PDF generation with CSS

Presenter:

Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu

Problems with XSL-FO PDF generation

You need:

- Good XSLT knowledge to make customizations

- Know how to create a plugin or a customization folder for the DITA OT

- Where to find various parameters and templates which need to be overwritten

- XSL-FO knowledge to make customizations

- Compatibility problems with various DITA OT versions

Our particular problem with XSL-FO PDF generation

- Lots of customization questions from our users

- For each question we lose a lot of time hunting down particular changes which should be made

- Some users just give up either the publishing or the editing solution

CSS-based PDF publishing

Advantages:

- Lots more end users are comfortable with CSS

- Use CSS both for XHTML and PDF styling

- CSS is much easier to learn than XSL-FO

- Majority of users do not need completely different styling for the Web or print outputs

- Consistent editing/publishing solution
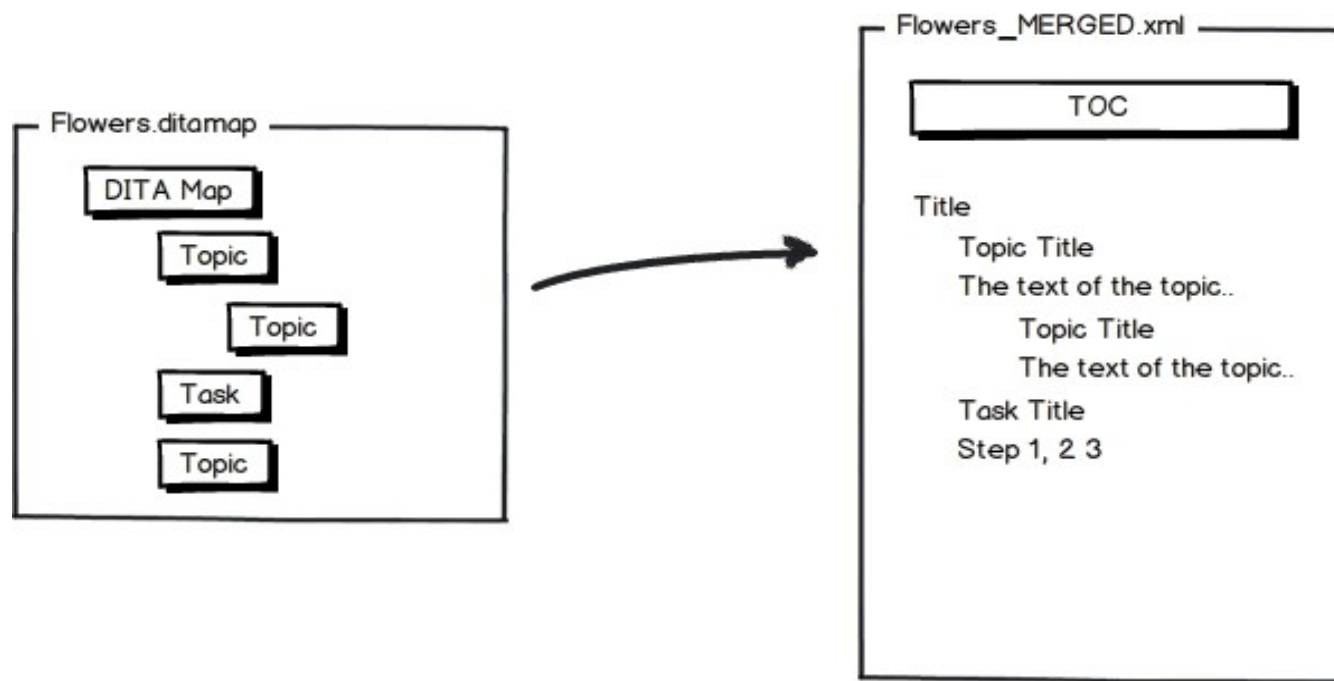
CSS-based PDF publishing

Disadvantages:

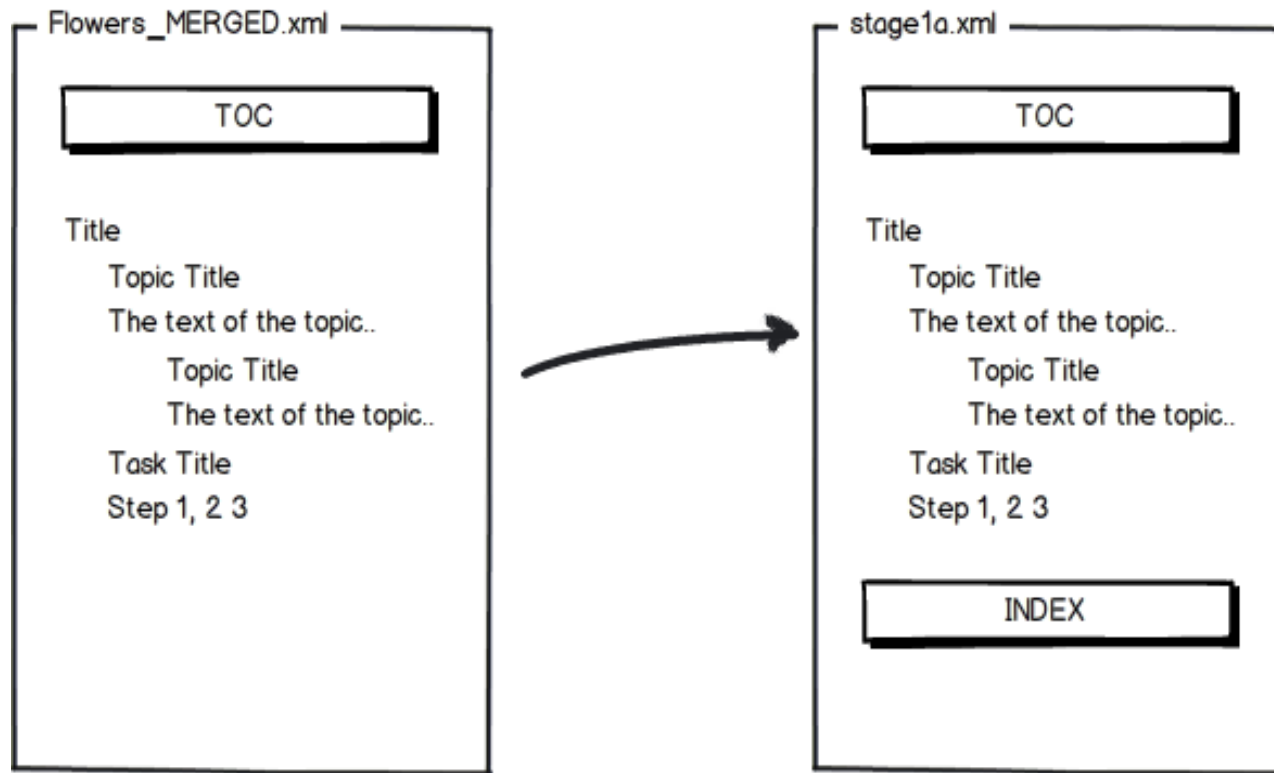- Only commercial engines like Antenna House and Prince XML are available

