



Windows[®] Embedded Compact

Getting Started with RemoteFX in Windows Embedded Compact 7

Writers: Randy Ocheltree, Ryan Wike

Technical Reviewer: Windows Embedded Compact RDP Team

Applies To: Windows Embedded Compact 7

Published: January 2012

Abstract

This implementation overview of Microsoft RemoteFX on a Windows Embedded Compact powered thin client provides the following information:

- A description of both the server and client features of RemoteFX
- Hardware recommendations for thin client devices
- Design guidelines for display drivers
- Implementation fundamentals for display drivers
- Licensing requirements

Contents

Introduction to Using RemoteFX in Windows Embedded Compact 7	3
RemoteFX Overview	3
Windows Server RemoteFX Support	4
Windows Server RemoteFX Features	4
Windows Embedded Compact 7 Remote Desktop Client RemoteFX Support	5
Windows Embedded Compact 7 RemoteFX Features	5
Client-Side Software Decoding for RemoteFX.....	5
Client-Side Hardware-Accelerated Decoding for RemoteFX.....	5
Differences Between RemoteFX Support in Windows 7 and Windows Embedded Compact 7	6
Recommendations and Requirements	7
Remote FX Hardware Requirements	7
Windows Server Hardware Requirements	7
Windows Embedded Compact 7 Thin Client Hardware Recommendations.....	7
RemoteFX Software Requirements	8
Designing and Implementing a Thin Client RemoteFX Display Driver	8
Using a Coprocessor for Hardware Acceleration.....	9
Display Driver Design Guidelines	10
Display Driver Design Choice: Window-Frame or Full-Screen Display Mode	10
Performance Optimization for the RemoteFX Thin-Client Display Driver	12
Implementing a Thin-Client RemoteFX Display Driver	13
Conclusion	14
Additional Resources	15

Introduction to Using RemoteFX in Windows Embedded Compact 7

Microsoft RemoteFX adds new functionality to both the server and client sides of the Remote Desktop experience. RemoteFX is a new feature in Windows Server 2008 R2 with Service Pack 1 (SP1) and also in the Windows Embedded Compact 7 (Compact 7) implementation of the Remote Desktop Protocol (RDP) 7.1. RemoteFX introduces a set of enhancements to RDP that benefit the users of your clients. These enhancements provide a full desktop environment over a local area network (LAN) to users of a range of client devices, which include rich clients, thin clients, and ultrathin clients.

A Compact 7 thin client with RemoteFX support provides a full remote desktop experience when you use your thin client together with a RemoteFX-enabled Windows server. RemoteFX uses complete server-side rendering of a Remote Desktop session, which allows a RemoteFX-enabled client that runs on Compact 7 to provide users with a full RDP experience, such as high-quality full-screen video and the Windows Aero theme.

With Compact 7, you can use the new RemoteFX enhancements for RDP to build a low-cost RemoteFX-enabled thin-client device. This article describes what you need to start developing a RemoteFX-enabled thin client by giving you the following information:

- An overview of the RemoteFX technology in [RemoteFX Overview](#).
- The necessary hardware, software, and licensing requirements for using RemoteFX in [Recommendations and Requirements](#).
- Guidance on developing a display driver for your thin client in [Designing and Implementing a Thin Client RemoteFX Display Driver](#).

When you use RemoteFX and Compact 7 to build a thin client, you receive these benefits:

1. Client-side hardware acceleration.
2. Minimal hardware requirements for the client device, which provides the opportunity for a new category of extremely low-cost clients.
3. Easy integration into existing Microsoft network infrastructures, for example, integration into a secure enterprise network.
4. The ability to create customized device-based software.

RemoteFX Overview

Microsoft RemoteFX introduces a set of user experience enhancements in RDP that enable a full, local-like desktop environment for remote desktops. RDP is designed to provide remote display and input capabilities over network connections for Windows-based applications that run on a server.

