



# Web Author

Empower SMEs to Create High Quality  
Structured Content

Cristian Talau

[cristi\\_talau@oxygenxml.com](mailto:cristi_talau@oxygenxml.com)

<https://github.com/ctalau/>

# Oxygen XML Web Author



# Oxygen XML Web Author



uses



# Oxygen XML Web Author



# Subject Matter Experts

Je ne parle pas  
XML!



Ich spreche kein  
XML!

# Task 1: Access XML content

- Support for new file servers:
  - Perforce
  - CMIS: configurable icon and tab name
- Link to open folder in Web Author
- Anonymous Git access



On-premise

# Task 1: Access XML content

- Git Submodules:
  - Images repository:
    - synchronized from a Digital Assets Management system
  - Legal content:
    - Copyright, Various warning messages required by law
    - Reused in all projects across company

-----  
git clone git://github.com/foo/bar.git  
... Parts of content are missing, google it ...  
git submodule update --init --recursive

# Task 2: Read XML content

- The best presentation depends on the user's task
  - View content to review / search for something
  - View titles to work on the TOC
  - Edit topicrefs in-place



# Task 2: Read profiled XML

- Profiled content is confusing without editor support:

## Memory



- Minimum - ▷2 GB of RAM◁
- Recommended - ▷4 GB of RAM◁▷2 GB ◁

# Task 2: Read profiled XML

- Profiling: displayed inline & greyed out

## Memory



- Minimum - >2 GB of RAM<
- Recommended - >4 GB of RAM (if product is 'author' or 'developer' or 'editor') ✎ ◀ | >2 GB (if product is 'svnClient') ✎ ◀

# Task 2: Read content with vars

- Content with variable names:

It is possible for your organization to use your own `cfEnterprise` server to store uploaded content rather than the server hosted by the `cfPublic`.

# Task 2: Read content with vars

- Content with expanded variable content:

It is possible for your organization to use your own `>Oxygen Content Fusion Enterprise<` server to store uploaded content rather than the server hosted by the `>Oxygen Content Fusion Demo Service<`.

# Task 3: Write text

- Custom term dictionaries

```
<Dictionary>  
  <Forbidden>  
    <w>Github</w>  
    <w>GitHUB</w>  
  </Forbidden>  
  <Learned>  
    <w>GitHuB</w>  
  </Learned>  
</Dictionary>
```

# Task 3: Write text

- Autocorrect

```
<replacements>
```

```
  <addition match="oXygen" substitution="Oxygen"/>
```

```
</replacements>
```

# Task 3: Write text

- Spell checking with custom engines
  - API for implementing custom ones
  - Language Tool:
    - <https://github.com/oxygenxml/web-author-languagetool-plugin>
    - Spanish, Simplified Chinese, Italian, Dutch, Japanese
- Acrolinx

# Task 3: Custom placeholders

- `<?oxy-placeholder  
content="Describe in less than 50 words what task does"?>`

**Enter the task title**

**Short Description:** Describe in less than 50 words what this task does

**About this task:**

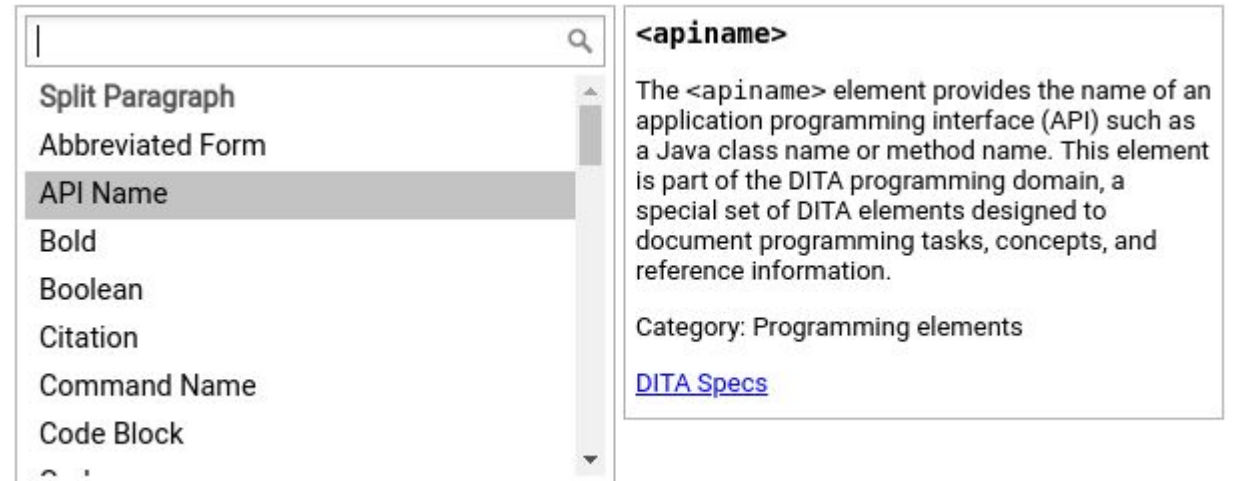
Specify the task context

1. Enter the step action



# Task 4: Add markup

- Inserting elements:
  - Toolbar buttons
  - Enter
  - Content completion
    - Includes documentation
    - Wizards (e.g. for images)
    - Available in text mode

The image shows a screenshot of the Oxygen XML Editor's content completion feature. On the left, a dropdown menu is open, listing various elements. The 'API Name' element is highlighted in grey. To the right of the dropdown, a panel displays the documentation for the selected '<apiname>' element. The documentation includes a description of the element's purpose in DITA programming, its category, and a link to 'DITA Specs'.

