Practical workflow
for
technical documentation

Dag Wieërs - dag@wieers.com

Goals

From manageable sources to professionally styled documents from the command line

Which basically means ?

- Sources need to be clear text
- Output to various formats (PDF, DOC, HTML)
- Easy to modify style/output (for non-developers)
- Applying styles in the process
- Instruct process from the command line/Makefile
- Make writing documentation **as easy as possible**

Text source format

- No shortage in options
  - Markup languages
    - DocBook / XML, X-HTML, SGML
  - Light-weight markup languages
    - ReStructuredText, Markdown, AsciiDoc, Wiki, ...
- Not everything fits common needs
  - Focus on content, simple to write
  - Easy to version (eg. Git or subversion)
  - Support for advanced “book” features: e.g. inline comments, footnotes, admonitions, positioning, complex nesting, indexes, cross references, ...

AsciiDoc as source format

- Light-weight markup language
  - Simple syntax that reads like ascii text
- Syntax feels natural, even in source format
- Maps 1:1 with “Simplified DocBook”
  - Same functionality as used for e.g. O'Reilly books
- Can be converted to various formats
  - X-HTML, DocBook, HTML, LaTeX, Slidy, Wordpress
- Pluggable config, back-ends, themes and filters
  - e.g. code-highlighting, mathml
  - Lots of filters offer interesting possibilities

AsciiDoc demonstration

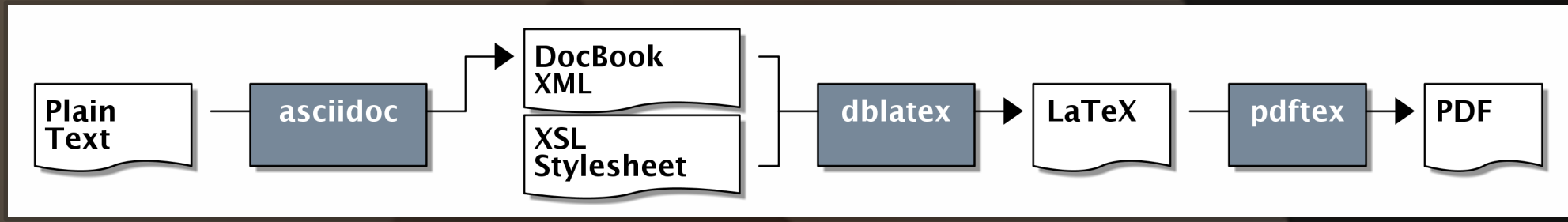
- `vim` *`file.txt`*
  - *show AsciiDoc source*
- `asciidoc` *`file.txt`*
  - *produces file.html*
- `asciidoc -b docbook` *`file.txt`*
  - *produces file.xml*
- `asciidoc -b html -a icons -a toc` *`file.txt`*
  - *produces file.html with icons and table-of-contents*

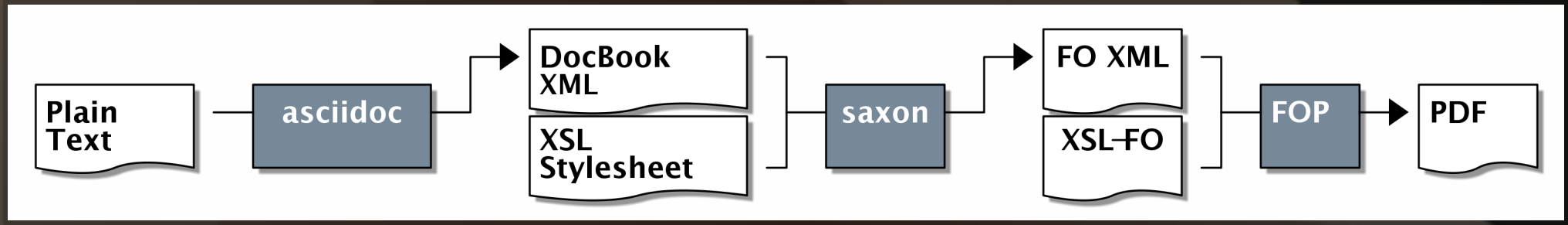
Creating PDF output

- dblatex / pdflatex
  - From DocBook XML to LaTeX to PDF
- FOP
  - From DocBook XML to XSL-FO to PDF
- Firefox
  - From HTML to PDF
- LibreOffice
  - From various formats to PDF

