

org-change: Track changes in org-mode

Stefano Ghirlanda

April 12, 2023

1 Introduction

org-change is a minor mode that (ab)uses the org-mode link syntax for a simple “track changes” feature. It defines a **change** link type to mark additions, deletions, and replacements. For example:

```
[[change:old text][new text]]
```

Because org-mode (typically) hides the part of the link within the first pair of brackets, you only see **new text**. To indicate an addition, just omit **old text**:

```
[[change:][new text]]
```

To indicate a deletion, use ****DELETED**** as **new text**:

```
[[change:old text]**DELETED**]]
```

Two features provided by org-change make this syntax useful:

1. Key bindings to manipulate **change** links.
2. Export filters to render **change** links appropriately.

Right now, only exporting to Latex is supported.

2 Installation

Manually from here for now, from MELPA later.

3 Key bindings

C-‘ a for additions.¹

C-‘ d for deletions.

C-‘ r for replacements.

You can mark a region and then use these keys to mark it as an addition, deletion, or replacement. In the case of replacement, you will be prompted for the new text. You can also use C-‘ a without marking a region. In this case, you will be prompted for the new text.

Key bindings are also provided to accept and reject changes:

C-‘ k to accept the change at point.

C-‘ x to reject the change at point.

To accept means to delete the change link (including any comments) and insert the new text (nothing if the change is a deletion). To reject means to delete the change link and insert the old text (nothing if the change is an addition).

4 L^AT_EX export

When exporting to L^AT_EX, org-change uses the **changes** package, which must be included:

```
#+latex_header: \usepackage{changes}
```

org-change will then use the commands `\added`, `\deleted`, and `\replaced` provided by this package.

org-change supports some additional features of the **changes** package. You can add a comment to any edit like this:

```
[[change:old text][new text**A comment**]]
```

which is exported to

```
\replaced[comment=A comment]{new text}{old text}
```

¹The quote in C-‘ is a left-quote character. It is not pretty, but practically every other ctrl prefix that does not also need shift is taken. It's the curse of keymensionality.

The comment must be at the end of `new text`. You can add other fields supported by `changes` by tagging them to the end of the comment. For example, you can indicate the author of the comment:

```
[[change:old text][new text**My comment,author=SG**]]
```

which is exported to:

```
\replaced[comment=My comment,author=SG]{new text}{old text}
```

5 Customization

The key bindings and the face used to display change links can be changed through the customize interface:

```
M-x customize-group RET org-change
```

6 Bugs and limitations

- Text inside a `change` link is exported as-is. For example, org-ref links are not interpreted.
- Link hiding is sometimes inaccurate in org-mode. You may see stray brackets, especially with link that span multiple lines. Often M-q takes care of this.

7 Planned features

- More export filters
- Enable users to add export filters
- A convenient way to see the hidden part of a link
- Function `org-change-accept-all` to process all change links in the buffer

8 Notes

I described the features of org-change to ChatGPT and asked to get the corresponding code. It was mostly wrong, but a good starting point.