[redacted]

Windows Store Apps Succinctly

by John Garland

Syncfusion | Technology Resource Portal

Chapter 1 Core Concepts

Introducing Windows Store Apps

Figure 1: The Windows 8 Start Screen

Windows Store apps are a new kind of application that run on Microsoft's most recent generation of operating systems. Currently, this includes Windows 8, Windows RT, and Windows Server 2012. When installed, an app can have one or more tiles pinned in user-selected positions on the Windows Start screen. Users can launch the app by simply tapping or clicking one of its tiles. Additionally, some applications can be launched by Windows itself as a result of user interaction with common Windows interface elements, including the charms bar, which provides a focal point for accessing common functions such as in-app searching (Search charm), app-to-app data exchange (Share charm), hardware interaction (Device charm), and configuring settings and preferences (Settings charm). Apps can even be launched as a result of scenarios where they have elected to participate in Windows-brokered interactions that are actually initiated from within other applications.

Windows Store apps and the Windows environment they run in feature a new user experience. Apps occupy a single window, and run either full-screen or in a secondary, fixed-size reduced screen known as the "snapped" view. This user experience follows a set of published Microsoft design principles. A summary of these principles includes:

- Content before chrome: Elimination of any frivolous elements that take away from the display of the information or controls being presented to users.

- Fast and fluid: Response to user interactions should be quick and intuitive, and the UI should not “lock up” to support data processing or other background activities.
- Support for multiple view states: The app should handle being displayed in different screen modes, whether it is running as the primary landscape app, in the afore-mentioned snapped landscape view, or full-screen in portrait orientation.
- Support for the right contracts: The app can interact with other Windows components or other apps via the provided contracts and extensions, and should do so to foster and reinforce a powerful and familiar user experience across all apps.
- Live tiles: Even when the app isn’t running, both its primary launching tile and any secondary tiles used to launch the app can come alive and be used to provide app-related information to users.
- Settings and user context roam via the cloud: Apps now have the option to tap into support for moving settings beyond just the local machine, potentially providing users with continuity within the app regardless of what machine they run it from.

Windows provides a controlled environment for Windows Store apps to run in—sometimes known as a “sandbox”—which allows Windows to protect system resources and state from defective or malicious programs. Apps submitted to the Windows Store are qualified against a published set of requirements as part of a certification process that helps to ensure that customers’ systems are not adversely affected by defective or malicious apps. Windows Store apps are digitally signed to provide verification of their authenticity and integrity. Apps published to the store can be offered free of charge; include time-based trials, feature-based trials, or both; or be sold for a fee. In-app purchases and subscriptions are also available for Windows Store apps.

As the name implies, most Windows Store apps are made available for purchase from the centralized Windows Store. This provides development efforts of all sizes, from single hobbyist developers to large corporate concerns, the opportunity to reach a global marketplace of customers with their apps to realize revenue or recognition—or both! However, Windows Store app distribution isn’t limited to the Windows Store—several line-of-business deployment scenarios exist, including deployment via enterprise management systems. This process of deploying an app through a means other than the Windows Store is known as “sideloading.”

A thorough overview of Windows Store apps has been published by Microsoft and can be found at http://msdn.microsoft.com/en-us/windows/apps/hh852650.aspx.

The Windows Runtime

Windows Store apps run on top of a new runtime API called the Windows Runtime, or WinRT. This is a fundamental shift in Windows development, where for quite some time development has occurred on top of one version or another of the Win32 APIs (albeit the presence of Win32 has sometimes been abstracted away by development tools and runtimes, such as MFC, Visual Basic 6, or even .NET). In contrast to the C-style approach afforded by Win32, WinRT provides a more modern object-based surface for application development.