

# The numerology of the build, redux

 [devblogs.microsoft.com/oldnewthing/20160330-00](http://devblogs.microsoft.com/oldnewthing/20160330-00)

March 30, 2016



Raymond Chen

I noted some time ago that starting in Windows Vista, final build numbers must be a multiple of 16. Why is that?

The version number comes in four parts, each consisting of a 16-bit integer: major.minor.build.revision. The major and minor numbers correspond to the programmatic operating system version. For example, poseurs refer to Windows 7 as version 6.1. because its programmatic major version is 6 and its programmatic minor version is 1. Nobody at Microsoft calls it that; we just call it Windows 7.

The build number generally increases by one each time the build lab spins up a build, although there are occasional jumps and gaps, as I noted in the linked article.

The revision number is under the control of the servicing team. And here's what they needed to encode:

- Four bits for the service pack number, where zero represents RTM.
- One bit to indicate whether this is a non-milestone build. (And no, I don't know what that means.)
- One bit to indicate whether this is a build intended to be released to the public.
- Two bits to indicate which release track the build is from. 0 = GDR, 1 = LDR, 2 = (unused), 3 = private build.
- Twelve bits to encode the servicing build number.<sup>1</sup>

Add this all up and you'll see that this requires 20 bits. But the revision field is only 16 bits wide. Therefore, the servicing team needs to find 4 additional bits somewhere, and by requiring that the RTM build number be a multiple of 16, they get their extra four bits.

One of the changes to the servicing stack in Windows 10 is that the servicing team redesigned their version numbering system so that they need only 16 bits of information rather than 20. I don't know the details of how they accomplished it; all I know is that they no longer need to reserve the bottom four bits of the build number. In other words, the build number no longer needs to be a multiple of 16.

Most people in the Windows division were unaware of this change to the servicing stack. When the announcement came down that the release management team were hoping that build 10584 would be the final build for the November 2015 update, many people didn't believe it. "That can't be the final build. It's not a multiple of 16!"<sup>2</sup>

<sup>1</sup> The fact that this provides enough daily builds to cover the five-year mainstream support lifecycle is probably not a coincidence.

<sup>2</sup> Yes, the actual final build for the November 2015 update is 10586. I guess the release management team decided to take two more fixes.

Raymond Chen

**Follow**