DocBook toolchains

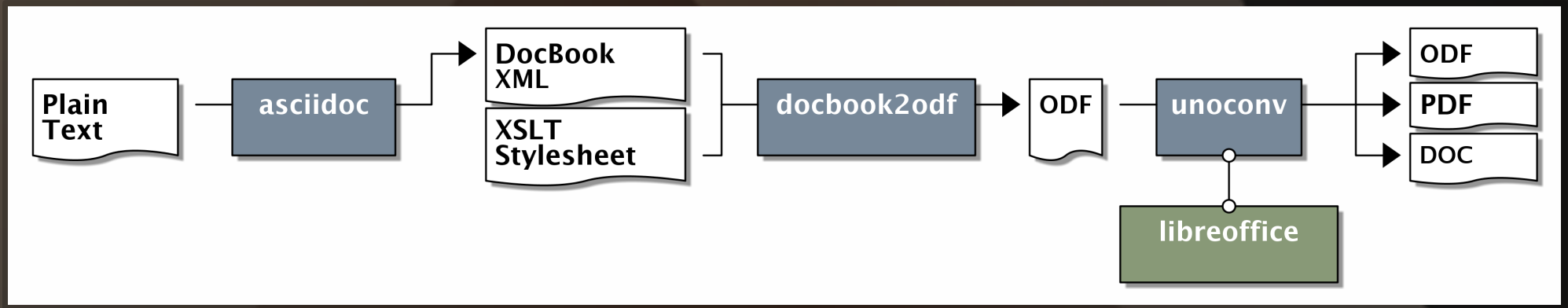
- DocBook output needs additional processing
  - dblatex or FOP requires XSL-FO/XSLT skills
  - XSL-FO and XSLT is programming in XML (ugh)
- If people need to be retrained: FAIL





docbook2odf toolchain

- docbook2odf converts to ODF using XSLT
  - Incomplete implementation
  - Hard to find skilled XSLT people to help
- Use unoconv to produce PDF, DOC, …
  - Apply ODF style during conversion



unoconv tool

Command line tool to do non-interactive (batch) conversions of documents using LibreOffice import and export plugins.

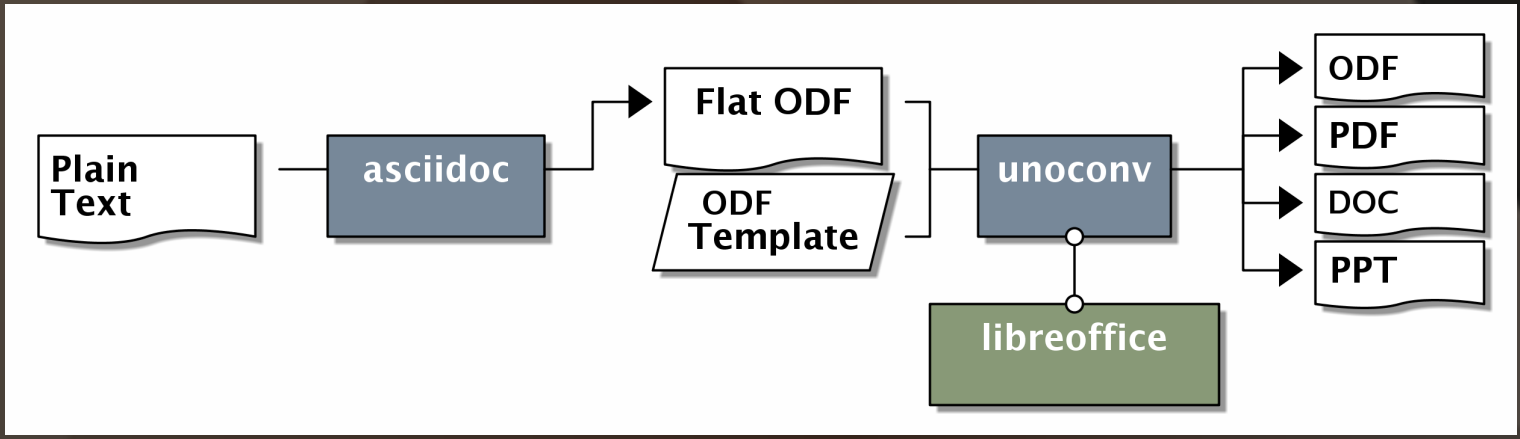
- So in essence, you can use unoconv in Makefiles and scripts to automatically convert files as part of the build-process of your project
- Or use it when doing migrations from MS Office to LibreOffice
- Or as a back-end tool for a conversion application or service

unoconv features

- Supports all import and export filters (+100 formats)
  - Influencing the conversion process through options
- Styles applied during conversion (corporate identity)
  - Either by providing another document or template
- LibreOffice instance is managed by unoconv
  - Using an existing instance or starting a new one
  - Starting a listener on the network is possible too
- Works on Linux, MacOS X and Windows