By using advanced codecs and virtual graphic resources on a server that runs Windows Server 2008 R2 with SP1, RemoteFX adds support for most types of graphical applications, which include three-dimensional (3-D), and also adds rich multimedia support for virtual desktops.

RemoteFX technology is different from traditional software-only codecs because it uses the hardware on servers that run Windows Server 2008 R2 SP1 to efficiently encode data before the data is sent to a client. The RemoteFX codec was designed to efficiently compress data, a software design that satisfies the goals of high quality and low latency while it uses a modest amount of computer resources.

Windows Server RemoteFX Support

When using RemoteFX for Windows Server 2008 R2 with SP1, you can deploy centralized Windows virtual desktops that are hosted by a Hyper-V server. RemoteFX-enabled clients can provide a full, high-fidelity user experience. RemoteFX uses server hardware to encode data from either the RD Virtualization Host or the RD Session Host (a role service in Remote Desktop Services) before the data is sent to a client. RemoteFX supports scalable virtual desktop deployments by using graphics processing units (GPUs) and GPU virtualization. RemoteFX for Windows server provides other RDP performance enhancements that are listed in the following section.

Windows Server RemoteFX Features

RemoteFX for Windows Server 2008 R2 with SP1 reduces the workload on RemoteFX-enabled RDP clients by using these features:

1. **Host-side rendering**, which uses the Windows host server, not the RDP client, to render graphics.
2. **Graphics processing unit (GPU) virtualization**, which shares GPU hardware resources on the host server with RDP clients. GPU virtualization only works with virtual machines running on a Hyper-V server.
3. **Intelligent screen capture**, which checks screen content between frames and transmits for encoding, only the changes that the user made to the screen. It also tracks and adjusts network speed according to available bandwidth.
4. **RemoteFX encoder**, which encodes bitmaps for transfer from the RD Session Host or RD Virtualization Host server to the RDP client. The RemoteFX encoder compresses screen data by using the host CPU or GPU, or other dedicated hardware, such as an application-specific integrated circuit (ASIC).

For more information about RemoteFX features in Windows Server 2008 R2 SP1, see [Microsoft RemoteFX](http://go.microsoft.com/fwlink/?LinkId=208193) (<http://go.microsoft.com/fwlink/?LinkId=208193>).

Windows Embedded Compact 7 Remote Desktop Client RemoteFX Support

RemoteFX on Compact 7 supports both hardware-accelerated decoding and software decoding. Hardware-accelerated decoding offers the best performance, especially on low-end hardware. We recommend that you use hardware decoding on your thin clients.

Software decoding requires a CPU processor that is enabled with the Streaming SIMD Extensions 2 (SSE2) instruction set and a large amount of system memory. Ultrathin clients typically do not have the required processing power to achieve a quality remote access experience with RemoteFX software decoding. We recommend that you use hardware-accelerated decoding when developing your thin client.

Windows Embedded Compact 7 RemoteFX Features

The following features on the Compact 7 RemoteFX-enabled client, work together with a server that runs Windows Server 2008 R2 SP1 with RemoteFX features to provide a full remote computing experience:

- **RemoteFX decoding**, which decodes the bitmaps that are transferred from the virtual desktop on the client computer by using software in the CPU or by using a hardware decoder. The flexibility to use the CPU or a hardware decoder helps provide a RemoteFX experience to a variety of client devices that range from rich clients to ultrathin clients.
- **Hardware-accelerated decoding**, which uses coprocessor hardware such as ASIC on the RemoteFX-enabled client device to decompress RemoteFX-compressed data. The coprocessor hardware does the work instead of the client CPU, which increases client CPU performance.

Client-Side Software Decoding for RemoteFX

Software decoding requires a CPU processor that is enabled with an SSE2 instruction set and a large amount of system memory. When you use software decoding, the client CPU handles RemoteFX data decompression. Software decoding does not require any additional processors or driver interfaces.

