Hog Bay Software Home About Posts Support

Posts › Bike: Innovative Rich Text Editing

Bike: Innovative Rich Text Editing

11 Oct 2022

Bike now supports rich text!

Download

Bike’s rich text editing should be familiar. There’s also innovation[1]. I’m trying to improve the fundamentals of how rich text editing works.

Watch Bike’s Rich Text Movie

Inspired by Markdown

Bike’s rich text editing is inspired by Markdown.

I like Markdown because the formatting is directly accessible. The formatting characters are always right there to guide my edits.

I dislike Markdown for the same reason. Those formatting characters are *always* visible. They make my text look messy.

I want clean looking outlines, so Bike uses rich text. I would also like my outlines easy to edit. That’s where I think rich text has problems and Bike’s *typing affinity* is a good solution.

Editing Puzzles

Rich text is difficult when you need to insert text at formatting boundaries. Markdown has formatting characters to guide you. In rich text there are no such characters, and so you must rely on these conventions:

- Use formatting from text before the caret
- At start of line use formatting from text after caret
- Some formatting (links for one) might use different rules

If you need to perform an edit that doesn’t fit into these conventions then rich text editing becomes painful.

Puzzle 1:

Start with “Hello” in bold. Then add " world" in normal text. In Markdown this is easy, just place " world" outside the **s:

```
From: **Hello**
  To: **Hello** world.
```

Now try the same thing in a rich text editor.

Not as easy. You need to exit from bold typing mode. Probably you will do `Command-B` to do that. How you exit depends on the previous formatting–in this case it’s `Command-B`, but if the formatting was different then you would need a different command. There’s no consistent way to end formatting, and go back to typing unformatted text.

Puzzle 2:

Insert "cool " at the start of a link. In Markdown it is straight-forward. You have the syntax characters to guide you, insert inside the braces:

```
Start: [links][...]
  End: [cool links][...]
```

Now try the same thing in a rich text editor.

Painful. You are editing against the conventions, "cool " doesn’t want to be part of the link. Best approach I’ve found is to select the first letter in the link “l” and then type to replace it to *catch* the formatting. We shouldn’t have to do these editing gymnastics.

Bike’s Solution: Typing Affinity

Bike’s *typing affinity* solves both editing puzzles easily. At each formatting boundary (including start and end of line) Bike’s text caret shows a tail. Use the arrow keys to point the tail at the formatting you want. Thats it.

Bike’s approach is just as precise as Markdown and requires fewer keystrokes. It does this without littering formatting characters throughout your document.

Give typing affinity a try. Perform the above editing puzzles in Bike and see how clean and predictable rich text editing can be.

1. Some of this innovation is rediscovered ↩