



# IXIASOFT

## Create Custom Review PDFs

DITA-OT Day, Berlin

29 October 2017

Leigh W. White, DITA Specialist

# The challenge

- You have a 100-topic publication map
- You need to send 30 topics for review
  - To multiple SMEs, in different combinations
  - Some SMEs want to see only content assigned to them
  - Other SMEs want to see everything
  - But they want to know what's specifically assigned to them so they can skim the rest
  - You really want to give each SME what they want to encourage them to review
- How do you assemble this content in all these different ways for each SME?

# Manual solution

- Multiple temporary review maps?
  - One per SME
  - With just the content for that SME—or everything, depending on the SME’s preference
  - If everything, then somehow tag topics per SME & use ditaval to highlight, color-code, etc.
  - Also tag topics per status (& use ditaval) so SME can ignore “in progress” or “complete” topics
  - Then, when review is complete, remove tags so content is clean for next review cycle & delete temporary maps
- Grim, even if scripted
- But, if you have a capable CMS...use it!

# Leverage your CMS and the DITA OT

- This solution uses the IXIASOFT DITA CMS
  - Because it lends itself perfectly to this kind of thing
  - And because I work for them
- But it's not a DITA CMS demo
  - The heavy lifting is done by an OT plugin
  - The DITA CMS just provides some information
  - If other CMS's can provide the same info, this principle should work with them as well

# DITA CMS Collaborative Review

- DITA CMS includes collaborative review feature
  - Web/HTML-based only (for now)
    - Some companies require PDF review
  - Allows multiple reviewers to simultaneously review content and see each other's comments
  - Does not allow for per-SME output
  - Does not allow for on-the-fly filtering
    - Can filter but applies to all reviews
  - What I'm showing could just as easily be done in HTML outputs
  - We might incorporate some of this in either medium
  - Or...you can do it yourself!

# DITA CMS information (1)

- Topic filename
- Recorded in the object's properties file
  - [@oid].properties
  - Which is sent to the Output Generator's temp job directory **so it's available to the transform**
    - (Simplistically, the Output Generator is a “shell” around the DITA-OT that the DITA CMS uses to communicate with the DITA-OT)

# .properties file

- (abridged)

```
<documentproperties>  
  <systemproperties>  
    <collection>/content/authoring/</collection>  
    <name>ahz1498765840476.xml</name>  
    <version>1</version>  
    <size>19293</size>
```

# DITA CMS information (2)

- Topic status
- Assigned reviewers
  - Date review is due
  - Comments to reviewer
- Recorded in the object's custom properties file
  - `[@oid].customproperties`



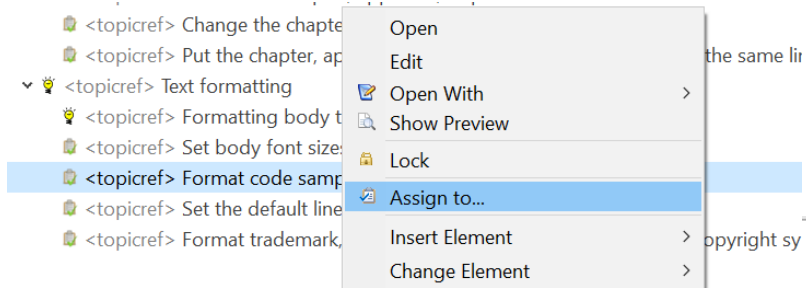
# .customproperties file

- (abridged)

```
<customproperties>
  <status>Authoring:review</status>
  <userproperties>
    <userproperty type="originalRelativePath">./</userproperty>
    <userproperty type="originalFilename">txt_format_commands_and_messages.xml</userproperty>
    <userproperty type="originalId">t20130227234227589</userproperty>
  </userproperties>
  <assignment assignedBy="Leigh White [leigh.white@ixiasoft.com]" assignedOn="2017-06-29 20:13:20"
    dueDate="2017-11-03" role="Reviewer" url="" votingSystem="">
    <assignedTo>Jean-François Ameye [jean-francois.ameye@ixiasoft.com]</assignedTo>
    <assignedTo>Keith Schengili-Roberts [keith.roberts@ixiasoft.com]</assignedTo>
    <assignedTo>Nolwenn Kerzreho [nolwenn.kerzreho@ixiasoft.com]</assignedTo>
    <assignedTo>Boris Aguilar [boris.aguilar@ixiasoft.com]</assignedTo>
    <comment>Per conversation with Leigh; she could be completely wrong</comment>
  </assignment>
</customproperties>
```