RemoteFX software decoding uses the SSE2 instruction set for acceleration of signal-processing operations. Because of this feature design, software decoding of RemoteFX data can only be performed on a CPU that supports this instruction set.

Client-Side Hardware-Accelerated Decoding for RemoteFX

RemoteFX hardware-accelerated decoding offloads data decompression to a coprocessor, which reduces the workload on the CPU of the client device. By using hardware-accelerated decoding, you can develop a Compact 7 thin client that uses less powerful hardware compared to the typical thin clients that are used with a remote server.

For example, the client receives drawing commands from the remote computer and then processes these commands to accurately decode what is being sent from the server. When you use hardware-accelerated decoding, the client can simply pass along a compressed bitmap that is received from the

server to specialized RemoteFX hardware. The RemoteFX-enabled hardware decompresses the data and displays it, pixel-for-pixel, as it is seen on the remote computer. Hardware acceleration, by using a coprocessor to display RemoteFX data, can significantly reduce the cost of remote-experience thin clients because relatively less-powerful processors and less RAM are required.

Differences Between RemoteFX Support in Windows 7 and Windows Embedded Compact 7

The following table lists the differences between RemoteFX support in Windows 7 and Windows Embedded Compact 7 (Compact 7).

Table 1: Differences in RemoteFX Support Between Windows 7 and Compact 7

Feature	Windows 7	Windows Embedded Compact 7
RemoteFX encoder (only supported on Windows Server 2008 R2 SP1)	Not supported	Not supported
DirectX 9 GPU decoding	Supported	Not supported
Full-screen mode	Not supported	Supported as follows: <ul style="list-style-type: none"> Offers a Windows Embedded Compact RDP option that you enable through the registry. Accommodates devices that have limited physical memory. Saves memory by reducing allocated Graphics Device Interface (GDI) resources.
RemoteFX can be removed from a Windows Embedded Compact OS image by using a SYSGEN variable	Not supported	You can remove RemoteFX from a Compact 7 OS image by using one of these methods: <ol style="list-style-type: none"> By unsetting the SYSGEN_RDP_REMOTEFX variable. By using Platform Builder to disable the RemoteFX Catalog Item in the Thin Client image.

Recommendations and Requirements

To configure your device as a RemoteFX-enabled Windows server and a Windows Embedded Compact 7 (Compact 7) thin client with RemoteFX decoding support, your device must meet certain hardware, software, and licensing requirements.

Remote FX Hardware Requirements

A RemoteFX-enabled Windows server and a RemoteFX-enabled thin client must meet the following hardware requirements. For more information about the hardware requirements for deploying RemoteFX, see [Hardware Considerations for RemoteFX](http://go.microsoft.com/fwlink/?LinkId=208194) (<http://go.microsoft.com/fwlink/?LinkId=208194>).

Windows Server Hardware Requirements

A RemoteFX-enabled server must meet the following hardware requirements:

- The server runs Windows Server 2008 R2 Service Pack 1 (SP1) with Hyper-V.
- The server also meets the hardware requirements that are listed in the [Windows Server 2008 R2 Technical Library](http://go.microsoft.com/fwlink/?LinkId=208195) (<http://go.microsoft.com/fwlink/?LinkId=208195>).

Windows Embedded Compact 7 Thin Client Hardware Recommendations

The following configuration is recommended for thin-client hardware so that your devices can take advantage of RemoteFX functionality when they connect to virtual desktops through an RD Virtualization Host or RD Session Host server:

- Display chipset that supports 32 bits per pixel (BPP) color depth.
- 100 megabits per second (Mbps) or greater, wired network adapter. For better performance, use a 1 gigabit per second (Gbps) network adapter.
- Human interface device (HID) mouse and keyboard.
- 128 MB or larger primary storage volume.
- 512 MB RAM or greater.
- 32 BPP hardware cursor support.

