Collabora Productivity

Editing ReqIF-XHTML fragments with Writer

By Miklos Vajna

Software Engineer at Collabora Productivity

2018-09-28

About Miklos

From Hungary

- More details: https://vmiklos.hu/

Google Summer of Code 2010 / 2011

- Rewrite of the Writer RTF import/export

Then a full-time LibreOffice developer for SUSE

Now a contractor at Collabora

Editing ReqIF-XHTML fragments with Writer

Motivation

Requirements Interchange Format

- (Zipped) XML file format
- Can be used to exchange requirements along with associated metadata
- Values can be XHTML fragments

More than XHTML

- Writer is relevant as an editor here due to e.g. embedded objects
- Those objects are frequently Office documents
- Best is Writer / LibreOffice handles everything

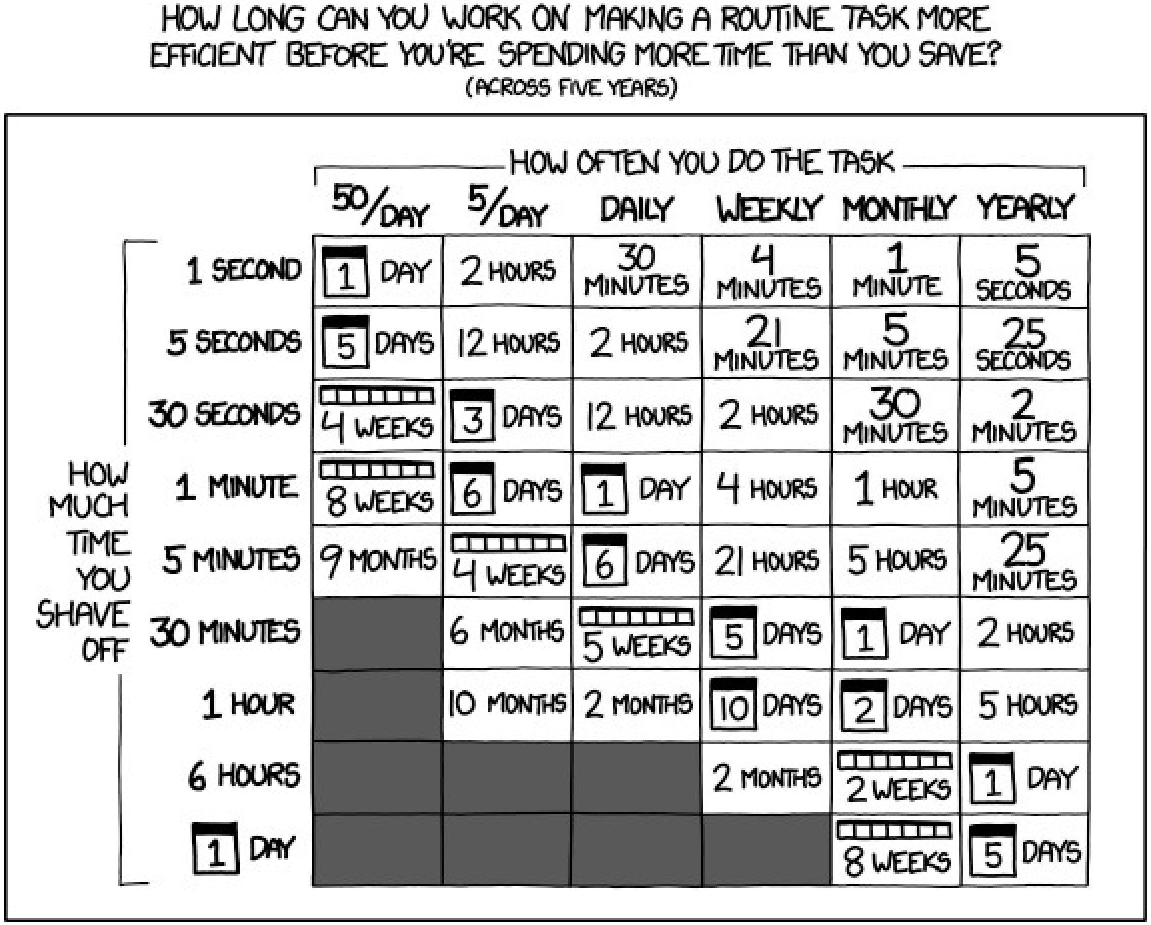
But we already have an XHTML export

A dreaded XSLT-based one

- Hard to change anything
- No random access to the document model
- No import
- Slow

We have a first-class HTML filter already

- Can’t we use that instead?



