

Cost Effective Solutions for Technical Documentation

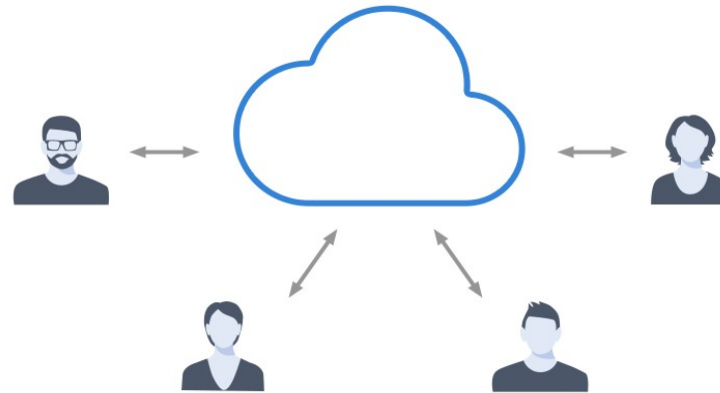
Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu



Details

- Storage
- Collaboration
- Project structure
- Managing links and reused content
- Deliverables

Storage

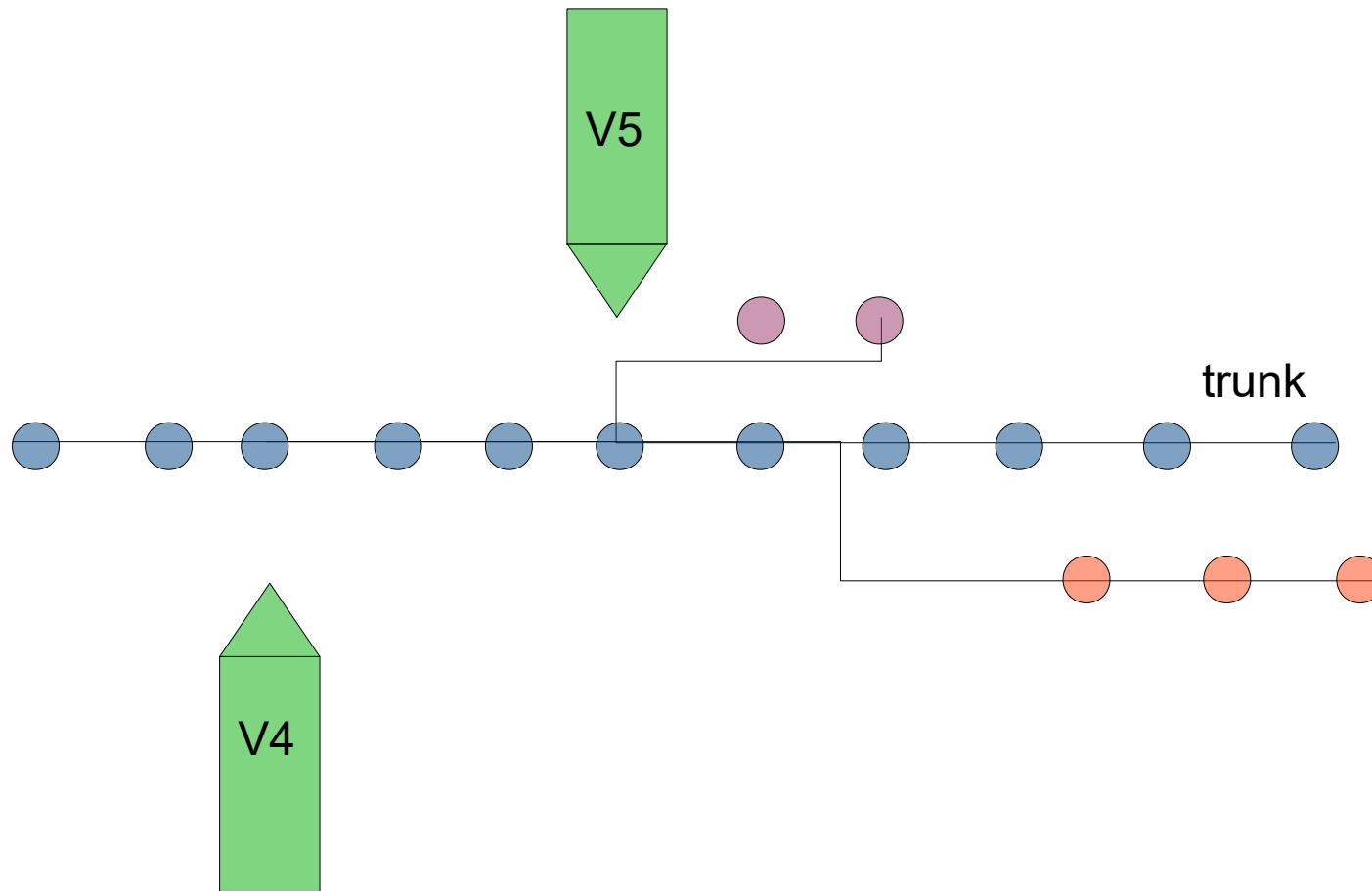


- Commercial content management systems (CMS).
- Open Source version control systems: Git, Subversion, CVS

