Pushing square pegs into round holes

Vanity slide

Software Developer at Wunderdog

Not affiliated with OASIS TC

Started work on Markdown for DITA-OT plug-in in early 2015

Goals of this presentation

Refresher on LwDITA and Markdown (5)

Explanation how the LwDITA for DITA-OT plug-in works (5)

Share pain points and sources of frustration (10)

Invitation to contribute to the code (1)

There's really nothing new here

At DITA-OT Day 2017 I presented on DITA-OT Markdown support

The foundations are still the same

Adoption has increased and resulted in more feature requests

Markdown

Lightweight markup language targeting readability

Uses "ASCII art" for markup, Markdown document is an ASCII preview of itself

Common structures standardized into CommonMark

Multiple incompatible flavours exists by design

DITA

XML

According to haters, a complex mess of way too many elements and attributes

Apparently no one can write DITA documents

HTML5

Kinda like SGML or XML

Web browser driven

Can be written directly but not that common

LwDITA

Content model with three syntaxes… syntaxia… syntaxen 🤷‍♀️

Syntax first development

Retrofitting DITA into Markdown, HTML5, and DITA

HDITA

HDITA specifies a subset of Markdown

No DTD but HTML5 validator can be used

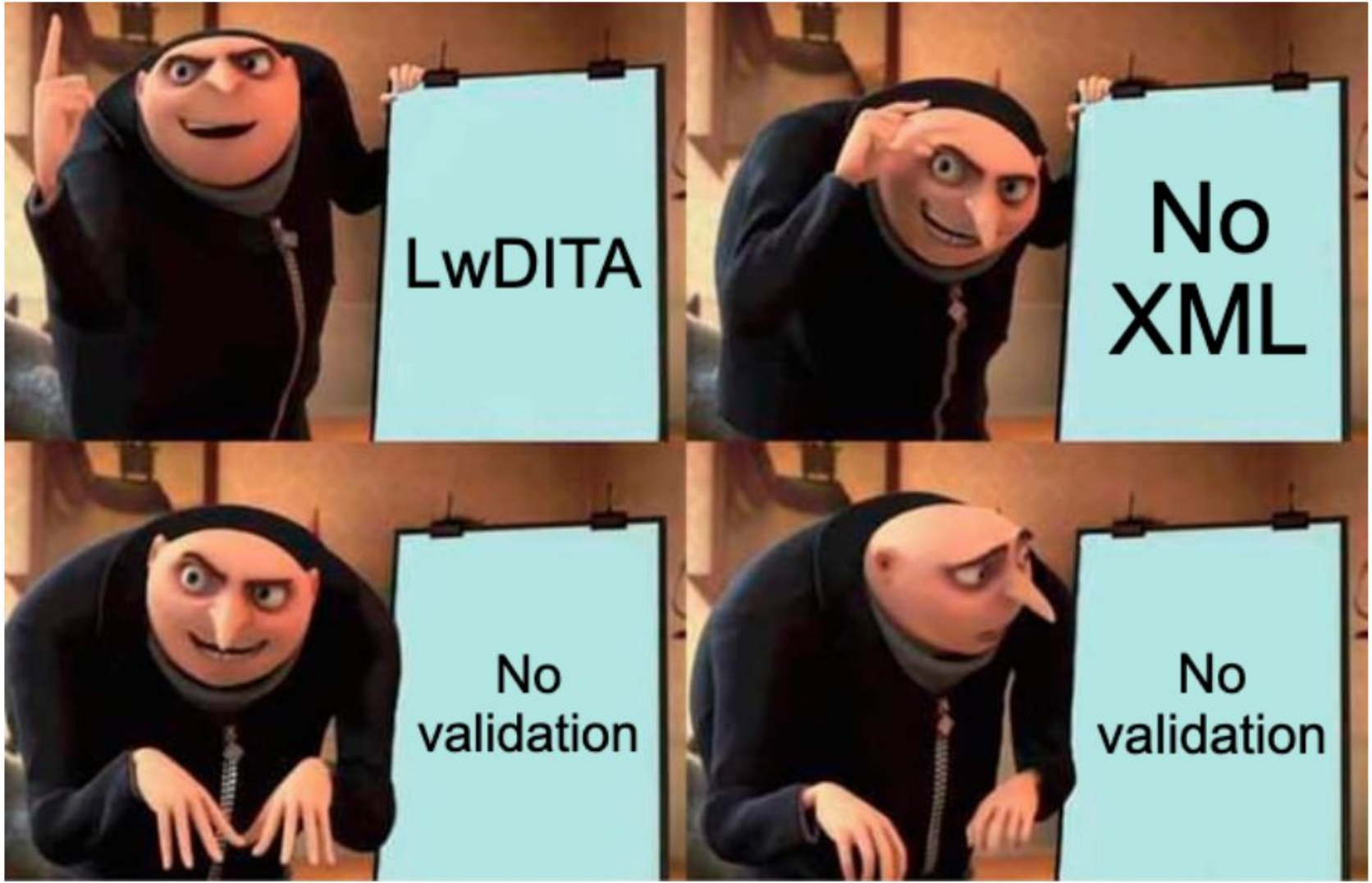
Uses HTML5 custom data attributes for DITA attributes

MDITA

MDITA specifies a subset of Markdown

Core profile is based on GFM

Extended profile cherry-picks structures from other flavours and falls back on HDITA


LwDITA
No XML
No validation
No validation

MDITA

No validation except lax parsing

In practise preview based validation

Linters to the rescue

XDITA

XDITA specifies a subset of DITA

Has its own DTD, not just restricting specialization

IMO building block between LwDITA and DITA, not intended to be used directly

LwDITA plug-in

Supports both LwDITA input and output

Uses Markdown and HTML5 parsers

Not extensible but has few configuration options

Markdown

RFC 7764: "Guidance on Markdown: Design Philosophies, Stability Strategies, and Select Registrations"

"You can't just invent syntax and expect it to work."

"Now that I have read the spec, I think you didn't."

Which Markdown

As the only developer, I have to decide what `markdown` means

What I think might not be the best answer

Luckily DITA-OT doesn't (currently) have extension points for runtime configuration

Choose Your Own Adventure

You can shoot yourself in the foot if you want

Creating a new Markdown `@format` can be done with DITA-OT plug-in configuration

Not shareable like DITA specializations

LwDITA is not 1.0 yet

LwDITA is still at draft stage

I follow what happens in LwDITA Git repository and implement the latest commit

Releases of LwDITA plug-in contain unfinished LwDITA features, bugs in the spec draft, and features that will be removed from the final spec

Superscript

A superscript is text that is printed above the line. It is frequently used in chemical and mathematical formulas.

Syntax

XDITA

```
<sup>
```

HDITA

```
<sup>
```

MDITA (extended profile)

There is no specific support in MDITA core profile. If needed, use an HDITA snippet.

Attributes

The available attributes vary based on the authoring format:

XDITA

The following attributes are available on this element: localization attributes, universal attributes, and `@keyref`.

HDITA

The following attributes are available on this element: localization attributes, universal attributes, and `@keyref`.

MDITA

For the MDITA core profile, the equivalent of the XDITA `@keyref` attribute is supported. For the MDITA extended profile, attributes can be specified by using the HDITA representation.

Examples

Figure 1. XDITA example

The following example demonstrates the use of superscript in an XDITA topic.

TODO

All of HDITA in MDITA

Maps in MDITA or HDITA

Support every Markdown flavour

Better error messages

Improve performance

More configurability via SAX features

`Some validation`

Thank you