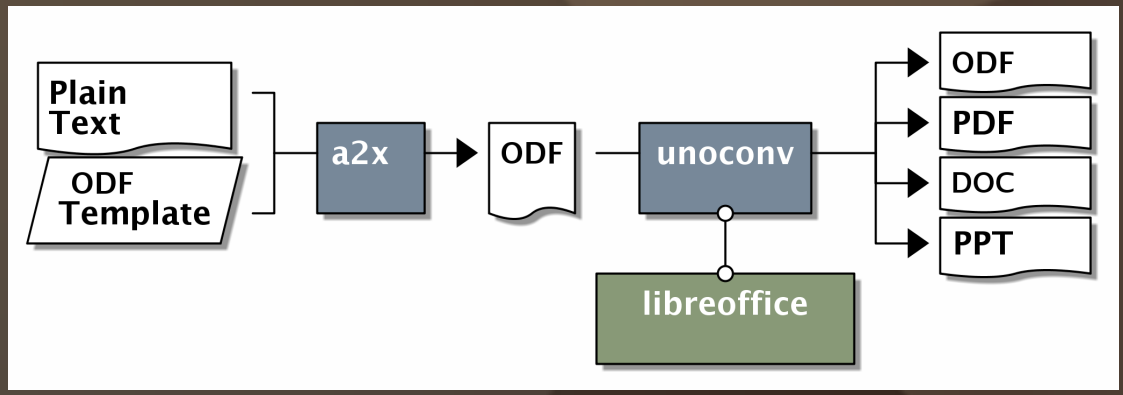
ODF back-end for AsciiDoc

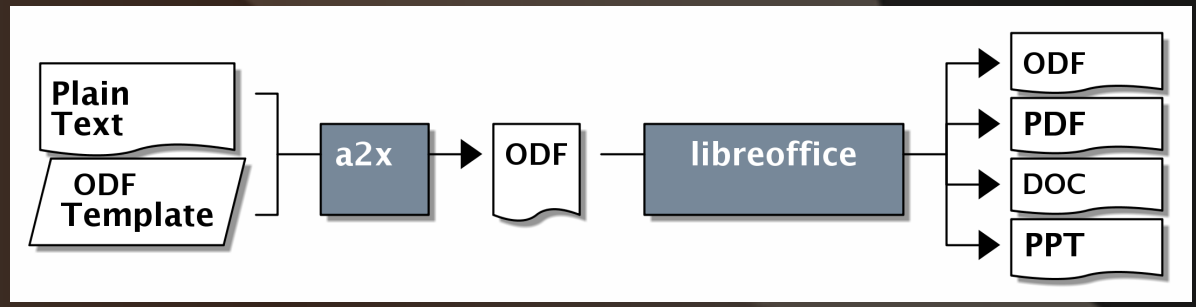
- Directly convert AsciiDoc to ODF
- Apply *corporate identity*
  - Use an ODF *theme* or a styled ODF (LibreOffice)
- LibreOffice can be used to export to PDF, DOC, …
  - Use unoconv for command line conversion
  - Use soffice.bin –convert-to pdf *file.odf*



But we're not done yet...

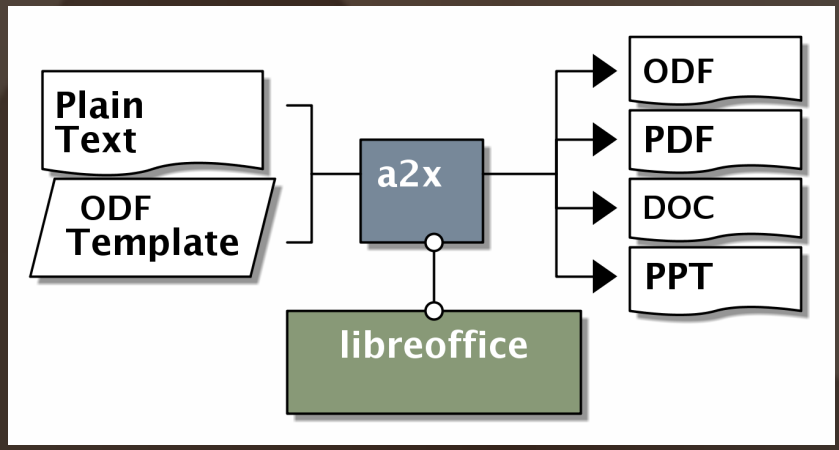
- Integrating into a2x
  - outputs ODF directly
- Improve LibreOffice
  - allow for ODF-styling
  - improve cmdline
  - stabilize
  - get rid of unoconv





Dreams and aspirations

- Creating full-featured ODP files from AsciiDoc
  - With support for custom slides, transitions, etc...
- Filter to create ODG/SVG files from markup/ascii-art
  - With a lot more options than ditaa
- LibreOffice integration into a2x directly



asciidoc-odf project

- Started only 3 months ago
- Implements the basic DocBook constructs
  - Lacks complex tables, ...
  - Default stylesheet is not 100% complete
  - Only basic ODP (presentation) support
- Various limitations to overcome
  - ODP vs other AsciiDoc slide backends
  - ODF vs DocBook constructs

asciidoc-odf demonstration

- `asciidoc -b odt` *`file.txt`*
  - *produces file.fodt*

- `a2x -f odt` *`file.txt`*
  - *produces file.odt*

Links

- AsciiDoc
  - http://www.methods.co.nz/asciidoc/
- asciidoc-odf
  - http://github.com/dagwieers/asciidoc-odf
- unoconv:
  - http://github.com/dagwieers/unoconv
- Libreoffice:
  - http://libreoffice.org/

Thank you for listening

Any questions, ideas, pull-requests ?