Windows 7 Developer Guide

Published October 2008

Updated June 2009; May 2009; February 2009; December 2008; November, 2008;

RICHER APPLICATION EXPERIENCES..8

CONTENTS

ABSTRACT

Building applications that are easy to use, visually appealing, and offer high performance is a challenge that developers face every day. Innovative applications can greatly improve the user experience, empowering companies to differentiate their services and solutions. However, developers are increasingly asked to do more in less time, while also optimizing the power and performance requirements of their applications.

The Windows 7 platform makes it easy for developers to create engaging, user-friendly applications by providing familiar tools and rich development features that allow them to take advantage of the latest PC capabilities.

INTRODUCTION

Developing software has never been more exciting. The world has never been more connected, interactive and mobile. User experiences and content have never had higher fidelity. Advances in connectivity, mobility, natural interfaces, graphics and media are enabling new scenarios and opportunities for developers. At the same time, expectations have never been higher for performance, flexibility and interoperability. A world-class developer platform must provide a flexible and complete foundation for developers and their solutions, empowering them to build applications that set them apart from the competition while maximizing their investments.

The Windows® 7 operating system is the essential platform for developers. Listening to our customers, looking ahead to the scenarios of tomorrow, and building on the capabilities of Windows Vista®, Microsoft offers developers a wide variety of choices and capabilities within the Windows developer platform, while empowering developers to deliver creative solutions that are mobile-aware, connected, high-fidelity, and provide a highly intuitive user experience. Most importantly, developer platform fundamentals such as security, performance, and compatibility are top priorities in Window 7.

This guide summarizes the key developer advances in each of the following three areas:

Solid Foundation

Great user experiences start with a solid foundation. Windows 7 delivers a solid development platform and innovative tools that give you more options, increased application compatibility, better performance, and sophisticated document support. This results in a simple, more reliable environment for your applications and a familiar, intuitive one for your users.

We've kept the user interface consistent, predictable and easy to use, while adding multi-touch support, scrolling, and other intuitive interaction features that take advantage of the latest PC and mobile device capabilities. With Windows 7, developers have the power to build the right user experiences for their applications. For more information see Windows User Experience Interaction Guidelines (http://go.microsoft.com/?linkid=9669659).

Richer Application Experiences

Windows 7 enables developers to build applications today that will run on the PCs of tomorrow. Service-enabled software and devices, such as mobile phones, portable media players, and digital cameras, demand constant connectivity and advanced applications. Windows 7 delivers a platform for both, making it easy for developers to take advantage of the functionality and features of next generation hardware, while ensuring that users are always on, and always connected. For more information see Windows Hardware Developer Central (http://go.microsoft.com/?linkid=9669663)

The Best of Windows and the Web

With Windows 7, it's easy to get connected and stay connected. Windows networking offers developers options for better caching and sharing of data to improve network performance. Network diagnostics in Windows 7 give developers relevant information for monitoring network issues. Internet Explorer™ 8 is the essential platform for building fast, rich Web experiences. Whether the goal is to connect devices, connect to a network, or connect to the Internet, Windows 7 provides an enhanced platform. For more information see Web Development in the MSDN Library (http://go.microsoft.com/?linkid=9669664)

SOLID FOUNDATION

Windows 7 provides a highly productive developer platform and tools that deliver on core operating system fundamentals. Windows 7 builds upon the Windows Vista platform, giving developers the power to build applications that are compatible with both platforms while limiting application compatibility issues. Windows 7 dramatically improves performance and power management so that you can create applications that optimize the mobile experience. As a result, your applications will be more visually appealing, easier to create, and support a wider range of international standards. Advances have also been made in global support, accessibility, and application deployment.

Compatibility and Reliability

Windows 7 is designed to run on the same hardware as Windows Vista, and to be compatible with applications and device drivers that work with Windows Vista.

Windows 7 is the most reliable version of Windows yet. Designed on an improved technology foundation, Windows 7 allows users to reliably start up, shut down, or hibernate their computers without having to worry about losing valuable work. Furthermore, Windows 7 makes it easier than ever to back up and restore data to network drives or DVDs. Windows 7 also improves upon print reliability and performance. For more information see Windows 7 Application Quality Cookbook (http://go.microsoft.com/?linkid=9669662).

Applications

To help ensure compatibility, Windows 7 has been designed in close partnership with software vendors and PC manufacturers. Early engagement has enabled Microsoft to build a comprehensive list of the most widely used applications. Automated testing cycles ensure that compatibility issues are detected and fixed early in the development cycle. For more information see Windows Application Compatibility (http://go.microsoft.com/?linkid=9669665).

Drivers

The Windows Driver Kit (WDK) Version 7.0.0 provides the build environment, tools, documentation, and samples that

developers need to create quality drivers for Windows. The WDK 7.0.0 supports static source code analysis, using PREfast to detect certain classes of C and C++ coding errors. PREfast includes a specialized driver component, known as PREfast for Drivers (PFD), which detects errors in kernel-mode driver code. In addition, the WDK has been enhanced by annotating all kernel header files for PFD support. New sample drivers have been added that demonstrate new technologies, and the documentation has been expanded.

Windows 7 supports a large variety of software and hardware products designed to integrate seamlessly with the platform. Drivers that were created for Windows Vista should not require updating to run correctly in Windows 7. For more information see the Windows Driver Kit (http://go.microsoft.com/?linkid=9669667).

Devices

Windows 7 provides flexible, robust support for a wide variety of applications and devices, including music players, storage devices, mobile phones, and other types of connected devices. Automatic testing of these devices is used to ensure that compatibility issues are fixed early in the development cycle. For more information see Windows Device Class Fundamentals (http://go.microsoft.com/?linkid=9669666).

Reliability Access Monitor

Reliability Analysis Component is an in-box agent that provides detailed customer experience information on system usage and reliability. This information is exposed through a Windows Management Instrumentation (WMI) interface, making it available for consumption by Portable Readers Systems. By exposing Reliability Analysis Component through a WMI interface, developers can monitor and analyze their applications, increasing reliability and performance,

Windows 7 uses the built-in Reliability Analysis Component to calculate a reliability index which provides information about your overall system usage and stability over time. Reliability Analysis Component also keeps track of any important changes to the system that are likely to have an impact on stability, such as Windows updates and application installations. You can use the Reliability Monitor snap-in to see trends in your system's reliability index correlated with these potentially destabilizing

events, making it easy to trace a reliability change directly to a particular event. For more information see MountVHD Function (http://go.microsoft.com/?linkid=9669668).

Management and Deployment

IT professionals or developers preparing to deploy Windows 7 will have increased confidence and experience a shorter evaluation cycle due to improvements in imaging features and tools. These include support for managing applications, drivers, and operating systems in offline image files. Additionally, image creation and management will be easier and will be available to a broader range of IT organizations. Deploying Windows 7 to business PCs will also be easier and faster because of new IT migration tools and automated deployment technologies.

Windows PowerShell 2.0

Windows PowerShell™ is a complete .NET managed scripting language with both an interactive command line shell and a graphical Integrated Scripting Environment (ISE). It supports branching, looping, functions, debugging, exception handling, and internationalization. PowerShell 2.0 is part of Windows 7 and delivers many enhancements and a growing set of cmdlets for Windows Diagnostics, Active Directory®, Internet Information Services (IIS) and more.

The PowerShell 2.0 remoting feature now allows users to run commands on one or more remote computers from a single computer running Windows PowerShell. Developers can also host Windows PowerShell on Internet Information Services to access and manage their servers.

> **Windows PowerShell is a complete .NET managed scripting language with both an interactive command line shell and a graphical Integrated Scripting Environment (ISE)**

PowerShell 2.0 supports partitioning and organizing PowerShell scripts using modules that can be distributed and deployed as self-contained, reusable units. It also includes transactions support in the PowerShell engine and APIs, which means that developers can start, commit, and rollback transactions using built-in transaction cmdlets. Further, the PowerShell engine

```
Windows PowerShell Integrated Scripting Environment (ISE)

File   Edit   Debug   View   Help

Script

if(("-?","-help") -contains $args[0])
{
    Write-Host "Description: This script runs a consistency check on all the actively protected datasources " `
        "which are not in valid state."
    Write-Host "Usage: Validate-InvalidDatasources.ps1 [-DPMServerName] <Name of the DPM server>"
    Write-Host "Example: Validate-InvalidDatasources.ps1 mohitc02"
    exit 0
}
if (!$DPMServerName)
{
    $DPMServerName = Read-Host "DPM server name"
    if (!$DPMServerName)
    {
        Write-Error "Dpm server name not specified."
        exit 1
    }
}
if (!(Connect-DPMServer $DPMServerName))
{
    Write-Error "Failed to connect To DPM server $DPMServerName"
    exit 1
}
$jobList = @()
foreach ($datasource in @(Get-Datasource -DPMServerName $DPMServerName | ? {$_.Protected -and $_.State -eq "Invali
{
    $jobList += @{Job = Start-DatasourceConsistencyCheck -Datasource $datasource; Datasource = $($datasource.Logic
}
$completedJobsCount = 0

PS C:\Users>
>

Completed                                                                        12
```

includes eventing support for listening, forwarding, and acting on management and system events. PowerShell applications can be written to subscribe to certain events for synchronous or asynchronous processing. For more information see Windows PowerShell (http://go.microsoft.com/?linkid=9669669).