Steps of current DITA OT PDF publishing solution

1. Creates a single file containing the resolved, or **merged** DITA map with all its topics

2. Creates a file derived from the "merged" DITA map file, containing also the indexterms

3. Apply XSLT processing to get the final FO file and uses PDF processor to convert the FO file into PDF.

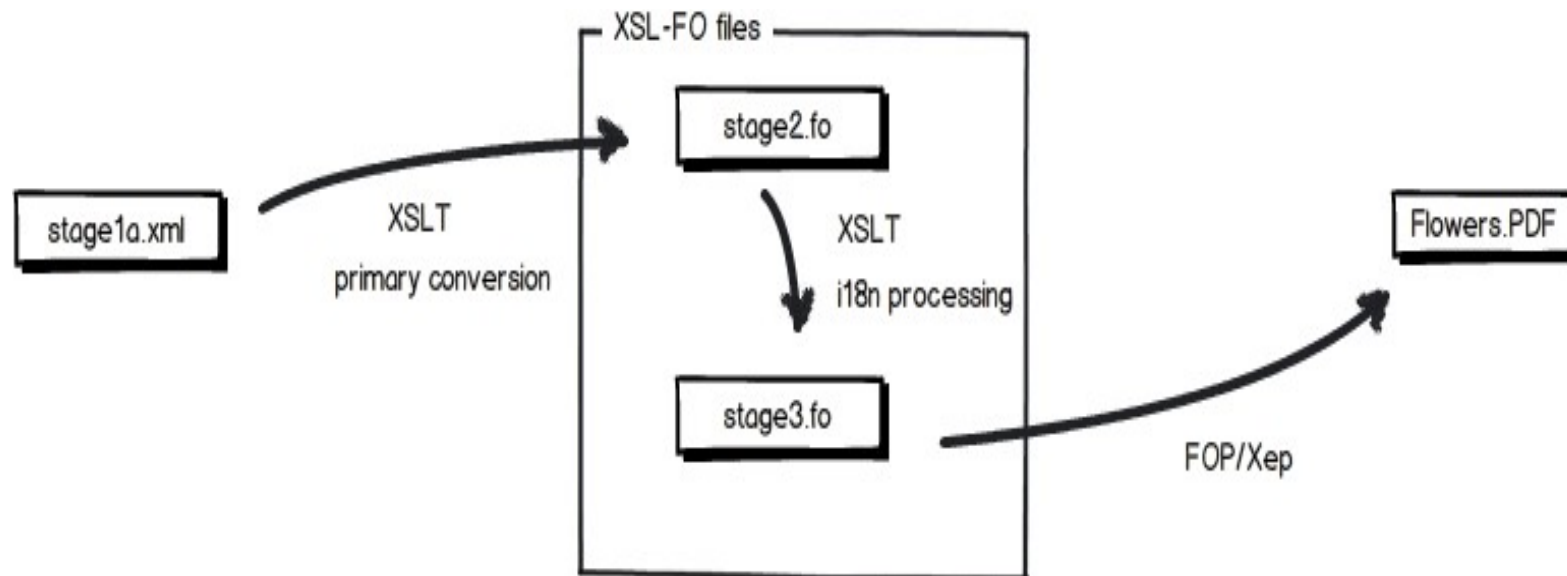# 1. Create a single file containing the resolved, or "merged" DITA map with all its topics

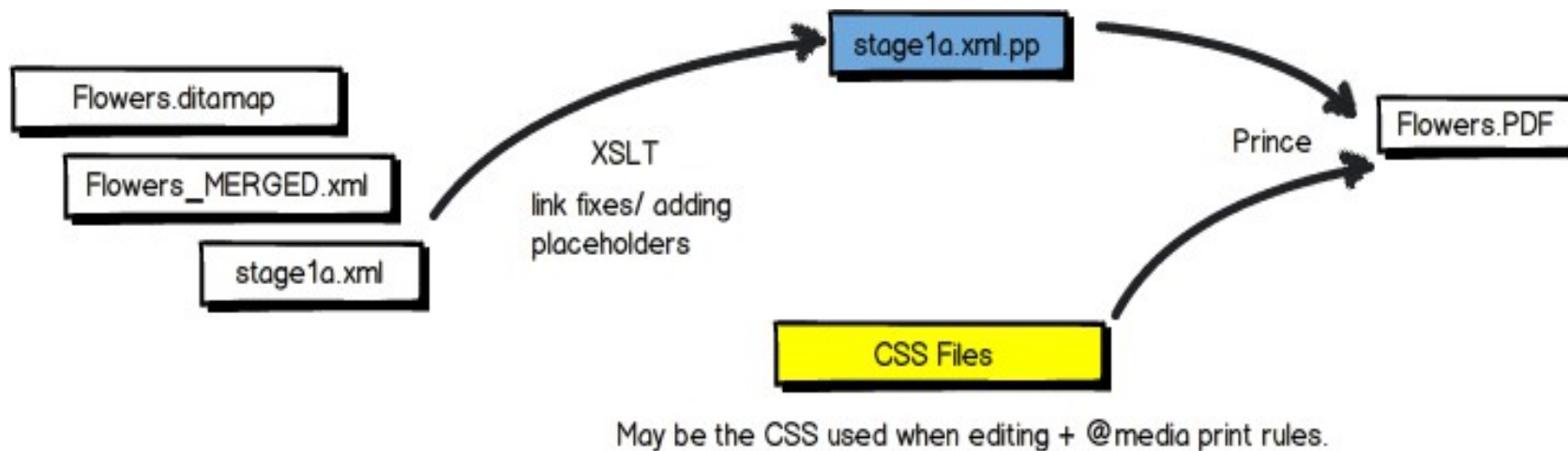# 2. Create a file derived from the "merged" DITA map file, containing also the indexterms