## <apiname>

The <apiname> element provides the name of an application programming interface (API) such as a Java class name or method name. This element is part of the DITA programming domain, a special set of DITA elements designed to document programming tasks, concepts, and reference information.

Category: Programming elements

[DITA Specs](#)

# Task 4: Work with tables

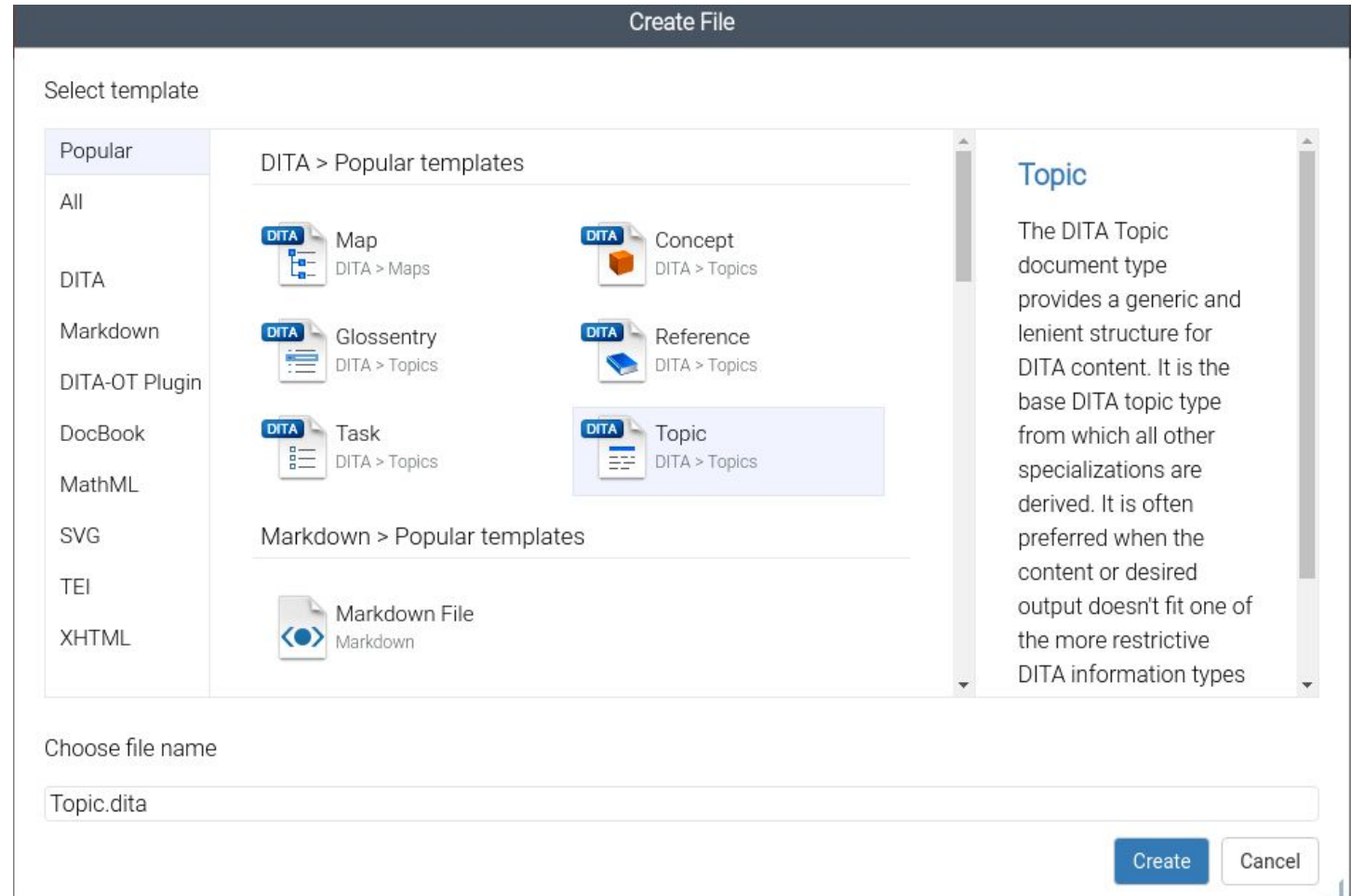
- Reorder rows
- Duplicate a column and modify its content
- Bold the feature names
- Join cells

# Task 4: Reuse content

- Automatically insert DITA key-based references instead of path-based references
- Reusing content - variables

# Task 5: Create new docs

- Template chooser
- Draft in Markdown



Create File

Select template

Popular

All

DITA

Markdown

DITA-OT Plugin

DocBook

MathML

SVG

TEI

XHTML

DITA > Popular templates

Map  
DITA > Maps

Concept  
DITA > Topics

Glossentry  
DITA > Topics

Reference  
DITA > Topics

Task  
DITA > Topics

Topic  
DITA > Topics

Markdown > Popular templates

Markdown File  
Markdown

Topic

The DITA Topic document type provides a generic and lenient structure for DITA content. It is the base DITA topic type from which all other specializations are derived. It is often preferred when the content or desired output doesn't fit one of the more restrictive DITA information types

Choose file name

Topic.dita

Create Cancel

# Task 6: Create new projects

- Project templates
  - Paper template: XML Prague Paper template
    - <https://github.com/ctalau-oxygen/prague-template>
  - Wiki:
    - <https://ghd-wiki-ctalau.netlify.com/>
    - <https://github.com/ctalau/ghd-wiki>

# THANK YOU!

**Any questions?**

Cristian Talau

[cristi\\_talau@oxygenxml.com](mailto:cristi_talau@oxygenxml.com)

<https://github.com/ctalau>