Windows Installer

Windows Installer has been updated to increase developer efficiency by reducing the amount of custom code required to create an installation package and create true per-user software installations.

Multiple Package Transaction allows developers to create a single transaction from multiple packages, using a "chainer" to dynamically include packages in the transaction. If one or more of the packages do not install as expected, simply roll back the installation.

Embedded UI Handler makes custom UIs easier to integrate by embedding a custom user interface handler in the Windows Installer package.

Embedded Multiple Package Chainer allows developers to enable installation events across multiple packages. For example, they can enable install-on-demand events, repair events, and uninstall events across multiple packages.

New features also enable the creation of true per-user installations, including support for per-user program files and "elevate now" functionality, and provide support for offline software inventory and patch applicability checks through Deployment Image Servicing and Management. For more information see What's New in Windows Installer 5.0 (http://go.microsoft.com/?linkid=9669677).

Security

Windows 7 includes new and improved security features that make it easier for developers to improve, use, and manage the security of their applications. It comes with a variety of new security features that not only help protect against threats but also limit the damage that attackers can do if they gain access to a computer.

Enhancements to the Windows Filtering Platform allow developers to create applications that interact with the packet processing in the networking stack of the operating system. Network data can be filtered and also modified before it reaches its destination.

Also, due to changes to the Windows privilege model, system security is more manageable by both developers and their end users. New improvements make it easy to identify critical prompts to ensure that users can access the applications and features they need without compromising their systems. For more information see the MSDN Security Developer Center (http://go.microsoft.com/?linkid=9669672).

Windows Filtering Platform

In Windows 7, the Windows Filtering Platform has been enhanced to give developers more control over firewall functionality. The level of filtering has been increased and ISVs can now plug in custom protection and detection at lower levels. In addition, firewall developers can selectively turn parts of the Windows Firewall on or off.

Using Windows Filtering Platform, developers can build firewalls, intrusion detection systems, antivirus programs, network monitoring tools, and parental controls into their applications. Windows Filtering Platform integrates with and provides support for a wide variety of firewall features, including authenticated communication and dynamic firewall configuration based on applications' use of sockets API (application-based policy). Windows Filtering Platform also provides infrastructure for policy management, change notifications, network diagnostics, and stateful filtering.

The initial architecture of Windows Filtering Platform in Windows Vista® provided capabilities for IP-based traffic. Other non-IP protocols—such as such as Address Resolution Protocol (ARP) and media access control (MAC)-layer protocols for network management and authentication—also require filtering, inspection, or logging. In Windows 7, an NDIS inspection layer that supports MAC and ETHERNET filtering has been provided to satisfy this need. For more information see Windows Filtering Platform (http://go.microsoft.com/?linkid=9669676).

User Account Control

User Account Control is a security component in Windows 7 that allows developers to build applications that enable users to perform common tasks as non-administrators. Developers can reduce security risks by running applications under a standard user token, reducing the risks of mistakes or attacks.

User accounts that are members of the local Administrators group will run most applications as a standard user. By separating user and administrator functions while enabling productivity, User Account Control gives developers greater control over the level of access that users have over protected areas of an application. User Account Control requests credentials in a Secure Desktop mode, where the entire screen is protected to prevent spoofing of the user interface or the mouse. For more information see User Account Control Dialog Updates (http://go.microsoft.com/?linkid=9669674) and User Account Control and WMI (http://go.microsoft.com/?linkid=9669675).

Performance

Windows 7 maximizes hardware energy efficiency and scalability while maintaining high performance. Energy efficiency is improved through reduced background activity and new support for the trigger starting of system services. Windows 7 also offers improvements in the Windows kernel that enable applications and services to scale efficiently between platforms.

Performance of many features and APIs is improved in Windows 7 versus Windows Vista. For example, driver performance on servers is optimized by new user-mode and kernel-mode topology APIs. Graphics rendering is considerably smoother and faster. Accessibility performance is also significantly faster than before.

Building Power-Efficient Applications

Building energy efficient applications that take advantage of the latest power management technologies is a significant challenge developers are facing today. Typically, processor and device manufacturers get all of the attention as their latest offerings are measured and benchmarked. However, a single application can easily prevent the latest generation of hardware from realizing its energy-efficiency potential. For example, a single application that increases the platform timer resolution can decrease battery life by 10 percent.

Extended operation on battery power and the use of energy efficient technologies are key requirements for today's developers. Windows 7 greatly reduces the number of activities that the operating system performs that prevent use of power-saving modes. It also supports the trigger-starting of system services to enable processors to become idle more often and stay idle longer, which decreases power consumption. In addition, Windows 7 takes advantage of the latest energy-efficient hardware, including network adapters, storage devices, and graphics cards.

Windows 7 provides the infrastructure and tools that make it easy for developers to determine the energy impact of their applications. A set of event callbacks enable applications to reduce their activity when the system is on battery power and automatically scale up when the system is on AC power. For applications that involve a background process or service,

Windows 7 features new infrastructure to automatically enable background tasks when most appropriate in order to maximize energy efficiency. For more information see WHDC Performance Central (http://go.microsoft.com/?linkid=9669670) and Power Management in Windows 7 Overview (http://go.microsoft.com/?linkid=9669671).

Service Control Manager

The Windows 7 Service Control Manager has been extended so that a service can be automatically started and stopped when a specific system event, or trigger, occurs on the system. Trigger-start capabilities remove the need for services to start up automatically at computer startup and then poll or wait for an event to occur, such as device arrival. Common trigger events for services include:

Device-class interface arrival: Start a service only when a certain type of device is present or attached on the system.

Domain join: Start a service only if the system is joined to a Windows domain.

Group policy change: Start a service automatically when group policies are refreshed on the system.

IP address arrival: Start a service only when the system is connected to the network.

Software developers can use the predefined trigger types for Windows 7 and the configuration options to enable trigger-start capability. The Windows 7 Service Control Manager exposes a new set of APIs that enable a service to register for specific custom trigger events. For more information see Service Control Manager (http://go.microsoft.com/?linkid=9669673).

Windows Troubleshooting Platform

Windows 7 delivers a comprehensive and extensible Troubleshooting Platform that uses a PowerShell-based mechanism to troubleshoot and resolve problems. The key components of the Troubleshooting Platform include a troubleshooting package, troubleshooting engine, and troubleshooting wizard. The troubleshooting pack is a collection of PowerShell scripts and relevant metadata. The troubleshooting engine launches a PowerShell runtime to

Troubleshoot problems with your computer
Click on a category or use the Search box to find problems to troubleshoot.

Programs
Run programs made for previous versions of Windows
Browse the web with Internet Explorer
Display my music and video library in Windows Media Player

Hardware and Sound
Configure a device | Use a printer | Record sound | Play sound

Network and Internet
Connect to the Internet | Access shared files and folders on other computers

Appearance and Personalization
Show the Aero theme

System and Security
Run maintenance tasks | Improve power usage | Make your computer run faster

Updated troubleshoting content is available online.
Use online content ▼

execute a troubleshooting pack, and exposes a set of interfaces to control troubleshooting pack execution.

The troubleshooting wizard provides a consistent experience across troubleshooting packs, communicating with the troubleshooting engine to troubleshoot and resolve problems that are specified in a troubleshooting pack. Execution of a troubleshooting pack can also be controlled through a set of PowerShell commandlets.

The Troubleshooting Platform seamlessly integrates with the Windows 7 PC Solution Center, enabling other applications to execute diagnostics in a similar manner as part of their PC management regimen. The Troubleshooting Platform is configurable by IT professionals through Group Policy for use within the enterprise, and a Windows Troubleshooting Toolkit that allows developers to author troubleshooting packs is also available. For more information see Windows Troubleshooting Platform (http://go.microsoft.com/?linkid=9669687).

> **The Troubleshooting Platform seamlessly integrates with the Windows 7 PC Solution Center**

Documents and Document Peripherals

Windows 7 provides developers with a robust platform for working with documents and integrating document peripherals. Two new document and storage technologies were introduced in Windows Vista: the XML Paper Specification (XPS) and Open Packaging Conventions (OPC). These technologies, which were available in Windows Vista only to developers of managed-code applications through the .NET Framework, are now available in the Windows 7 SDK for use by developers of unmanaged code.

Open Packaging Conventions

Windows 7 supports all Open Packaging Conventions file formats, including those from Microsoft as well as those from third parties. OPC is a component of the Office Open XML (OOXML) international specification defined through ISO/IEC DIS 29500 and ECMA-376. Based on the ZIP file format, OPC enables applications to store a combination of data items within a single package file. Application developers can use the Packaging APIs in Windows 7 to create, read, and manipulate multiple data elements in OPC-based files.

Using the Packaging APIs in Windows 7, developers can create new package formats to accommodate application-specific data storage requirements.