3. Apply XSLT processing to get the final FO file

Steps of the CSS PDF generator plugin

- Duplicate and re-use first two steps from the existing PDF publishing

- Extra XSLT stage

- Add CSSs to the merged XML output and process

XSLT stage

TOC processing:

- Mark every topic reference to a chapter

- Cleaning the TOC, keeping only the navtitles.

- Creates an entry in the TOC pointing to the ID of the index section.

XSLT stage

Content processing:

- Mark each chapter

- If there are relative image references, makes them absolute

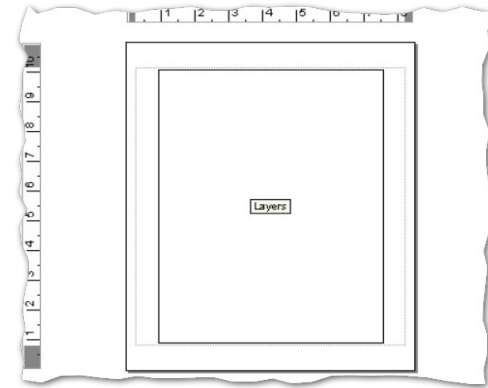- Convert the comments and change tracking processing instructions to elements

XSLT stage

Index preparation:

- Generates an ID to each of the indexterm from the content.

- Makes a reference to the generated ID from the index structure.
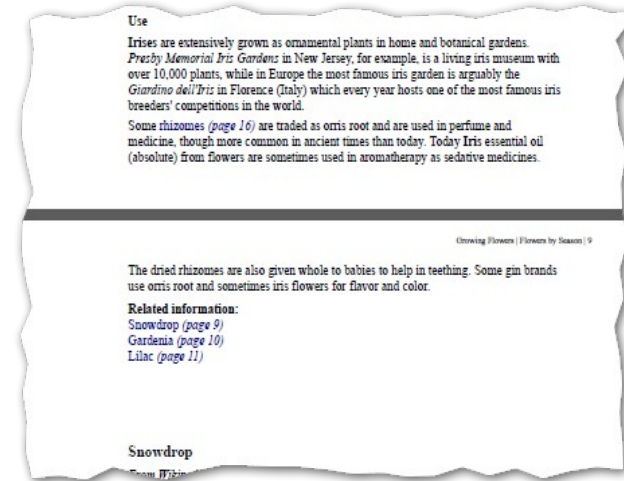
CSS styling examples (1)

Setting page margins and size:

```
@page{
    padding-top:0.2em;
    padding-bottom:0.2em;
    size: 8.27in 11.69in;
    margin: 0.75in 0.25in 0.75in 1in;
}
```