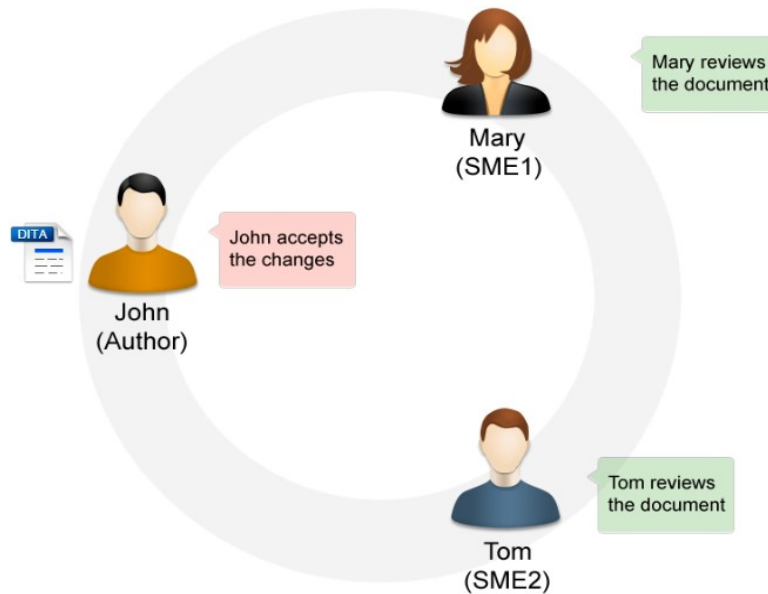
Possible affordable content repositories

- Git (eg: GitHub, GitLab)
- Subversion (eg: Apache Subversion).

Version Control



Collaboration



Open Source version control systems – Local working copies, no editing restrictions.

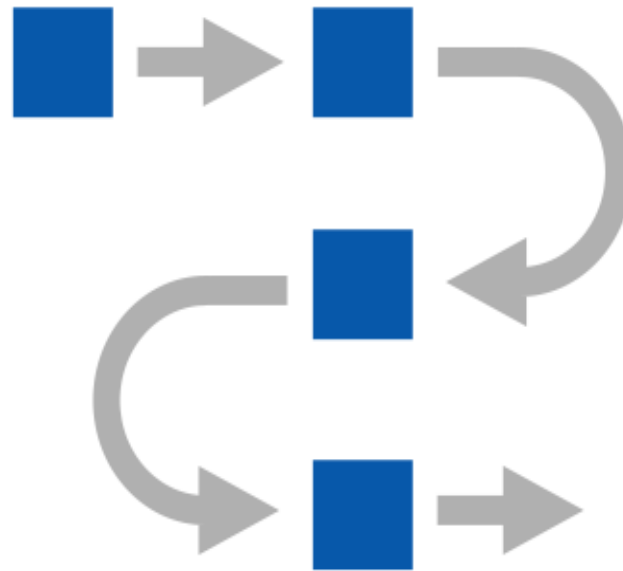
Example:

Using Oxygen Git plugin or SourceTree to manage GitHub project.

