Seamless Content Management with OpenOffice and Cocoon

Christian Egli

egli@apache.org

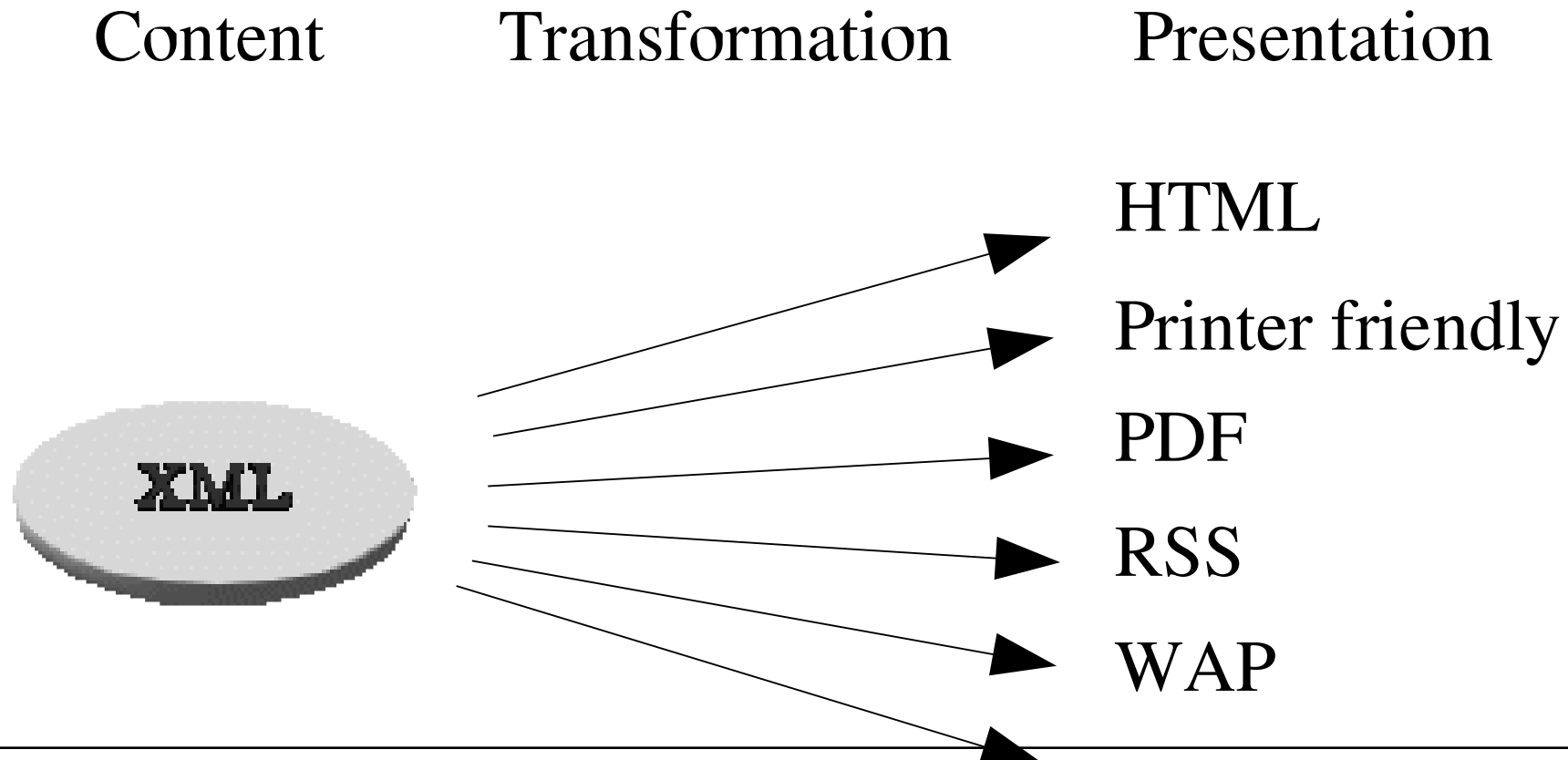# Outline

○ Multi-Channel publishing today

○ XML, XSLT as enabling technology

  ▷ Apache Cocoon

○ Content Creation

  ▷ OpenOffice.org

○ Is there more to Content Management?