HOW LONG CAN YOU WORK ON MAKING A ROUTINE TASK MORE EFFICIENT BEFORE YOU'RE SPENDING MORE TIME THAN YOU SAVE?
(ACROSS FIVE YEARS)

		HOW OFTEN YOU DO THE TASK					
		50/DAY	5/DAY	DAILY	WEEKLY	MONTHLY	YEARLY
HOW MUCH TIME YOU SHAVE OFF	1 SECOND	1 DAY	2 HOURS	30 MINUTES	4 MINUTES	1 MINUTE	5 SECONDS
	5 SECONDS	5 DAYS	12 HOURS	2 HOURS	21 MINUTES	5 MINUTES	25 SECONDS
	30 SECONDS	4 WEEKS	3 DAYS	12 HOURS	2 HOURS	30 MINUTES	2 MINUTES
	1 MINUTE	8 WEEKS	6 DAYS	1 DAY	4 HOURS	1 HOUR	5 MINUTES
	5 MINUTES	9 MONTHS	4 WEEKS	6 DAYS	21 HOURS	5 HOURS	25 MINUTES
	30 MINUTES		6 MONTHS	5 WEEKS	5 DAYS	1 DAY	2 HOURS
	1 HOUR		10 MONTHS	2 MONTHS	10 DAYS	2 DAYS	5 HOURS
	6 HOURS				2 MONTHS	2 WEEKS	1 DAY
	1 DAY					8 WEEKS	5 DAYS

XHTML mode for the HTML filter

HTML filter in general

- Shared feature, not only in Writer
- Not only export, import as well

XHTML: XML and XML namespace

- Biggest difference is that the output has to be well-formed XML
- Also: explicit XHTML namespace: <reqif-xhtml:p>, etc.

ReqIF: inline CSS

No old-style formatting

- All formatting has to be done using CSS
- We had some support for this already

CSS has to be inline, though

- No complex CSS inheritance rules
- Inline CSS is also limited, e.g. no table border options

ReqIF: image support

By default, only PNG images are allowed

- Everything else has to be an object instead

Image objects

- JPG, GIF, SVG, etc.
- Native data is the original image data
- And a PNG replacement
- Using nested <object> XHTML markup