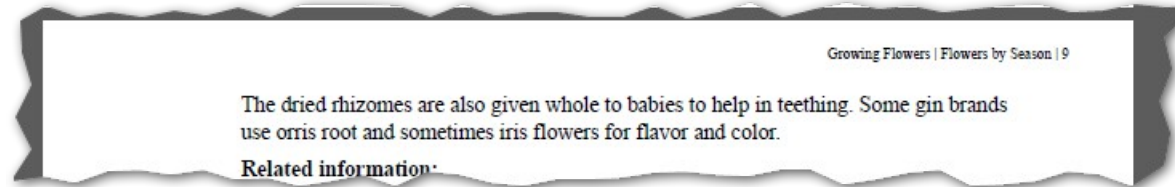# CSS styling examples (2)

Page breaks:


```
*[class ~= "topic/topic"][is-chapter = "true"]{
  page-break-before:always;
}
```

CSS styling examples (3)

Page numbers:

Growing Flowers | Flowers by Season | 9

The dried rhizomes are also given whole to babies to help in teething. Some gin brands use orris root and sometimes iris flowers for flavor and color.
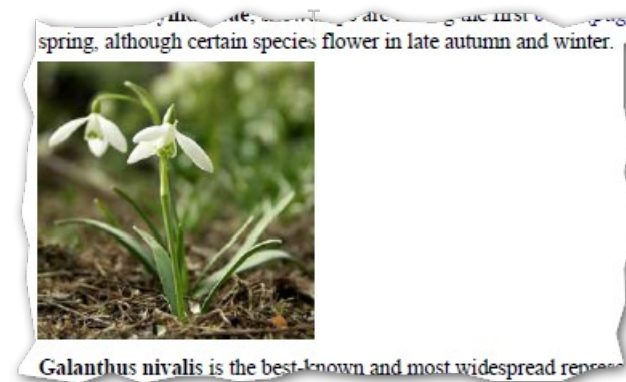
Related information:

```css
@page :left{
   @top-left {
     content: string(maptitle) " | " string(chaptertitle) " | "
counter(page);
     font-size:8pt;
   }
 }

@page :right{
   @top-right {
     content: string(maptitle) " | " string(chaptertitle) " | "
counter(page);
     font-size:8pt;
   }
}
```
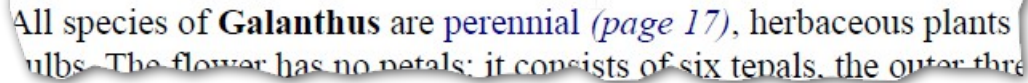
CSS styling examples (4)

Setting the image resolution:


```
@media print{
  *[class ~= "map/map"] {
     prince-image-resolution: 196dpi;
  }
}
```
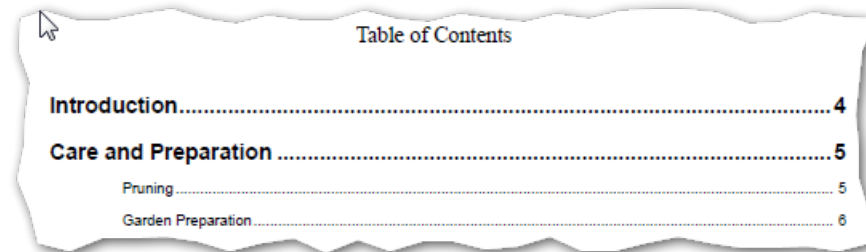
CSS styling examples (5)

Links:

All species of **Galanthus** are perennial *(page 17)*, herbaceous plants bulbs. The flower has no petals; it consists of six tepals, the outer three

```css
*[class ~= "topic/xref"][href]:after {
    content: " [page " target-counter(attr(href), page) "]";
    link: attr(href);
}
```

CSS styling examples (6)

Table of contents:

```
*[class ~= "map/map"] > map {
  display: block;
  page: table-of-contents;
  counter-reset: page 1;
}
```

Publishing Capabilities

- Publish with Prince XML.

- Publish with Antenna House Formatter.

Availability

https://github.com/oxygenxml/dita-css

Future plans

- Try to establish a community to share the work load

- Bring the new plugin's output quality to the same level of compliance with the current PDF generation plugin

- Write some decent documentation for it

Thank You!

Questions?

Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu