 소프트웨어가 수행되는 시점부터 실제로 디바이스가 연결되는 과정을 도식화 하고 있다.



<그림 4> 중계 소프트웨어의 디바이스 연결 요청 수행

## 4. 평가

### 4.1. 구현 및 수행 환경

#### 4.1.1 중계 소프트웨어

중계 소프트웨어는 iOS 4.1 기반 스마트폰 iPhone 3GS 에서 구현되었으며, 사용자 인터페이스는 <그림 5>와 같다. 현재 스마트폰 주변에서 탐색된 디바이스 목록이 나타나며, 간단한 스위치 조작을 통해 가상 데스크탑에 연결 및 해제 작업을 수행한다.

#### 4.1.2 디바이스 서버

디바이스 서버는 VIA C7 1GHz 프로세서를 탑재한 VIA ARTiGO 모델을 사용하고 있으며, 1GB 메모리, Linux Kernel 2.6.26을 구동하고 있다.



<그림 5> 사용자 인터페이스

#### 4.1.3 디바이스 클라이언트

디바이스 클라이언트는 Intel Dual Core CPU E6300 2.80GHz 프로세서, 4GB 메모리를 탑재하고 있으며, Xen 3.1 Official distribution을 구동하며, 가상 데스크탑은 Windows XP가 구동된다.

### 4.2 중계 소프트웨어의 디바이스 인식 시간

디바이스 서버에 USB 타입 키보드를 연결한 후, 해당 키보드 디바이스가 스마트폰 중계 소프트웨어에 인식되기 까지는 약 2초 정도의 시간이 소요된다. 소요 시간의 대부분은 실제 디바이스 서버에서 USB 타입의 디바이스 인식을 위해 사용하는 Linux Hald[11] 와 Dbus[12] 인터페이스에서 사용됨 (약 1.7 초 정도) 을 확인하였다. 이는 중계 소프트웨어와 디바이스 클라이언트가 연결되어 있는 상태에서의 소요시간을 나타내며, 실제 중계 소프트웨어는 3초의 주기로 디바이스 클라이언트 정보를 Advertisement 한다.

## 5. 향후 연구 방향 및 결론

본 연구에서는 주변 디바이스와 원격지의 가상 데스크탑을 연계하기 위해, 스마트폰에서 구동되는 중계 소프트웨어를 설계 및 구현하였다. 이를 통해 기존의 연구의 문제점인 동일 서브넷 내의 디바이스 간의 연결을 스마트폰이라는 매개체를 통해 로컬-원격지 연결로 확장할 수 있게 되어, 원격 가상 데스크탑을 더욱 효과적으로 지원할 수 있는 장점을 가지고 있다.

현재 중계 소프트웨어의 역할은 실제 디바이스 서버와 디바이스 클라이언트의 연결을 위한 정보 전달 및 디바이스 연결 및 해제 명령 전달로 제한적인데, 사용자가 항상 휴대하는 장치로써, 사용자의 요구사항 또는 사용자의 디바이스 선호도에 따라 복수개의 디바이스 중 최적의 조합을 지능적으로 찾아낼 수 있는 시스템으로의 확장에 대한 추가 연구가 필요하다.

[11] HALd, Hal daemon,  
<http://www.freedesktop.org/wiki/Software/hal>

[12] Dbus library,  
<http://www.freedesktop.org/wiki/Software/dbus>

### <참고문헌>

[1] 박세진, 강상우, 박찬익, 레거시 디바이스의 네트워크 디바이스화를 지원하는 프레임워크의 설계 및 구현, 2009 한국 컴퓨터 종합 학술대회(KCC 2009)

[2] 박세진, 강상우, 박찬익, 가상머신의 원격 자원 입출력 지원을 위한 디바이스 드라이버 기반 프레임워크의 개발, 2009 차세대 컴퓨팅 추계학술대회

[3] T.Richardson, et al., "Virtual Network Computing" IEEE, Internet Computing, Vo1.2, 30.1, 1998.

[4] Microsoft, Remote Desktop Protocol, RDP, <http://msdn.microsoft.com/en-us/library/aa383015> (VS.85).aspx

[5] VMWare, ESX server, <http://www.vmware.com/>

[6] Dragovic, B., et al., Xen and the art of virtualization. In Proceedings of the ACM Symposium on Operating Systems Principles (October 2003)

[7] Desktone, <http://www.desktone.com>

[8] SmartBusiness, IBM,  
<http://ibm.com/smartbusiness>

[9] Gartner News Room, 2009. 3. 26

[10] Fabrice Bellard, "QEMU, a Fast and Portable Dynamic Translator", Usenix Annual Technical Conference, 2005