ReqIF: embedded object support

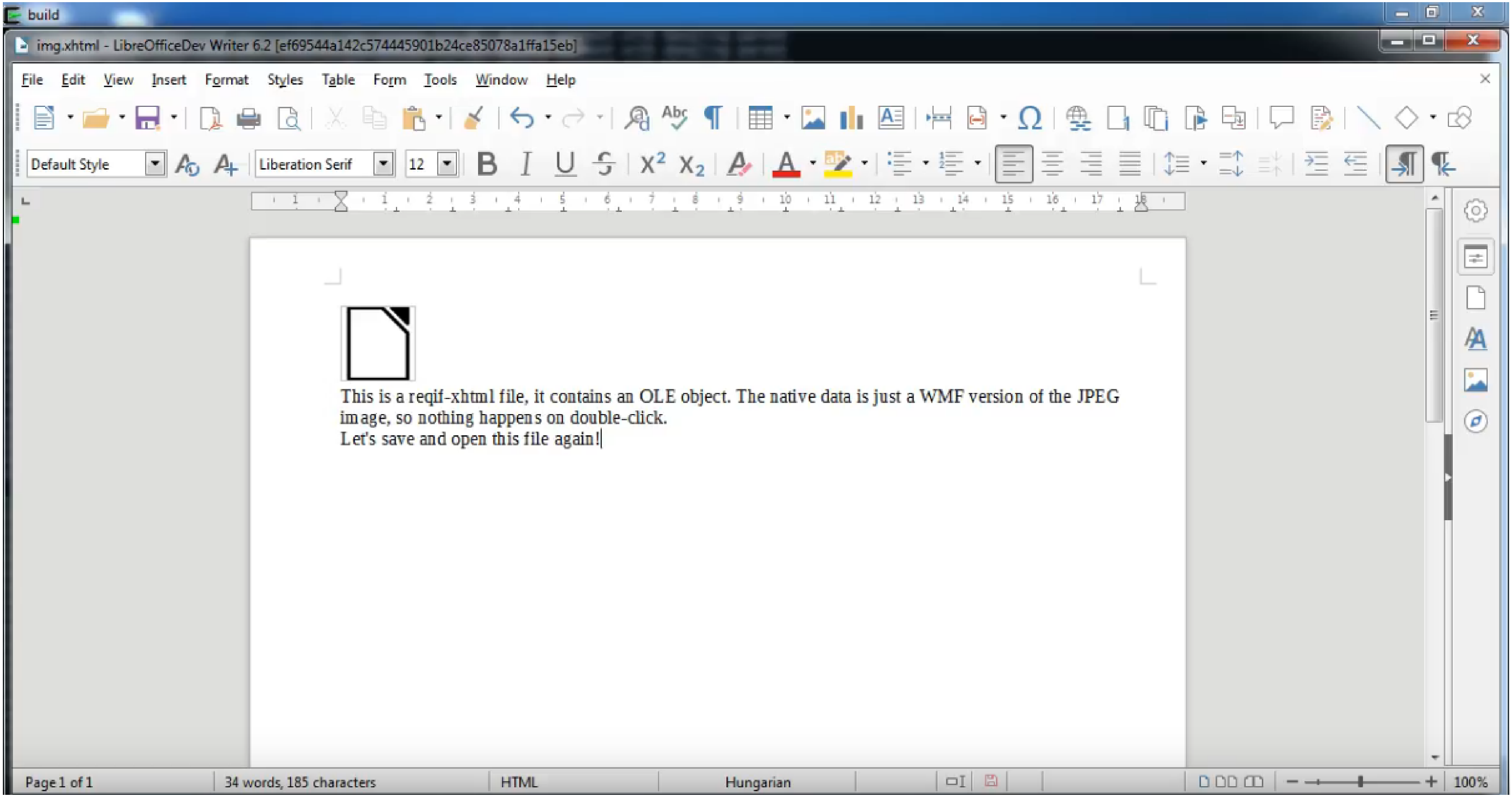
Fake objects

- The object is in fact an image

Real objects

- Either edited directly inside LibreOffice:
  - Writer, Calc, Impress
- Or edited by some external 3rd-party application
- Full wrapping/unwrapping using OLE and RTF markup