# Assigned reviewers

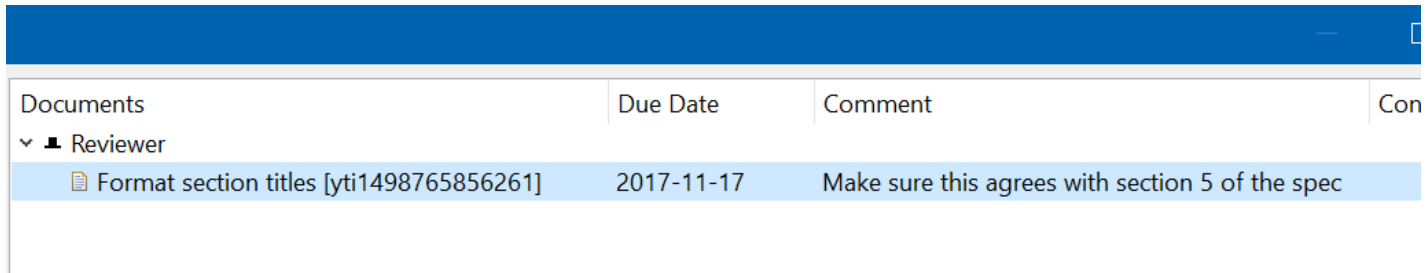
- Topics are assigned to different combinations of reviewers



	Boris	Jean-Francois	Keith	Nolwenn
Format section titles			x	x
Format the chapter, appendix, or part autonumber			x	
Change the chapter number label and format	x			x
Put the chapter, appendix or part label, number, and title on the same line	x			
Format code samples and messages	x	x	x	x
Format trademark, registered trademark, service mark, and copyright symbols		x		

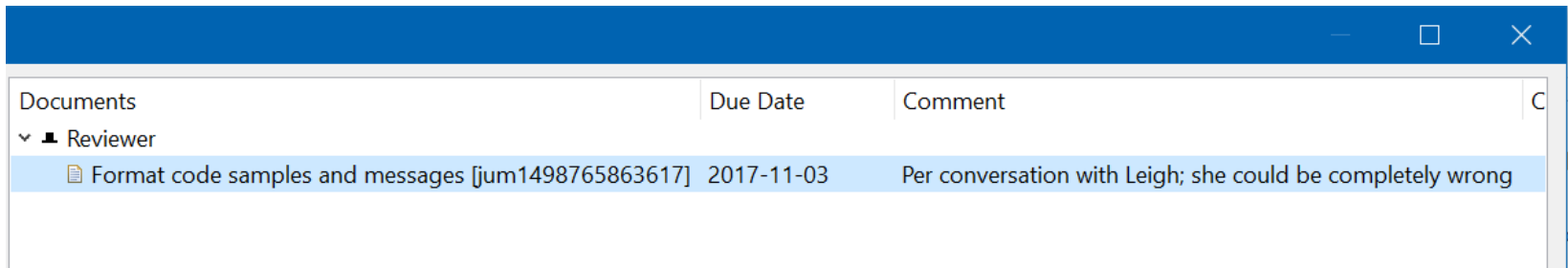
# Review due date and comments

- Some review requests have associated due date and comments that should appear in review PDF



A screenshot of a software interface showing a table with three columns: Documents, Due Date, and Comment. The table is filtered by a reviewer. One row is highlighted, showing a document titled 'Format section titles [yti1498765856261]' with a due date of '2017-11-17' and a comment 'Make sure this agrees with section 5 of the spec'.

Documents	Due Date	Comment
▼ Reviewer		
Format section titles [yti1498765856261]	2017-11-17	Make sure this agrees with section 5 of the spec



A screenshot of a software interface showing a table with three columns: Documents, Due Date, and Comment. The table is filtered by a reviewer. One row is highlighted, showing a document titled 'Format code samples and messages [jum1498765863617]' with a due date of '2017-11-03' and a comment 'Per conversation with Leigh; she could be completely wrong'.

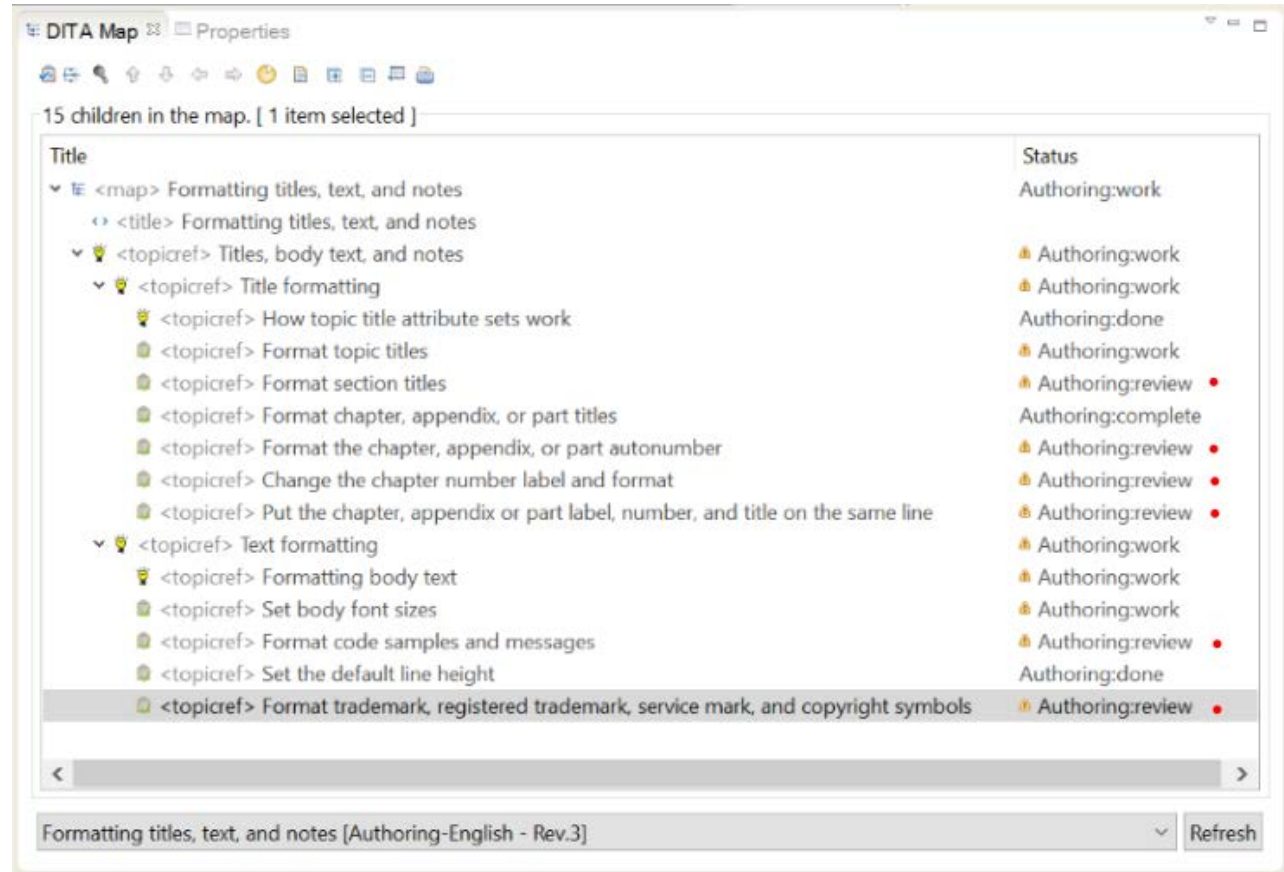
Documents	Due Date	Comment
▼ Reviewer		
Format code samples and messages [jum1498765863617]	2017-11-03	Per conversation with Leigh; she could be completely wrong

# Stuff from the topic

- oXygen Track Changes comments
- <draft-comment>

# Topic statuses

- Only topics at Authoring:review need to be reviewed



The screenshot shows the 'DITA Map Properties' window. The main area displays a tree view of 15 children in the map, with 1 item selected. The selected item is '<topicref> Format trademark, registered trademark, service mark, and copyright symbols', which has a status of 'Authoring:review'. Other topics in the list include 'Formatting titles, text, and notes' (Authoring:work), 'Titles, body text, and notes' (Authoring:work), and various 'Title formatting' and 'Text formatting' topics with statuses ranging from 'Authoring:work' to 'Authoring:done'.

Title	Status
<map> Formatting titles, text, and notes	Authoring:work
<title> Formatting titles, text, and notes	
<topicref> Titles, body text, and notes	Authoring:work
<topicref> Title formatting	Authoring:work
<topicref> How topic title attribute sets work	Authoring:done
<topicref> Format topic titles	Authoring:work
<topicref> Format section titles	Authoring:work
<topicref> Format chapter, appendix, or part titles	Authoring:review
<topicref> Format the chapter, appendix, or part autonumber	Authoring:complete
<topicref> Change the chapter number label and format	Authoring:review
<topicref> Put the chapter, appendix or part label, number, and title on the same line	Authoring:review
<topicref> Text formatting	Authoring:work
<topicref> Formatting body text	Authoring:work
<topicref> Set body font sizes	Authoring:work
<topicref> Format code samples and messages	Authoring:review
<topicref> Set the default line height	Authoring:done
<topicref> Format trademark, registered trademark, service mark, and copyright symbols	Authoring:review

Formatting titles, text, and notes [Authoring-English - Rev.3] Refresh

# oxygen Track Changes comments

- Some topics have oxygen comments that should appear in the PDF

\*Format the chapter, appendix, or part autonumber

task taskbody context p

```
<indexterm> (parts ( <indexterm> format label  
number <indexterm> ) ) <indexterm>  
<indexterm> (labels ( <indexterm> format number  
in chapter, appendix, part <indexterm> ) )  
<indexterm>  
<indexterm> (autonumbering) <indexterm> <keywords>  
<metadata> <prolog>
```

<taskbody> <context> **About this task:**

<p> By default, the PDF plugin generates a two-part autonumber for chapter, appendix, and part titles. The autonumber consists of a label (the words Chapter, Appendix, Part, or any alternatives you choose) and the number. Each part has its own attribute set. <p> <context>