○ Conclusion

○ Question & Answers

Multi-Channel publishing today

○ Content mixed with presentation

  ▷ HTML, Word, LaTeX

○ Publish multiple formats from single source

  ▷ HTML, PDF, Printer friendly, RSS

○ Does not scale

Separate content from presentation

○ Single source

○ Multiple presentation descriptions for each output format

Content   Transformation   Presentation

XML

HTML

Printer friendly

PDF

RSS

WAP

Enabling Technology

○ XML (eXtensible Markup Language)

▷ universal, presentation-free content format

▷ perfect for media-independent publishing and data interchange

▷ lingua franca for office tools such as OpenOffice.org

○ XSLT

Enabling Technology

○ XML

○ XSLT (Extensible Stylesheet Language Transformations)

  ▷ language for transforming XML documents
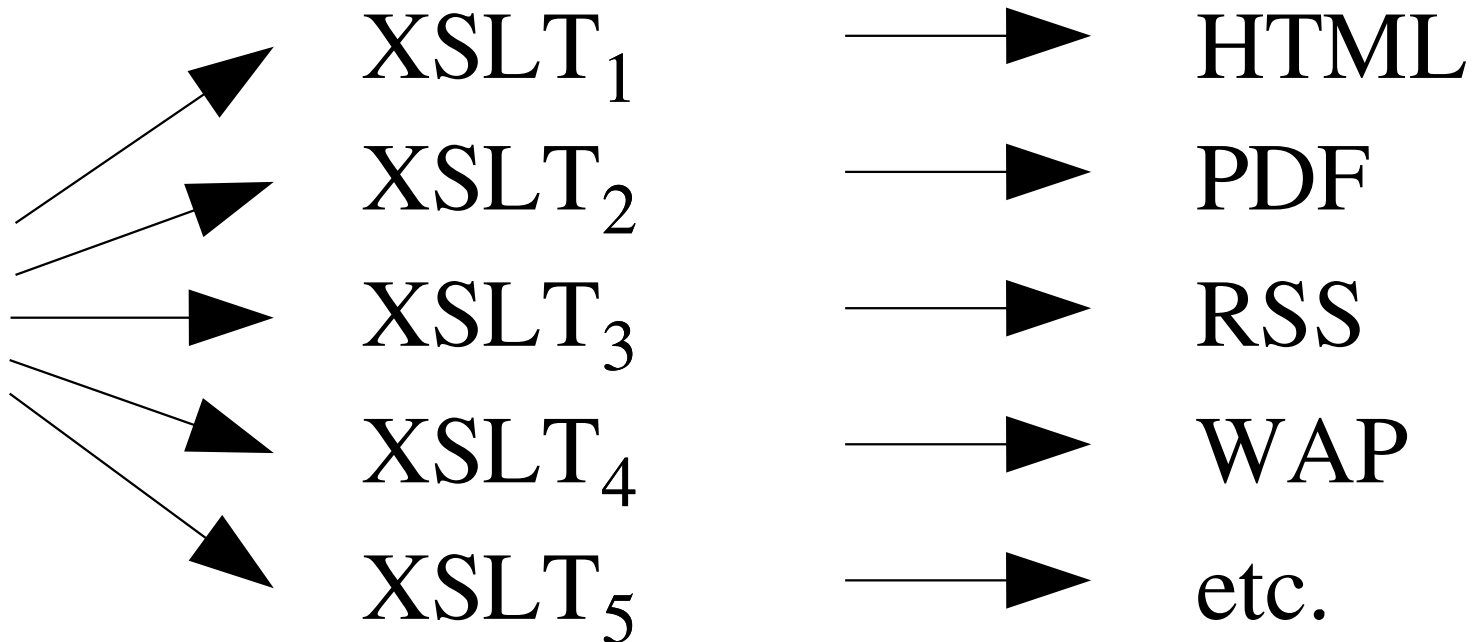
  ▷ transformation of one XML format into another