X509 digital signatures are also supported by the Packaging APIs. Developers can use the digital signature features to sign and validate selected parts of an OPC package or the entire package. Applications can give their documents an added level of security by using digital signatures to detect when the content of an OPC-based file has been altered after the file was signed. For more information see Open Packaging Conventions Overview (http://go.microsoft.com/?linkid=9669685).

XPS Documents

Windows application developers can create applications that produce XML Paper Specification documents with Windows 7. This enables them to integrate tightly with the document peripheral ecosystem (devices like scanners and printers) and

to work with secure electronic paper to support publication and archiving.

In previous versions of Windows, XPS was not supported for Win32 developers. XPS was introduced in Windows Vista but the API surface was limited to .NET developers working with managed code. With Windows 7, Win32 developers can use the new XPS Document APIs to reduce the amount of work required when working with XPS. Since XPS is the foundation for the new Windows print platform, that's a significant benefit.

In previous versions of Windows, access to the XPS Print Path from Win32 applications was limited to driver escapes. This significantly reduced the utility of the print path for developers not using managed code

For Win32 developers, the new XPS Print API reduces significantly the amount of work required to benefit from the advantages of the XPS Print Path and eliminates the need for parallel print code.
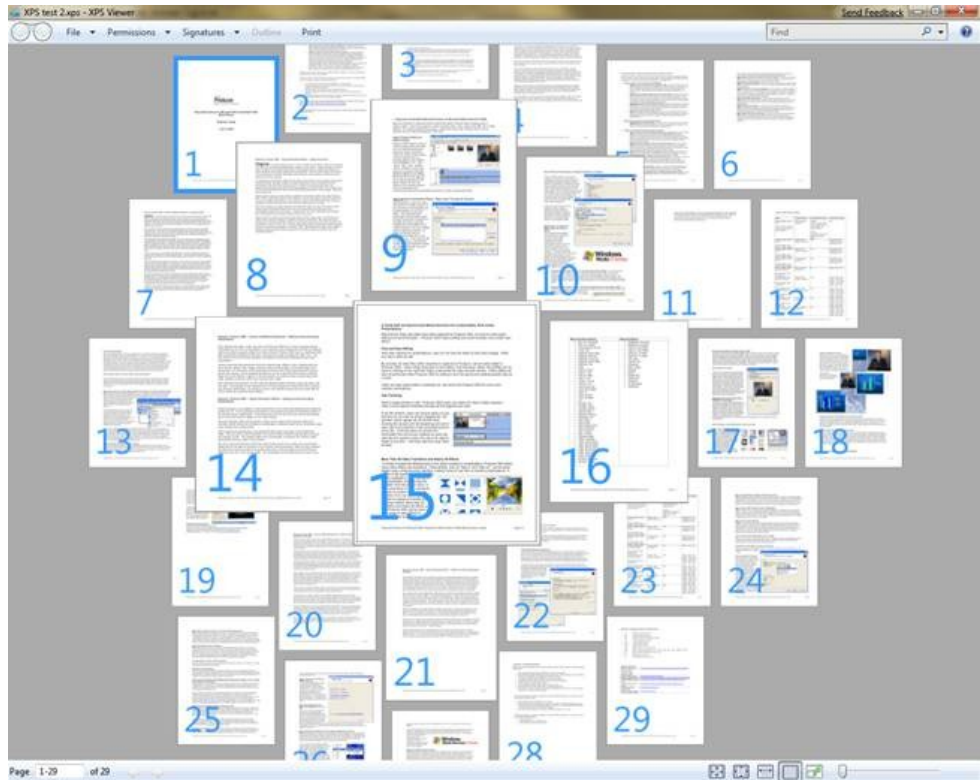
Application developers can use XPS documents to share and archive content as electronic paper in a high-fidelity, efficient, and trustworthy format. Just like Windows Vista, the print path in Windows 7 is built on the XPS format to provide enhanced printing capabilities. The XPS document APIs in Windows 7 give developers the power to create, access, and manipulate XPS documents easily. For more information see XPS Document Programming Guide (http://go.microsoft.com/?linkid=9669688).

> **Windows application developers can create applications that produce XML Paper Specification documents with Windows 7**

Accessibility and Global Support

The Windows 7 platform makes it easier to build solutions that are accessible to more users and that meet or exceed accessibility compliance standards. The Assistive Technology Vendor (ATV) community can now build solutions for a broader variety of client applications, and application developers will find it easier to build and validate accessible user interfaces.

Windows 7 also makes supporting multiple global languages easier than in previous versions of Windows. From the time a user selects a language and location, Windows 7 presents dates, numbers, calendars, collations, and other information using the cultural conventions that customers expect.

Windows Automation

Windows 7 delivers a rich, standards-based automation layer that is extended for native applications. It builds on Microsoft Active Accessibility and Microsoft UI Automation. It's also designed to work with industry standards such as the W3C Web ARIA (Accessible Rich Internet Application) and Section 508 Specifications.

UI Automation offers improved performance by introducing faster unmanaged automation proxies for Win32 controls and legacy Microsoft Active Accessibility (MSAA) applications, and better and faster UI Automation event and proxy registrations. New extensibility features extend control patterns, properties, and custom events. For more information see Windows Automation API: Overview (http://go.microsoft.com/?linkid=9669686).

Accessibility Support Tools

The UI Accessibility Checker is a convenient graphical user interface tool that enables developers and testers to rapidly verify whether their UI conforms to key accessibility requirements, such as MSAA (which verifies child-parent relationships or bounding rectangles) and UI Automation programmatic access, event generation, layout, and keyboard navigation. For more information see UI Accessibility Checker (http://go.microsoft.com/?linkid=9669680).

UIA Verify is a test automation framework that facilitates manual and automated testing of the Microsoft UI Automation Provider implementation of a control or application. These two new tools enable developers to test accessibility implementations and functionality in applications that use either MSAA or UI Automation. Both tools are available via CodePlex (http://www.codeplex.com), a Web site that Microsoft created to host open-source projects and to better serve the developer community. For more information see UI Automation Verify (UIA Verify) Test Automation Framework (http://go.microsoft.com/?linkid=9669681).

Improved Multi-Language User Interface Support and Linguistic Services

Windows 7 provides developers with a standard method to prepare their applications for the international market by delivering an improved multi-language user interface support and linguistic services that they can use in their applications.

Extended Linguistic Services is a new feature in Windows 7 that allows developers to use the same small set of APIs to leverage a variety of advanced linguistic functionality. By using Extended Linguistic Services APIs in Windows 7, developers can auto-detect the language of any piece of Unicode text and use that information to help make smarter user experience choices for customers around the world. Extended Linguistic Services also offers built in transliteration support that converts text from one writing system to another. For example, developers can now auto-convert text between Simplified and Traditional Chinese to help people communicate with each other across linguistic boundaries. By using Extended Linguistic Services APIs, developers will be able to use existing Extended Linguistic Services as well as pick up new services in the future without learning new code. For more information see Extended Linguistic Services (http://go.microsoft.com/?linkid=9669682).

RICHER APPLICATION EXPERIENCES

Windows 7 enables developers to create distinctive and intuitive applications that significantly enhance discoverability, usability, and sheer enjoyment. New methods of desktop integration put application functionality right at the user's fingertips, and Windows Explorer and Libraries provide easy access to high-value information. The Scenic Ribbon control and animation framework make it easier to build interactive and appealing user interfaces. New touch APIs enable natural interactions through multi-touch and finger-panning, and manipulation and inertia APIs enable impressive visual effects.

Rapid advances in hardware and software technology are also driving higher-fidelity user experiences. Windows 7 brings these advances under developer control with new and flexible APIs that take full advantage of the technology, while making it even easier to develop compelling applications.

Intuitive User Experience

For the first time, Windows 7 allows developers and their end-users to control their computers by touching the screen. Touch and multi-touch features provide a natural, intuitive way for users to interact with PCs. The developer platform includes high-level gesture APIs, as well as low-level touch messages and touch input APIs. The top-level UI elements, such as the Start menu and taskbar, have larger targets than previous Windows releases, making them easier to select with a finger instead of a mouse. Visual feedback is provided for tap and double tap. Windows Explorer and Internet Explorer 8 are both touch friendly and easily integrated with Windows 7 applications.

Multi-Touch Gestures, and Manipulation and Inertia APIs

Windows 7 features improved touch and gesture support, empowering developers to quickly and easily create unique application experiences that go beyond simple mouse pointing, clicking, and dragging. The new multi-touch APIs support rich gestures, such as pan, zoom, and rotate. All gestures provide direct visual feedback, and interact with underlying content in a natural and intuitive manner. For example, a zoom gesture centers the view at the location of the gesture. Lower-level touch input APIs are also available for custom gesture definition and advanced touch-response experiences. Windows 7 provides a development platform that gives developers the tools they need to develop creative applications for multi-touch input devices, by processing user input from multi-touch devices and improving the user interface. The result is more intuitive environments, which enable innovations in PC interaction.

