

Marking Down DITA

 *DITA-OT Markdown plugins*

infotexture
Information Architecture & Content Strategy

Roger W. Fienhold Sheen

Agenda

Markdown – Web Writing Simplified

Mobile Authoring & Lightweight Content

Markdown Meets DITA

The DITA-OT Markdown Plugin

The “Markdown DITA” Format

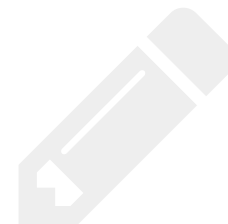
Benefits & Use Cases



Markdown – Web Writing Simplified

Created in 2004 by John Gruber & Aaron Swartz, [Markdown](#) is:

1. a plain text formatting syntax; *and*
2. software ... that converts the plain text formatting to HTML



The overriding design goal for Markdown's formatting syntax is to make it as readable as possible. The idea is that a Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions.

So while Markdown was originally designed to make it easier to write for the web without worrying about angle brackets and tags, it's proving useful for more than just websites...

Mobile Authoring & Lightweight Content



The rise of mobile devices sparked renewed interest in lightweight content formats and ease of use as authors began looking for ways to take their writing (*and their tools*) with them on the go.

Authors can capture notes with a smartphone on the go, flush out the draft back at their desk, and proofread the final result on a tablet later without copying-and-pasting or converting to other file formats along the way.

Since writing in Markdown encourages authors to focus on structure rather than presentation, it's a good match for structured authoring scenarios in which minimal markup is sufficient.

Markdown Meets DITA

The learning curve associated with XML dialects like DITA presents a barrier to broader adoption of structured authoring.



But why ask subject matter experts to struggle with XML or learn a new tool before they can provide input to our publications?

Shouldn't we just let people write and let the tools figure out what to do?

Several new solutions embrace this notion, many of which rely on Markdown:

- **Lightweight DITA**
- **DITA Glass**
oXygen's URL-based on-the-fly conversion from various file formats to DITA

Some use `h2d` to convert to HTML as an interim format, and onward to DITA. This limits vocabulary to the original Markdown syntax — *but one goes further...*

The DITA-OT Markdown Plugin



Jarno Elovirta's **DITA-OT Markdown** plugin extends the toolkit so you can use Markdown files directly in topic references. Install it with the `dita` command:

```
dita -install https://github.com/jelovirt/dita-ot-markdown/releases/ ↵  
download/1.1.0/com.elovirta.dita.markdown_1.1.0.zip
```

To add a Markdown topic, set the `@format` attribute to `markdown` so the plugin will recognize the source file as Markdown and convert it to DITA:

```
<map>  
  <topicref href="markdown-dita-topic.md" format="markdown"/>  
</map>
```

The **DITA-OT Markdown** plugin not only enables the DITA-OT to *read* Markdown, it also provides a new `markdown` transformation type that can be used to publish existing DITA content in Markdown format.

The “Markdown DITA” Format

The DITA-OT Markdown plugin introduces a new Markdown flavor called “*Markdown DITA*”, a representation of DITA content in Markdown.

The shortcut reference link syntax is used to represent DITA key references, so you can just write `[key]` to create a cross-reference like `<xref keyref="key"/>`.



Definition lists use the [PHP Markdown Extra](#) format, so

```
Term
: Definition.
```

becomes a DITA definition list:

```
<dl>
  <delentry>
    <dt>Term</dt>
    <dd>Definition.</dd>
  </delentry>
</dl>
```

Syntax Extensions



Tables use the [MultiMarkdown](#) table extension format, and Pandoc's [header attributes](#) can be used to define `id` or `outputclass` attributes, so `# Topic title { #carrot .juice}` becomes:

```
<topic id="carrot" outputclass="juice">
  <title>Topic title</title>
```

Where necessary, *Markdown DITA* establishes a few conventions of its own to support additional DITA features:

- Specify the information type of the generated DITA topic with a header attribute like `{.task}`
- Generate `<section>` and `<example>` elements with the `{.section}` and `{.example}` attributes.

The plugin's [syntax reference](#) provides an overview of the supported constructs and illustrates how DITA's XML structures are represented in *Markdown DITA*.

Benefits & Use Cases

- Markdown becomes a first-class citizen
- Makes it easier to contribute to DITA publications
- Facilitates review processes with less technical audiences
- Feed DITA content into Markdown-based publishing systems



Workflow Considerations

1. **Avoid roundtripping.**

2. *Once complex content is converted to DITA, it stays in DITA.*

If the input is a one-off contribution, use the Markdown file as raw material that is easily converted to DITA and enriched with conditional processing attributes, conkeyrefs or other more complex semantics.

3. *Simpler content stays Markdown.*

Simple content authored collaboratively over multiple versions is kept in Markdown, extended with *Markdown DITA* conventions and combined as necessary with more complex content maintained in DITA XML.

Resources

- <http://daringfireball.net/projects/markdown/>
- <https://github.com/jelovirt/dita-ot-markdown/>
- <http://infotexture.net/2015/04/dita-ot-markdown-plugin/>