Object: image

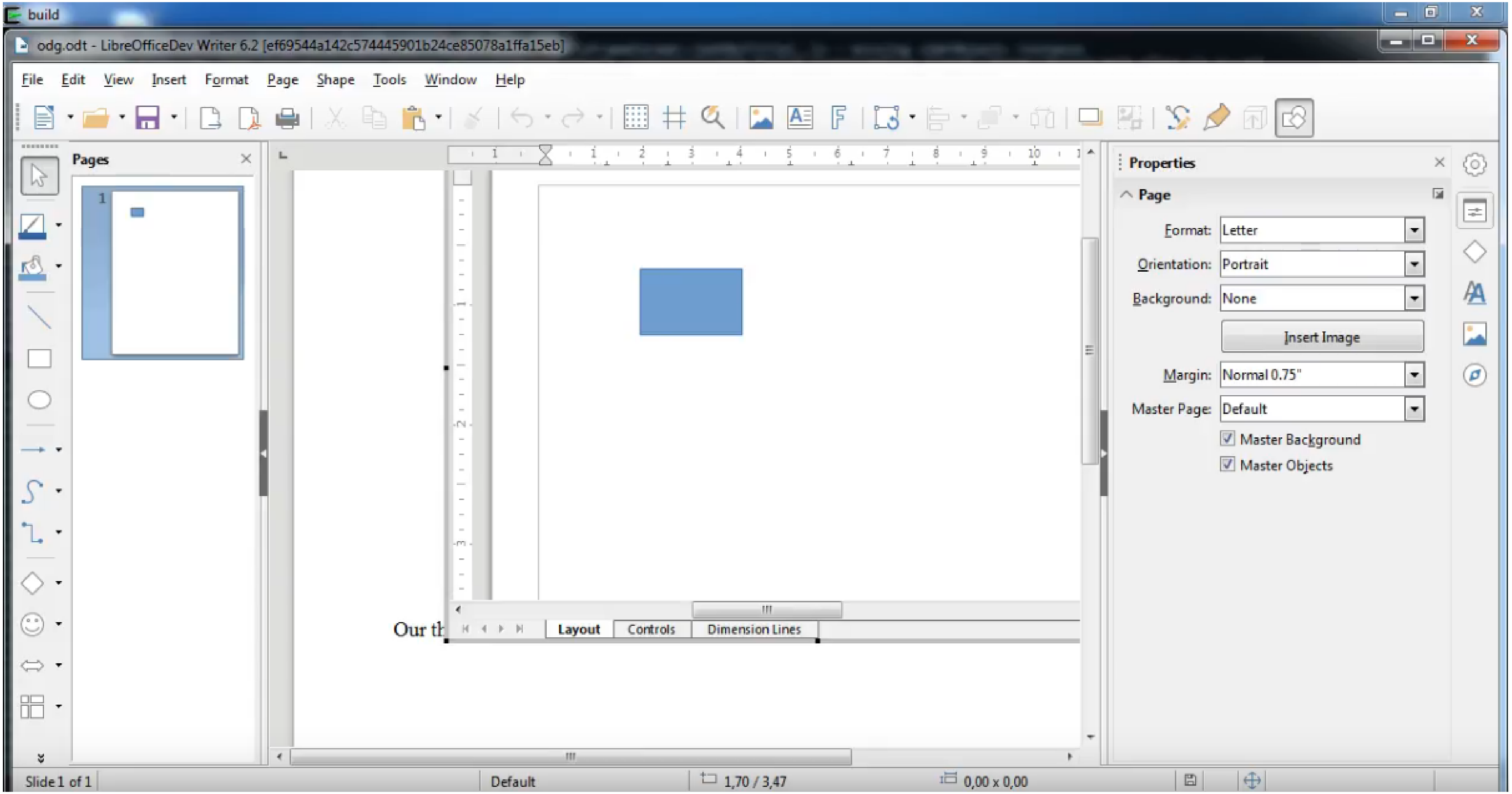

build
img.xhtml - LibreOfficeDev Writer 6.2 [ef69544a142c574445901b24ce85078a1ffa15eb]
File Edit View Insert Format Styles Table Form Tools Window Help
Default Style
Liberation Serif
12

This is a reqif-xhtml file, it contains an OLE object. The native data is just a WMF version of the JPEG image, so nothing happens on double-click.
Let's save and open this file again!

Page 1 of 1
34 words, 185 characters
HTML
Hungarian
100%

Object: external application (e.g. PPSX)

Object: own application (e.g. ODG)



odg.odt - LibreOfficeDev Writer 6.2 [ef69544a142c574445901b24ce85078a1ffa15eb]

Format: Letter

Orientation: Portrait

Background: None

Margin: Normal 0.75"