Windows 7 also provides platform support for object manipulation and inertia processing. A rich set of manipulation functions enable you to stretch, resize, or rotate multiple objects concurrently and in very fine granularity. For example, multiple digital photographs could be cropped, resized, and rotated in a single session using touch based gestures.

Windows 7 includes inertia APIs which simulate inertia when objects are moved, working hand-in-hand with the manipulation APIs. For example, in a photo application, you can use the manipulation APIs to let users rotate, resize, and move photos. Similarly, if a user "tosses" a photo, the inertia APIs provide

natural interaction and enable the photo to coast to a stop or bounce off the borders of the application's window. For more information see Windows Touch Programming Guide (http://go.microsoft.com/?linkid=9669683) and Windows Touch: Developer Resources (http://go.microsoft.com/?linkid=9669684).

Single-Finger Panning

In many common applications, touch features are more useful for navigation than for text selection. With extended touch APIs, a developer's application can choose to enable panning rather than dragging. For example, if you created an application that uses multi-touch gestures for users playing music, you could allow these users to simply slide a finger up or down to adjust the volume, change songs, or download a file. No scrolling required.

Windows 7 provides endless opportunities for developers who are interested in creating applications for next-generation PCs. Best of all, it does the hard work of checking for scroll bars and implementing the panning semantics. Applications also receive a richer set of events and feedback for customized control of gestures than they did in previous versions of Windows. For more information see Improving the Single-Finger Panning Experience (http://go.microsoft.com/?linkid=9669697).

Raw Touch Input Data

In Windows 7, new touch experiences are enabled by interaction models that access lower-level touch input messages, and provide customized responses to combinations of touch messages. The platform supports receiving raw touch input data for scenarios like multi-touch painting applications and custom gestures within an application. You can use the platform support for touch or create your own original, multi-touch experiences. For more information see WM_TOUCH Message (http://go.microsoft.com/?linkid=9669696).
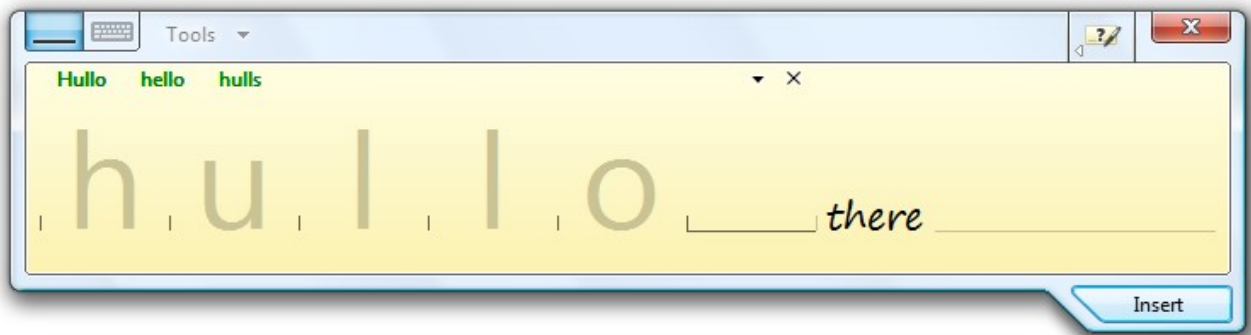
Handwriting and Ink

With the proliferation of Tablet PCs in the market, Tablet features are becoming part of mainstream computing. In Windows 7, touch and writing are first-class user experiences. Windows 7 improves the pen experience by providing greater accuracy and speed. Handwriting input is improved and more languages are supported. The Text Input Panel offers predictive text for greater speed of input and correction. Handwriting accuracy is improved through personalization in all languages, custom dictionaries, and breakthroughs in East Asian language recognition. The improved interaction model delivers a better reading experience on the small, high-resolution screens common in portable computers. For more information see Programming the Text Input Panel (http://go.microsoft.com/?linkid=9669692).

> **The Text Input panel features easy text correction**

Math Recognition

The new Math Recognition feature enables users to enter math into applications by means of handwriting—the most natural and efficient way of entering mathematical expressions. The functionality is provided by two UI components. Math Input Panel is a stand-alone Windows accessory that works with any math-aware application. Math Input Control is integrated into applications through its API.

Underlying the UI components is the Math Recognizer. This engine recognizes handwritten mathematical expression and translates the result into MathML format for applications to use. The correction experience has been improved to help users
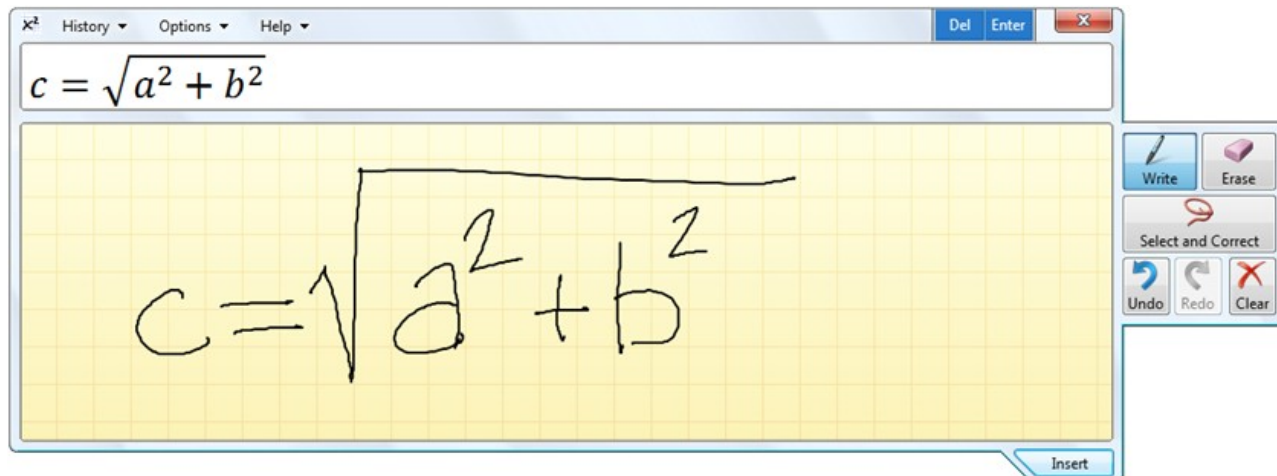
make corrections faster. For more information see Programming the Math Input Control (http://go.microsoft.com/?linkid=9669695).

> **Math Recognition enables users to enter math into applications by means of handwriting**

Handwriting with Personalized Custom Dictionary

For many scenarios, good handwriting accuracy requires a dictionary tailored to the domain of use. Windows 7 introduces custom dictionaries, which enable better handwriting recognition for specialized vocabularies. Developers who are writing vertical applications — for example, a medical prescription notepad — can now add specific terms to their application, such as drug names. For more information see Developer Enhancements to Windows Touch and Tablet PC (http://go.microsoft.com/?linkid=9669693).
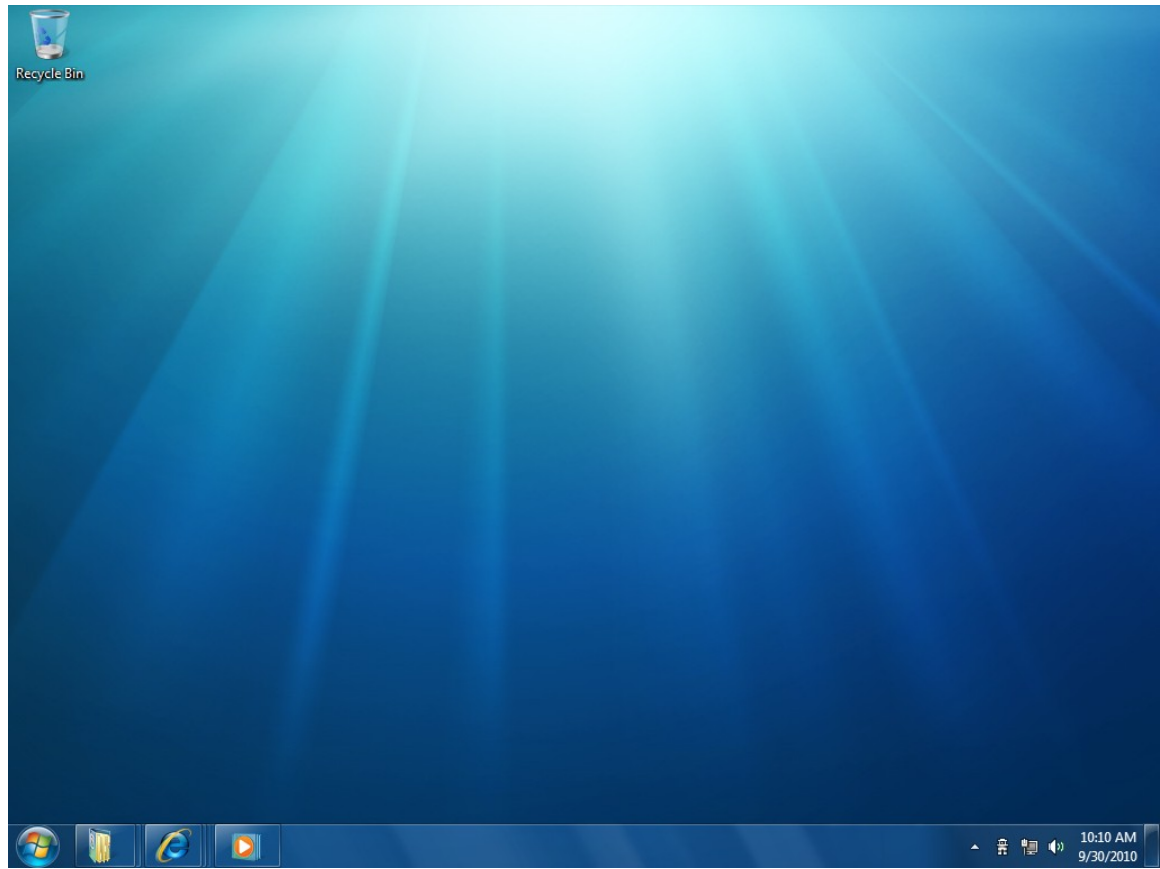
The Desktop Experience

The new Windows 7 desktop brings your applications to life. Applications are now more discoverable, informative, and interactive. Modern and intuitive user interfaces are easier to develop with Windows 7. New desktop and application experiences include the following:

The enhanced taskbar introduces interactive thumbnails, and enables animation and interaction for minimized applications.

The Destinations concept lets users jump with one click to the files, locations, or tasks that they use most often.

New controls and APIs for the Ribbon, based on the Office Fluent UI, are available for easily adding Ribbon-style controls, menus, and galleries to your applications.

An animation framework helps you enhance custom animations.

Enhancements to the gadgets platform enable applications to install companion gadgets during the setup or first-run experience.

> **The new Windows 7 desktop brings your applications to life**

Jump Lists—Getting Users into Your Application Quickly

Jump Lists help users get to where they want to go faster. Jump Lists are files, URLs, tasks, or custom items that open within the application. The new Jump Lists menu in the Start menu and taskbar makes common destinations and key tasks available with a single click. The Jump Lists menu is automatically populated based on how frequently and how recently items have been used. Developers can provide custom Jump Lists based on their own semantics. Applications can also define Tasks to appear in their menus—these are actions of the application that users want to access directly, such as composing an e-mail. For more information see Taskbar Extensions (http://go.microsoft.com/?linkid=9669694) and ICustomDestinationList Interface (http://go.microsoft.com/?linkid=9669711).

Jump Lists help users get to where they want to go faster

Enhanced Taskbar

With the new taskbar in Windows 7, applications can provide more information to the user in more intuitive ways. For example, applications can show progress bars in their taskbar buttons so that users can stay aware of progress without having to keep the window visible. This is useful for tracking time consuming operations such as file copying, downloads, installations, or media burning. Icon overlays can be displayed on the lower-right area of the application's taskbar button, and are used to convey status or notifications (such as new mail). New thumbnail APIs enable an application to define child windows and corresponding thumbnail images for those windows. The thumbnail toolbar provides a place to control

common actions without requiring window restoration, such as Play/Stop for media. For more information see Taskbar Extensions (http://go.microsoft.com/?linkid=9669694) and Windows 7 Taskbar: Developer Resources (http://go.microsoft.com/?linkid=9669706).

Gadgets Platform

Gadgets are a popular feature of the Windows Vista desktop, and in Windows 7, it's even easier for applications to install gadgets. In Windows 7, an application can programmatically add a gadget to the Windows Desktop during application setup or first run. This means that an application's out-of-the-box experience can include a simple checkbox, for example, to install a companion gadget that is available on the desktop as soon as the application is ready to be used. For more information see Introduction to the Gadget Platform (http://go.microsoft.com/?linkid=9669708).

> **In Windows 7, it's even easier for applications to install gadgets**

Windows Ribbon

Windows 7 features the Ribbon interface from Office 2007 throughout the operating system, enabling improved user interface development on the platform. This means that developers can eliminate much of the drudgery of Win32 UI development and deliver a rich, graphical, animated, and highly
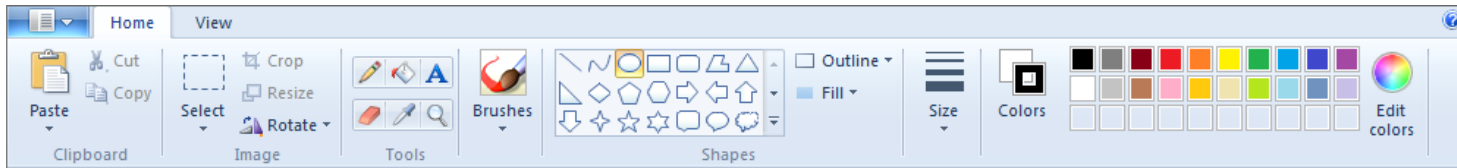
familiar user interface by using markup based UI and a small, high-performance, native code runtime.

The Windows Ribbon control helps developers improve usability by exposing your application's most frequently accessed features directly to end users. The Ribbon makes it easier for end users to find and use application features because less functionality is hidden, leading to increased productivity. The Windows Ribbon is designed as an intent-based alternative to the command presentation model of menus, toolbars, task panes, and dialog boxes in standard Windows applications.

The Windows Ribbon controls consist of a set of Microsoft Win32 APIs that override the top-level menu bar functionality and render a ribbon-style command UI instead. It is similar in functionality and appearance to the Ribbon in Office 2007. The UI is composed of a number of sub-controls that include the following:

Application button (or pearl)

Quick-access toolbar

Ribbon control of contextual tabs

Mini-toolbars

Style galleries

Templates and markup authoring are available to developers for rapid development and integration of Ribbon functionality. For more information see Windows Ribbon Framework (http://go.microsoft.com/?linkid=9669714) and Windows Ribbon Framework: Developer Resources (http://go.microsoft.com/?linkid=9669715).

> **The Windows Ribbon control helps developers improve usability by exposing your application's most frequently accessed features**

Animation

Smooth animations are fundamental to many graphical UI applications, and Windows 7 introduces a native animation framework for managing the scheduling and execution of animations. The animation framework supplies a library of useful mathematical functions for specifying behavior over time and also lets developers provide their own behavior functions. The framework supports sophisticated resolution of conflicts when multiple animations attempt to manipulate the same value simultaneously. An application can specify that one animation must be completed before another can begin and can force completion within a set time. The new framework also helps animations determine appropriate durations. For more information see Windows Animation Manager (http://go.microsoft.com/?linkid=9669704).

Managing Files and Data

Users have easier access to files and data in Windows 7. New APIs make files and views more informative, enabling applications to deliver relevant and distinctive information to Windows Explorer. In addition, applications benefit from the new Libraries model—a useful, more abstract notion of user storage space than folders—and can also participate in common libraries of similar file types that are shared by different applications.

Libraries

Windows 7 introduces the concept of Libraries as destinations where developers and end-users can find and organize their data as collections of items that can span multiple locations on the local computer as well as on remote computers.

The Library APIs provide a straightforward way for developers to create applications that create, interact with, and support Libraries as first-class items within applications. Libraries can also be selected by using the folder picker dialog box. Applications can enumerate relevant library scopes, or they can use the library directly as a folder.

Windows 7's new approach solves many of the previous API compatibility issues that previous versions struggled with, offering new API frameworks and supporting legacy frameworks through monolithic Libraries. This allows legacy applications to
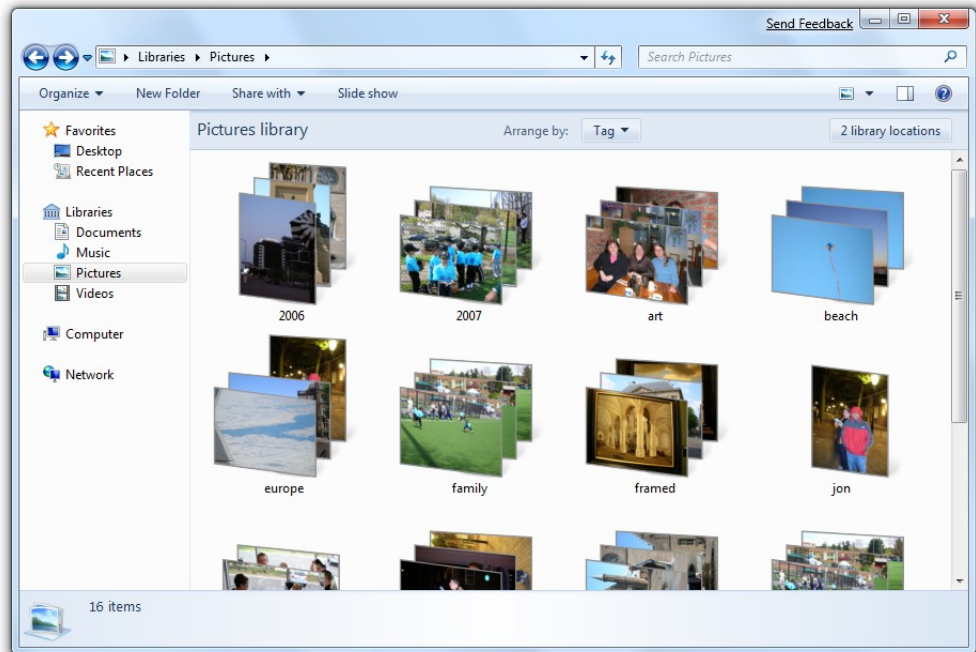
run smoothly, without compromising new performance enhancements for applications created for the Windows 7 platform. Ultimately, users can quickly and seamlessly upgrade without having to worry if their favorite programs will still work. For more information see Windows Libraries (http://go.microsoft.com/?linkid=9669712) and Windows 7 Libraries: Developer Resources (http://go.microsoft.com/?linkid=9669713).