Issue tracking

- Using issue management systems like Bugzilla or Atlassian JIRA or GitHub.
 - Linking the product development with the documentation development.

Workflow

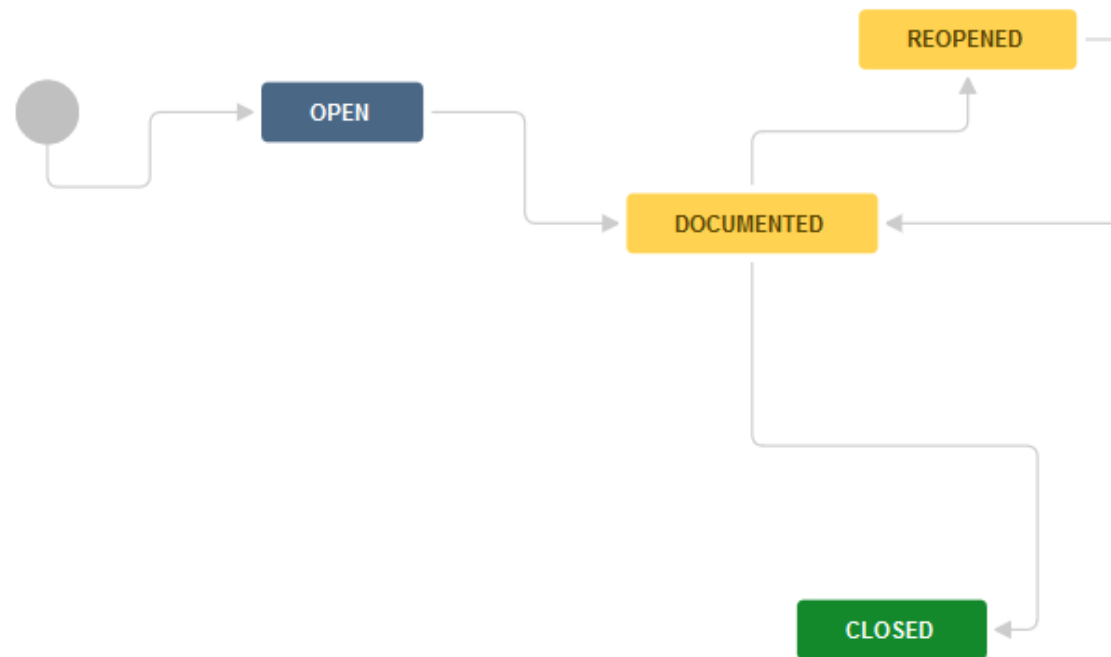


Issue Tracking

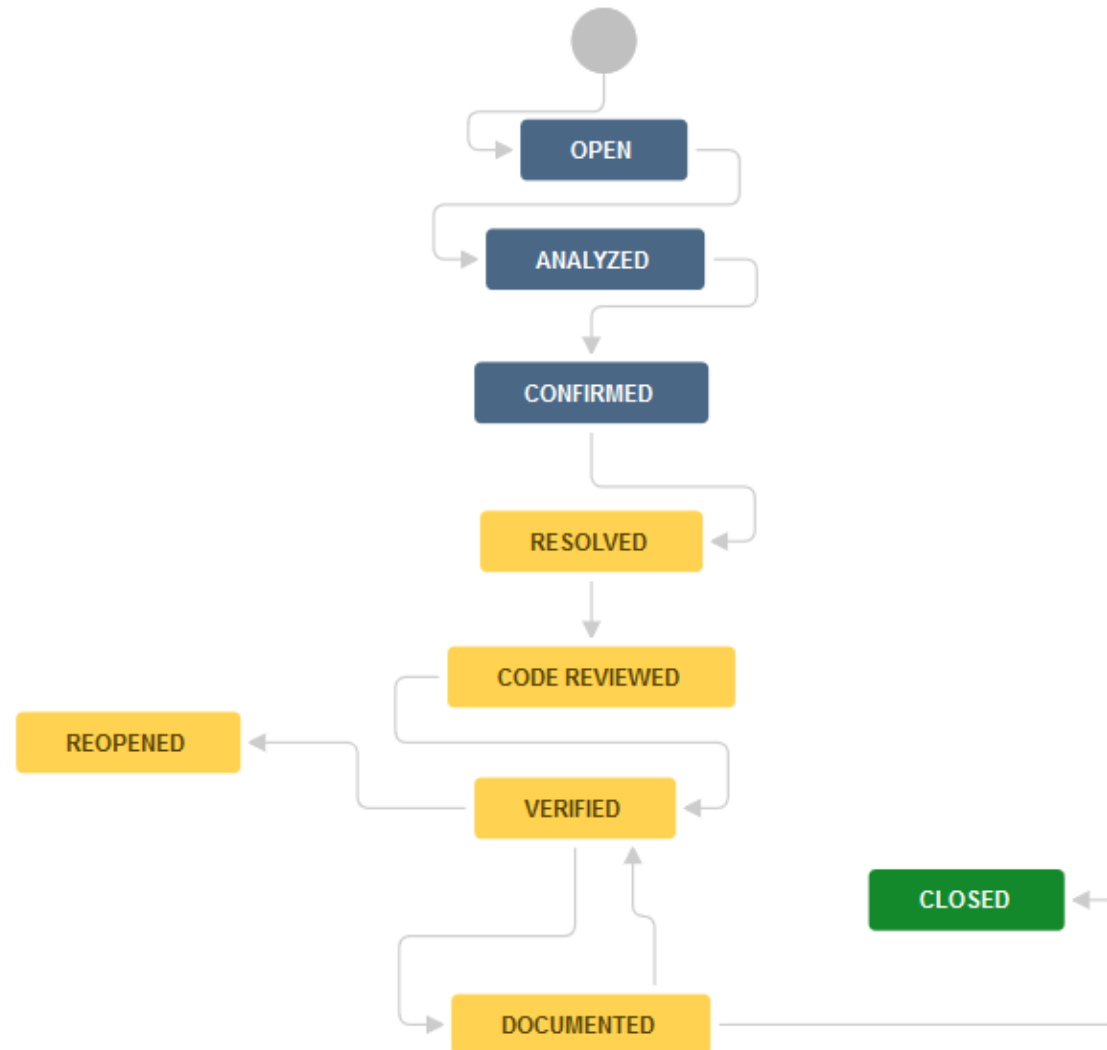
Custom workflows

- Documentation task specific workflow
- Integrate QA and documentation in software development process

Issue tracking – Simple Documentation Workflow



Issue tracking – Development and Documentation Workflow




Issue Tracking and Storage Integration

- Jira and SVN/GIT integration
 - Provides a single place where you can monitor an issue from start to end, including
 - Issue description
 - Who worked on that issue
 - What was changed in the application
 - What was changed in the documentation

Documentation Changes

Repository	Revision	Date	User	Message
Site and Documentation	#28921	Thu Sep 24 09:45:38 EEST 2015	steven_higgs	EXM-34450 : Added more info to a feature in What's New in 17.1 section
				Files Changed
				MODIFY /www.oxygenxml.com/branches/Release_17.1/xml/site.xml

**StevenHiggs** 24/Sep/15 9:46 AM Forks 2

EXM-34450: Added a note about DITA OT 2.x templates and revised the DITA preference page View full commit

dev

- +6 -1 DITA/topics/author-dita-templates.dita
- +20 -16 DITA/topics/preferences-dita.dita

Repository	Revision	Date	User	Message
Site and Documentation	#28922	Thu Sep 24 09:49:54 EEST 2015	steven_higgs	EXM-34450: Added more info to a feature in What's New in 17.1 section
				Files Changed
				MODIFY /www.oxygenxml.com/branches/Release_17.1/xml/site.xml