The following configurations are also recommended:

- **Local time of both the server and client match:** The method for setting the accurate time in the operating system at run-time. The operating system time can be set by a real-time clock (RTC), a network time service, or another method.

 **Important**

When you establish a secure RDP connection to the Windows server, if the local time of the operating system does not match the time stamp of the certificate being issued by the server, the Windows server rejects the RDP session request.

- **Latency of LAN network:** RemoteFX is optimized for a LAN network. A LAN network that has a latency of 20 milliseconds or less is recommended.

In addition, the following decoding requirements and recommendations apply:

- **RemoteFX software decoding:** Requires a CPU processor that is enabled with an SSE2 instruction set. For best results, use a 2 gigahertz (GHz) CPU. If the CPU speed is lower than 400 megahertz (MHz), a RemoteFX session is not established. Hardware-accelerated BitBlt in the device driver is recommended, which helps reduce workload on the CPU.
- **RemoteFX hardware-assisted decoding:** Requires a 1 GHz ARM or x86 CPU and a RemoteFX decoder, such as an ASIC or a digital signal processor (DSP), that has a corresponding RemoteFX display driver. For better performance, it is recommended that the display driver also support hardware-accelerated BitBlt.

RemoteFX Software Requirements

You must meet the following software requirements to take advantage of RemoteFX functionality when your device connects to virtual desktops or the RD Session Host server.

- **Windows server software requirements:** For a list of specific editions of Windows Server 2008 R2 SP1 that support RemoteFX, see [Microsoft RemoteFX](http://go.microsoft.com/fwlink/?LinkId=208193) (<http://go.microsoft.com/fwlink/?LinkId=208193>).
- **Windows Embedded Compact 7 thin client software requirements:** RemoteFX software requirements are Windows Embedded Compact 7 with support for Remote Desktop Connection 7.1.

Designing and Implementing a Thin Client RemoteFX Display Driver

RDP for Windows Embedded Compact 7 can decompress RemoteFX-compressed data in one of two methods:

- By using CPU-based software decoding, which requires a CPU processor that is enabled with the SSE2 instruction set. When you use software decoding, the client CPU handles RemoteFX data decompression. Software decoding does not require any additional processors or driver interfaces.
- By using hardware-accelerated decoding, which offloads RemoteFX data decompression to a coprocessor on the device. For example, the coprocessor might be an application-specific integrated circuit (ASIC) or a digital signal processor (DSP).

Using a Coprocessor for Hardware Acceleration

Client CPU performance is enhanced by using a coprocessor for hardware-accelerated decoding of RemoteFX data. Dedicated coprocessor hardware, such as an ASIC, can perform decoding of RemoteFX data. The coprocessor requires a less powerful CPU and less memory on your thin client. You can obtain significant performance improvements over CPU-based software decoding when you combine coprocessor offloading with the high-compression ratio of RemoteFX.