Enabling Technology

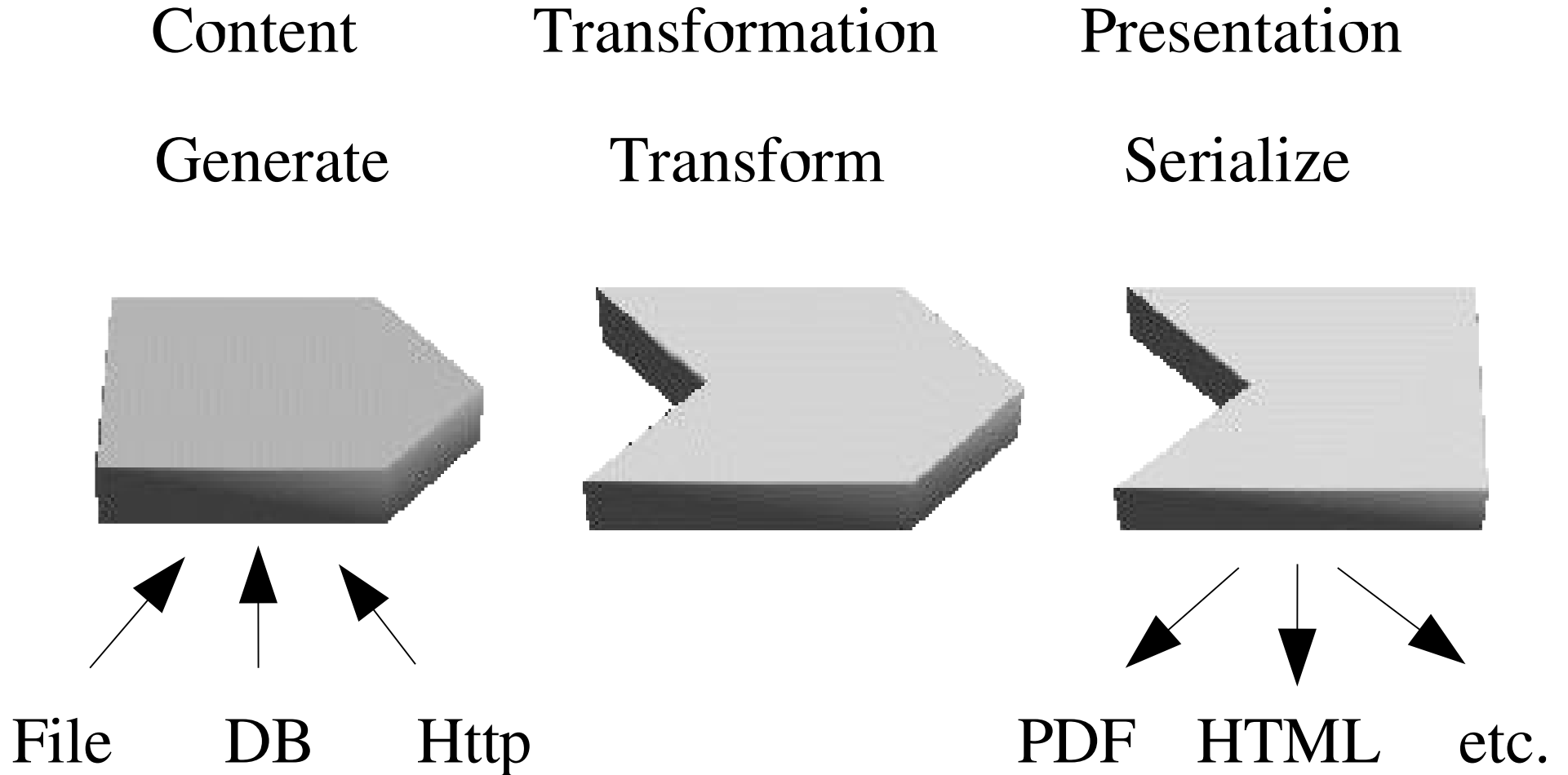| Content	Transformation	Presentation
	$XSLT_1$ →	HTML
	$XSLT_2$ →	PDF
XML →	$XSLT_3$ →	RSS
	$XSLT_4$ →	WAP
	$XSLT_5$ →	etc.

Apache Cocoon

○ highly flexible open-source web publishing framework

○ based around XML and XSLT

○ fundamental concept of the XML pipeline

XML pipeline

Content
Transformation
Presentation

Generate
Transform
Serialize

File   DB   Http

PDF   HTML   etc.

Cocoon XML Pipeline

```
<map:match pattern="hello.html">
  <map:generate src="content/hello.xml"/>
  <map:transform src="style/xsl/page2html.xsl"/>
  <map:serialize type="html"/>
</map:match>
```
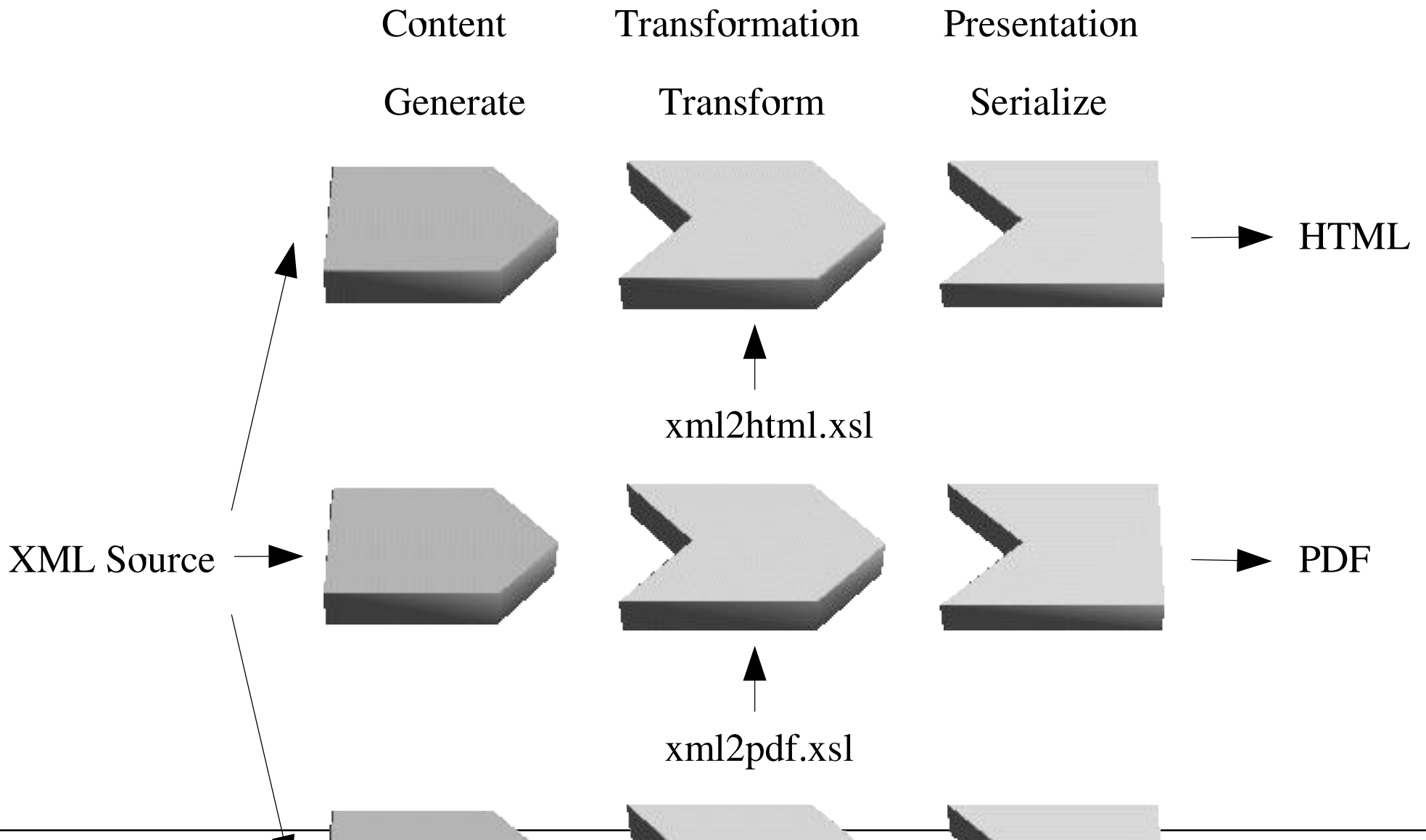
Cocoon Multi-Channel publishing