Involving reviewers

- Reviewing directly the DITA content
- Using change tracking and comment capabilities
- Give feedback on the published output

Review

- Comments/annotations
- Change tracking

◀ - The schema diagram editor will combine schemas imported by the current schema file into a single schema diagram. You can choose what happens if you try to edit a component from an imported schema. The options are:

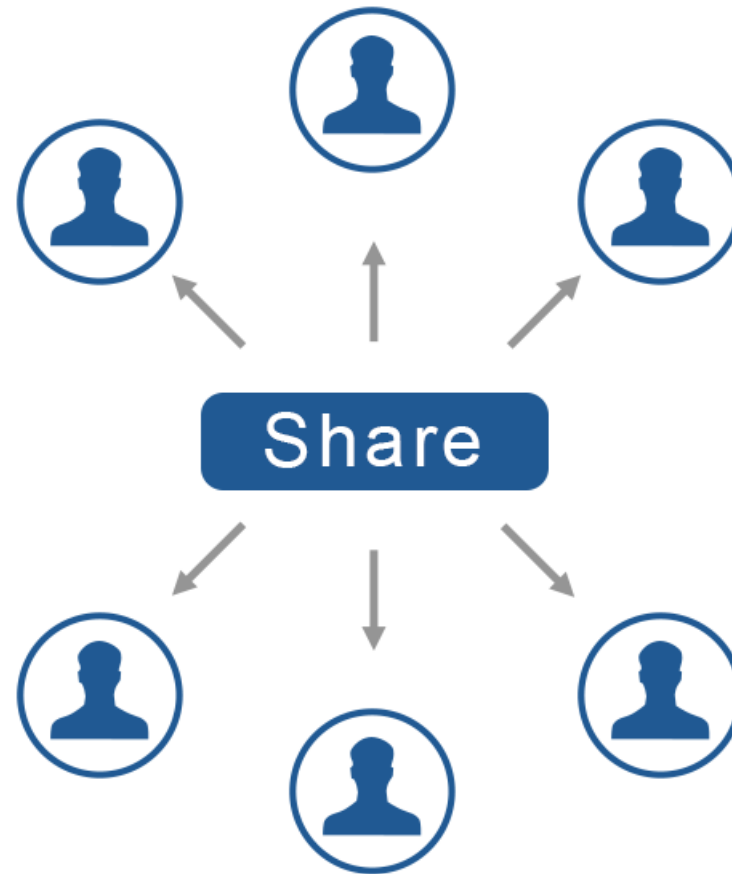
- ▶ **Always go to its definition** ▶ - [product] ▶ Oxygen XML Editor ▶ opens the imported schema file so that you can edit it.
- ▶ **Never go to its definition** ▶ - The imported schema file is not opened. The definition cannot be edited in place. Edits the component definition in the current file;
- ▶ **Always ask** ▶ - [product] ▶ Oxygen XML Editor ▶ asks if you want to

✓ Accept change(s)

✗ Reject change(s)

Commented [Mark]: To original document said that you can edit the imported schema in place, but I tested this and you can't. Am I missing something?

Sharing common settings



Sharing common settings

- Custom Spell and auto-correct dictionaries.
- Various Common preferences.
- Controlled attribute values.

Using the same terminology rules

<http://blog.oxygenxml.com/2017/06/checking-terminology-when-editing-in.html>

- Free to use Spell dictionaries.
- Possibility to create custom dictionaries (eg: medical terminology).
- Custom auto-correct mappings.
- Advanced free terminology checkers: LanguageTools.
- Building your own terminology checker:
<https://github.com/doctales/org.doctales.terminology>

Custom Style Guide

- Internal style guide, documentation about how to do documentation.
- Automatic checks based on the style guide.

Schematron Project-specific checks

Examples:

- Report invalid pattern for topic IDs.
- Report web links with redundant text content.
- Report codeblocks without outputclass attribute.
- Report empty paragraphs and list items.
- Report consecutive ordered/unordered lists.
- Report tables with more cells per row than specified column number.