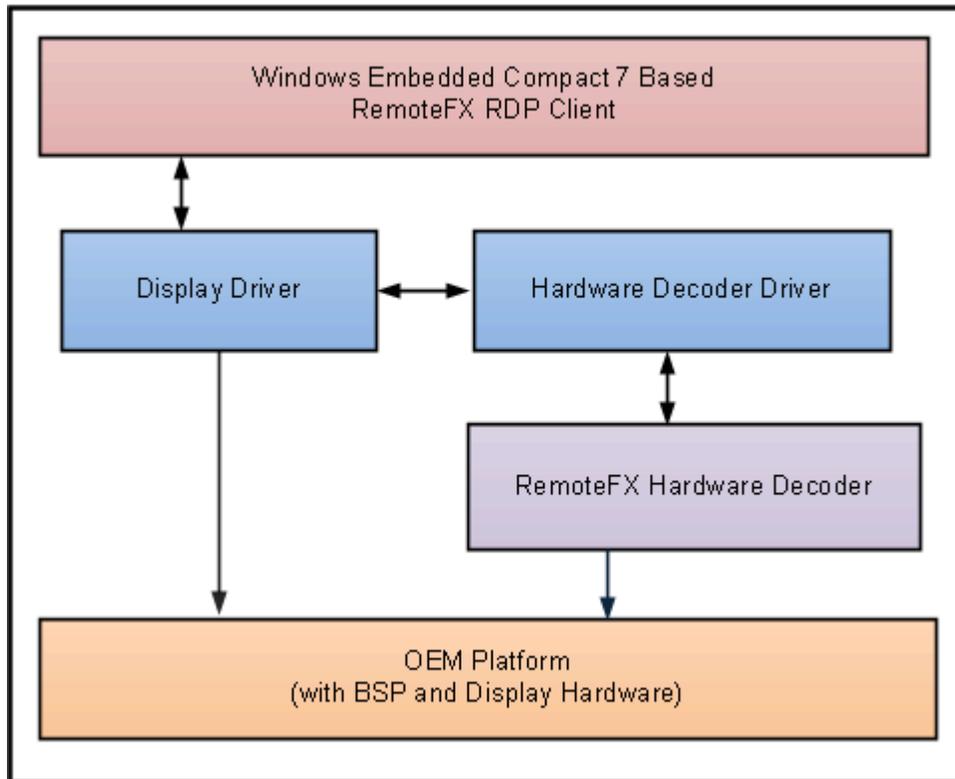
If you decide to build a thin client that uses coprocessor offloading to decompress RemoteFX compressed data, you also need to develop a display driver. Your display driver must support a specified set of RemoteFX interfaces and the driver must communicate with hardware that is capable of processing RemoteFX data.

When you use coprocessor offloading, the RDP client receives RemoteFX packets from the server, processes these packets, and then sends the data to a RemoteFX-enabled display driver. The display driver is responsible for processing the RemoteFX data and making the processed data available to the hardware decoder coprocessor. The RDP client then determines where to display the data on a rendering surface. The RDP client interacts with a display driver via a set of well-defined interfaces that use Graphics Device Interface (GDI) escape sequences.

The following diagram shows the relationships among the different elements that are required to create a RemoteFX-enabled thin client that uses hardware decoding. It also shows the dependent components that you need to create, such as the display driver and the hardware decoder driver that work with a RemoteFX-enabled client.

The RemoteFX-encoded data stream from the server is passed down through the RDP stack, sent to the hardware decoder for decoding, and then displayed through the display controller.

Figure 1: RemoteFX Thin Client Architecture with Hardware Decoder



System integrators who want to develop RemoteFX hardware or to implement the RemoteFX algorithm in a coprocessor need to obtain a RemoteFX Partner license. Partners who are interested in obtaining this information are encouraged to contact their Technical Account Manager to learn more about becoming a RemoteFX Partner. If you are not a partner, see the [Microsoft RemoteFX](http://go.microsoft.com/fwlink/?LinkId=215879) (<http://go.microsoft.com/fwlink/?LinkId=215879>) site for information about joining the partner program.

Display Driver Design Guidelines

Compact 7 uses a display driver model for hardware acceleration of RemoteFX because during a RemoteFX session, the RDP client might request decoding for large amounts of display data. When Compact 7 uses the display driver to process data, any decompressed data can be sent directly to a display controller, without being sent through numerous memory-transfer operations.

Display Driver Design Choice: Window-Frame or Full-Screen Display Mode

You can develop a RemoteFX display driver by using either of the following methods:

- As a full-featured display driver that supports RDP window-frame mode, where the display surface is shared between the RDP window and other local applications.

Figure 2: Window-Frame Display Mode



- As a full-screen-only display driver (only used for hardware-assisted decoding). You set full-screen mode in the registry of the client device by using the **FullScreenRFXOnly** subkey. A full-screen-only hardware-assisted display driver makes board development and prototyping faster because display data can be sent directly to the frame buffer from the decode buffer (see figure 4). A full-screen-only display driver also reduces CPU load at run time.

Figure 3: Full-Screen Display Mode



Performance Optimization for the RemoteFX Thin-Client Display Driver

To optimize the performance of your RemoteFX display driver, consider the following:

- Do not use a software Graphics Primitive Engine (GPE) because it puts more load on the CPU. You can obtain optimal display performance when the video card is capable of supporting the full set of Graphics Device Interface (GDI) raster operations on Compact 7.
- Use direct memory access (DMA) whenever possible to move the decoded data in video memory.
- Use a hardware-accelerated BitBlt operation.

Implementing a Thin-Client RemoteFX Display Driver

The RDP client interacts with a display driver via a set of well-defined interfaces that use GDI escape sequences. You need to use the GDI escape sequence interface to implement your thin-client RemoteFX display driver. In addition to supporting the typical graphics-rendering operations, the display driver must support the following four RemoteFX escape sequences.

Table 2: RemoteFX Escape Sequences That the Display Driver Must Support

Escape Sequence	Description
ESCAPE_GET_CAPABILITIES	Allows the codec library in the RDP client to query hardware capabilities (for example, the supported tile size) from the display driver.
ESCAPE_DEC3	Allows the codec library to decode compressed data into video memory buffer and optionally, to transfer decoded data back to system memory.
ESCAPE_COPY_2BMP	Allows the codec library to copy decoded data directly to the shadow bitmap.
ESCAPE_EXIT_CA	When full-screen mode is enabled and the RemoteFX session ends, the RDP client sends the ESCAPE_EXIT_CA escape sequence to the display driver. The display driver uses this notification to change back to non-RemoteFX display mode.

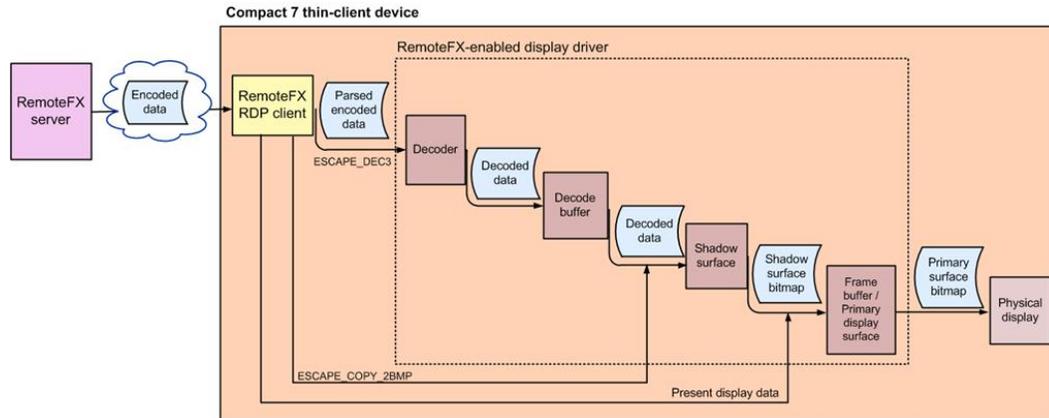
The process that is used by RemoteFX hardware-accelerated decoding to offload coprocessor work from the RDP client to the Windows server follows this sequence:

1. On initialization, the RDP client gets the hardware capabilities from the display driver by using the ESCAPE_GET_CAPABILITIES escape sequence and then negotiates RDP capabilities from the server.
2. At run time, the RDP client receives and parses the RemoteFX tile set data from the server.
3. The client offloads the encoded tile set data to the display driver for decoding by using the ESCAPE_DEC3 escape sequence.
4. The decoder processes the data and outputs the decoded data to the decode buffer.
5. The RDP client uses the ESCAPE_COPY_2BMP escape sequence to move decoded data from the decode buffer to the shadow surface. Hardware acceleration is recommended to transfer the decoded bitmap to the shadow surface.

