

Difference between 'Edit current' and 'Edit latest...' tabs

- [The Functionality](#)
- [The Potential Problem](#)
- [The Fix](#)
- [History](#)



Always use "Edit latest..."

When editing content users should use the "Edit latest unpublished revision" tab whenever it is present.

The Functionality

"Edit current" **always** creates a new pending revision, regardless of whether one already exists or not.

"Edit latest unpublished revision" opens the most recent pending revision, if one exists. If not, it creates one.

The Potential Problem

Because "Edit current" is always present (see History, below), even when there is a pending revision, it is possible to create multiple pending revisions. This can lead to confusion about which pending version has the most recent updates and may result in inaccurate or incomplete content being published.

The Fix

If you have multiple pending revisions, you will need to make sure the top-most revision under the Revisions tab has the latest edits, then save that revision with a log message indicating that it is the correct version. For non-catalog content, you may then publish that revision, which will force the incorrect revision to be archived. For catalog content, this will happen once the next academic catalog is published.

History

In Release 1.13, the label of the "Edit latest" tab was changed to "Edit latest unpublished revision".

Why?

The short version:

The label was not completely accurate. In almost all cases, if the "Edit latest unpublished revision" tab is visible, use it rather than "Edit current".

However, there are some important (and confusing) situations where "Edit latest" is not accurate (see 'long version' below), so the text was changed to reflect what actually happens when that tab is clicked.

The long version:

Previous to the CMS upgrade in October 2013, a given published node had EITHER an "Edit current" tab if there was no pending revision after the currently published one OR an "Edit latest" if there was a pending revision. Since only one tab was active at a time, this kept most editing simple.

Following the CMS upgrade, however, the module governing those tabs changed, forcing the presence of the "Edit current" tab at all times, regardless of whether a pending revision was present. For nodes with no pending revision, there was still only the "Edit current" tab, but for nodes with a pending revision, both the "Edit current" and "Edit latest" tabs were visible.

This caused confusion for several users, who clicked the "Edit current" button, because it seemed that any previous edits they had made and saved were 'missing'. They weren't actually missing, but the change in the process from the CMS upgrade made it seem so.

Clicking "Edit current" begins a new pending revision, regardless of whether or not there are any other pending revision(s) already present. For nodes with no pending revisions, this acts as expected, creating a new pending revision. For nodes with pending revisions already present, however, this results in a new pending revision based on the currently published version, with no awareness of any changes made in other pending revisions (imagine this newest revision 'leapfrogging' over any earlier ones). Once saved, the Revision List screen will show the yellow published revision, as normal, with two (or more) pink pending entries above it.

Clicking either of the pending revisions will take you to that revision, but neither revision will have the changes made in the other.

The 'simple' fix is to revert to the pre-upgrade display (either "Edit current" or "Edit latest"), but there are 2 problems with this.

1. Most importantly, the 'leapfrogging' functionality is:
 - a. In some cases desired (when users want to compare the differences between a set of edits), and

b. Still possible even without the "Edit current" button, by navigating to the Revision List for a node and selecting the current published revision

2. Secondly, what looks like a simple fix is actually a very complicated recoding effort, and could be impacted by future module upgrades.

We are currently trying to figure out a better solution to this issue, including alternate modules, custom development or revised workflows.