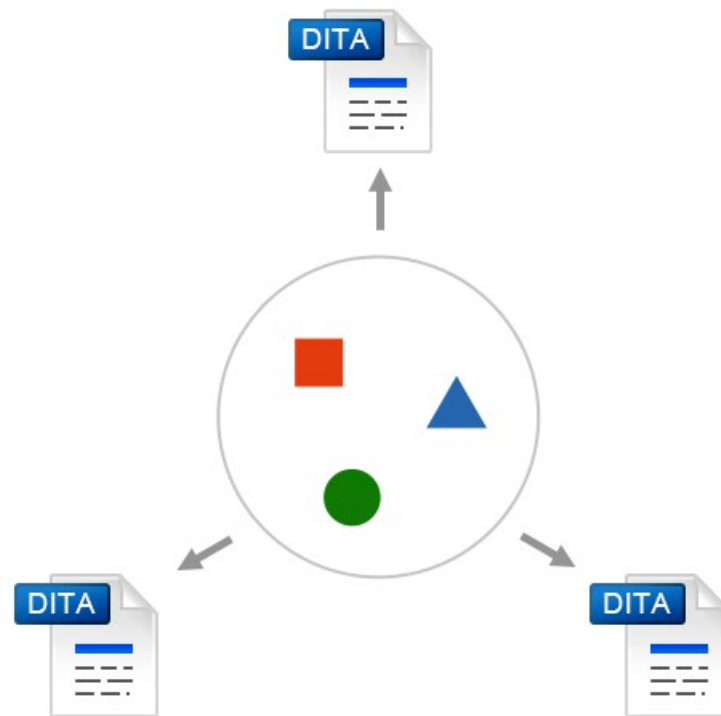
Schematron Quick Fixes

Examples:

- Inserting index terms in topic.
- Set the proper ID to a topic.
- Remove redundant text from web link.

Reusing Content

<http://blog.oxygenxml.com/2015/11/dita-reuse-strategy>

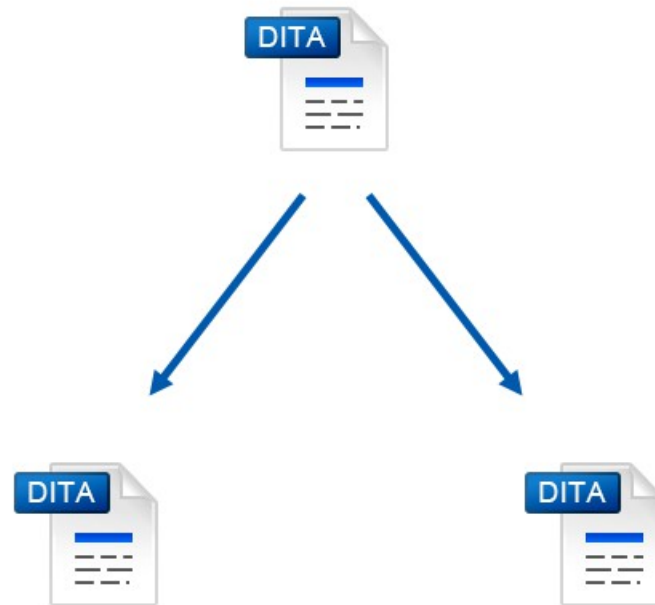


Managing Content Reuse

- Keep dictionaries of reusable components
- Prefer indirect references (conkeyrefs)
- Use Oxygen tools like “Open/Find Resource” or “DITA Reusable Components” to locate reused content.

