

# 멀티미디어 서비스를 위한 씬클라이언트 컴퓨팅의 성능평가 및 비교

김병길<sup>o</sup> 정인범<sup>o</sup>  
강원대학교 컴퓨터정보통신공학과  
bgkim@snslab.kangwon.ac.kr<sup>o</sup>, ibjung@kangwon.ac.kr

## Performance Evaluation and Comparison of Thin-Client Computing for Multimedia service

Byeonggil Kim<sup>o</sup> Inbum Jung<sup>o</sup>  
Dept. Computer Information & Telecommunication Engineering

### 요 약

최근 임베디드 컴퓨팅의 눈부신 발전으로 개인이 사용하는 PC의 단위는 점점 작아지는 추세에 있으며 이로 인해 소형 컴퓨터와 단말기에 대한 수요가 증가하고 있다. 하지만 시스템의 소형화로 인해 이러한 시스템들은 일반적인 애플리케이션은 물론 동영상과 같은 미디어 데이터를 처리할 만한 성능을 갖지 못하게 되었다.

본 논문은 VNC(Virtual Network Computing)를 이용하여 다양한 운영체제를 기반으로 미디어 데이터의 처리 성능을 측정하여 비교 연구를 하였다. 이를 통해 멀티미디어 서비스에 적합한 플랫폼을 제시하고 향후 씬클라이언트 컴퓨팅의 개발 방향을 제안하고자 한다.

### 1. 서 론

과거의 컴퓨팅 모델이었던 중앙집중식 컴퓨팅은 오늘날 데스크탑 컴퓨팅 모델로 변화되었으며 이러한 개인 데스크탑 컴퓨터들은 분산이 되어 그것들을 관리하기가 어려워지게 되었다. 최근 이러한 문제들로 인해 컴퓨팅 모델은 다시 과거의 모습으로 되돌아가려는 시도를 하고 있으며 단지 과거의 텍스트 기반의 지루한 형태가 아닌 사용자의 요구에 맞는 그래픽 환경 기반의 터미널 형태로 만들고자 하는 것이다. 하지만 오늘날의 사용자의 요구는 그래픽 환경에만 그치지 않고 멀티미디어에 대한 요구가 증가되어 가고 있다.

현재 이러한 요구를 충족시킬 만한 기술이 바로 AT&T에서 최초 개발한 VNC이며 공개소스의 원칙을 가지고 있기 때문에 본 논문에서 측정 대상이 되었다. 이러한 VNC 기술은 저사양의 씬클라이언트를 이용했을 경우 하드웨어의 저성능으로 인하여 미디어 데이터를 받아 화면에 처리할 만한 성능이 부족할 뿐만 아니라 멀티미디어에 필수적인 음성처리를 위한 기술을 지니지 못했다[2][3].

본 논문에서는 윈도우즈와 리눅스 기반의 플랫폼에서 네트워크와 CPU 등 서버의 자원 이용률을 측정하여 성능 비교를 통해 VNC의 적합한 개발 플랫폼을 제시한다. 또한 VNC 기반으로 미디어 데이터의 처리 성능을 측정하여 성능에 제약이 되는 원인을 규명하고 동영상 재생이 가능한 VNC 개발을 위한 방향을 올바르게 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문과 관련된

연구에 대하여 설명한다. 3장에서는 측정에 이용된 시스템과 실험환경에 대해 설명한 후 측정 결과 그래프들을 보여줄 것이다. 4장에서는 성능 측정에 대한 결과를 비교분석하여 결과를 설명한다. 5장에서는 본 논문의 결론을 맺고 향후 연구 계획을 설명한다.

### 2. 관련 연구

고성능의 집중된 자원을 네트워크를 통하여 그래픽 유저 환경으로 사용할 수 있는 VNC 시스템은 이미 많은 연구가 이루어져 활성화가 되어 있으며 상업용으로 개발하여 이미 판매가 이루어지는 제품도 있다[1][3].

저사양의 씬클라이언트 시스템 기반에서 VNC를 이용했을 때 성능 제약으로 작용하는 원인을 분석하여 VNC의 전체 성능을 지배하는 인자들을 알아내고자 하는 연구도 있다 [2]. 하지만 이러한 노력에도 불구하고 VNC를 이용하여 사용자들의 관심이 되는 멀티미디어를 아무런 불편함 없이 이용할 수 있도록 하는 연구는 아직 많이 부족한 실정이다. 따라서 이동 단말기와 같은 저사양의 소형 컴퓨터에서 VNC를 이용하여 멀티미디어를 제공할 수 있는 방안에 대한 연구가 필요하다.

### 3. 성능 측정

#### 3.1 시스템 및 실험 환경

실험에 사용된 프로그램은 AT&T에서 개발되어 VNC 표준으로 자리 잡혀 있는 RealVNC 프로그램을 사용하였다.

서버 시스템은 RealVNC 서버 프로그램을 이용하여 클라이언트의 접속을 대기하며 클라이언트의 요청 시 해당 데스크탑 정

\* 본 연구는 한국과학재단 목적기초연구(R05-2003-000-12146-0) 지원으로 수행되었습니다.

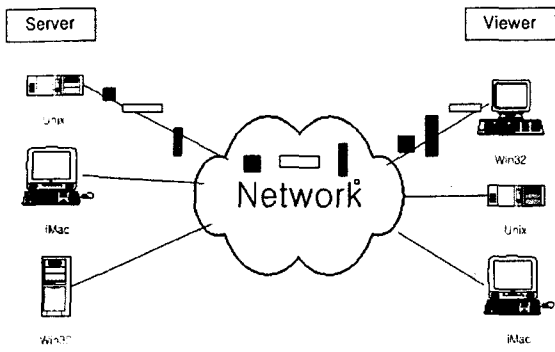


그림 1 시스템 구성도

보통 클라이언트에게 전송하는 역할을 담당하며 클라이언트 시스템은 RealVNC 뷰어 프로그램이나 웹 브라우저를 이용하여 서버에 접속해 데스크탑 정보를 수신하여 클라이언트 모니터에 보여주는 역할을 한다. 이러한 시스템의 전체적인 구성도는 그림 1 과 같다.

시스템 성능에 따른 측정을 위해 3가지 타입의 시스템인 펜티엄3, 펜티엄4, AMD 듀얼 서버에서 측정 하였으며 각각의 세부적인 사양은 표 1 과 같다.

측정에 사용된 미디어 플레이어는 리눅스기반 플랫폼을 위해 MPlayer(버전 0.90rc5)를 이용하였고 윈도우기반 플랫폼을 위해 마이크로소프트 윈도우즈 미디어 플레이어(버전 9.00.00.3008)를 사용하였다. 측정 비디오는 장면의 변화가 많은 액션 영화인 터미네이터3를 이용하였으며 서버에서 실행시켜 클라이언트에 재생되도록 하였다. 측정에 사용된 비디오 클립에 대한 세부적인 정보는 표 2 와 같다.

	Type A	Type B	Type C
CPU	P3 866 Hz	P4 1.8 Hz	애슬론 MP 2000
메모리	256M	256M	1G
디스크	WD 40G ( IDE )	IBM 40G ( IDE )	Seagatee 36G ( SISC )
네트워크	100Mbps	100Mbps	100Mbps

표 1 시스템 사양

프레임 해상도	480*208
컬러 수	16bit
프레임 속도	25fps
상영시간	6048(s)

표 2 측정 미디어 정보

### 3.2 성능 측정 방법

측정 방법은 RealVNC 서버에 클라이언트가 접속하여 미디어를 재생 했을 때 요구되는 네트워크량과 CPU 사용량을 측정 하였으며 클라이언트 수를 늘려가며 변화량을 살펴보았다. 정확한 측정을 위해 미디어 재생 화면 이외에는 어떠한 스크린 정보도 전송되지 않도록 하였으며 영화가 시작될 때부터 종료될 때까지의 실행시간과 네트워크의 최대와 평균 사용량, CPU

사용량에 대해 측정하였다. 최종 측정값은 3회 반복하여 평균한 값을 사용하였다.

측정 플랫폼은 윈도우즈와 리눅스 시스템에 대해 각각 성능 측정을 하였다.

### 3.3 성능 측정 결과

그림 2 는 접속한 클라이언트에게 서버가 변경된 스크린 데이터를 전송했을 때의 네트워크 사용량을 보여준다. 그래프에서 노드는 서버에 접속한 클라이언트를 말하며 Type A, B, C 는 리눅스 기반의 서버 시스템으로 표 1 과 같고 윈도우즈는 Type B 시스템을 사용하였다. 한 개의 클라이언트를 서비스 할 경우 윈도우즈는 초당 700K의 대역폭을 요구하지만 리눅스 시스템의 경우 윈도우즈에 비해 6배가량 높은 4M 정도의 대역폭이 요구되는 것을 알 수 있다. 로컬에서 표 2와 같은 사양의 비디오 클립을 정상 재생했을 때 복호된 데이터 크기는 4992K(480\*208\*16\*25)이므로 이 수치에 비해 윈도우즈와 리눅스는 각각 14%와 80%의 데이터를 이용하여 화면에 보여주는 것이다.

이러한 결과는 업데이트 데이터를 이용하여 화면에 그려주는 방식에 있어 윈도우즈는 작은 사이즈의 데이터를 요구하지만 X 라이브러리를 사용하는 리눅스의 경우는 많은 데이터를 필요로 한다는 것으로 분석되었다. 따라서 비디오 화질 측면에서는 리눅스가 훨씬 좋은 성능을 보여주었다.

그림 3은 윈도우즈 기반의 플랫폼이 작은 크기의 데이터를 이용한다는 것을 더욱 명확하게 보여준다. 그림 3에서 DL, DW, SL은 리눅스 기반의 Type A 시스템, 윈도우즈 기반의 Type B 시스템, 리눅스 기반의 Type C 시스템을 각각 의미하며 서버-클라이언트 구조로 다양한 시스템 형태를 구성하여 성능을 측정했다. 그림에서 보듯이 적은 대역폭을 요구하는 시스템 구조는 윈도우즈 기반의 플랫폼이 시스템 구성요소들 이루는 대역폭 사용량이 현저히 낮아지는 것을 볼 수 있다.

네트워크의 최대 가능한 대역폭을 기준으로 했을 때 서비스 가능한 노드의 수는 Type A의 경우 3개에서, Type B와 C의 경우는 6개의 노드 접속시 12M의 최대 대역폭에 가까워졌으며 반면 윈도우즈의 경우는 노드가 6개 일 경우에도 4M를 넘지 않을 것이다. 하지만 윈도우즈는 그림 4에서 보듯이와 같이 노드 2개 이상이 접속 했을 때는 서버의 CPU 사용량이 거의 한계에 도달하게 되어 서버에서 재생 중인 플레이어조차 정상적으로 재생되지 못하는 현상이 발생했다. 하지만 리눅스의 경우 Type A의 경우를 제외한 Type B와 C는 CPU 자원에 제약을 받지 않았다.

이러한 결과는 Type A의 경우는 CPU가 성능 제약의 원인이 된다는 것을 의미하며 이로 인해 그림 2에서 보듯이 노드 4개에서 변곡점이 발생하였다. 하지만 Type B 이상의 성능을

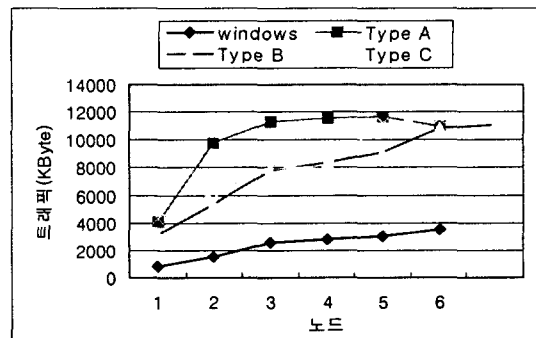


그림 2 노드 증가에 따른 평균 대역폭 사용량

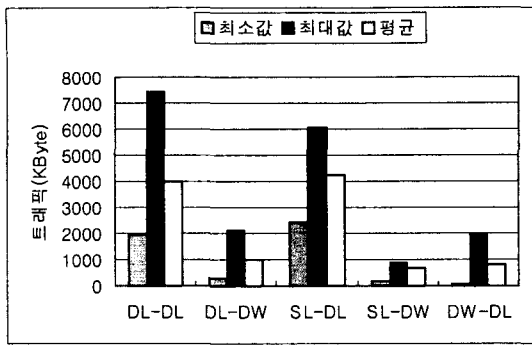


그림 3 플랫폼에 따른 대역폭 사용량

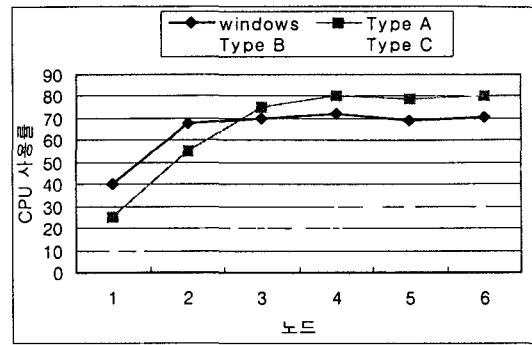


그림 4 노드 수에 따른 CPU 사용률

가진 시스템에서는 네트워크 대역폭이 최고점에 도달한다 해도 CPU가 성능 제약의 원인이 되지 않으며 대역폭이 성능 제약의 일차적인 원인이 될 수 있다.

