Survey of Windows update formats: The Full update

devblogs.microsoft.com/oldnewthing/20200210-00

February 10, 2020

Raymond Chen

In August 2018, Microsoft announced a new design for how updates are delivered. How did they manage to get the size of the updates so much smaller while simultaneously reducing memory usage and update negotiation time?

To understand what changed, you first need to understand how things were.

Let's focus only on changes that apply to a specific major release of Windows. (You can repeat the exercise for each major release.) Let's call the initial release *M0*. And suppose there have been five monthly cumulative updates, call then *M1*, *M2*, *M3*, *M4*, and *M5* (the latest version).

There are two general mechanisms that Windows Update uses to update a file: One way is to send an entire replacement file. Another is to send a patch (sometimes called a *delta*) that updates the file currently on the system to the latest version.

There are obvious trade-offs here. The full file works regardless of what version the customer is upgrading from, whereas the patch works only if the customer has the specific version that the patch was designed to upgrade. On the other hand, the full file is large, whereas the patch is much smaller.

	Full file	**Patch**
Size	Large	Small
Applicability	Can update any version	Can update only one version

Suppose that the file `F` was updated in M1 and M2, was unchanged in M3, and was updated again in M4 and M5. Here are all the possible full files and patch combinations.

Update	**Full file**	**Patch base**				
		M0	**M1**	**M2**	**M3**	**M4**

M1	M1	M0 to M1				
M2	M2	M0 to M2	M1 to M2			
M3						
M4	M4	M0 to M4	M1 to M4	M2 to M4		
M5	M5	M0 to M5	M1 to M5	M2 to M5		M4 to M5

Note that the rows and columns for M3 are empty because there was no change to `F` in M3. Updating to M3 is the same as updating to M2.

The simplest kind of update is the **Full update**. It contains a copy of every component that has changed since the last major release. In our example, this would be the components that changed between M0 and M5. It guarantees that you can get from any version to the latest version, but it's also very large (around 1GB).

For the file `F`, the corresponding Full updates would contain the files from the **Full file** column.

		Patch base				
Update	**Full file**	**M0**	**M1**	**M2**	**M3**	**M4**
M1	M1	M0 to M1				
M2	M2	M0 to M2	M1 to M2			
M3						
M4	M4	M0 to M4	M1 to M4	M2 to M4		
M5	M5	M0 to M5	M1 to M5	M2 to M5		M4 to M5

The breakdown of files for Full updates is as follows:

Full update	**Contents**
M1	M1
M2	M2
M3	M2
M4	M4

M5	M5

Note that Full update M3 contains a copy of `F` from M2. That way, if somebody currently running M1 needs to update to M3, the Full update will get them the M2 version of the file, which is the correct version for a system that has been updated to M3.

Feature summary of Full updates:

- Full updates can successfully update all customers, even those who installed a hotfix outside of the monthly servicing cycle.[1] Since full copies of modified files are delivered, it doesn't matter what your starting point is. You always end at the same place.
- Full updates are very large (1GB).
- Full updates require very little negotiation with the server. Every customer downloads the same update.
- Full updates are cache-friendly, because every customer downloads the same update. Therefore, caching features like caching proxies, BranchCache, and peer-to-peer delivery are effective.
- Full updates do not require significant server support. Once the package is negotiated, it is delivered in its entirety.

Next time, we'll look at the **Delta update**, which despite its name doesn't actually contain deltas.

Bonus chatter: If a week of articles about Windows update formats is too tedious, you can read this short version: Windows 10 quality updates explained and the end of delta updates.

[1] By policy, every out-of-cycle hotfix is rolled into the next cumulative update, so installing a cumulative update will never cause a hotfix to be lost.

Raymond Chen

Follow