Managing Links

<http://blog.oxygenxml.com/2017/06/dita-linking-strategies.html>



Managing Links

- Linking is a form of reuse
- Reuse small pieces of content instead of linking to them
- Avoid too much linking (Linking is disruptive)
- Prefer indirect links
 - Reuse link targets
 - Re-direct links depending on publication
- Use relationship tables

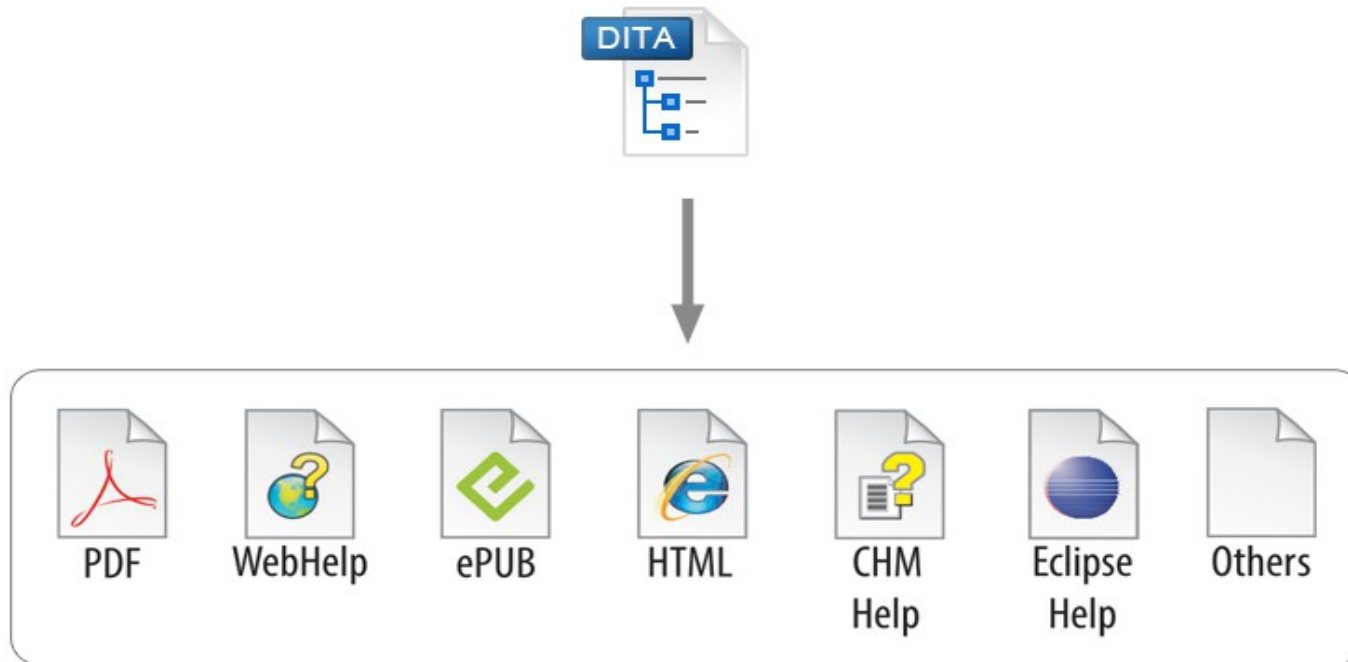
Map-wide Validation and Consistency Checks

- Validate each topic according to DITA standard.
- Check for broken links, key references and content references.
- Check for missing images or referenced resources.
- Check for broken links to remote web sites.
- Check for broken links in the context of profiling filters.
- Find unreferenced resources.

Project-wide refactor operations

- Change XML structure in topics from the entire project.
 - Example: Replace image @alt attribute with element.
- Convert between various topic types.
- Rename or move one or more topics.

Producing the deliverables



Producing the deliverables

- Use the DITA Open Toolkit open source processor to publish to PDF and XHTML-based output formats.
- Use the large DITA Users community to ask for help.
- Use additional output formats provided free to use from inside Oxygen (WebHelp Responsive/Classic/Mobile).

Producing the deliverables

- Checking the project before publishing.
- Sharing publishing customizations
- Automatic production of deliverables via an automated open source server (Jenkins).

Cost effective DITA projects can still:

- Be manageable
- Allow for scalability
- Allow for easy collaboration
- Allow for detection and correction of mistakes before the deliverables are published
- Allow for correction of mistakes after the deliverables are published

Thank You!

Questions?

Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu