

# Re: Rules to setup project version numbers?

---

*Source:*

<http://coding.derkeiler.com/Archive/Delphi/borland.public.delphi.non-technical/2007-03/msg05484.html>

---

- *From:* "Doug Chamberlin" <dchamberlin at andoversoftware dot org>
  - *Date:* 31 Mar 2007 12:59:46 -0800
- 

Tiger wrote:

- 1) When do you increase the build number?
- 2) When do you increase the release number?
- 3) When do you increase the minor number?
- 4) When do you increase the major number?

Here's what I have gathered from watching LOTS of projects from many vendors and talking to many developers:

1) Build number is automatic. Increments when each build is created. This is important for tracking the EXACT executable that someone is running and being able to debug it by matching it to a specific collection of source files. This is critical.

Build numbers are often omitted from the more visible places the program's version is displayed. However, SOMEWHERE the build number must be accessible by users so they can accurately report it to support. For example, in the About dialog it might say "Version 3.2" but if you click the "Support Info" button there it produces another window that shows Windows version and build, important DLL versions, and this program's build number.

2) Release number is the next release you are working toward. Lots of builds will have the same release number until the release is published.

3) Minor version is usually a marketing thing (meaning developers do not get to decide when this number is incremented). Increments when a new build is to be published that includes minor changes that are often included in support plans. (Customers on support do not get charged for changes in minor version numbers.)

4) Major version number. Also a marketing thing. Increments when major new features are added that are important enough to warrant a big public splash. Also marks a point at which existing customers usually have to pay for an upgrade.

## Re: Rules to setup project version numbers?

So, an example release history might be:

0.9.0.1789 Last test release before going public.

1.0.0.1790 First version 1 release for sale

1.0.1.1791 – 10.0.1.1845 Lots of testing for bug fixes.

1.1.0.1792 First public update. Now currently shipping version 1.

1.1.1.1793 – 1.1.1.1831 Lots of test builds. Marketing decides to announce 1.2 with all accumulated fixes.

1.2.0.1832 Version 1.2 published.

1.2.1.1833 Work begins on version 2.

1.2.1.1834 – 1.2.1.1855 Lots of builds including major new features.

2.0.0.1835 Version 2 published.

2.0.1.1836 First fixes for version 2. Includes serious bug fix so gets published as 2.0.1

2.0.2.1837 – 2.0.2.1849 Lots of builds with additional fixes.

.