

12345678
 uvwxyzabc
 uvw abc

0.0.1 MKTS Markup: Three Layers

MingKwai TypeScript uses three layers of markup.

- (1) A macro language which uses double pointy brackets—so called macro brackets—to escape into a meta-language;
- (3) MarkDown as parsed by `markdown-it`¹ (which basically means it should be quite close to the CommonMark²) specification); and
- (3) HTML markup.

The ordering in the list above reflects the precedence of the layers: the macro language takes the highest precedence and HTML constructs the lowest. Because MarkDown proper takes higher precedence than HTML, its constructs are parsed even if they appear between a pair of HTML tags. Likewise, because the macro language takes the highest priority, it is not possible to inhibit its interpretation by putting a macro inside HTML tags or inside a MarkDown fenced code block.

- ★ MKTS allows to use **HTML** in the markup.
- ★ **Regions** are used to markup stretches within a document, be it inline spans, blocks of text or longer portions like sections and chapters. Regions use parentheses to indicate the start and end points.

★ Full Notation: `<<(name)>>...<<(name)>>`

★ Short Notation: `<<(name)>>...<<>>`

When used in the long form, the name used in the start tag must match the one in the end tag. As with HTML tags, regions must be properly nested and must not overlap.

- ★ **Actions** allow to execute code snippets inside the document. They come in two flavors: ‘silent’ and ‘vocal’.
- Silent actions do not leave a direct trace in the document unless their code calls an API function to do that; vocal

actions are replaced by their evaluated value (i.e. whatever `eval("some code")` returns).

★ Silent Actions

★ Full Notation: `<<(.)>>some code<<.>>`,
`<<(js)>>some code<<js>>`

★ Short Notation: `<<(.)>>some code<<>>`,
`<<(js)>>some code<<>>`

★ Vocal Actions

★ Full Notation: `<<(:)>>some code<<:>>`,
`<<(:js)>>some code<<:js>>`

★ Short Notation: `<<(:)>>some code<<>>`,
`<<(:js)>>some code<<>>`

As with regions, the rule is that wherever the more explicit long form is used, the action type marker (. (dot) or : (colon)) and the action name of the start and end tags must match.

- ★ To execute an action without further logic code (i.e. a simple function call) in the manuscript, either a silent action `<<(.)>>makeitso 42<<.>>` or the short **Command** notation `<<!makeitso 42>>` can be used. Observe that commands do not allow for a language annotation; whatever is inside the macro tag will be parsed as CoffeeScript.
- ★ To interpolate the value of a variable into the document, either a vocal action `<<(:)>>foo<<:>>` or the shorter **Value** notation `<<$foo>>` can be used.
- ★ Finally, there are **Raw** regions that give authors an opportunity to talk directly to the \LaTeX typesetting system. Raw regions are surrounded by triple pointy brackets, e.g. `<<<...raw material...>>>`.

¹ <https://github.com/markdown-it>

² <http://commonmark.org/>