WPF: Hardware vs. Software Acceleration

WPF

- It uses hardware optimizations where possible, but it has a software fallback for everything
- WPF **offers some sort of hardware acceleration to all WDDM drivers and to XPDM drivers** that were created after November 2004
- When WPF infrastructure **first starts up**, it evaluates your video card and assigns it a **rating** from 0 to 2
  - **RenderCapability.Tier** property: provides level of hardware acceleration that's available in the client
    - in **System.Windows.Media** namespace
    - *int renderingTier = (RenderCapability.Tier >> 16);*
- The goal of WPF **is to off-load as much of the work as possible on the video card** so that complex graphics routines are render-bound (limited by the GPU) rather than processor-bound (limited by your computer's CPU).
- WPF recognizes three rendering tiers (<WPF 4):
  - **Rendering Tier 0**. The video card will not provide any hardware acceleration. This corresponds to a DirectX version level of less than 7.0.
  - **Rendering Tier 1**. The video card can provide partial hardware acceleration. This corresponds to a DirectX version level greater than 7.0 but less than 9.0.
  - **Rendering Tier 2**. All features that can be hardware accelerated will be. This corresponds to a DirectX version level greater than or equal to 9.0.

- Starting in the .NET Framework 4, rendering tier 1 has been redefined to only include graphics hardware that supports DirectX 9.0 or greater. *Graphics hardware that supports DirectX 7 or 8 is now defined as rendering tier 0*.
- **Factors**:
    - The amount of RAM on the video card
    - Support for pixel shaders (built-in routines that calculate per-pixel effects such as transparency)
    - Support for vertex shaders (built-in routines that calculate values at the vertexes of a triangle, such as the shading of a 3-D object).
    - **Multitexture Support:** Multitexture support refers to the ability to apply two or more distinct textures during a blending operation on a 3D graphics object

Feature	Tier 1	Tier 2
DirectX version	Must be greater than or equal to 9.0.	Must be greater than or equal to 9.0.
Video RAM	Must be greater than or equal to 60MB.	Must be greater than or equal to 120MB.
Pixel shader	Version level must greater than or equal to 2.0.	Version level must greater than or equal to 2.0.
Vertex shader	No requirement.	Version level must greater than or equal to 2.0.
Multitexture units	No requirement.	Number of units must greater than or equal to 4.

Graphics Rendering Registry Settings

WPF provides four registry settings for controlling WPF rendering:

Setting	Description
Disable Hardware Acceleration Option	Specifies whether hardware acceleration should be enabled. HKEY_CURRENT_USER\SOFTWARE\Microsoft\Avalon.Graphics\DisableHWAcceleration
Maximum Multisample	Specifies the degree of multisampling for antialiasing 3-D content.

Value	HKEY_CURRENT_USER\SOFTWARE\Microsoft\Avalon.Graphics\MaxMultisampleType
Required Video Driver Date Setting	Specifies whether the system disables hardware acceleration for drivers released before November 2004.
Use Reference Rasterizer Option	Specifies whether WPF should use the reference rasterizer. HKEY_CURRENT_USER\SOFTWARE\Microsoft\Avalon.Graphics\UseReferenceRasterizer

Guidelines for troubleshooting graphic issues in WPF applications

- http://support.microsoft.com/kb/963021
- Graphic issues in WPF applications can include any of the following symptoms:
    - A WPF window that fails to refresh.
    - A WPF window that contains distorted and corrupted visuals.
    - On Windows Vista, the screen flickers.
    - On Windows XP, a blue screen crash sometimes occurs.
    - The occurrence of a Timeout Detection and Recovery (TDR).
- Steps to resolve graphic issues in WPF applications
    - The first step is to install the most recent version of the .NET Framework.
    - The next step is to obtain and install the most recent drivers for your graphics card.
    - If the issue persists, try launching your application on another computer with a different model/brand of graphics card and the most up-to-date drivers. This will indicate whether the issue is due to bugs in the display drivers. Occasionally, the most recent display drivers may not contain a fix for graphic issues.
    - If updating the display driver does not resolve the issue, and if you do not have a second computer to debug the issue, try disabling hardware acceleration and forcing your WPF application to use software rendering.
    - If the issue is resolved by using software rendering and it is not possible for you to upgrade the physical graphics card or its drivers, then it is recommended that you disable hardware acceleration and force your WPF application to use software rendering as a workaround for your application.
    - If none of the previous steps resolves the issue, then send the DirectX Diagnostics log from your computer to Microsoft so that it can be further analyzed.

References

http://msdn.microsoft.com/en-us/library/ms742196(VS.100).aspx

http://msdn.microsoft.com/en-us/library/aa970912.aspx

http://support.microsoft.com/kb/963021

Matthew MacDonald, Pro WPF in C# 2010: Windows Presentation Foundation in .NET 4.0