

OOXML Interoperability

LibreOffice Conference 2013, Milan

Adam Fyne



- Senior Software Engineer
- Love tackling technologies
- .Net → Open-Source
- Passion for poker
- Adam.Fyne@cloudon.com

what is OOXML INTEROPERABILITY



Microsoft Office



LibreOffice



Microsoft Office

Why Bother ?

to disrupt Microsoft !

- MS too sticky



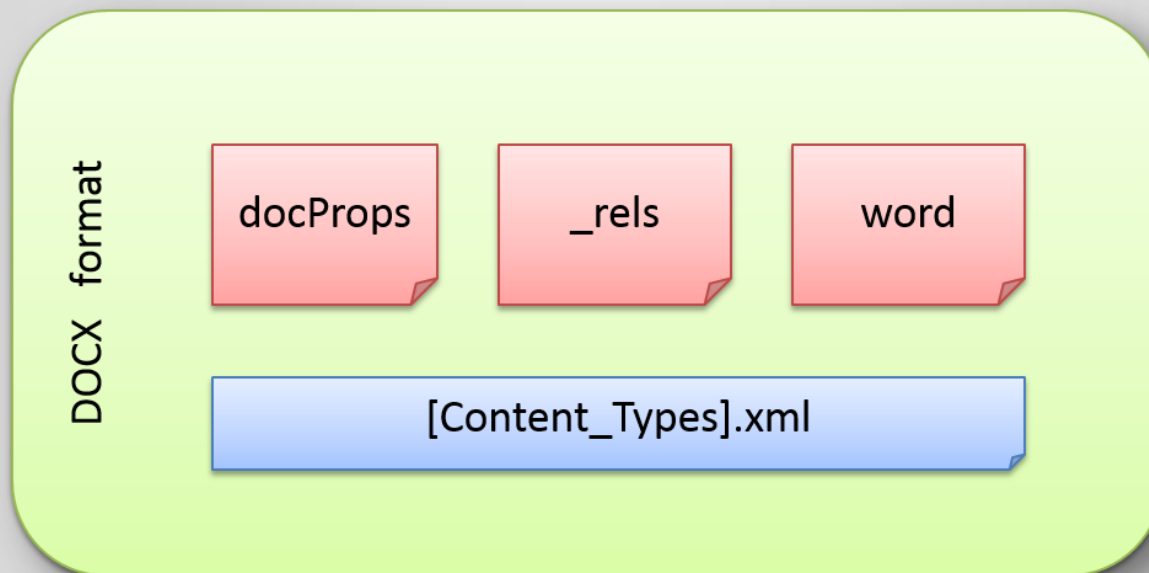
- Adoption barriers



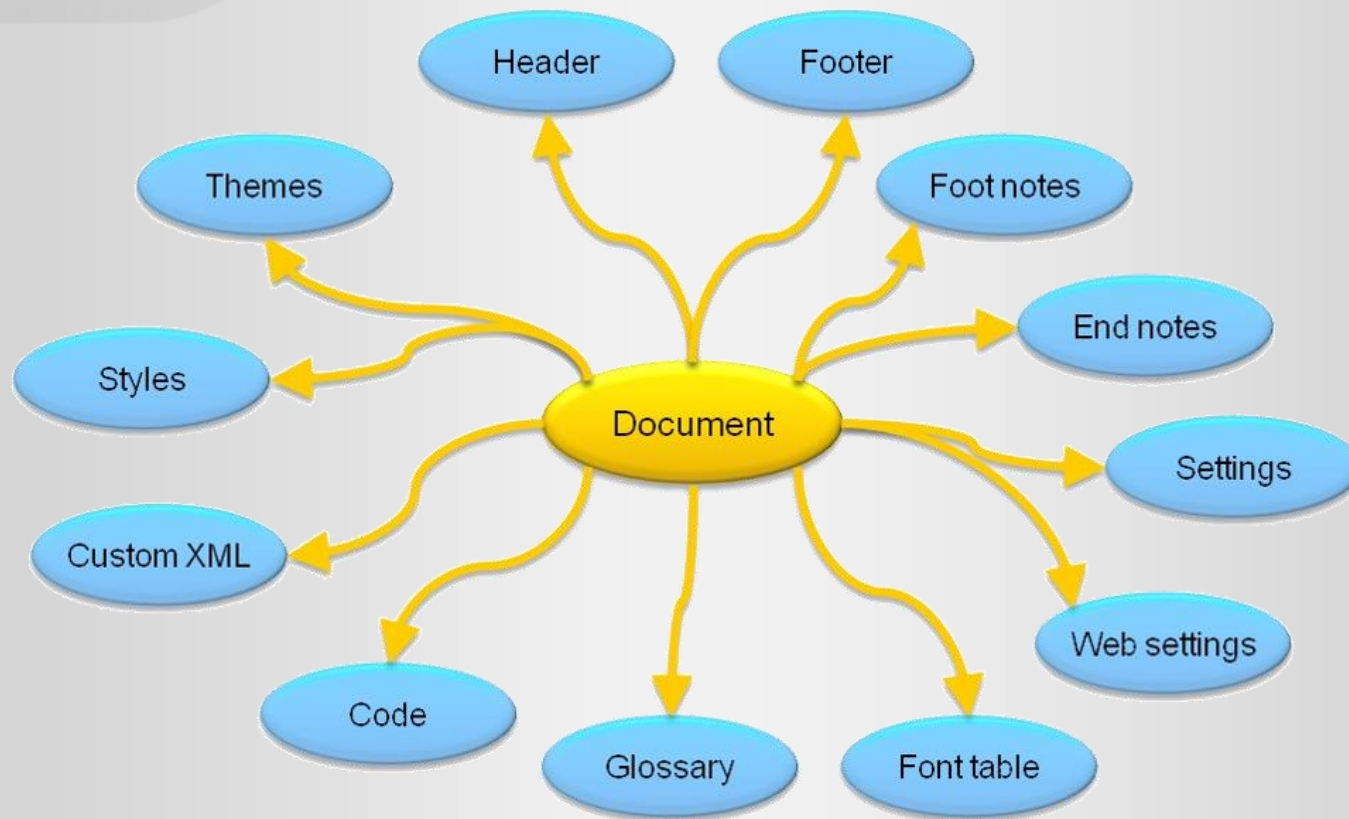
now is the time !

DOCX

DOCX is a ZIP



document.xml



What's on the menu ?

- How bad is it ?
- Fixing considerations
- Stuff we did
- Beasts to tackle



where are we ?



