

Windows 8/RT and Wine

Admittedly not usually ARM-specific

Ridiculous Terminology

- MS introduced a mess of new terms for Windows 8, and I have to go through them so we don't get confused.
- Windows RT: An edition of Windows 8 that runs on ARM devices. Microsoft has not defined what RT stands for.
 - Formerly: Windows on ARM
 - Not to be confused with: WinRT
 - In addition to running on ARM, this version refuses to run non-MS executables on the desktop. But an easy jailbreak is available.
- App container: A special security token that represents an application running as a specific user. App containers have limited permissions, will be suspended/terminated at the system's whim, and are exempt from signing requirements.

Ridiculous Terminology (part 2)

- Windows Store Environment: The part of Windows 8 that isn't the desktop.
 - This might not be the official name.
 - Alternative names: Metro, Windows Store, Touch-Optimized Interface
- Immersive Process: An instance of a program that runs in the Windows 8 store environment. AFAICT these must run inside an app container.
- Charms Menu: That silly thing on the right side of the screen that no one figures out they need the first time they want to turn off their Windows 8 computer.

I guess this is also sort of an overview?

- App Contract/Extension: Capabilities of an application that it registers with Windows and that Windows will activate under specific circumstances.
 - Example: The search contract allows Windows to list the application on the Search charm and activate it when the user selects it to search.
 - The launch tiles on the start screen are another example.
 - Research opportunity: Do these have to be inside an app container?
- Package Manager: The Windows component that unpacks an application package (appx file) and installs it, which includes creating an app container and registering contracts/extensions.

More Win8 overview stuff

- Windows Store: The primary official path for obtaining and installing application packages. It's a storefront similar to iTunes app store, fully controlled by Microsoft.
- Sideload: The alternative path for installing application packages. This requires a developer license, which is free but expires after 3 months. Permanent sideloading is available to corporations through some undisclosed process.
- WinRT: The Windows Runtime, which is some new thing that is sort of COM and sort of .NET. No, I don't get it either.
 - Not to be confused with: Windows RT

Can Wine run in Windows RT?

- No, for the same reason it can't run on x86 windows.
- Windows RT is missing the following components: D3DX, DirectInput, DirectMusic, DirectPlay, OpenGL, DirectShow.
- All of those except OpenGL could be replaced by a Wine version. ANGLE might replace OpenGL if we have D3DX.
- It's not clear that Windows desktop applications using OpenGL or DirectX would be useful on the sort of low-power device that would run Windows RT.

But what about using Wine to run x86 apps?

- Normally I'd say no, but someone released an interesting proof-of-concept for this without using Wine at all.
- It includes an x86 emulator, PE loader, and thunk dll's for a small subset of the Windows API.
- I don't think the thunk dll approach will scale, but a mix of thunks and Wine dll's may be feasible.
- I can't imagine why we would put in the work to actually do this.

Can Wine run application packages?

- Yes, but it requires a lot of architecture we don't have yet.
- WinRT can be used from desktop applications, so in theory we will need it.
- App containers, a package manager, and the XAML toolkit for immersive processes should not be needed by desktop apps.
- Because the primary method of distributing appx packages is the windows store, which we can't access, we would be limited to packages that have been distributed for sideloading and packages obtained from a real Windows RT machine.
- It's unlikely we'll ever see the windows store gain the level of dominance Win32 once had.