a000clown wrote on Thu, 29 July 2010 00:11Dover wrote on Tue, 27 July 2010 21:21
R315r4z0r wrote on Tue, 27 July 2010 17:41Dover wrote on Tue, 20 July 2010 08:43That's cool. I wonder what would happen if real, non-shitty developers did the same thing and sat on their laurels after they put out a partially-complete beta.
I posted before, but this really didn't hit me until just now.

There is no such thing as a 'complete beta.'

The point of a beta is to give out a work-in-progress in order to receive feedback on how they should adjust things for the final product.

A 'complete beta' wouldn't be a beta, it would just be the entire game.

I think you're confusing the term 'beta' with 'demo,' which the Infantry Beta was not.

I'm not. I'm drawing paralells with the StarCraft II beta which went on recently, which resembles what a beta should be. That wasn't a 'demo' either, but it was a more-than-half done product with all the serious bugs ironed out already and it resembled what the final version would be. None of that could be said for the reborn infantry beta.

There are minimum standards for how much work should be done for something to be considered a beta. "Beta" is the second letter of the greek alphabet. Second, that means something had to have come before it. And taking a look at Reborn infantry beta, I'm not sure there was.
Everyone uses different terms for preview releases, there's no standard for naming something an alpha/beta/rc etc. However, I'd say it's reasonable to assume their internal builds can be considered private alpha milestones.
That said, I do agree that the "beta" should resemble what the developers intended as being the end result, opposed to leaving out a lot of content (vehicles) which is a major defining aspect of the game.

Edit: Oh and unless the code is very minimal, software is very rarely "complete". There's always bugs to be squashed and features to be added, which is why we have product release cycles with maintenance and security updates on a (hopefully) regular basis.
Version 1.0 is usually just a milestone when the developers reach their intended features and decide the only remaining issues are minor and can thus be used by end-users; It does not signify a "finished" product that is totally void of bugs. Again though, everyone uses their own version numbering, some being more common than others.

For the purposes of this discussion, I was ignoring service updates. Obviously most things are patched post-1.0, but they have to actually reach 1.0 status first.