Content Transformation Presentation

Generate Transform Serialize

HTML

xml2html.xsl

XML Source

PDF

xml2pdf.xsl

Multi-Channel publishing Pipeline

```
<map:match pattern="hello.xml">

    <map:generate src="content/hello.xml"/>

    <map:serialize type="xml"/>

</map:match>


<map:match pattern="hello.html">

    <map:generate src="content/hello.xml"/>

    <map:transform src="style/xsl/page2html.xsl"/>

    <map:serialize type="html"/>

</map:match>
```

Multi-Channel publishing with Cocoon

○ Demo

▷ One source XML file

▷ Pipelines in sitemap

▷ Multiple output formats

Content creation

○ How to create the XML content?

▷ human-readable

▷ With Emacs, with vi :-(

○ "Office Application"

▷ WYSIWYG

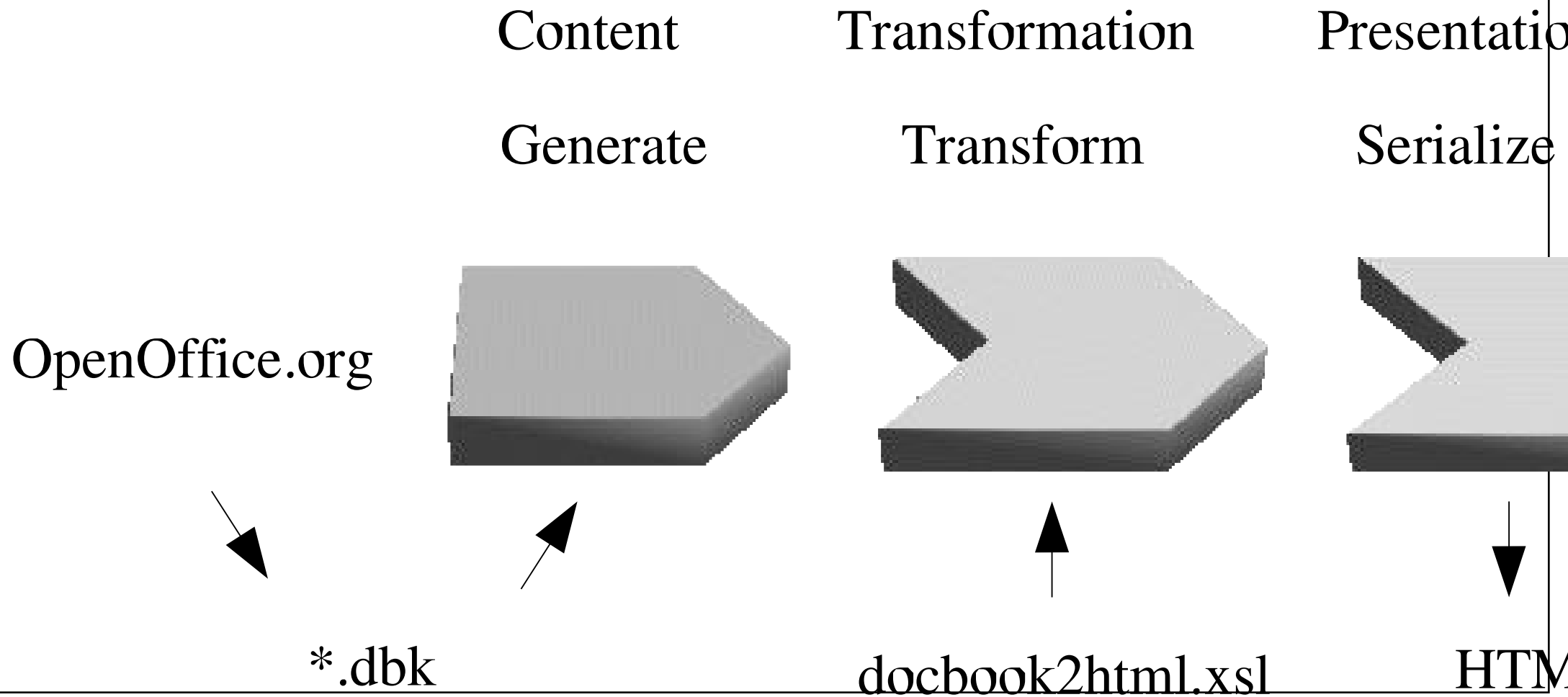▷ seamlessly integrated in the desktop

○ OpenOffice.org to the rescue

OpenOffice.org

- a free multi-platform office productivity suite

- includes the key desktop applications

  ▷ word processor, spreadsheet, presentation manager, and drawing program

  ▷ user interface and feature set similar to other office suites'
- XML as its native file format
- excellent DocBook support