> **Pictures Library shows your pictures no matter where they are stored**

File Formats and Data Stores

In Windows 7, Windows Explorer makes file management and manipulation easier for the user in several ways:

The preview for your application's file type is more accessible with a new button that lets users show and hide the preview pane.

Immersive visual stacks aggregate thumbnail images for file types in a view.

Windows Explorer views show useful information based on properties written with your property handler.

Document snippets and hit highlighting use your IFilter interface implementation to make searching and finding files easier.

Context-menu verbs and commands are easier than ever to implement.

By implementing all of the appropriate format handlers for the items returned from your protocol handler, search results from your custom data store can be as rich as search results from files. Libraries are automatically created for your protocol handlers so users can scope their searches easily. And the logic for creating Libraries can be easily customized through the registry. For more information see Developing Filters for Windows Search (http://go.microsoft.com/?linkid=9669707).

> **In Windows 7, Windows Explorer makes file management and manipulation easier**

High DPI

As display technologies advance, manufacturers have increased the number of pixels supported by their displays. While text, images, and user interface elements look much sharper and more readable on high resolution displays, the

operating system must scale up to support the visual experience; otherwise, everything just looks smaller.

Windows 7 supports high DPI displays. Market data suggests that deployments of high DPI screens (120-144 DPI) will increase in the Windows 7 timeframe. When running native resolutions on these screens, many applications appear very small unless they use High DPI. Some applications (such as Internet Explorer) have font scaling features that allow users to zoom in and out, but many applications do not. The High DPI feature in Windows 7:

Ensures that Windows and application experiences are optimal on standard hardware (DPI settings are optimized to match the capabilities of the hardware).

Enables the Windows Shell and other Windows-based applications to look good with different DPI settings.

Respects default DPI settings based on hardware specifications and capabilities.

Enables users to personalize DPI settings without rebooting.

Ensures the screen is always set to native resolution.

The Windows DPI scaling feature scales fonts and user interface elements (such as buttons, icons, and input fields) by a calculated percentage, as specified by the DPI setting. This is different from the scaling that occurs when the display resolution is lowered. In the case of DPI Scaling, Windows provides fonts and user interface elements that are drawn with more pixels, resulting in a larger, higher fidelity, and sharper Windows experience. Third-party Windows applications can leverage High DPI settings and adjust the user interface accordingly, by declaring themselves High DPI aware. Application developers should no longer assume that 96 DPI is the ideal resolution for all applications.

For more information see the High DPI node in the MSDN Library (http://go.microsoft.com/?linkid=9669709).

High-Fidelity Graphics with DirectX

Windows application developers have long used DirectX® to provide high-quality, hardware-accelerated, 3D graphics. When the technology debuted in 1995, developers could provide high-

quality 3D graphics for games and engineering applications for gamers and professionals willing to pay extra for a 3D-graphics board. Now, even the most inexpensive PCs include capable 3D-graphics hardware.

To take advantage of these graphics capabilities, Windows Vista introduced the Windows Display Driver Model (WDDM) infrastructure for DirectX that enabled multiple applications and services to share the resources of the GPU. The Desktop Window Manager (DWM) uses this technology to animate task switching in 3D, provide dynamic thumbnail images of application windows, and to provide Windows® Aero® glass effects for desktop applications.

Windows 7 puts even more graphics capability into the hands of application developers. Through a new set of DirectX APIs, Win32 developers can take advantage of the latest innovations in GPUs to add fast, scalable, high-quality, 2D and 3D graphics, text, and images to their applications. On the latest LCD displays, DirectX APIs can display desktop and window content using color depth greater than 8 bits per color component.

With DirectX, Win32 developers can also use the GPU's parallelism for general-purpose computation such as image processing, and can render to DirectX 10 hardware, DirectX 9 hardware, the CPU, or to a remote Windows computer. These technologies were designed to interoperate with GDI and GDI+, ensuring that developers can easily preserve their existing investments in Win32 code. For more information see What's New in the March 2009 DirectX SDK (http://go.microsoft.com/?linkid=9669710).

These enhanced graphics capabilities are provided by the following COM-based APIs:

Direct2D for drawing 2D graphics.

DirectWrite for arranging and rendering text.

Windows Imaging Component for processing and displaying images.

Direct3D® 10 for drawing 3D graphics.

Direct3D 11 for drawing 3D graphics, and providing access to next-generation GPU technologies, such as tessellation, limited support for texture streaming, and general purpose computing.

DirectX Graphics Infrastructure (DXGI) for managing devices and GPU resources, and providing interoperability between DirectX and GDI.

Direct2D

Built on Direct3D 10, Direct2D offers Win32 developers immediate-mode, resolution-independent, 2D APIs that use the power of next-generation graphics hardware, yet interoperate well with today's GDI/GDI+ applications and Direct3D 10 applications. Direct2D provides high-quality 2D rendering with performance superior to GDI and GDI+. It provides Win32 developers finer control over resources and their management. For more information see Direct2D in the MSDN Library (http://go.microsoft.com/?linkid=9669716).

DirectWrite

Many of today's applications need to support high-quality text rendering, resolution-independent outline fonts, and full Unicode text and layout support. DirectWrite, a new DirectX component, provides these features and more:

A device-independent text layout system that improves text readability in documents and in UI.

High-quality, sub-pixel, ClearType® text rendering that can use GDI, Direct2D, or application-specific rendering technology.

Hardware-accelerated text, when used with Direct2D.

Support for multi-format text.

Support for the advanced typography features of OpenType® fonts.

Support for the layout and rendering of text in all supported languages.

GDI-compatible layout and rendering.

The DirectWrite font system enables "any font anywhere" font usage, where users don't have to perform a separate installation step just to use a font, and an improved structural hierarchy of font grouping to help with manual or programmatic font discovery. The APIs support measuring, drawing, and hit-testing of multi-format text. DirectWrite handles text in all supported languages for global and localized applications, building on the

key language infrastructure found in Windows 7. DirectWrite also provides low-level glyph rendering APIs for developers who want to perform their own layout and Unicode-to-glyph processing. For more information see DirectWrite in the MSDN Library (http://go.microsoft.com/?linkid=9669718).

Windows Imaging Component

In Windows Vista, the Windows Imaging Component introduced an extensible framework for working with images and image metadata. The image formats supported by Windows Imaging Component include JPEG, PNG, and TIFF, and the supported metadata formats include XMP and EXIF. With Windows 7, Windows Imaging Component broadens its standards compliance by providing support for progressive image decoding, expanded PNG features, GIF metadata, and metadata that spans APPn segments. For more information see What's New for WIC in Windows 7 (http://go.microsoft.com/?linkid=9669722).

Direct3D 11

Direct3D 11 extends the functionality of the Direct3D 10 pipeline and provides Windows 7 games and high-end 3D applications with efficient, robust, scalable access to the upcoming generation of GPUs and multi-core CPUs. In addition to the functionality found in Direct3D 10, Direct3D 11 introduces several new features.

Geometry and high-order surfaces can now be tessellated to support scalable, dynamic content in patch and subdivision surface representations.

To make good use of the parallel processing power available from multiple CPU cores, multithreading increases the number of potential rendering calls per frame by distributing the application, runtime, and driver calls across multiple cores. In addition, resource creation and management has been optimized for multithreaded use, enabling more efficient dynamic texture management for streaming.

New general-purpose compute shaders have been created for Direct3D 11. Unlike existing shaders, these are extensions to the programmable pipeline that enable your application to do more work completely on the GPU, independent of the CPU.

DrawAuto, which was introduced in Direct3D 10, has been extended to interact with a compute shader.

Several improvements have been made to the high-level shading language (HLSL), such as a limited form of dynamic linkage in shaders to improve specialization complexity, and object-oriented programming constructs like classes and interfaces. For more information see What's New in the March 2009 DirectX SDK (http://go.microsoft.com/?linkid=9669710).

Direct3D 10 improvements

Direct3D 10 includes a redesigned graphics pipeline with programmable shader stages and immutable state objects for initializing the fixed-function stages. The state objects simplify the pipeline and improve performance by minimizing the number of state changes required. Programmability of shader stages now offers high-level shading language extensions to support unlimited shader instructions, generalized shader resources, and integer and bitwise calculations.

The pipeline also introduces the geometry shader stage, which offloads work entirely from the CPU to the GPU. This new stage enables you to create geometry, stream the data to memory, and render the geometry with no CPU interaction.

Several other improvements are designed specifically for faster performance. Predicated rendering performs occlusion culling to reduce the amount of geometry that is rendered. Instancing APIs can dramatically reduce the amount of geometry that needs to be transferred to the GPU by drawing multiple-instances of similar objects. Texture arrays enable the GPU to do texture swapping without CPU intervention.

Several additions have been made to Direct3D 10 and Direct3D 11 to extend the gamut of configurations that can be targeted with these APIs. The Windows Advanced Rasterization Platform (WARP) implements fast, multi-core scalable CPU rendering for Direct3D 10, enabling full-featured graphics rendering on systems without graphics hardware. The addition of new "Feature Levels," specifically called Direct3D 10 Level 9, allow the Direct3D 10 and Direct3D 11 APIs to drive Direct3D 9-class hardware, expanding the number of configurations a Direct3D 10 or Direct3D 11 application can target to nearly every

computer system on the market. For more information see Direct3D 10 Graphics (http://go.microsoft.com/?linkid=9669717).

DirectX/GDI Interoperability

In Windows Vista, the behavior of an application that uses both DirectX and GDI to render to a shared surface is different depending on whether Desktop Window Manager (DWM) is on or off. In addition, when DWM is on, applications that use both DirectX and GDI behave differently on Windows Vista than on Windows XP. This caused many ISVs to disable DWM when running their applications on Windows Vista to ensure consistent behavior. With the improvements to DirectX in Windows 7, an application can now freely mix DirectX and GDI without disabling DWM. Windows 7 also features improved performance for scenarios that require interoperation between DirectX and GDI by utilizing the more efficient Direct3D 10 APIs. For more information see Direct2D and GDI Interoperation Overview (http://go.microsoft.com/?linkid=9669719).

Media Platform

Media Foundation and DirectShow® provide the basis for media support in Windows. Media Foundation was introduced in Windows Vista as the replacement for DirectShow. In Windows 7, Media Foundation has been enhanced to provide better format support, including MPEG-4, as well as support for video capture devices and hardware codecs.

Format Support

In Windows 7, Media Foundation provides extensive format support that includes codecs for H.264 video, MJPEG, and MP3; new sources for MP4, 3GP, AAC audio, and AVI; and new file sinks for MP4, 3GP, and MP3. For more information see Supported Media Formats in Media Foundation (http://go.microsoft.com/?linkid=9669720).

Hardware Devices

Media Foundation now supports the following types of hardware devices in the audio/video pipeline:

UVC 1.1 video capture devices, such as webcams

Audio capture devices

Hardware encoders and decoders

Hardware video processors, such as color-space converters

Hardware codecs can perform very fast video transcoding. For example, suppose you want to transfer a Windows Media Video (WMV) file to a cell phone that supports only 3GP files. With a hardware encoder, the file can be transcoded "as needed," immediately before transferring it to the device.

Hardware devices are represented in Media Foundation by a proxy object, and are used in the pipeline just like software-based components. For more information see What's New for Media Foundation (http://go.microsoft.com/?linkid=9669721).

Simplified Programming Model

In Windows Vista, Media Foundation exposed a relatively low-level set of APIs. These APIs are flexible, but may not be appropriate for performing tasks. Windows 7 adds new high-level APIs that make it simpler to write media applications in C++. These new high-level APIs include:

MFPlay. These APIs are designed for audio and video playback. They support the typical playback operations (stop, pause, play, seek, rate control, audio volume, and so forth), while hiding the details of the low-level APIs (the session and topology layers).

Source Reader. You can use these APIs to pull raw or decoded data from a media file, without knowing anything about the underlying format. For example, you can get a thumbnail bitmap from a video file or get live video frames from a webcam.

Sink Writer. You can use these APIs to author media files by passing in uncompressed or encoded data. For example, you can re-encode or remix a video file.

Transcode. The Transcode APIs target the most common audio and video encoding scenarios.

Platform Improvements

Windows 7 includes numerous enhancements to the underlying Media Foundation platform APIs. Advanced applications can use these APIs directly; other applications will get the benefits indirectly. These benefits include:

Improvements in the video pipeline to reduce power consumption and video memory usage.

New DVXA video processing APIs, which use a more flexible compositing model and are better suited for HD video formats.

Improvements to the way in which plug-ins (sources and decoders) are enumerated and managed.

Devices

Devices are a fundamental part of the PC experience, and Windows 7 enables new possibilities for developers of applications that interact with devices. The Device Experience Platform enables the association of applications and services with a particular device, so that users can get the maximum benefit from their peripherals immediately upon connection. The Sensor Platform provides a set of APIs for discovery of and communication with sensor devices that will enable a new generation of applications that are aware of their environments. The Location Platform provides new APIs for using location data from a global positioning system (GPS) receiver or other services that enable location-specific application behavior for mobile users. For more information see Device Fundamentals – Overview (http://go.microsoft.com/?linkid=9669725).

Device Experience Platform

Windows 7 combines software and services to create exciting new experiences for mobile phones, portable media players, cameras, and printers. Windows 7 makes it easier to use these devices directly from the Windows desktop. It also provides device makers with prominent placement on the Windows desktop, with branding opportunities and a simple interface for presenting the functionality and services that the device supports.

Through the Device Experience Platform, every Windows session becomes a portal for customers to get more value from their devices. The Device Experience Platform enables users to connect with the device manufacturer, discover and use related services, and learn about accessories. Because the device experience is connected to Microsoft's Web services, device companies can update the experience even after devices have been shipped to consumers. The Device Experience Platform can generate an application-like experience for Windows Logo'd devices, such as a mobile phones.

The Device Experience Platform gives applications access to devices such as mobile phones and media players that implement services through the Media Transfer Protocol (MTP) or the Windows Portable Devices driver model.

To enable the synchronization of personal information between a PC and a device, the Device Experience Platform hosts a new synchronization platform for connected devices, and provides a user interface for selecting target applications for data synchronization such as Contacts, Calendar, and Tasks. For more information see Windows Device Experience (http://go.microsoft.com/?linkid=9669724).

Windows Biometric Framework

The Windows Biometric Framework (WBF) provides an API which enables applications to use fingerprint devices to enroll, identify, and verify user identities without gaining direct access to any biometric fingerprint hardware or samples. You can use WBF with fingerprint devices that have Windows Biometric Device Interface (WBDI) drivers. WBF is extensible through plug-in adapters that manage sensor communications, biometric matching, and template storage. This ensures that WBF can be used with a wide range of fingerprint sensors. In Windows 7, fingerprint readers can use WBF for authentication during UAC and Windows logon. For more information see the Windows Biometric Framework API in the MSDN Library (http://go.microsoft.com/?linkid=9669733).

THE BEST OF WINDOWS AND THE WEB

With Windows 7, it's easy to get connected and stay connected. Windows networking offers developers options for better caching and sharing of data to improve network performance. Network diagnostics in Windows 7 give developers relevant information for monitoring network issues. Whether the goal is to connect devices, connect to a network, or simply understand the status of your connection, Windows 7 provides an enhanced platform.

Services

Windows 7 provides a powerful, highly extensible, and manageable platform for building and integrating the Web services and applications of the future.

Windows 7 offers both managed-code APIs and native APIs for building and running web services. A variety of new features are built on top of a new extensibility layer that allows developers to extend all APIs, in native code or within the .NET Framework.

Windows 7 also lets developers take advantage of better caching and searching capabilities. With these enhancements, developers can retrieve data faster and reduce network bandwidth usage.

Windows Web Services

With Windows Web Services, you can create applications that communicate easily with a local computer or a remote web service. Windows Web Services is a native-code implementation of SOAP and provides core network communication by supporting a broad set of the web services (WS) family of protocols. Windows Web Services is a peer to Windows Communication Foundation (WCF – managed-code web services), and provides a high-performance subset of WCF functionality. Windows Web Services provides the following benefits:

The ability to build native code web services in C/C++ in Windows   client and server.

Extensive integration with Windows Communication Foundation services.

The ability to build web services with minimal startup time.

The ability to build services based on the core WS family of protocols and W3C standards.

The ability to use web services in resource-constrained environments.

For more information see the Windows Web Services API (http://go.microsoft.com/?linkid=9669736) and Windows Web Services API: Developer Resources (http://go.microsoft.com/?linkid=9669737).

Distributed Routing Table

Windows 7 makes it easier to build sophisticated peer-to-peer applications like distributed file systems and content distribution networks with the Distributed Routing Table. The Distributed Routing Table provides a secure, scalable mechanism for publishing and searching for keys in a peer-to-peer system. It

can be used to build distributed hash tables and construct topologies for overlay networks. For more information see Distributed Routing Table API (http://go.microsoft.com/?linkid=9669727).

Windows BranchCache™

Windows 7 improves application responsiveness between central servers and branch-office computers. In today's networks, communication between central servers and branch offices is often congested, which leads to slower performance for applications in the branch office. With Windows BranchCache, clients can retrieve data from other clients in their own branch that have already downloaded the data, instead of having to retrieve the data over remote servers. As a result, WAN link traffic decreases and application responsiveness improves. The cache keeps a copy of all content that clients in the branch have requested and ensures that only the clients that are authorized by the content server can access the requested data, while preserving end-to-end encryption of the data.

