

CriticMarkup

(/)

Plain Text Editing Markup for Humans



Get the Toolkit (<https://github.com/CriticMarkup/CriticMarkup-toolkit/archive/master.zip>)



Visit us on Github

(<https://github.com/CriticMarkup/CriticMarkup-toolkit/>)

Critic Markup is intended to provide basic editorial change tracking in plain text files. The syntax is compatible with Markdown, MultiMarkdown and HTML.

The Three Laws

1. Critic Markup shall be human readable. A human with a simple text editor can easily read and comprehend any text containing Critic Markup.
2. Critic Markup shall be computer readable except where it conflicts with rule 1. Markup syntax should be easily parsed with simple regular expressions to support a wide variety of implementations.
3. Critic Markup shall be compatible with existing markup syntax for Markdown, MultiMarkdown and HTML except where it conflicts with rules one or two. Many users of plain text write in combinations of Markdown and HTML. Critic Markup should work alongside that syntax.

The Goal

Critic Markup can be used in any writing environment without special applications or tool kits. While we have supplied processors and plugins that are compatible with popular apps, they are not required.

Critic Markup should be readable inline and clearly indicate the intent of the editor and author.

Critic Markup should support change tracking with multiple authors and editors through the use of comments.

Plain Text and HTML

Critic Markup may be used without a conversion to HTML. However, as with Markdown, HTML may be a desirable format for a more stylized presentation. We have several recommendations for how Critic Markup may be converted to HTML. The intent is to allow the HTML to be used without custom CSS. However, custom CSS may be used to enhance the review process.

The Basic Syntax

There are five types of Critic marks:

- Addition `{++ ++}`
- Deletion `{-- --}`
- Substitution `{~~ ~> ~~}`
- Comment `{>> <<}`
- Highlight `{== ==}{>> <<}`

Using these five basic marks you can successfully copy edit in plain text.

Additions

Additions are inserted inline by surrounding the desired text with curly braces and double plus marks as shown:

```
Lorem ipsum dolor{++ sit++} amet...
```

A space character and “sit” are to be added at the position of the left (or right) most curly brace. The additions may be rendered as `<ins>` tags in the processed HTML:

```
Lorem ipsum dolor<ins> sit<ins> amet...
```

Paragraphs may be inserted in the same manner.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. Mauris massa sem,
tempor sed cursus et, semper tincidunt lacus.{++

++}Praesent sagittis, quam id egestas consequat, nisl orci vehicula libero, quis
ultricies nulla magna interdum sem. Maecenas eget orci vitae eros accumsan mollis. Cras
mi mi, rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo
eget, facilisis eget nunc. Cras tincidunt auctor varius.
```

To ensure the Markdown processor outputs valid HTML, the `<ins>` tag encloses a non-breaking space and is followed by two newlines. The result is that both paragraphs render separately. The non-breaking-space gives the tag enough content to render properly, and we assign `class=break` to

allow for custom styling.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. Mauris massa sem,
tempor sed cursus et, semper tincidunt lacus.

```

```
<ins class="break">&nbsp;</ins>
```

```

Praesent sagittis, quam id egestas consequat, nisl orci vehicula libero, quis ultricies
nulla magna interdum sem. Maecenas eget orci vitae eros accumsan mollis. Cras mi mi,
rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo eget,
facilisis eget nunc. Cras tincidunt auctor varius.

```

Rules for proper use of the `<ins>` element can be found in the HTML 4 spec (<http://www.w3.org/TR/REC-html40/struct/text.html#h-9.4>).

Deletions

Deletions are denoted with a similar syntax. The text to be deleted is surrounded with curly braces and double hyphens.

```

Lorem{-- ipsum--} dolor sit amet...

```

The word “ipsum” and a space character are marked for deletion in the above example. These deletions are rendered as `` tags in the processed HTML.

```

Lorem<del> ipsum</del> dolor sit amet...

```

Paragraphs may also be deleted.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. Mauris massa sem,
tempor sed cursus et, semper tincidunt lacus.{--

--}Praesent sagittis, quam id egestas consequat, nisl orci vehicula libero, quis
ultricies nulla magna interdum sem. Maecenas eget orci vitae eros accumsan mollis. Cras
mi mi, rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo
eget, facilisis eget nunc. Cras tincidunt auctor varius.

```

The newlines will be removed by the processor and replaced by an inline `` tag. Again, the non-breaking-space gives the tag enough content to render properly.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. Mauris massa sem,
tempor sed cursus et, semper tincidunt lacus.<del>&nbsp;</del> Praesent sagittis, quam
id egestas consequat, nisl orci vehicula libero, quis ultricies nulla magna interdum

```

sem. Maecenas eget orci vitae eros accumsan mollis. Cras mi mi, rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo eget, facilisis eget nunc. Cras tincidunt auctor varius.

Rules for proper use of the `` element can be found in the HTML 4 spec (<http://www.w3.org/TR/REC-html40/struct/text.html#h-9.4>).

Substitutions

Substitutions combine a delete with an insert in one snippet, and are written as curly braces and double tildes. A squiggly arrow made up of a tilde and greater-than symbol separates the old and new text. The characters to be deleted always occur to the left of the squiggly arrow.

```
Lorem {~~hipsum~>ipsum~~} dolor sit amet...
```

Despite the unique syntax, substitutions should render as a deletion followed by an insertion.

```
Lorem <del>hipsum</del><ins>ipsum</ins> dolor sit amet...
```

Newlines are treated the same as a deletion or insertion, depending on where they're located.

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. {~~Mauris massa
sem, tempor sed cursus et, semper tincidunt lacus.~>
~~}Praesent sagittis, quam id egestas consequat, nisl orci vehicula libero, quis
ultricies nulla magna interdum sem. Maecenas eget orci vitae eros accumsan mollis. Cras
mi mi, rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo
eget, facilisis eget nunc. Cras tincidunt auctor varius.
```

After rendering this example, the deletion and insertion are added to the end of the first paragraph and two newlines ensure the proper paragraph break.

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla. <del>Mauris massa
sem, tempor sed cursus et, semper tincidunt lacus.</del><ins class="break">&nbsp;</ins>
Praesent sagittis, quam id egestas consequat, nisl orci vehicula libero, quis ultricies
nulla magna interdum sem. Maecenas eget orci vitae eros accumsan mollis. Cras mi mi,
rutrum id aliquam in, aliquet vitae tellus. Sed neque justo, cursus in commodo eget,
facilisis eget nunc. Cras tincidunt auctor varius.
```

Comments

Critic Markup supports generic comments for metadata. A comment may include a note, time stamp, author initial or similar annotation.

Comments are added via a set of curly braces and double greater-than/less-than symbols.

```
Lorem ipsum dolor sit amet.{>>This is a comment<<}
```

The contents of a metadata field should render as `` after the relevant change.

```
Lorem ipsum dolor sit amet.<span class="critic comment">This is a comment</span>
```

Metadata may be used however you like, whether as explanations for the changes, time stamps or more. The `` element is for inline content only, so all newlines will be stripped during conversion to HTML.

Rules for proper use of the `` element can be found in the HTML 4 spec (<http://www.w3.org/TR/html401/struct/global.html#edef-SPAN>).

Highlights

Highlights may be added as required by an editor or author, and the text to be outlined is marked by matched curly braces and double equal signs. Since a highlight on its own is fairly meaningless, it's always followed by a comment related to the highlighted passage.

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. {==Vestibulum at orci magna.
Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla.==}{>>confusing<<}
Mauris massa sem, tempor sed cursus et, semper tincidunt lacus.
```

Highlights should be rendered as `<mark>` tags in the processed HTML.

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. <mark>Vestibulum at orci
magna. Phasellus augue justo, sodales eu pulvinar ac, vulputate eget nulla.</mark><span
class="critic metadata">confusing</span> Mauris massa sem, tempor sed cursus et, semper
tincidunt lacus.
```

Rules for proper use of the `<mark>` element can be found in the HTML 5 spec (<http://www.w3.org/html/wg/drafts/html/master/single-page.html#the-mark-element>).

Putting it all together

When used in combination the marks can indicate more complex changes.

```
Don't go around saying{-- to people that--} the world owes you a living. The world owes
you nothing. It was here first. {~~One~~>Only one~~} thing is impossible for God: To
find {++any++} sense in any copyright law on the planet. {==Truth is stranger than
fiction==}{>>strange but true<<}, but it is because Fiction is obliged to stick to
possibilities; Truth isn't.
```

The above paragraphs should render to HTML in the following manner.

```
Don't go around saying<del> to people that</del> the world owes you a living. The world
owes you nothing. It was here first. <del>One</del><ins>Only one</ins> thing is
impossible for God: To find <ins>any</ins> sense in any copyright law on the planet.
<mark>Truth is stranger than fiction</mark><span class="critic comment">strange but
true</span>, but it is because Fiction is obliged to stick to possibilities; Truth
isn't.
```

Caveats

There are a few limitations to consider when using CriticMarkup.

Avoid Newlines in CriticMarkup

Newlines should be avoided as much as possible within CriticMarkup tags. Many of the HTML elements we use are inline elements only. While we've crafted CriticMarkup with as much care as possible, it is possible for you to break the syntax and output invalid HTML if you try hard enough.

Wrap Markdown Tags Completely

While it may support incomplete Markdown tags in the future, the CriticMarkup processor currently chokes on them. Avoid this:

```
I really love *italic {~~fonts*~>font-styles*~~}.
```

Instead, wrap the asterisks completely:

```
I really love {~~*italic fonts*~>*italic font-styles*~~}.
```



Download (<https://github.com/CriticMarkup/CriticMarkup-toolkit/archive/master.zip>)



User's Guide (</users-guide.php>)



Tools (</tool-list.php>)



The Spec (</spec.php>)



About (</about.php>)



Support (<http://criticmarkup.tenderapp.com/>)

CriticMarkup was designed and developed by Gabe Weatherhead (<http://macdrifter.com>) and Erik Hess (<http://themindfulbit.com>) with a lot of help (</about.php#acknowledgements>). Say hi to Gabe (<http://twitter.com/macdrifter>) and Erik (<http://twitter.com/themindfulbit>) on Twitter: @criticmarkup (<http://twitter.com/criticmarkup>). The project and all code are open source under the Apache License, Version 2.0 (<http://opensource.org/licenses/Apache-2.0>).