Apache Cocoon + OpenOffice.org

○ OpenOffice.org for content creation

○ Cocoon for content publication

○ DocBook as the interface glue

▷ XML dialect well suited for documentation
▷ widely used and supported

Apache Cocoon + OpenOffice.org

Content

Transformation

Presentatio

Generate

Transform

Serialize

OpenOffice.org

*.dbk

docbook2html.xsl

HTM

Apache Cocoon + OpenOffice.org

○ Demo

   ▷ Edit docbook with OpenOffice.org

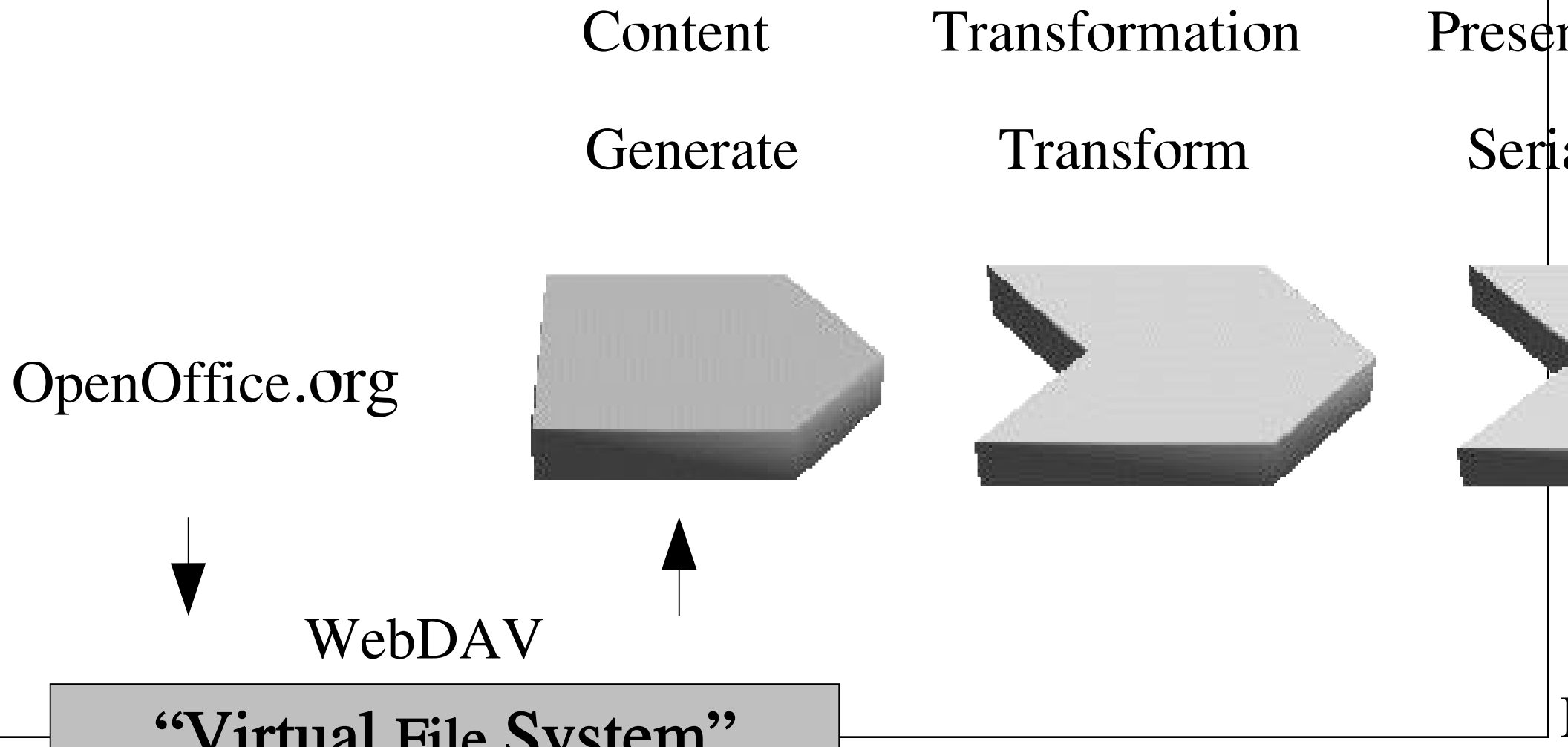   ▷ Publish docbook with Apache Cocoon

Problems

○ WYSIWYG - Really?

○ OpenOffice.org Styles

○ DokBook and Cocoon

WebDAV

○ Web-based Distributed Authoring and Versioning

○ edit and manage files on remote web servers collaboratively

WebDAV

Content

Transformation

Preser

Generate

Transform

Seri

OpenOffice.org

WebDAV

"Virtual File System"

Is this Content Management?

○ Solution for content creation and delivery

▷ WYSIWYG

▷ seamlessly integrated in the desktop

○ Lacks support for

▷ workflow

▷ access control

▷ versioning (e.g. scheduling, expiration)

▷ personalization

▷ localization