<steps>

<step> 1. <command> In <filepath> <lot-attr-path> <keyword>

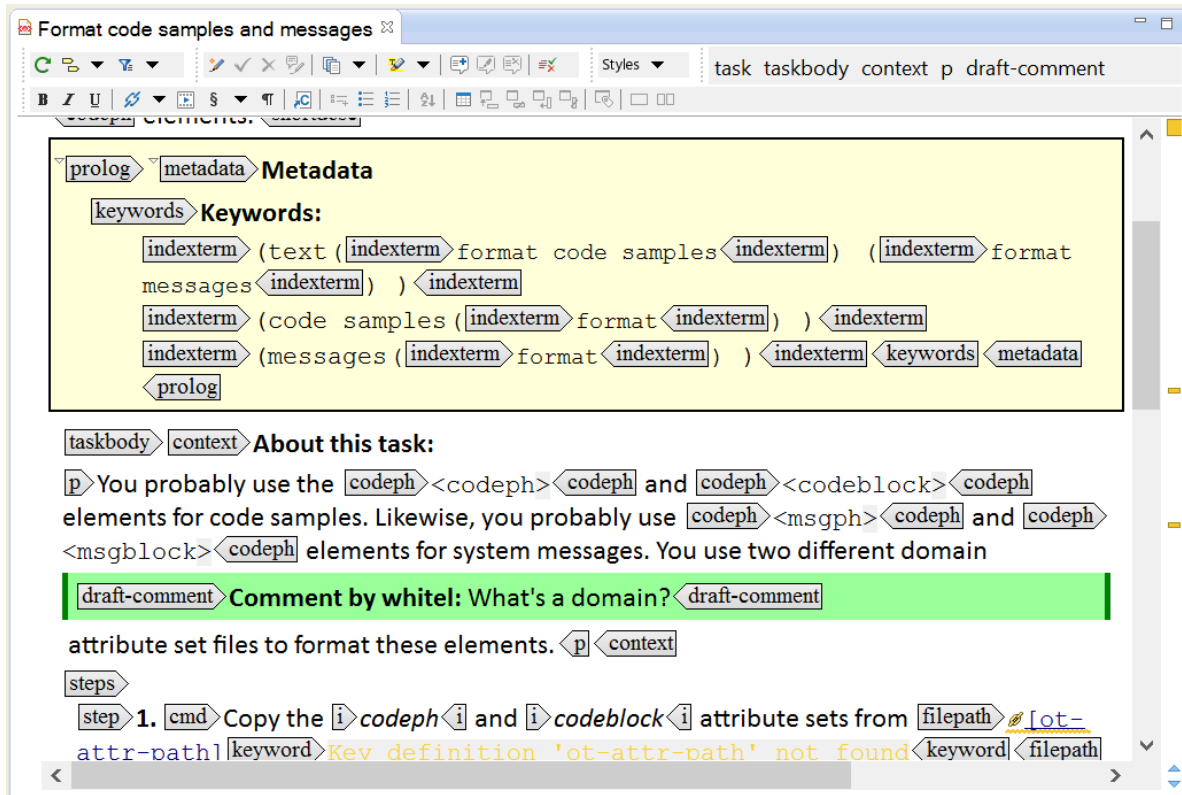
Key definition 'ot-attr-path' not found

<keyword> <filepath> <b> static-content-attr.xml </b>, find

whitel: Keith, could I get an explanation of how the numbering function works?

# <draft-comment>

- This is an OT freebie using the args.draft ANT parameter
- But we'll dress it up a little bit



Format code samples and messages

task taskbody context p draft-comment

**Metadata**

**Keywords:**

```
(text (format code samples) (format messages))
```

```
(code samples (format))
```

```
(messages (format))
```

prolog

**About this task:**

You probably use the `<codeph>` and `<codeblock>` elements for code samples. Likewise, you probably use `<msgph>` and `<msgblock>` elements for system messages. You use two different domain

**Comment by whitel: What's a domain?**

attribute set files to format these elements.

**steps**

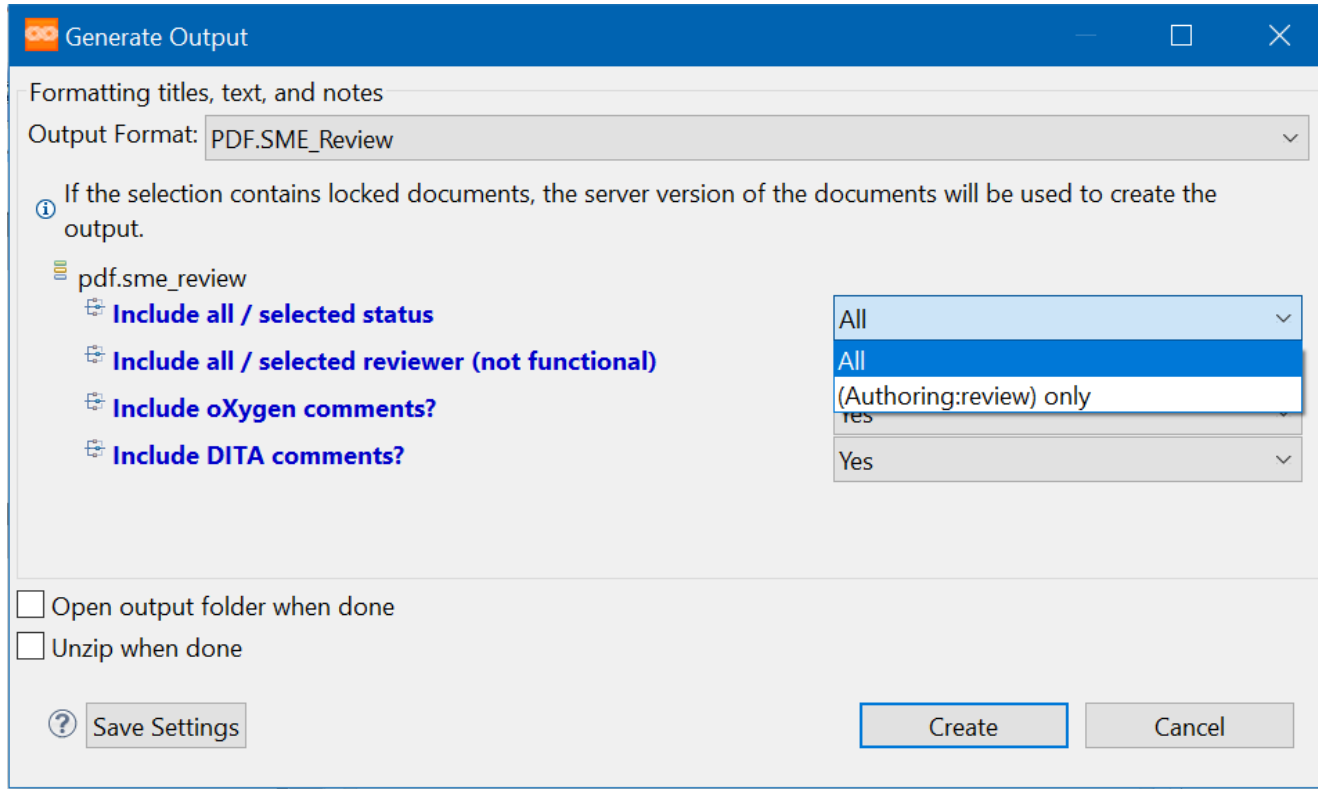
1. Copy the `codeph` and `codeblock` attribute sets from `filepath` `lot-attr-path` keyword. Keyword definition 'ot-attr-path' not found.