Windows BranchCache is already integrated with HTTP and SMB. If an application uses the Windows APIs for either of these protocols, Windows BranchCache can help increase the performance of this application on Windows 7 without making any changes to it.

If your application retrieves the same data multiple times from a server over a WAN link and is not automatically optimized using Windows 7, it is easy for you to use the Windows BranchCache APIs to optimize your application to work faster on Windows 7 and satisfy your branch users.

These new features help reduce wide area network (WAN) traffic and latency while ensuring compliance with security mandates. For more information see Peer Distribution (http://go.microsoft.com/?linkid=9669734).

Federated Search

Windows 7 supports searching for documents beyond the user's own PC. Developers and IT professionals can enable their search engines, document repositories, web applications, and proprietary data stores to be searched from Windows 7 without needing to write and deploy client code. This enables end users
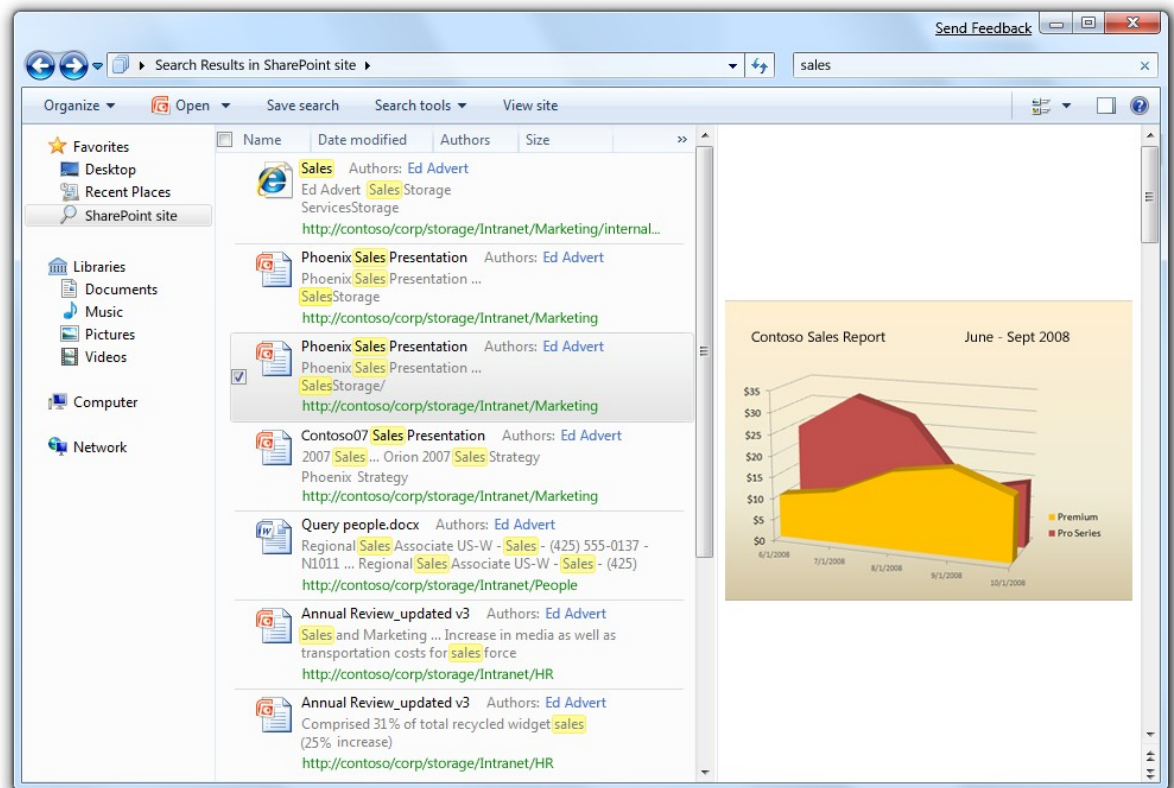
to search their corporate intranet or the web as easily as they can search for their local files—all from within the same familiar Windows interface.

Windows 7 compatible OpenSearch support can be added by developers and IT professionals to any existing searchable web application by adding RSS or ATOM output as defined in the public OpenSearch standard (http://www.opensearch.org). OpenSearch description files can then be authored by developers and advanced users to define the connection parameters that enable the desktop client to create a Search Connector for the service.

SharePoint® Search Server can also query these compatible OpenSearch services. This enables access to your search service from the SharePoint search center web interface as well. For more information see Federated Search in Windows (http://go.microsoft.com/?linkid=9669728).

Sensor Platform

Windows 7 has changed how developers use sensors. It includes native support for sensors, expanded by a new development platform for working with sensors, including location sensors, such as GPS devices.

Built on the Sensor platform, the Windows Location APIs are a new Windows 7 feature that enables application developers to access the user's physical location information. The Windows Location APIs can abstract hardware, simultaneously support multiple applications, and seamlessly switch between different technologies, relieving the application developer of the burden of managing these constraints. The Location APIs can be used by programmers through the C++ programming language (by programmers familiar with COM), or by using COM objects in scripting languages, such as JScript®. Scripting support gives easy access to location data for projects such as gadgets or web pages.

Windows 7 provides a solid, easy-to-use platform for using sensor devices, such as an ambient light sensor or a temperature gauge, to create environmental awareness in Windows applications. PCs can use sensors that are built into the computer, connected through wired or wireless connections, or connected through a network or the Internet.

The Sensor and Location APIs provide a standard way to discover sensors, and to programmatically access data that sensors provide

The Sensor control panel lets users enable or disable sensors, control access to sensors that might expose sensitive data, view sensor properties, and change the descriptions of sensors.

The Sensor Class Extension is a core part of the driver development model for the Sensor platform. It provides the following mechanisms, which are used when writing a User-Mode Driver Framework (UMDF) sensor driver:

Integration with the Sensor platform

Security enforcement

For more information see the Windows Sensor API documentation (http://go.microsoft.com/?linkid=9669730) and Windows Sensor and Location Platforms: Developer Resources (http://go.microsoft.com/?linkid=9669731).

Internet Explorer 8

The evolution of the Internet has introduced new sources of rich information and more ways to access it. This growth has created new opportunities, experiences, online services, and standards to the Web. With this intensity and reliance, Web developers face an evolving set of needs, which Internet Explorer 8 addresses in three key ways.

First, Internet Explorer 8 provides real-world interoperability with other browsers and compatibility for existing sites. It includes enhancements to the core platform and architecture, offering improved performance, safety, reliability, and compatibility. With previous versions of Internet Explorer, developers and designers have sometimes noted that Internet Explorer has had its own interpretation of Web standards and the way the browser handles HTML, Cascading Style Sheets (CSS), scripting, and so on. In some cases, interpretations were decided upon because Internet Explorer supported certain features before corresponding standards were finalized. If those standards change as they are finalized, Internet Explorer's implementation can vary from what the standard specifies. With Internet Explorer 8, the decision to support legacy behaviors versus strict standards will be put in the hands of developers by enabling you to select the rendering mode on a page-by-page basis.

Second, Internet Explorer 8 makes Web development faster and easier with built-in developer tools. With advancements in support for Ajax (Asynchronous JavaScript and XML) applications, as well as improvements that simplify the process of building cross-browser applications, Internet Explorer 8 enables developers to be more productive when building the most robust Web applications possible.

Finally, Internet Explorer 8 enables experiences that reach beyond the page through new browser features that effortlessly connect users to innovative Web services. Internet Explorer 8 features a set of integrated developer tools that are a component of the browser. As a result, performance is improved and no memory is used when the tools are not running. The integrated Developer Tools make it faster and easier for developers to develop and troubleshoot rich content sites.

Internet Explorer 8's Standards mode layout engine was also built with the CSS 2.1 spec in hand and full compliance in mind, helping developers to easily support the latest standards while having the option to fall back to Internet Explorer 7- compatible (or earlier) behavior if necessary. To assist developers in taking full advantage of the elements offered by HTML 4, Internet Explorer 8 provides upgraded support for several presentational elements. For example, the `Q` element represents an inline quoted string, and the `object` element may now represent any "object," including images. Through improved support for these and other HTML elements, Web developers can deliver more expressive and accessible HTML markup.

In addition to the improvements made to the underlying platform and developer experience, Internet Explorer 8 introduces new features that deliver new opportunities for Web developers to extend their experience beyond the Web page. These new features are enabled through community standards rather than building custom client code that is not interoperable across browsers. For more information see the following topics:

What's New in Internet Explorer 8 (http://go.microsoft.com/?linkid=9669729)

Silverlight (http://go.microsoft.com/?linkid=9669732)

Windows Presentation Foundation (http://go.microsoft.com/?linkid=9669735)

CSS Improvements in Internet Explorer 8 (http://go.microsoft.com/?linkid=9669723)

Discovering Internet Explorer Developer Tools (http://go.microsoft.com/?linkid=9669726)