Is this Content Management?

○ WebDAV has or will have

  ▷ access control

  ▷ versioning (partially)

○ Apache Cocoon has

  ▷ personalization

  ▷ localization

○ still missing

  ▷ workflow

  ▷ versioning (e.g. scheduling, expiration)

Choices

○ Use a "real CMS"

▷ Cocoon-based solution (leverage XML publishing strengths)

▷ No full-fledged Cocoon-based solution

○ Wait for WebDAV

▷ promises to standardize and implement versioning, configuration management and access control

○ Use one of up and coming contenders

▷ Apache Lenya (incubating at Apache Cocoon)

▷ Linotype

Conclusion

○ OpenOffice.org + Cocoon

  ▷ Simple + Powerful Solution for content creation and delivery

  ▷ Lacks some CMS features

○ More work needed (see Problems)

○ Other Choices

References

- Apache Cocoon: http://cocoon.apache.org

- OSCOM: http://www.oscom.org

- OpenOffice.org: http://www.openoffice.org

- W3C: http://www.w3.org

- WebDAV: http://www.webdav.org

- XML: http://www.xml.com/pub/a/98/10/guide1.html

- XSLT: http://www.w3.org/Style/XSL

- OO.org + Docbook:
  http://www.zzoss.com/projects/oowdbk