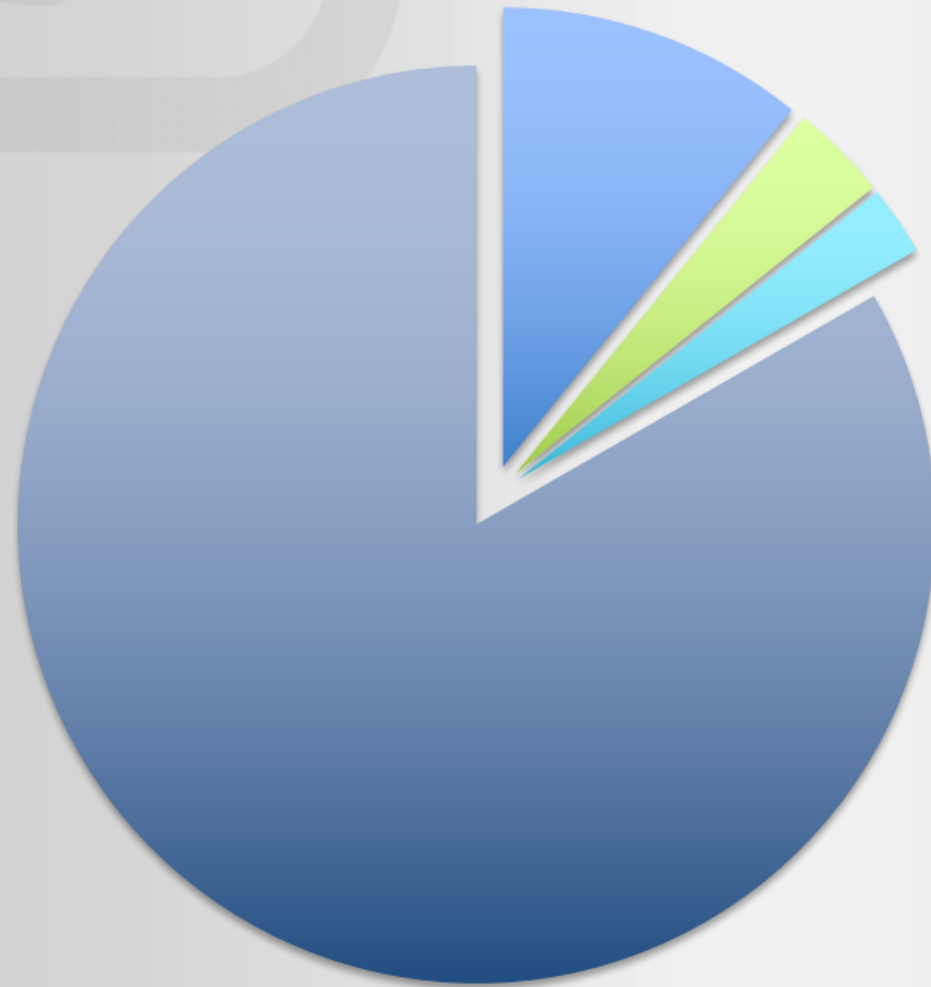
5,700

open bugs

* As of September 1st, 2013



MS Related Bugs

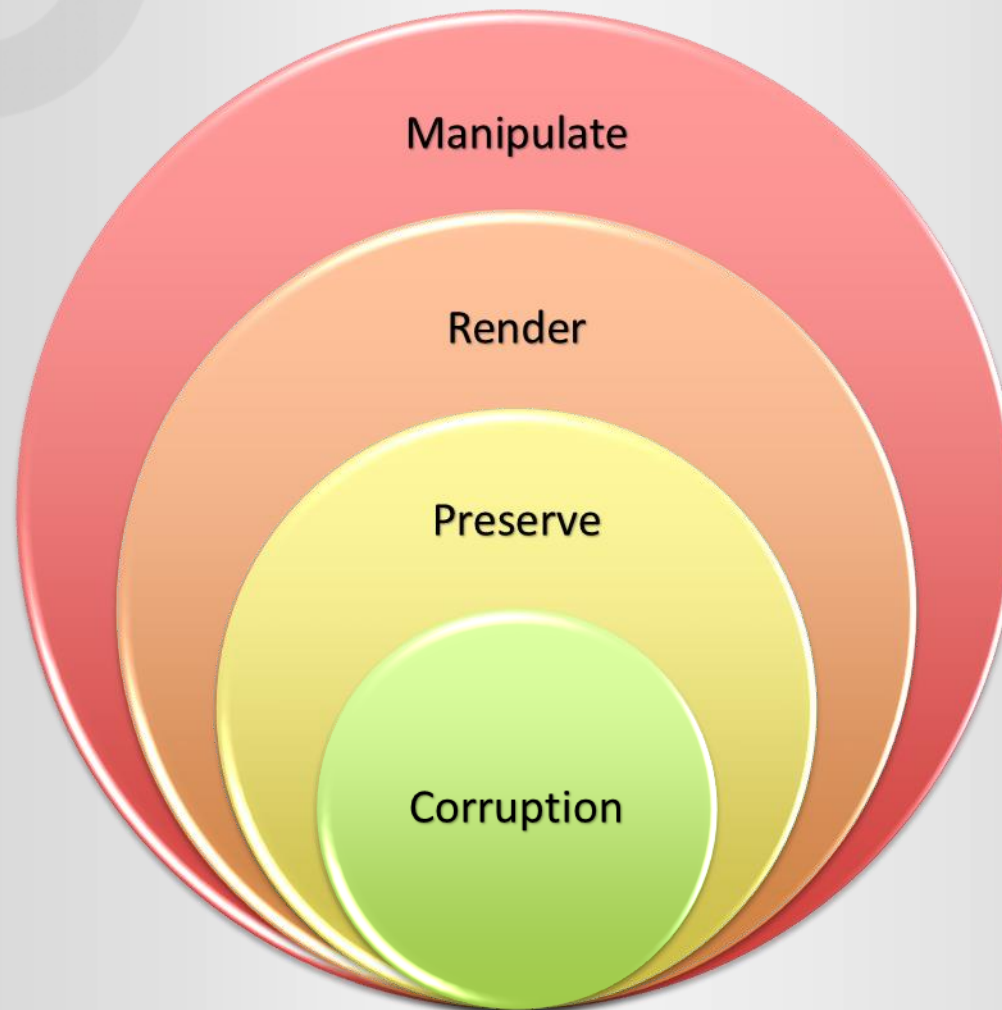


20%

- 620 bugs - Word
- 200 bugs – Excel
- 140 bugs – PowerPoint

■ Word Bugs ■ Excel Bugs ■ PowerPoint Bugs ■ Other Bugs

4 Layers



preserve

most important

[corruption goes without saying]



we found

61 preserve bugs

45 render bugs

90% unreported



Manual Issue Mapping

MS Word Roundtrip

- Methodically try features of MS Word
- For each feature – create DOCX
- Roundtrip DOCX in LO
- Compare in MS Word

For Each Problem

- Classify problem – ‘corruption’, ‘render’, ‘preserve’
- Minimize DOCX to bear-bones
- Create \ Find in Bugzilla

For Each Bug

- Create inner bug in CloudOn
- Compare XML contents of export & original
- Check if DOC filter works
- Find problematic area (import \ core \ export)

Scale Testing



work in progress

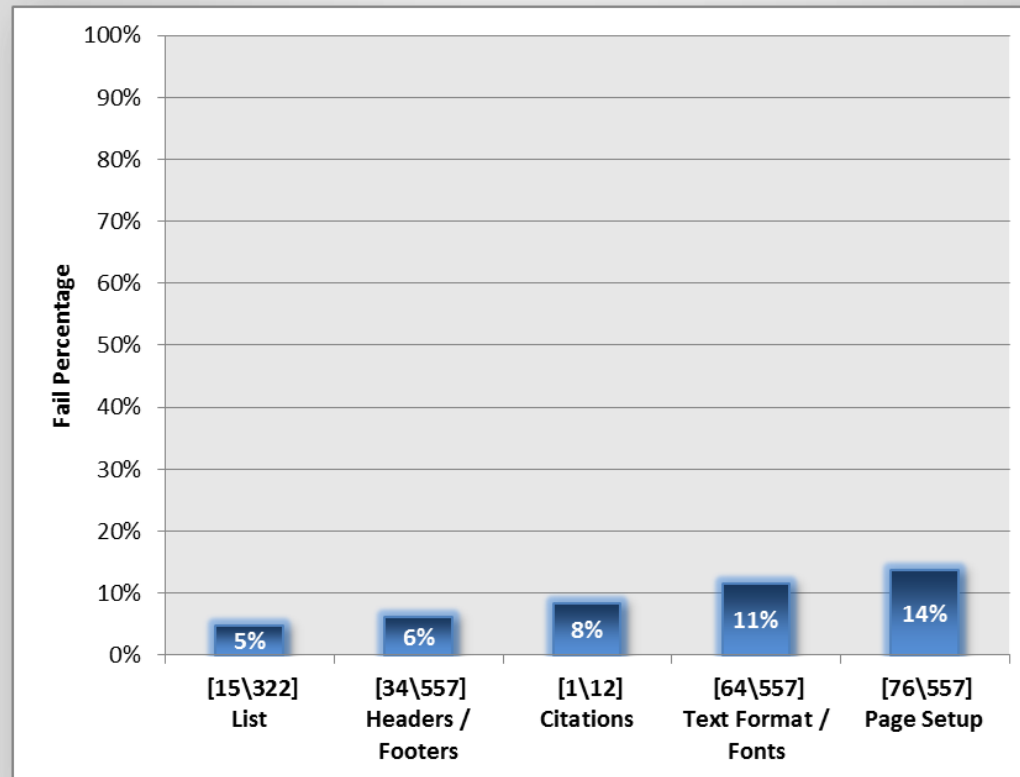


Microsoft → LibreOffice → Microsoft

- **Test thousands of documents automatically**
- **Full coverage of Office features**
- **produce a report that contains**
 - Which files differ
 - Which features are contained in each file

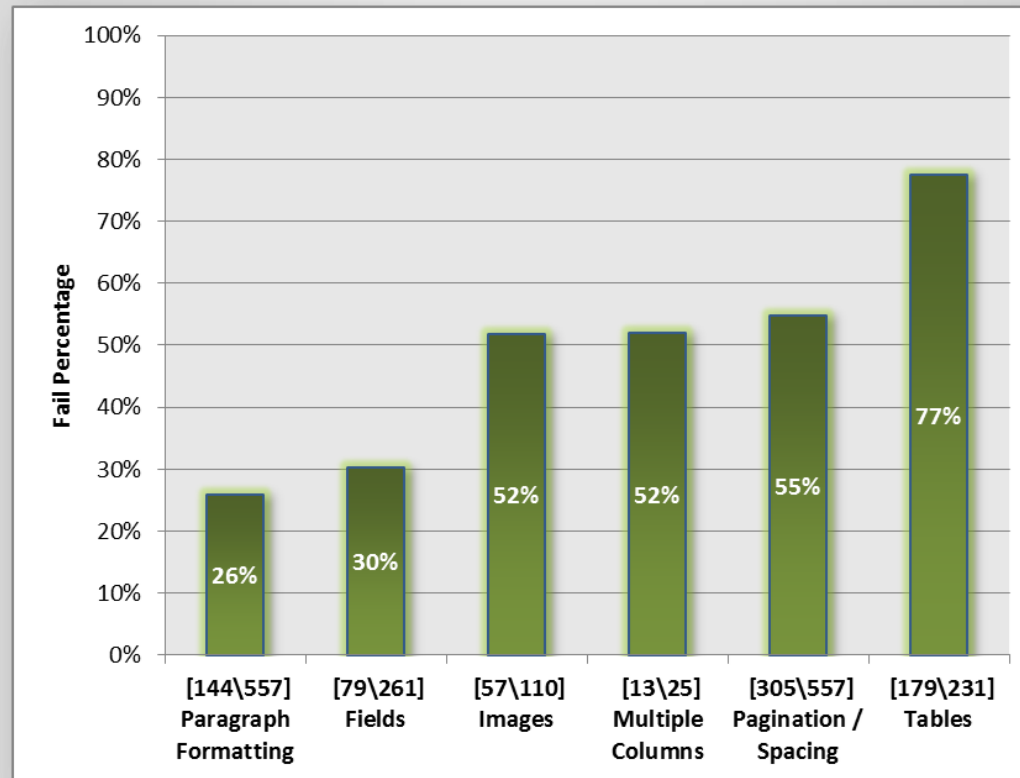
Scale Testing

The Good



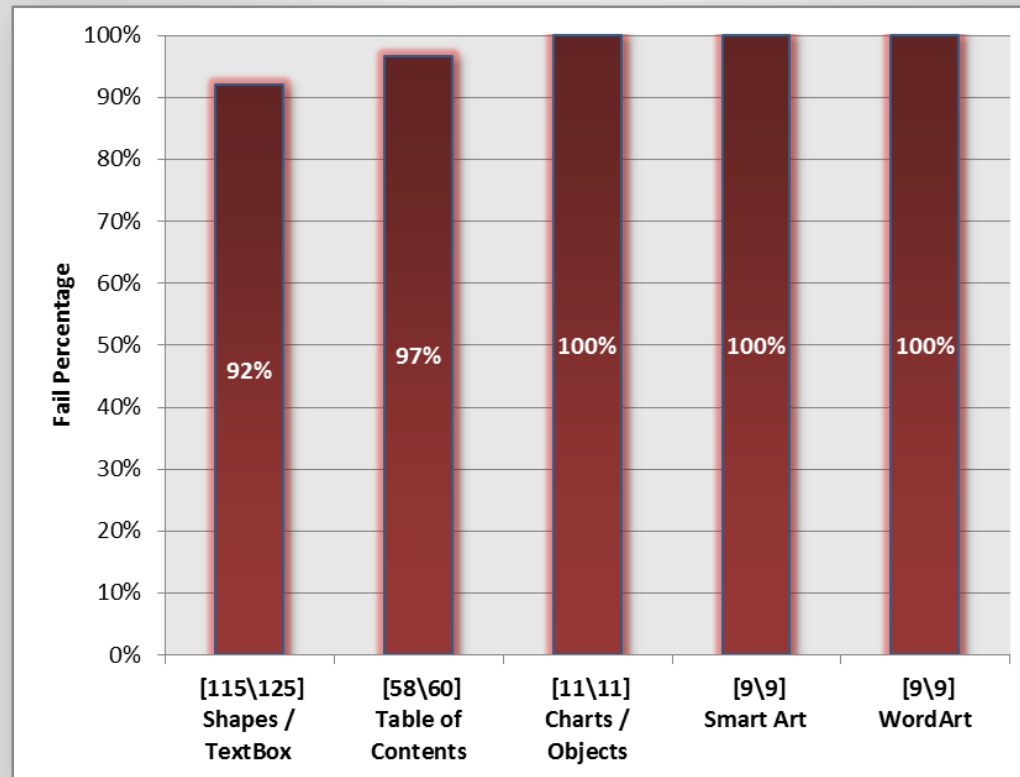
Scale Testing

The Bad



Scale Testing

The Ugly



The Goal

- Corruption - **zero** tolerance
- Preserve - **low** tolerance
- Render - **medium** tolerance
- Manipulation - **high** tolerance

100% ROUNDTRIP

how to solve interop ?

How to compare preservation?

XML Preservation

vs

Pixel Preservation

Example #1

XML - not preserved

Pixels - preserved



AND THAT'S OK !

Different Order - *Paragraph Properties*

- There is no meaning for order of XML nodes
- 99% of time - LO will export in different order than original

Style Names

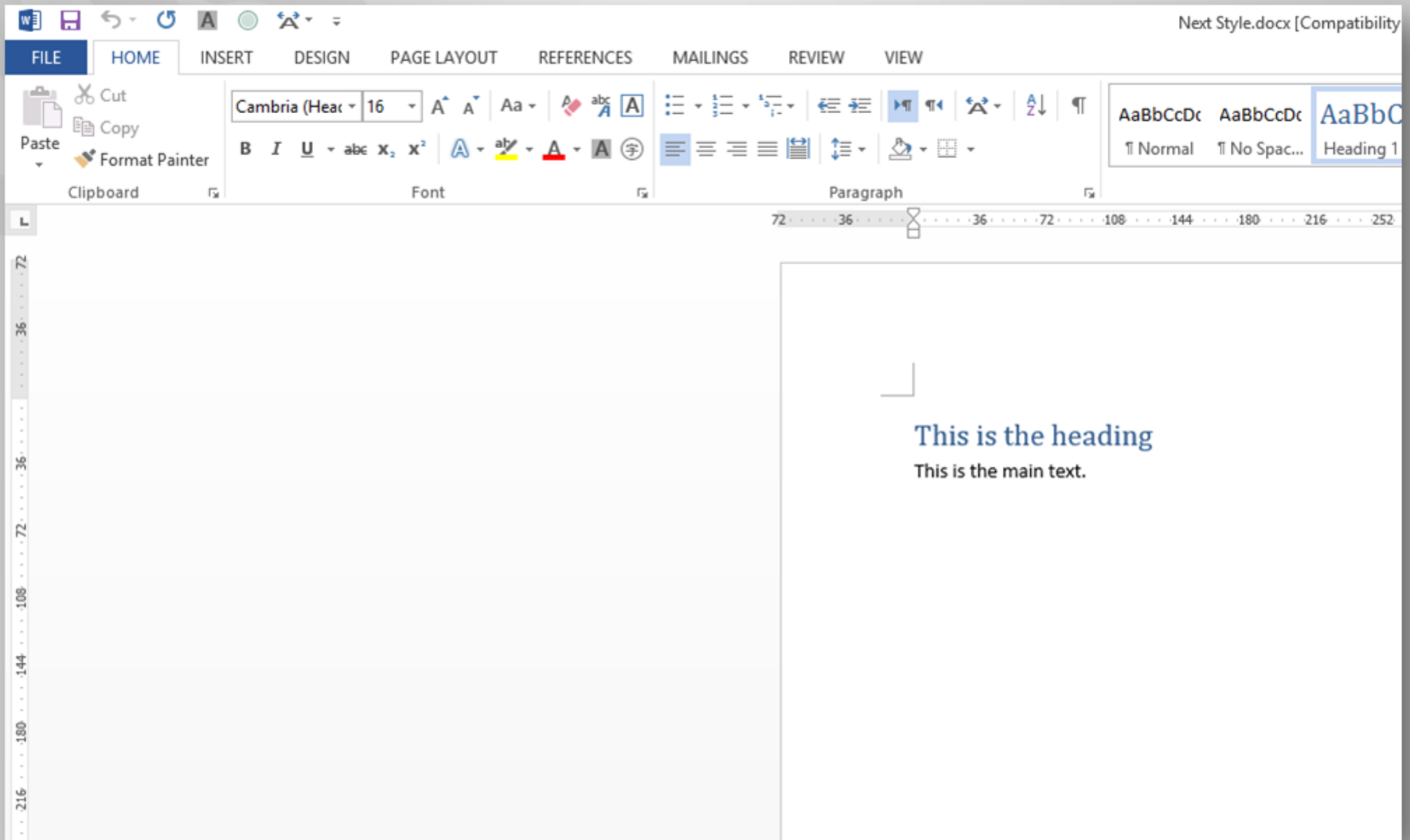
- No meaning for style name
- Can be different from original

Example #2

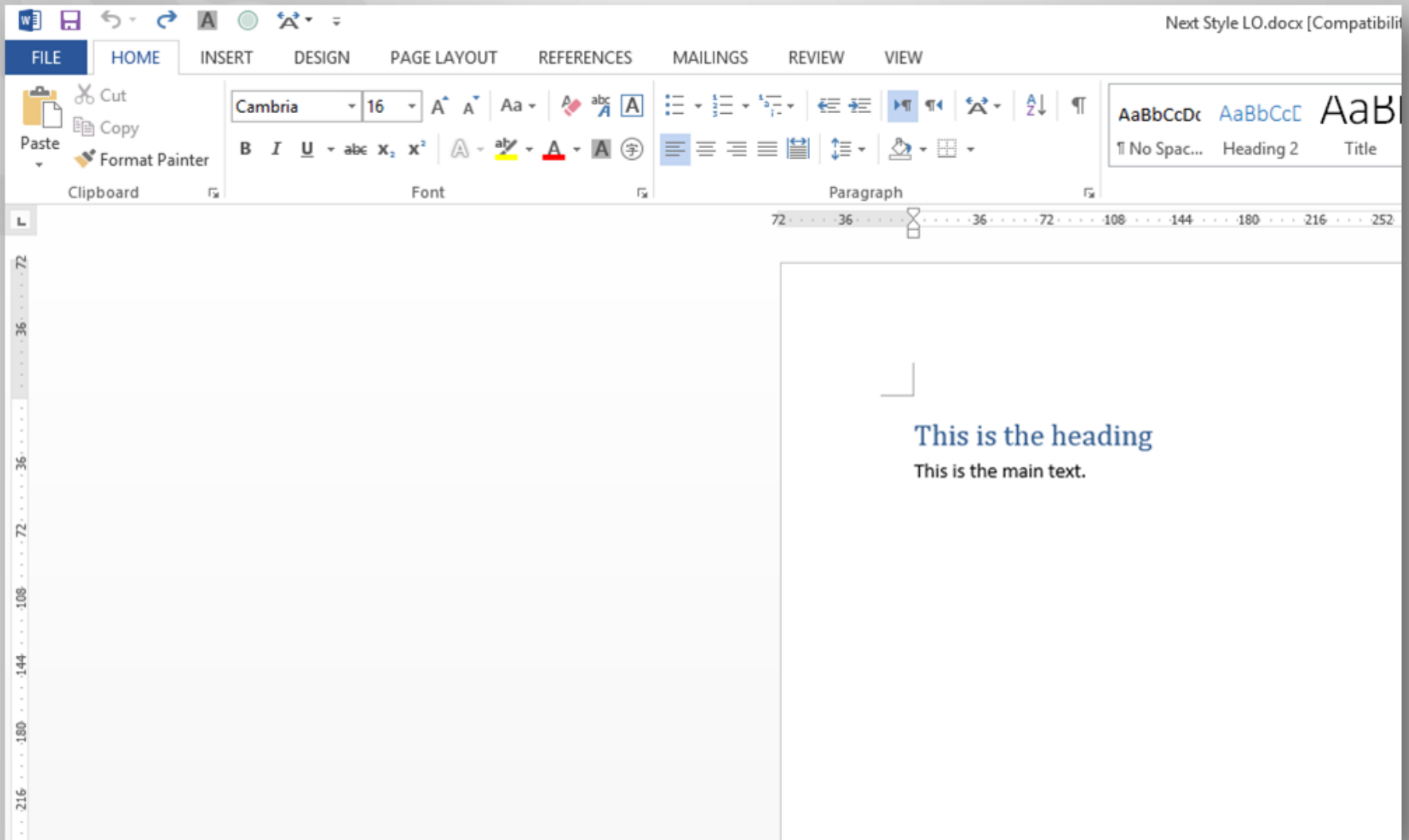
XML not preserved, Pixels Preserved

Different Behavior

NEXT Style



Original DOCX in Word



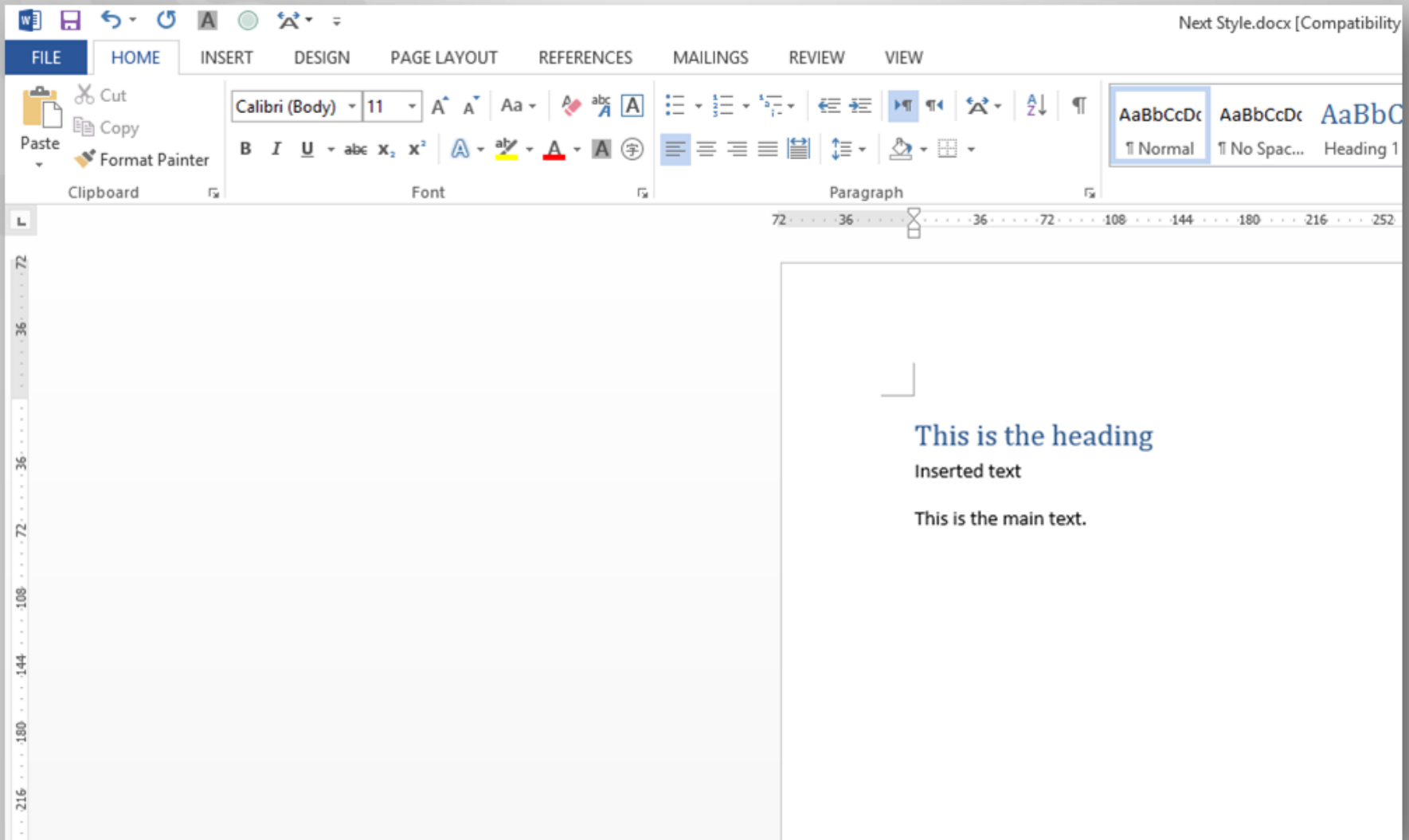
LO-Exported DOCX in Word

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<w:styles ...>
  ...
  <w:style w:type="paragraph" w:styleId="Heading1">
    <w:name w:val="heading 1"/>
    <w:basedOn w:val="Normal"/>
    <w:next w:val="Normal"/>
    <w:link w:val="Heading1Char"/>
    <w:uiPriority w:val="9"/>
    <w:qFormat/>
    <w:rsid w:val="005C517F"/>
    ...
  </w:style>
  ...
</w:styles>
```

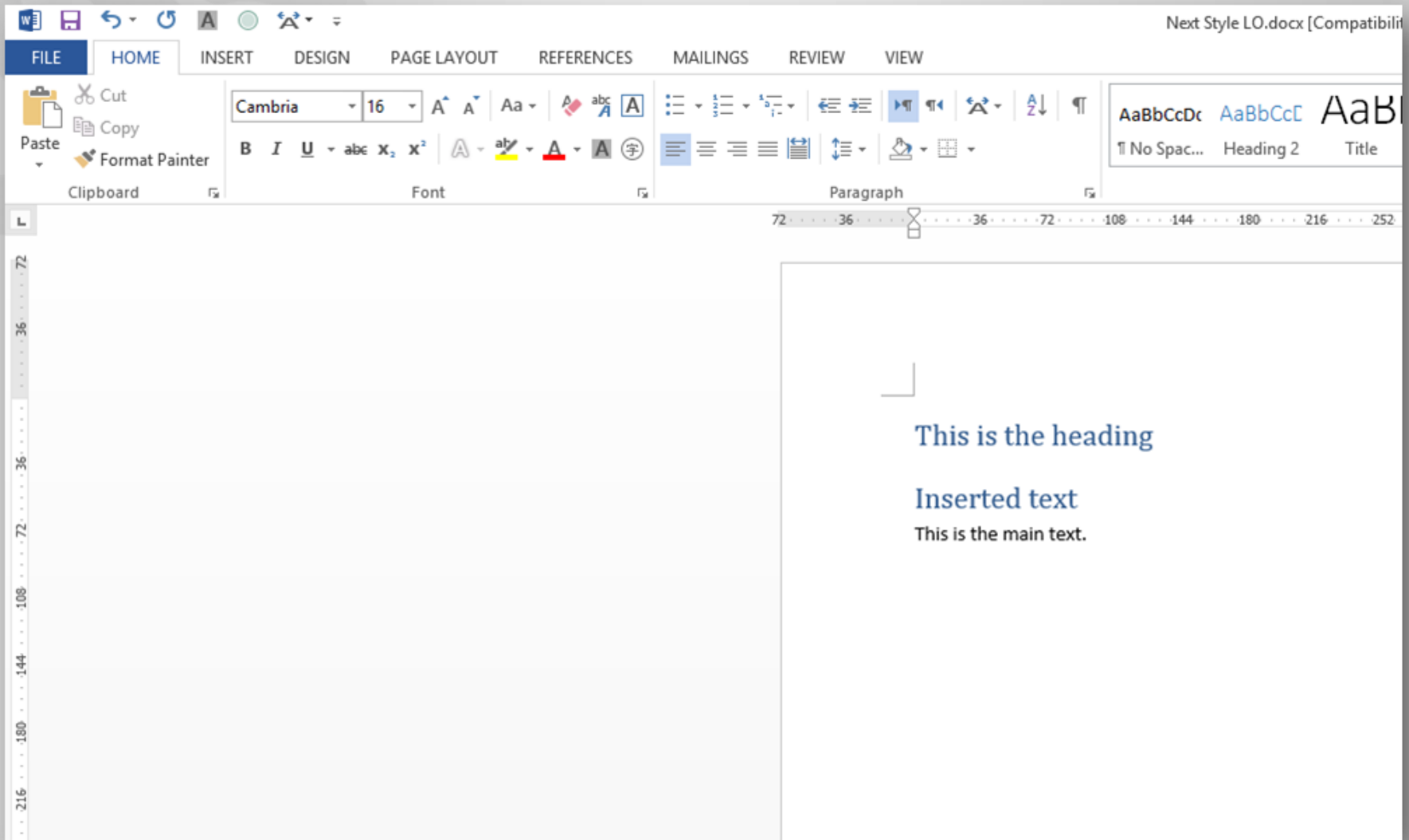
Original DOCX 'styles.xml'

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<w:styles ...>
  ...
  <w:style w:styleId="style1" w:type="paragraph">
    <w:name w:val="Heading 1"/>
    <w:basedOn w:val="style0"/>
    <w:next w:val="style1"/>
    ...
  </w:style>
  ...
</w:styles>
```

LO-Exported DOCX 'styles.xml'



Insert Text in Original DOCX



Insert Text in LO-Exported DOCX

Bottom Line

- **XML comparison** - not accurate
- **Pixel Comparison** - hides functionality differences

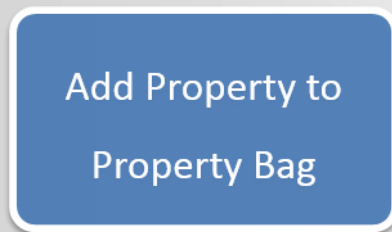
So How to Verify Preservation?

Compare in final product (MS Office)

- visually - detect obvious
- functionally

Preserving – How ?

The Generic Way

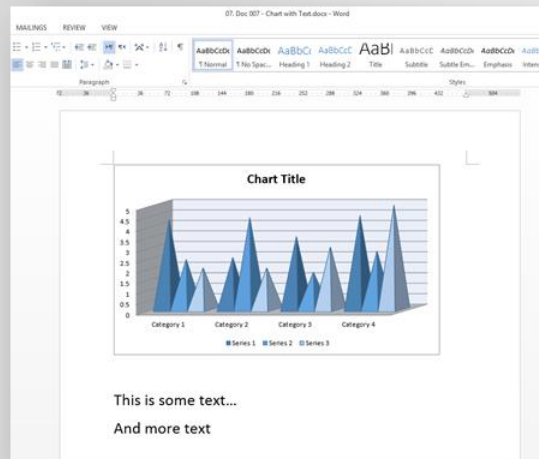


The Specific Way

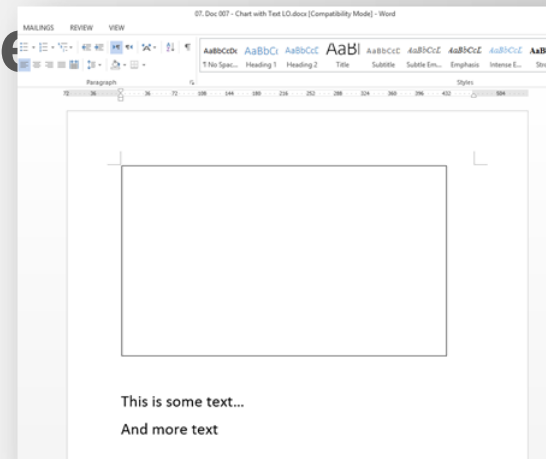


Preservation Issue

How to render something that only has 'preserve' logic?



lace



what is CloudOn doing?

