

DOCS AS CODE AND DITA

George Bina, oXygen XML Editor

george@oxygenxml.com

[@georgebina](#)

TCUK 2019

© 2019 Syncro Soft SRL. All rights reserved.



About me

#xml #oxygenxml #syncrosoft #opensource #jing #ditaot #dance #tango

Motivation

GitHub + Markdown

vs

CMS + **DITA**



Documentation follows product

An incremental update to product implies an incremental update to documentation

Integrated development

Product development and documentation creation should be integrated

Docs As Code

Use tools and processes from code development also for documentation

Typical docs as code components

- Non-proprietary / easily editable source formats
- Issue tracking tools
- Concurrent versioning systems
- Continuous Integration / Continuous delivery (automated publishing)
- Automated tests

Source formats

- Markdown
- AsciiDoc
- reStructuredText
- XML
 - **DITA**
 - DocBook

DITA source format

Checks all the docs as code requirements, better than Markdown or other formats

Non-proprietary

DITA is an OASIS standard

Easily editable

DITA = XML = text

Lightweight DITA = DITA = XML | Markdown | HTML = text

Multi-format support

You can work with other formats in DITA (Markdown, Excel, HTML)

Concurrent versioning systems

DITA works with any concurrent versioning system, Git in particular
XML/DITA aware comparison options are available for improved diff/merge

Automated tests

- Structural validation
- Business rules with quick-fixes
- Integrity checks

Automated publishing

DITA-OT can be used as a static site generator and for more output types
Transformations can be triggered using a CI server and deployed automatically

Publish to other formats

Markdown, HTML can be also target formats

Semantic layer

Basic structure and inline highlighting (similar to HTML/Markdown)

Specialized structure and many specific domains

Modularity and reuse

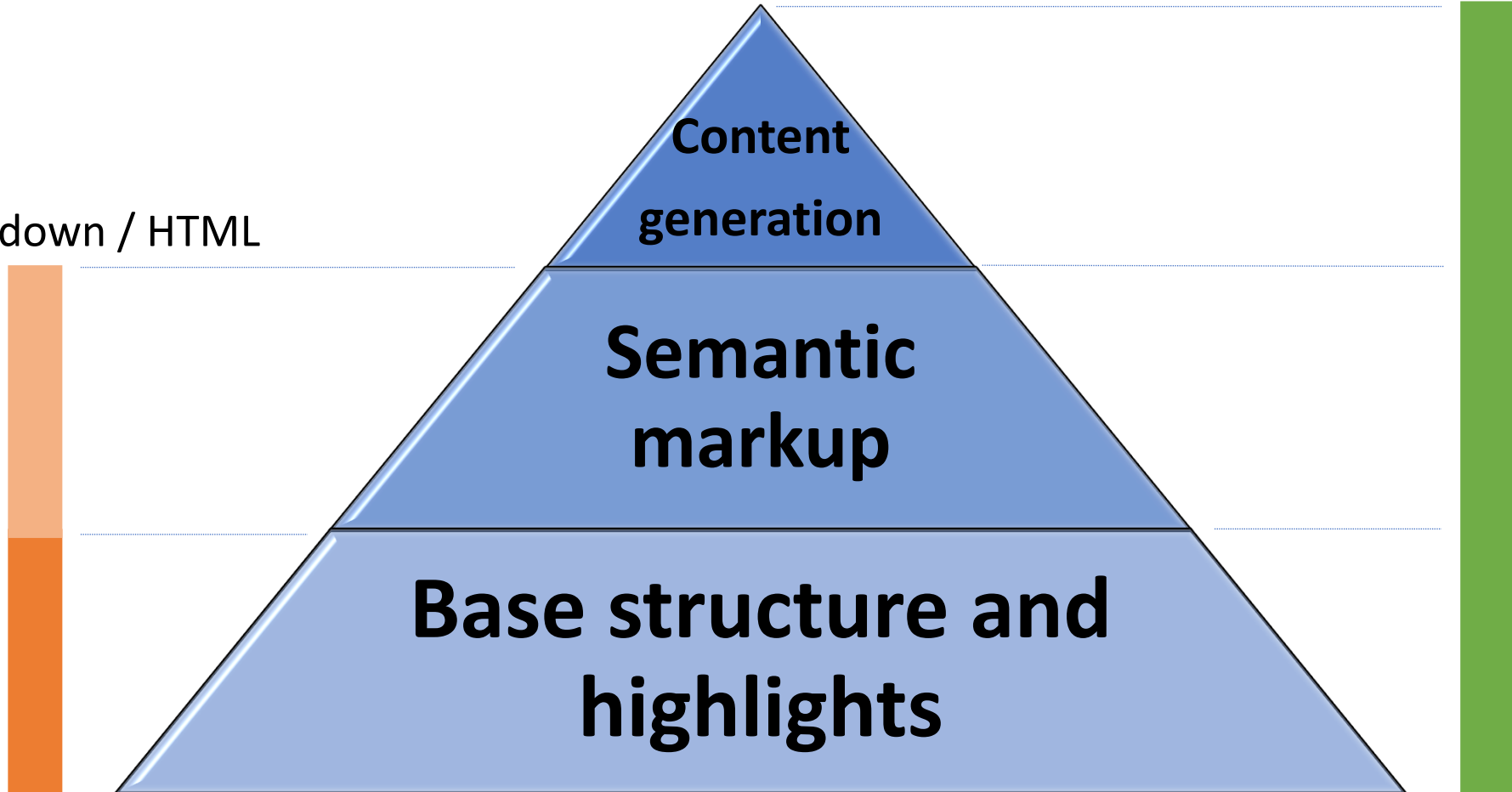
Products are modular and reuse components, the documentation should be similar

DITA provides modularity and reuse

Format capabilities

DITA

Markdown / HTML



+ Extensibility

External vocabularies (MathML, SVG), specialization mechanism

+ Metadata

Annotate topics with useful information

Docs as code and DITA examples

DITA-OT Day 2018

A number of sessions presented docs as code setups

https://www.oxygenxml.com/events/2018/dita-ot_day.html

DITA-OT

De-facto reference implementation of DITA

<https://www.dita-ot.org/>

DITA-OT documentation

Docs as code setup using DITA as source format, Jekyll for publishing

<https://github.com/dita-ot/docs>

Project management

GitHub

- Git repository
- Issue tracking
- Workflow

Build

Travis

- Gradle to build
- Generate HTML and use Jekyll to publish
- Publish on GitHub Pages / Netlify
- Edit link to allow online editing of latest DITA source

Integrations

- GitHub / Slack – messages on documentation changes

DITA features

- Profiling
- Controlled values
- Indirect references
- Variable text
- Semantic markup
- Push-modified topics

oxygen documentation

Another example of docs as code setup using DITA

<https://github.com/oxygenxml/userguide>

Project management

GitHub

- Public Git repository
- Private Git repository

Jira (on premise)

- Issue tracking
- Workflow

Build

Jenkins

- Ant to build
- Generate multiple formats using DITA-OT and plugins
- Publish on staging server automatically
- Publish on production website on request
- Edit link to allow online editing of latest DITA source

Test

- Structural validation
- Business rules using Schematron
- DITA completeness checks

Integrations

- GitHub / Jira – enable developers to review documentation
- GitHub / Slack – messages on documentation changes

DITA features

- Profiling
- Controlled values
- Indirect references
- Variable text
- Semantic markup

Final notes

- DITA, and XML in general, is already used in docs as code setup
- DITA handles complex scenarios but it can be simple for simple tasks
- DITA allows you to start simple and grow in a controlled, standard and tested way
- For publishing you benefit of both DITA and HTML/Markdown options

Image references

- [Apples and Oranges](#) by [MicroAssist](#) is licensed under [CC BY-SA 2.0](#)

Thank you!

Questions?

George Bina
george@oxygenxml.com
[@georgebina](#)