# Summary: what we're going to do

- Include (per parameter):
  - Status: All | Only “Authoring:review” (if all, include extra info in yellow highlighted block below title)
  - Reviewer: All | Only selected
  - oXygen comments: Yes | No
  - <draft-comment>: Yes | No (with nicer formatting)
- Always include
  - CMS file name
  - Due date and review comments (if any)



# Output Generator user parameters



# Output (1)

## Format code samples and messages

Filename: dvp1508514661744.xml | Status: Authoring:review | Due Date: 2017-11-03  
Reviewers: Boris Aguilar [boris.aguilar@ixiasoft.com], Nolwenn Kerzreho [nolwenn.kerzreho@ixiasoft.com], Keith Schengili-Roberts [keith.roberts@ixiasoft.com], Jean-François Ameye [jean-francois.ameye@ixiasoft.com]  
CMS comment to Reviewers: Per conversation with Leigh; she could be completely wrong  
Determine the appearance of text in the `<codeph>`, `<codeblock>`, `<msgph>`, and `<msgblock>` elements.

ID, status, due date  
assigned reviewers,  
review comment

You probably use the `<codeph>` and `<codeblock>` elements for code samples. Likewise, you probably use `<msgph>` and `<msgblock>` elements for system messages. You use two different domain

*Disposition: / Status:*  
What's a domain?

draft-comment

attribute set files to format these elements.

1. Copy the `codeph` and `codeblock` attribute sets from `DITA-OT/plugins/org.dita.pdf2/cfg/fo/attrs/pr-domain-attr.xml` to your copy of `pr-domain-attr.xml`.
2. Add the following attribute to both attribute sets.

```
<xsl:attribute name="font-size">9pt</xsl:attribute>
```

You also want `<codeblock>` to have a light gray background

# Output (2)

## Format the chapter, appendix, or part autonumber

Filename: ffx1508514660811.xml | Status: Authoring:review

Reviewers: Keith Schengili-Roberts [keith.roberts@ixiasoft.com]

Determine the appearance of the number generated for chapter, appendix, or part titles.

By default, the PDF plugin generates a two-part autonumber for chapter, appendix, and part titles. The autonumber consists of a label (the words Chapter, Appendix, Part, or any alternatives you choose) and the **[oXygen comment: "Keith, could I get an explanation of how the numbering function works?"]**number. Each part has its own attribute set.

oXygen comment\*

1. In `DITA-OT/plugins/org.dita.pdf2/cfg/fo/attrs/ static-content-attr.xml`, find these two attribute sets:
  - `__chapter__frontmatter__name__container`: formats the label and the number if no separate formatting is specified for the number
  - `__chapter__frontmatter__number__container`: formats the number



**Note:** The same two attribute sets also apply to appendix and part autonumbering.

In your PDF, the chapter title is 20pt Trebuchet regular black. You also want the autonumber to be Trebuchet regular black, but you want it to be 16pt.

2. Copy them into your `static-content-attr.xml`.
3. In `__chapter__frontmatter__name__container`, add or edit the attributes as follows:

```
<xsl:attribute name="font-size">16pt</xsl:attribute>
```

\*oXygen's `com.oxygenxml.pdf.css` plugin has much nicer formatting for all the oXygen Track Changes PIs

# What I changed

- commons.xml
  - `<xsl:template match="*" mode="commonTopicProcessing">`
- commons-attr.xml
  - `<draft-comment>` formatting
- basic-settings.xml
  - Added parameters
- REALLY easy!
  - You could add this stuff to your own PDF plugin