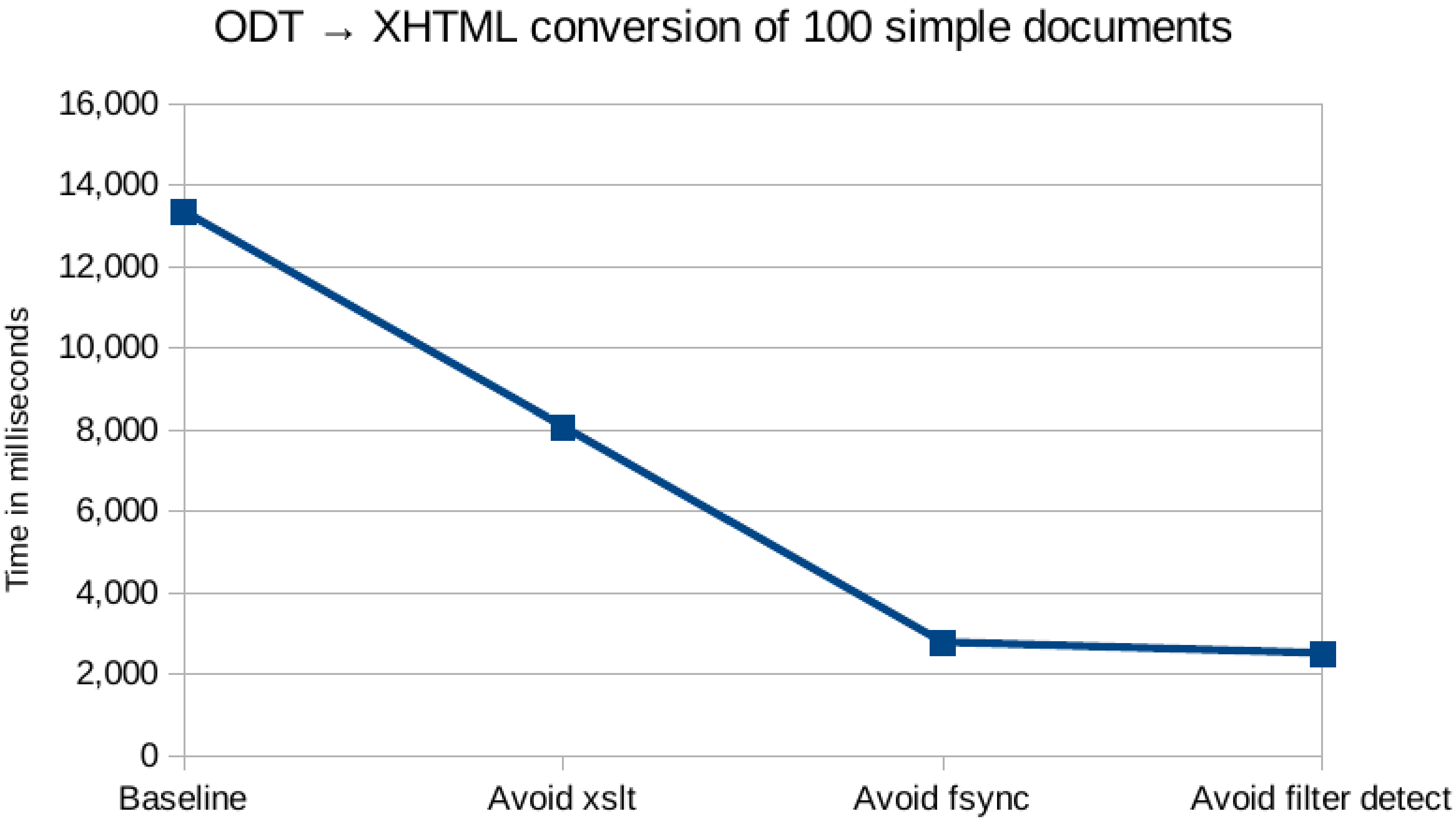
Master Page: Default

☑ Master Background

☑ Master Objects

Slide 1 of 1 | Default | 1,70 / 3,47 | 0,00 x 0,00

ReqIF: performance



Usage from UNO API

Import

- It’s your responsibility to extract the XHTML fragment from a .reqif/.reqifz file
- External entities are expected to be next to the XHTML fragment file (e.g. images)
- Set FilterName to “HTML (StarWriter)”
- Set FilterOptions to “xhtmlns=reqif-xhtml”

Export

- Same values for FilterName and FilterOptions
- No Writer/Web, no Web view

Usage from commandline

Open: explicit import filter

- --infilter=”HTML (StarWriter):xhtmlns=reqif-xhtml”
- No filter detection as these fragments don’t have a standard header

Save: explicit export filter

- --convert-to “xhtml:HTML (StarWriter):xhtmlns=reqif-xhtml”
- No UI here either, typical use-case is embedded LibreOffice anyway

How is this implemented?

