Notation for versioning of collaborative codes

Brendon Bradley
Jan 2018

<u>Context:</u>
- For highly-collaborative and integrated research we need a commonly-understood language when talking about software products that are being used

<u>Approach:</u>

a. To make versioning easy to understand the prefix of the versioning should be the year.month (e.g. 18.1 for Jan 2018).
b. Minor iterations (e.g. bug fixes etc) should be denoted with a suffix increase (e.g. a bug fix means to go from 18.1.1 to 18.1.2).

<u>Implications for backward compatibility</u>
a. Somewhat by definition, backward compatibility should be maintained in all minor iterations (18.1.1 to 18.1.2)
b. Backward compatibility may not be maintained for major versioning changes (e.g. 18.1.2 to 18.3.1).

<u>Implications for planning:</u>
a. Minor iterations (18.1.1 to 18.1.2) are expected to happen on the fly
b. Major revisions (e.g. to release a new version 18.3.X) should be explicitly planned in advance using Gantt charting and elicit input from multiple people in order to make strategic decisions around desired functionality and backward-compatibility.