



# DDI Mode & GDI Mode?

이 문서는 알서포트 기술연구소에서 리모트콜의 기술적인 부분을 분석, 설명하기 위해 작성한 기술 문서입니다.  
TECHDOC-20051020

## RemoteCall Ver 4.0 Technical Report

Page : 1

RemoteCall 4.0 버전은 이전 버전 보다 가장 기술적으로 진보된 부분이 원격제어 엔진입니다. 즉 DDI방식의 VRVD(Virtual Remote Video Driver)라는 가상 원격 비디오 드라이버를 이용한 원격제어 방식(DDI방식)으로 변경되었습니다. VRVD는 기존의 Message Hooking방식의 화면 제어 방식보다 속도가 30% 이상 빠르며 CPU가 40% 이상 감소된 획기적인 드라이버입니다. 이에 이전의 GDI방식과의 차이점을 쉽게 설명하여 기술적인 이해를 돕고자 합니다.

## DDI ( Device Driver Interface ) 방식 이란?

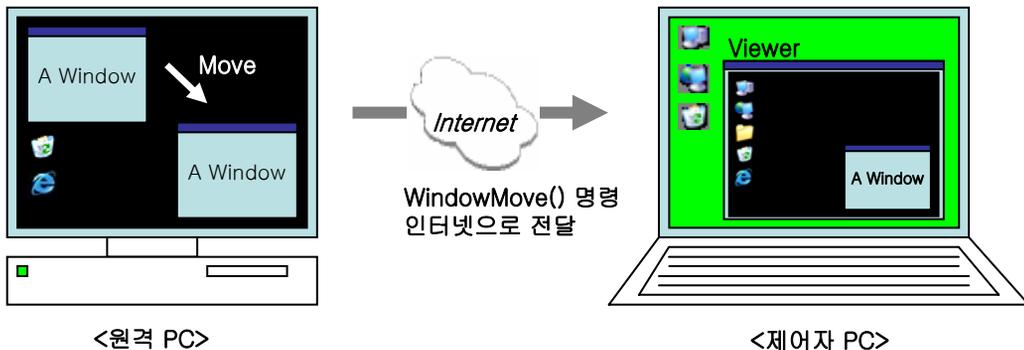
DDI방식이란 Windows OS 의 Kernel Level Driver 로 개발되어 Video Driver처럼 동작하게 만드는 방식으로 속도가 가장 빠르며 화면의 변화 량을 알아내기 위한 알고리즘이 복잡하지 않으므로 CPU 점유 량 등을 획기적으로 줄일 수 있는 기술 방식입니다. 그러나 이런 가상의 비디오 드라이버를 개발하기 위해서는 Windows OS에서 동작되는 실제 Video Driver와 Interface를 동일하게 만들어야 하기 때문에 많은 어려움이 있습니다. 그러나 RemoteCall에서는 이런 고도의 기술을 필요로 하는 DDI Driver 방식을 완벽하게 개발하여 원격제어 속도와 CPU속도 문제를 한꺼번에 해결할 수 있게 되었습니다. 즉 가상의 원격 비디오 드라이버(VRVD)를 OS에 Plug-In 함으로 최고의 원격제어 속도를 구현할 수 있는 것입니다.

- DDI 방식은 Kernel Level Driver 이므로 안정성이 보장되어야 함으로 개발이 어렵습니다.
- 모든 비디오 드라이버와의 호환성이 보장되어야 함으로 더욱 까다롭습니다.
- 전세계적으로 DDI방식을 지원하는 원격제어 프로그램은 몇 되지 않습니다. (알서포트가 세계최고 입니다)
- DDI 방식의 대표적인 예가 Window Terminal Service 입니다.

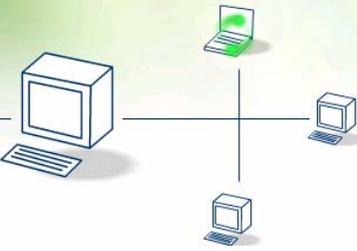
## 왜 원격제어에 DDI 방식이 빠른가?

원격제어는 원격지의 PC화면을 보여주며 제어가 가능하도록 하는 기능을 제공합니다. 즉 원격지의 컴퓨터에서 변화되는 윈도우 화면을 재빨리 알아낸 후 변화된 영역을 고속 압축하여 네트워크를 통하여 제어자 뷰어에 디스플레이 하는 기술입니다. 이런 변화된 윈도우 영역을 가장 빨리 알아내고 전송하는 방법은 비디오드라이버를 만들면 간단해 집니다. 즉 윈도우 OS는 자신의 OS에서 변화되는 윈도우를 표현하기 위해 비디오 드라이버에게 신호를 보내게 됩니다. 이때 OS에서 보내지는 명령을 가로채어 원격지의 제어자 뷰어에 표시하게 함으로써 가장 적은 데이터를 보냄과 동시에 빨리 표현할 수 있는 것입니다. 예를 들면 다음과 같습니다.

원격PC의 한 윈도우가 좌측 상단에서 우측하단으로 이동되었다면 OS는 WindowMove()와 같은 비디오 드라이버 함수를 호출합니다. 이때 원격 비디오 드라이버가 이 명령을 가로채어 원격지로 전송한다면 OS가 내린 명령만 네트워크로 전송하면 되기 때문에 아주 적은 패킷이 전송되며 빠른 윈도우 이동을 보장할 수 있게 되는 것입니다.



1. OS → Video Driver에게 이전에 표시된 윈도우를 메모리상에서 이동할 것을 명령
2. 이 명령은 이동될 위치를 포함하는 WindowMove()함수와 같음
3. VRVD엔진은 이 명령을 가로채어 네트워크로 전송 (Packet Data:약 20 Byte 이내)
4. 원격지 Viewer에서 동일명령을 수행



# DDI Mode & GDI Mode?

본 기술 문서에 대한 의문 사항은  
본사로 문의하시면 더욱 자세히  
답변해 드리겠습니다.

## RemoteCall Ver 4.0 Technical Report

Page : 2

### GDI ( Graphic Device Interface) 방식이란?

DDI 방식에 비해 GDI 방식은 RemoteCall 4.0 이전 버전에서 사용하던 방식입니다. GDI는 Application Level에서 Window Message를 처리하는 방식으로, OS에서 비디오투드라이버로 명령이 전달되고 디스플레이에 표시가 된 이후 Application에서 알 수 있기 때문에 원격제어 속도가 약간 느립니다. 그러나 개발하기 편하며 DDI와 같이 특정 비디오투드라이버등과 같이 특성을 타지 않기 때문에 비교적 쉽게 개발할 수 있습니다. 또한 이 방법은 CPU 사용량과 패킷처리량이 많아지기 때문에 시스템 성능이 좋지 않거나 네트워크 속도가 느린 경우 원격제어 성능이 떨어지는 경향이 있습니다. 대부분의 원격제어 프로그램이 이 방식을 사용하고 있습니다.

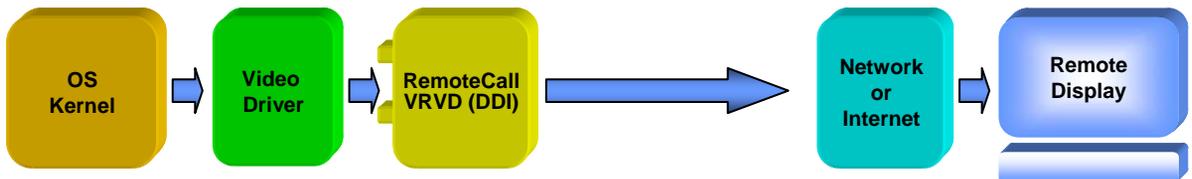
- 대표적으로 VNC라는 공개 소프트웨어가 이 방식을 사용합니다.
- 화면 변화 량을 알아내기 위한 복잡한 알고리즘이 필요하므로 CPU사용량이 높습니다.
- 성능이 떨어지는 PC나 전송속도가 느린 네트워크에서는 원격제어가 원활하지 못합니다.
- 변환된 화면 이미지를 전송함으로 네트워크의 패킷 량이 많습니다.
- 대부분의 원격제어 제품이 이 방식을 사용합니다.

그러나 RemoteCall에 적용된 GDI 엔진은 개선된 알고리즘 적용으로 일반적인 VNC방식의 제어속도보다 약 50% 이상 속도가 빠르며 완벽한 이미지 동기화로 전혀 사용에 불편함이 없도록 설계된 세계최고의 GDI엔진입니다.

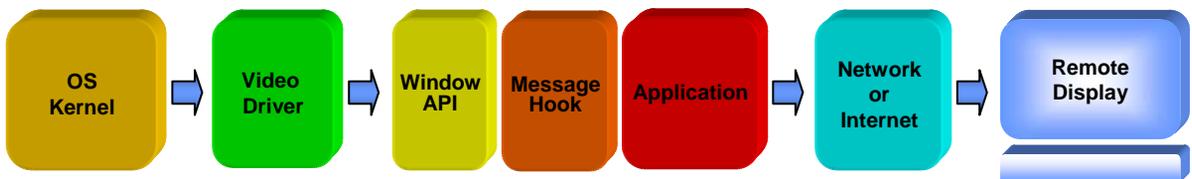
### DDI 와 GDI의 제어 흐름도

리모트콜은 DDI방식인 가상원격비디오투드라이버(VRVD)방식 과 일반적인 Message Hooking 방식인 GDI 방식 등 두 가지 제어 모드를 지원합니다. 각각의 제어 흐름도를 보시면 DDI와 GDI가 어떤 단계를 거쳐 제어가 되는지 쉽게 알 수가 있습니다.

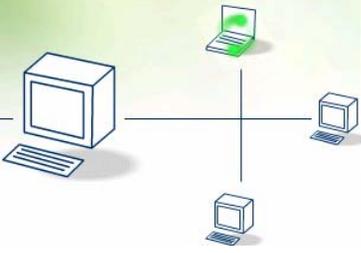
#### ● DDI 방식 제어 흐름도



#### ● GDI 방식 제어 흐름도



DDI방식은 간단히 가상 비디오투드라이버를 통과하도록 설계가 가능하기 때문에 많은 단계를 거치지 않습니다. 그러나 GDI방식은 윈도우의 변화를 알아내기 위하여 Message Hook Procedure 와 이와 연관된 Application Layer를 거쳐 변화 량을 확인하기 때문에 많은 시간이 걸리는 것입니다. 그러나 RemoteCall 에 적용된 GDI 엔진은 현존하는 GDI 엔진 중 가장 뛰어난 엔진입니다. RemoteCall 의 DDI방식에 비해 약 20% 정도 제어속도가 떨어지나 실제 제어자 입장에서는 크게 느낄 수 없습니다.



# DDI Mode & GDI Mode?

본 기술 문서에 대한 의문 사항은  
본사로 문의하시면 더욱 자세히  
답변해 드리겠습니다.

## RemoteCall Ver 4.0 Technical Report

Page : 3

### 그럼 RemoteCall은 왜 DDI 와 GDI를 둘 다 지원하는가?

RemoteCall에서 두 가지 제어모드를 지원하는 이유는 여러 가지가 있습니다. 사용자 계정에 따른 문제도 있으면 이전버전과의 호환성 문제도 있습니다. 다음은 두 가지 방식을 모두 지원하는 이유입니다.

- DDI방식은 Administrator 권한이 있는 계정에서만 동작 됩니다. User 권한에서는 GDI만 지원합니다.
- 즉 일반 계정에서는 GDI방식으로만 동작되기 때문에 모든 유저를 지원하기 위해서는 두가지 모두 필요합니다.
- AVI, WMV, ASF 와 같은 미디어 파일을 완벽하게 재생하기 위해서는 DDI방식이 필요합니다.
- DirectX 와 같이 비디오 드라이버에 직접 화면 데이터를 쓰는 프로그램일 경우는 GDI방식이 필요합니다.
- Windows 9x 계열은 GDI방식만을 지원합니다. 물론 RemoteCall은 자동으로 GDI방식으로 접속됩니다.

위와 같이 계정에 따라, 특정 프로그램에 따라 또는 윈도우 OS버전에 따라 사용할 수 있는 경우가 있고 사용할 수 없는 경우가 있습니다. 이를 모두 지원하기 위해서는 각각의 방식을 모두 지원해야 하는 것입니다. 그러나 대부분의 원격제어 제품은 단지 GDI방식만 지원하고 있습니다. 그래서 잘 살펴보면 어떤 화면은 원격제어에서 보이나 어떤 프로그램은 검은색으로 화면이 변해도 보이지 않는 경우가 있는 것입니다.

### DDI와 GDI의 기능 비교

원격제어 동작 기능	DDI방식	GDI방식
제어 속도	◎ (가장 빠름)	○ 보통
CPU 사용량	◎ (아주 적음)	△ (높음)
네트워크 트래픽 (윈도우 이동 시 실제 패킷 량)	◎ (20 Byte 이내)	△ (약 2K ~ 10K, 이미지 전송)
Caching 기능 (이전화면을 저장하여 더욱 빨리 보여주는 기능)	◎	X
동영상 화면 재생	○ (모두 재생)	△
모든 화면을 제어 가능	○	△ (윈도우 테두리 등 일부)
부드러운 화면 제어 (윈도우 스크롤 등)	◎ (아주 부드러움)	X (지원 안됨)
User 계정에서의 제어	△ (설치되어 있는 경우는 가능)	○
DirectX를 사용하는 프로그램 제어	△ (프로그램에 따라 다름)	○
Windows 9x 계열 지원	X (RemoteCall에서 지원안함)	○

리모트콜은 원격제어에 필요한 두 가지 방식을 모두 지원합니다. 즉 상황에 따라 자동으로 제어모드 방식을 변경합니다. 그러므로 사용자는 자신의 제어방식을 굳이 변경할 필요는 없습니다.