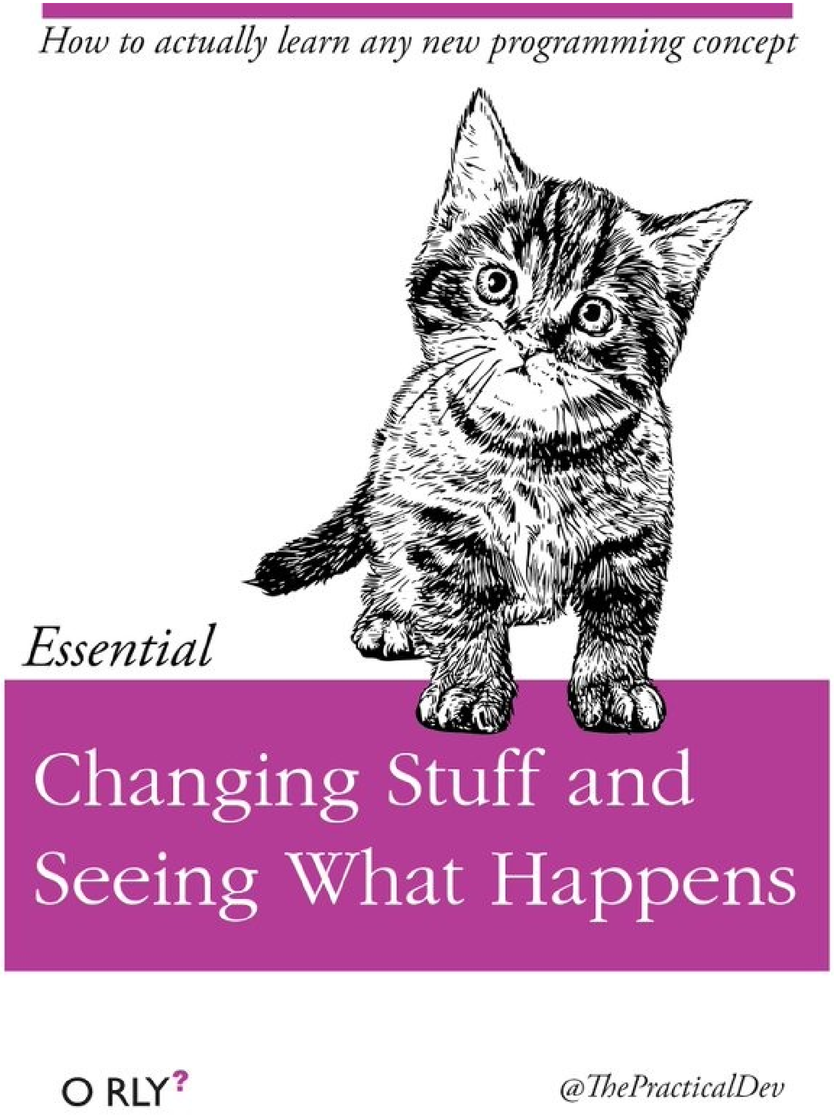
Architecture

svtools

- HTMLParser::maNamespace: expected XML namespace

sw

- SwHTMLParser::m_bXHTML
- SwHTMLParser::m_bReqIF
- SwHTMLWriter::mbXHTML
- SwHTMLWriter::mbReqIF


How to actually learn any new programming concept
Essential
Changing Stuff and Seeing What Happens
O RLY?
@ThePracticalDev

From HTML to XHTML

Parser

- PlainTextFilterDetect::detect() to accept XHTML as HTML
- The additional header: <?xml ...>
- The expected (common) HTML/XHTML header: <!DOCTYPE ...>
- Ignore the expected namespace in HTMLParser::GetNextToken_()

Export

- Entirely inside Writer, as most of the output is put together manually
- Change all code in SwHTMLWriter:
  - From OOO_STRING_SVTOOLS_HTML_foo
  - To GetNamespace() + OOO_STRING_SVTOOLS_HTML_foo

3 types of embedded objects

OleEmbeddedObject

- Has native data
- We try to let an external application handle that data

OCommonEmbeddedObject

- Has native data
- We loaded that into one of our own document models (Writer e.g.)

ODummyEmbeddedObject

- May or may not have native data
- If it has, we don’t understand that data at all
- Nothing happens on double-click on the object

Embedded objects code reuse

Layers

- .reqifz (ZIP)
- XHTML: refers to PNG (replacement image) + native data (RTF fragment)
- RTF: hexump of OLE1 container
- OLE1: wraps an OLE2 container
- OLE2: binary MSO document or ODF/OOXML

Binary MSO filters already support the anything-as-OLE2 feature

- Duplicating that in the HTML filter would be sad
- Import: SvxMSDffManager::GetFilterNameFromClassID()
  - And if it’s ODF: SvxMSDffManager::ExtractOwnStream()
- Export: it works out of the box, embeddedobj code does the hard work

Testing

ReqIF validator (Consequent)

- Nicely tests all aspects of the XHTML fragment
- http://formalmind.com/tools/consequent/

Our side

- CppunitTest_sw_htmlimport
- CppunitTest_sw_htmlexport
- Can parse the export result as an XML DOM tree
  - Then XPath asserts on it

Thanks

Collabora is an open source consulting company

- What we do and share with the community has to be paid by someone

Vector (Software + Services for Automotive Engineering)

- Sponsor of this work

Summary

HTML support in Writer is not dead

- An XHTML mode is here with new features
- Improved performance, compared to existing XSLT-based approach

Thanks for listening! :-)

- Slides: https://vmiklos.hu/odp