- The RDP client presents the shadow bitmap content into the frame buffer, which is then output to the physical display. Hardware-accelerated BitBlt is recommended to transfer the shadow bitmap to the frame buffer.

The following figure represents this process in a diagram.

Figure 4: RemoteFX Hardware Offload Flow Chart



System Integrators who want to develop RemoteFX hardware or implement the RemoteFX algorithm in a coprocessor need to obtain a RemoteFX Partner license. Partners who want to obtain this information are encouraged to contact their Technical Account Manager to learn more about becoming a RemoteFX Partner. If you are not a partner, see the [Microsoft RemoteFX](http://go.microsoft.com/fwlink/?LinkId=215879) (<http://go.microsoft.com/fwlink/?LinkId=215879>) site for information about joining the partner program. RemoteFX software-only decoding does not require a RemoteFX Partner license.

For specific performance guidelines, implementation details, and code examples, you should read [Designing and Implementing RemoteFX Display Drivers in Windows Embedded Compact 7](http://go.microsoft.com/fwlink/?LinkId=219976) (<http://go.microsoft.com/fwlink/?LinkId=219976>).

Conclusion

Microsoft RemoteFX adds new functionality to both the server and client sides of the Remote Desktop experience. RemoteFX is a new feature that is included in Windows Server 2008 R2 with Service Pack 1 (SP1) and in the Windows Embedded Compact 7(Compact 7) implementation of the Remote Desktop Protocol (RDP) 7.1.

RemoteFX introduces a set of RDP enhancements that benefit the users of your clients. These enhancements provide a full near-local desktop environment over a LAN to users of a range of client devices, which include rich clients, thin clients, and ultrathin clients.

You can use the new RemoteFX enhancements in Compact 7 to configure a low-cost thin client. A Compact 7 thin client with RemoteFX support provides a full rich remote desktop experience when you

use your thin client together with a RemoteFX-enabled Windows server. RemoteFX uses complete server-side rendering of the Remote Desktop session, which allows an RDP client that runs on Compact 7 to offer full remote computing experiences, such as high-quality full-screen video and the Windows Aero theme.

RemoteFX technology is different from traditional software-only codecs because RemoteFX uses hardware on servers that run Windows Server 2008 R2 SP1 to efficiently encode data from either the RD Virtualization Host or RD Session Host server before the data is sent to a client.

RemoteFX on Compact 7 supports both hardware-accelerated decoding and software decoding. Hardware-accelerated decoding offers the best performance, especially on low-end hardware. The flexibility to use the CPU or a hardware decoder helps provide a RemoteFX experience to a variety of client devices that range from rich clients to ultrathin clients.

Additional Resources

- [Windows Embedded Compact 7 Documentation](http://go.microsoft.com/fwlink/?LinkId=190787) (http://go.microsoft.com/fwlink/?LinkId=190787)
- [Microsoft RemoteFX](http://go.microsoft.com/fwlink/?LinkId=215879) (http://go.microsoft.com/fwlink/?LinkId=215879)
- [Designing and Implementing RemoteFX Display Drivers in Windows Embedded Compact 7](http://go.microsoft.com/fwlink/?LinkId=219976) (http://go.microsoft.com/fwlink/?LinkId=219976)
- [RemoteFX Technet Wiki](http://go.microsoft.com/fwlink/?prd=11857&pver=7.1&sbp=Windows%20Embedded%20Compact&plcid=0x409&clid=0x409&ar=Windows%20Embedded%20Compact%20TechNet&sar=RemoteFX%20TechNet%20Wiki) (http://go.microsoft.com/fwlink/?prd=11857&pver=7.1&sbp=Windows Embedded Compact&plcid=0x409&clid=0x409&ar=Windows Embedded Compact TechNet&sar=RemoteFX Technet Wiki)

This document is provided “as-is.” Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2011 Microsoft. All rights reserved.