- Automatic Testing
- Issue Mapping
- Bug Fixing

OOXML Workgroup

- CloudOn



- Collabora



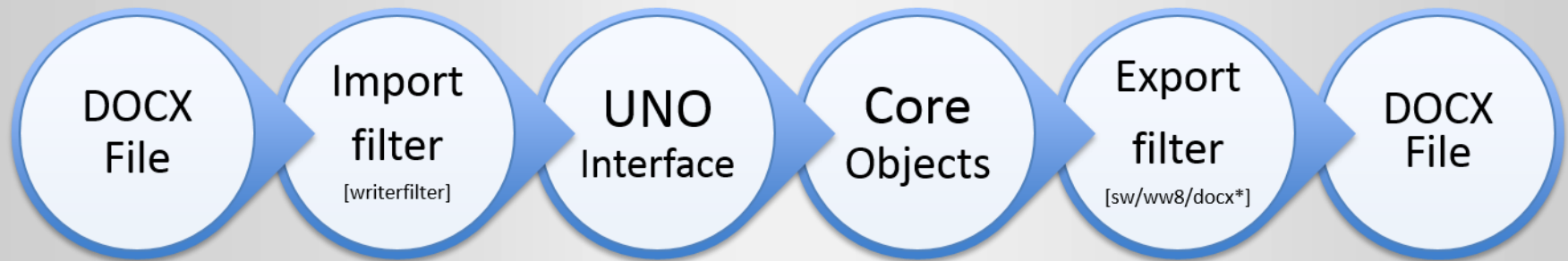
- Synerzip



- Igalia



Data Flow



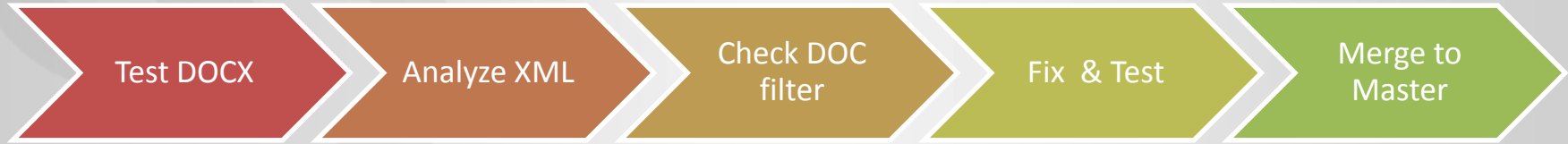
What We Fixed

&

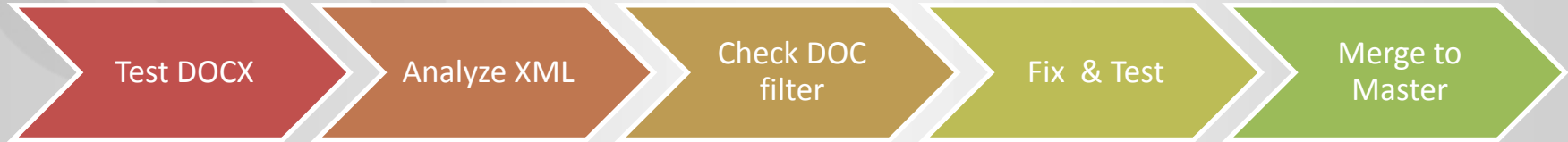
how we fixed it

Easy Hack – Background Color Export

Part 4 / What We Did



Easy Hack – Character Shading



you add to DOCX



you add to ODT

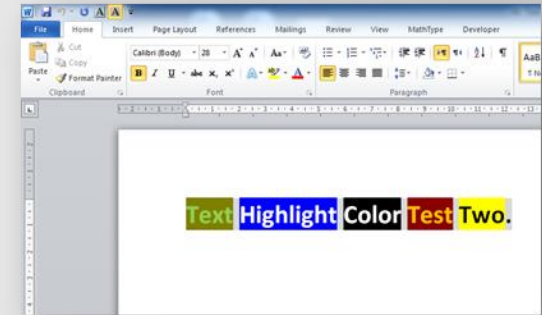
the big stuff



Issue #1 – Character Highlighting

Current

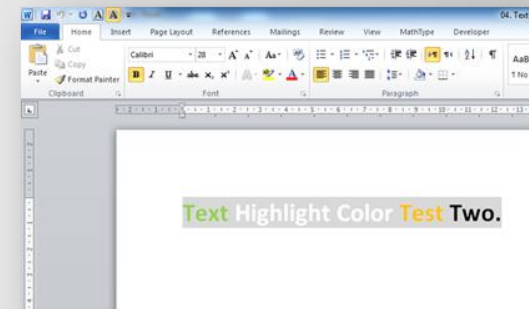
- Highlighting → Shading
- Ignored if Shading also exists



Original

Cause

- Core has only 1 background color for a character



LO-Export

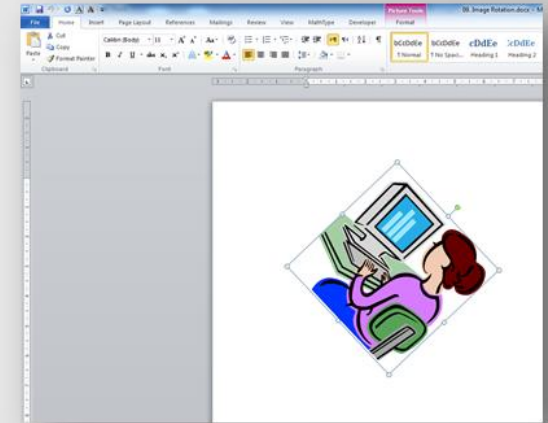
Issue #2 – Image Rotation

Current

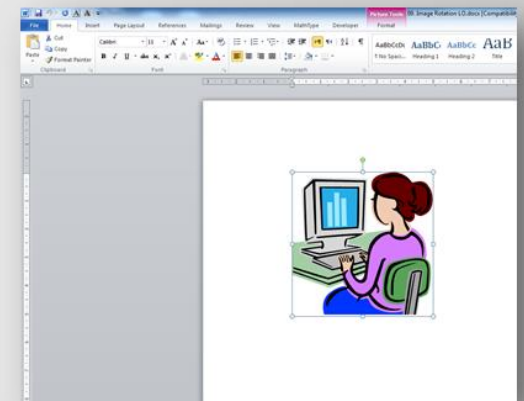
- Image rotation lost

To Do

- ‘Draw’ uses ‘XShape’ for image
- ‘Writer’ uses ‘SwXTextGraphicObject’ (loses information)
- SwXTextGraphicObject → XShape



Original



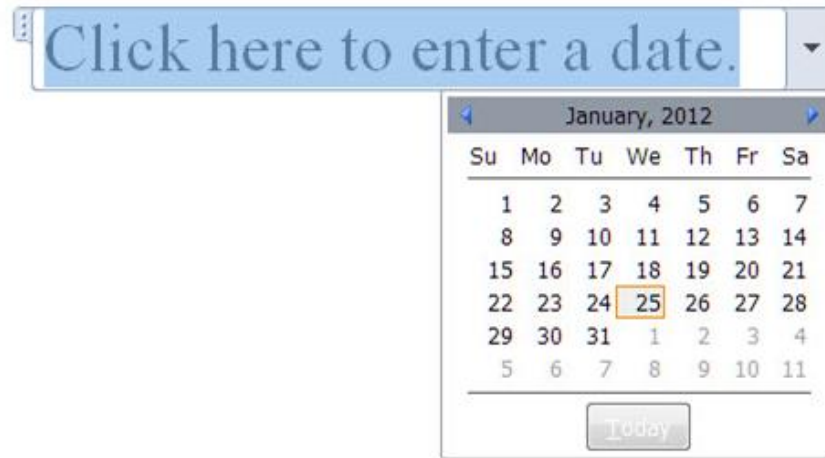
LO-Export

Issue #3 – Content Controls

Background

Content controls in documents

- UI that has
- Restrictions
-



Placeholder text

Issue #3 – Content Controls

Current

- Loss of content control structure
- Content controls turn to simple text

To Do

- 'sdt' node
- 'sdtPr' node
- 'sdtEndPr' node
- 'sdtContent' node

Issue #4 – Table of Contents

Current

- Loss of hidden structure tag
- Loss of ToC functionality
- ToC appearance changes (# of levels)
- ‘Hardcoded’ hyperlinks

To Do

- ToC field & flags
- PAGEREF field & flags

Contents	
Heading 1.....	2
Heading 2.....	2
Heading 3.....	3
Heading 4.....	4
Heading 5.....	4
Heading 5.1.....	5
Heading 6.....	6
Heading 6.1.....	7
Heading 6.2.....	8
Heading 7.....	9
Heading 7.1.....	10
Heading 8.....	11
Heading 9.....	12
Heading 10.....	12
Heading 11.....	12
Heading 12.....	12
Heading 13.....	13
Heading 14.....	14
Heading 15.....	15
Heading 15.1.....	15
Heading 15.2.....	15

Original

Contents	
Heading 3.....	3
Heading 4.....	4
Heading 5.....	4
Heading 6.....	6
Heading 7.....	9
Heading 8.....	11
Heading 9.....	12
Heading 10.....	12
Heading 11.....	12
Heading 12.....	13
Heading 13.....	14
Heading 14.....	15
Heading 15.....	16

LO-Export

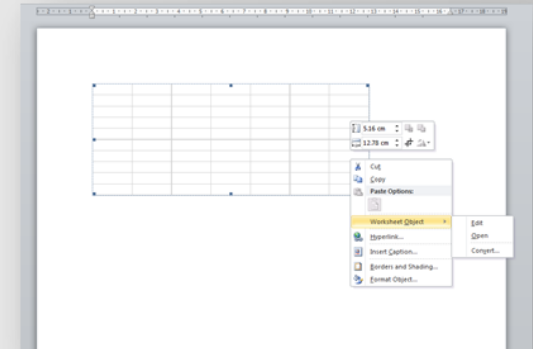
Issue #5 – Excel Embedded in DOCX

Current

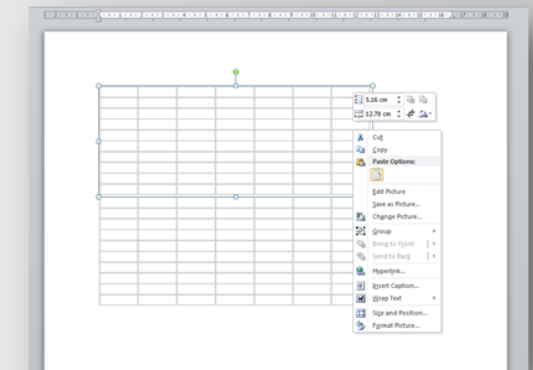
- Embedded Excel file lost
- Sheet turns to 2 pictures
- Import Works

