No need to update individual apps? Microsoft wants Windows Update to handle it all

It can be said that this is a bold ambition, especially in the context of Windows Update, which has inherent problems.

Microsoft has announced plans for the Windows Update orchestration platform, a big step toward making Windows Update the hub for almost all software updates, not just operating system updates. The idea is to end the chaos of each app updating itself with its own schedule and annoying notifications.

It's a bold ambition, especially given Windows Update's long history of problems, like the infamous 0x80070643 error that many people encountered with the WinRE update last March. Or an update that caused USB printers to print gibberish. But what's even more notable is that Microsoft says the new system is 'built on the Windows Update stack.' It will allow app developers to plug their updates into this centralized manager using new APIs and PowerShell commands.

The way the new Windows Update orchestration platform works isn't too complicated, but is based on a seamless process as follows:

1. Application registration : The developer or update manager needs to register the software with the coordinator.

2. Declare the scanning tool : They specify an executable file for the system to scan for new updates for the application.

3. Update Notification : Use the API to notify the orchestrator of an available update, including: name, version, package type, reboot requirement, and application deadline (for managed devices).

4. Handling Win32 applications : For non-MSIX/APPX applications (e.g. regular Win32), developers can:

1. Provide executable file for download and installation.
2. Provide a script to close/reopen apps if they block the update process.

5. Scheduling & Notifications: The dispatcher will intelligently schedule downloads/installations (taking into account the user's machine usage status) and display familiar Windows Update notifications when interaction is needed.

6. Reporting results: The application must report whether the update was successful or failed so that the system can retry if necessary.

In theory, the potential upsides are notable: Smarter scheduling that takes into account whether or not a user is actually using the PC, and providing a single, centralized place to manage update history in Settings. For developers, this means they can avoid having to build and maintain their own update system. But it's still in preview, and there's a lot of work to be done.