그림 5는 노드 증가에 따른 비디오 재생 시간을 기록한 그래프이다. 그래프에는 윈도우즈에 대한 측정값이 보이지 않을 것이다. 이는 윈도우즈 경우 노드 개수의 증가와 상관없이 서버의 CPU 자원 부족으로 비디오 프레임을 정상적으로 복호화하지 않고 버리게 됨으로써 본래의 상영시간을 항상 유지하는 결과를 보여주었기 때문이다. 리눅스의 경우 Type A는 3개 이상에서 Type B와 C는 5개 이상의 노드 접속 시 재생 화면이 느려지는 현상이 발생했다.

#### 4. 결과 분석

지금까지 윈도우 기반과 리눅스 기반에서의 VNC 성능을 비교하였으며 리눅스 기반에서 시스템 타입에 따른 성능을 비교하였다.

윈도우즈 기반은 모니터에 화면을 그리는 방식에 있어 적은 대역폭을 차지한다는 장점을 지니고 있지만 비디오 화질 측면에서는 매우 낮은 화질을 보여주기에 멀티미디어를 목적으로 하는 시스템에는 부적합함을 확인 하였다. 또한 서버의 CPU 이용률은 노드 개수가 불과 2개임에도 불구하고 거의 모든 자원이 소모되는 것을 확인하였다. 이러한 원인은 화면의 갱신이 빈번하게 발생하는 동영상 화면을 처리하는데 있어 업데이트 영역을 검출하기 위해 필요로 하는 처리량이 매우 높기 때문이다.

하지만 리눅스 기반에서는 X 서버에서 화면 업데이트를 위해

많은 데이터를 필요로 하므로 높은 네트워크 대역폭을 차지 하지만 비디오 화질은 로컬에서 재생되는 화질과 별 차이가 없을 정도의 매우 우수한 화질을 보여 주었다.

또한 Type B 이상의 시스템 성능에서는 최대 4개의 노드까지 정상적인 서비스가 가능함을 확인하였다.

#### 5. 결론 및 향후 연구

본 논문에서는 현재 사용가능한 VNC 기반에서 멀티미디어 기능에 대한 성능을 측정했다.

네트워크 기술의 발전으로 향후 기가망과 같은 네트워크가 구성되어 질 것으로 감안한다면 리눅스 기반에서 성능 제약이 되었던 네트워크의 제약은 곧 해소되어 질 것으로 예측할 수 있으며 압축 기법이나 프로토콜의 성능 향상으로도 대역폭을 줄여 나갈 수 있을 것이다. 그러므로 썬클라이언트의 개발 플랫폼으로 리눅스를 사용함으로써 훨씬 쉽고 빠른 개발이 이루어질 것이다.

멀티미디어에 있어 음성 또한 없어서는 안 될 중요한 요소이다. 하지만 VNC는 음성처리기술에 대한 언급이 전혀 없다. 향후에는 미디어 데이터를 재생함에 있어 음성기능이 지원되도록 하는 연구를 진행할 계획이다.

#### 6. 참고 문헌

- [1] Tristan Richardson, Quentin Stafford-Fraser, Kenneth R. Wood, Andy Hopper, "Virtual Network Computing", IEEE, 1998
- [2] S. Jae Yang, Jason Nieh, Matt Selsky, and Nikhil Tiwari, "The Performance of Remote Display Mechanisms for Thin-Client Computing", USENIX, 2002
- [3] Brian K. Schmidt, Monica S. Lam, J. Duane Northcutt, "The interactive performance of SLIM: a stateless, thin-client architecture", ACM Symposium, December 1999
- [4] S. J. Yang, J. Nieh, and N. Novik, "Measuring Thin-Client Performance Using Slow-Motion Benchmarking", USENIX, 2001
- [5] Alexander Ya-li Wong, Margo Seltzer, "Operating System Support for Multi-User, Remote, Graphical Interaction", USENIX, 2000
- [6] R. W. Scheifler and J. Gettys, "The X Window System", ACM Transactions on Graphics, 1986

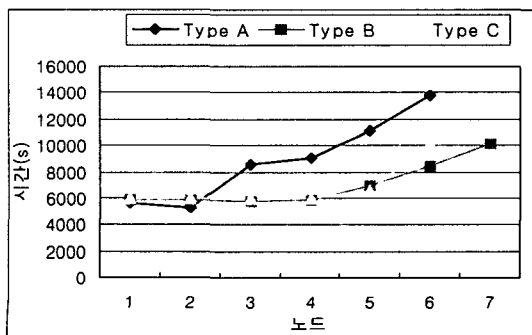


그림 5 노드 증가에 따른 영화 상영시간 변화량