To Do

- Fix Export



Original



LO-Export

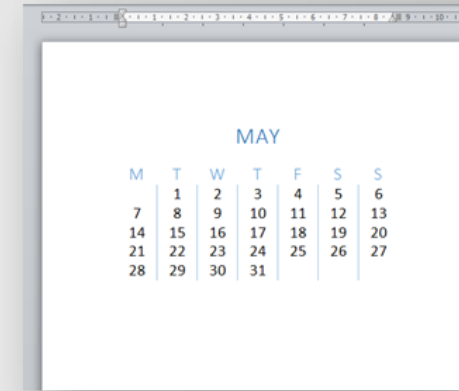
Issue #6 – Table Styles

Current

- Table header not capitalized
- Wrong font sizes
- Wrong font color

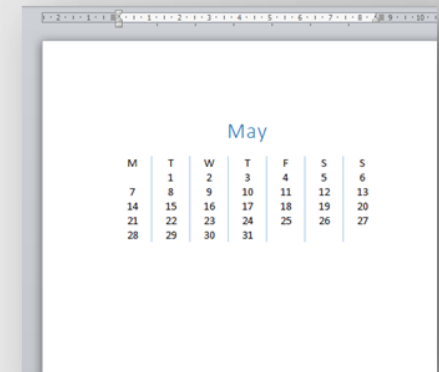
Cause

- Table styles not preserved
- Style attributes → direct formatting



MAY						
M	T	W	T	F	S	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Original



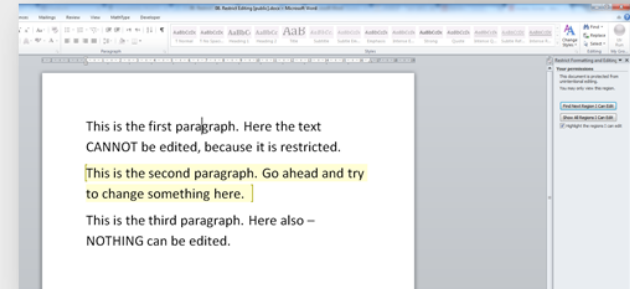
May						
M	T	W	T	F	S	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

LO-Export

Issue #7 – Restricted Editing

Current

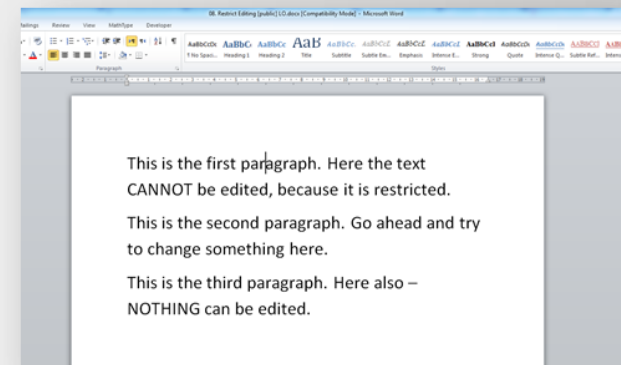
- Lost restrictions on paragraphs



Original

To Do

- ‘permStart’ *(‘document.xml’ file)*
- ‘permEnd’ *(‘document.xml’ file)*
- ‘documentProtection’ *(‘settings.xml’ file)*



LO-Export

Issue #8 – Password Protected DOCX

Current

- Password-Protected DOCX cannot be imported into LO
- Import support only 'old' decryption & encryption (DOC)

To Do

- Support import & export

Issue #9 – Themes

Current

- Themes transformed to direct formatting

GSoC 2013 Task

- Interoperability with MS Office
- Implementing user interface for the feature
- Implement the ODF support for that

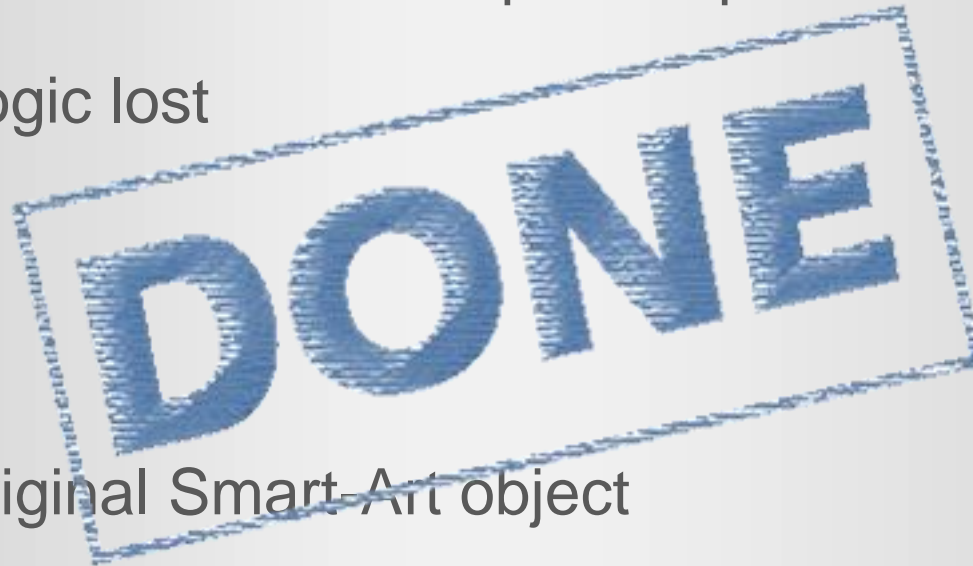
Issue #10 – Smart-Art

Current

- Smart-Art transformed to Simple Shapes
- Smart-Art logic lost

To Do

- Preserve original Smart-Art object



Issue #11 – Charts

Current

- Charts not imported in Writer
- Impress handles charts

To Do

- Make Writer handle like Impress



MOACI

Mother of All Compatibility Issues



Issue #12 – Alternating Content

Background

- Word team – smallest team in Microsoft
- Had least time before Office-2007 Launch
- Did not implement DrawingML, only VML

Issue #12 – Alternating Content

Result

- Word 2010 DOCX contain 2 sections per drawing:
 - Old style VML (for Word 2007)
 - Newer DrawingML (for Word 2010+)
- Writer only imports & export VML
- DrawingML + features lost
- Resulting DOCX appears & behaves different

Issue #12 – Alternating Content

Requirements

- Support **alternating content** mechanism
- Support “Word Processing **Shape**”
- Support “Word Processing **Group**”
- Support “Word Processing **Canvas**”
- Support “Word Processing **Drawing**”

Closing Argument

- Interoperability - huge problem
- Now is the time !
- CloudOn identified, classified, reported & fixed
- Major issues still lurking



“He who says he can and he who says he can't are both usually right.”

- Confucius





Questions ?

Adam.Fyne@cloudon.com

OOXML Hackathon

- **More than 40 attendees ! Amazing work done !**
- **Tablet Winners:**
 - **Miklos Vajna** - Vertical Text
 - **Caolan McNamara** - Combined Characters
 - **Zolnai Tamas** - Character Highlighting



Thank You



All text and image content in this document is licensed under the [Creative Commons Attribution-Share Alike 3.0 License](#) (unless otherwise specified). "LibreOffice" and "The Document Foundation" are registered trademarks. Their respective logos and icons are subject to international copyright laws. The use of these therefore is subject to the [trademark policy](#).