



Collabora Productivity

Exporting shapes to DrawingML

Bug fixing case study

Andras Timar

@timar +timar74

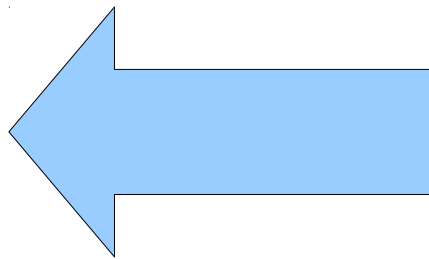


Introduction



Why?

- I was looking for a small problem that I can fix in 2 days (LibreOffice Hackfest 2015 FOSDEM)
- I was told that filter code is easy to hack
- I found a problem:



(draw an arrow, save as docx, reload)

Bugfixing



The problem and scope

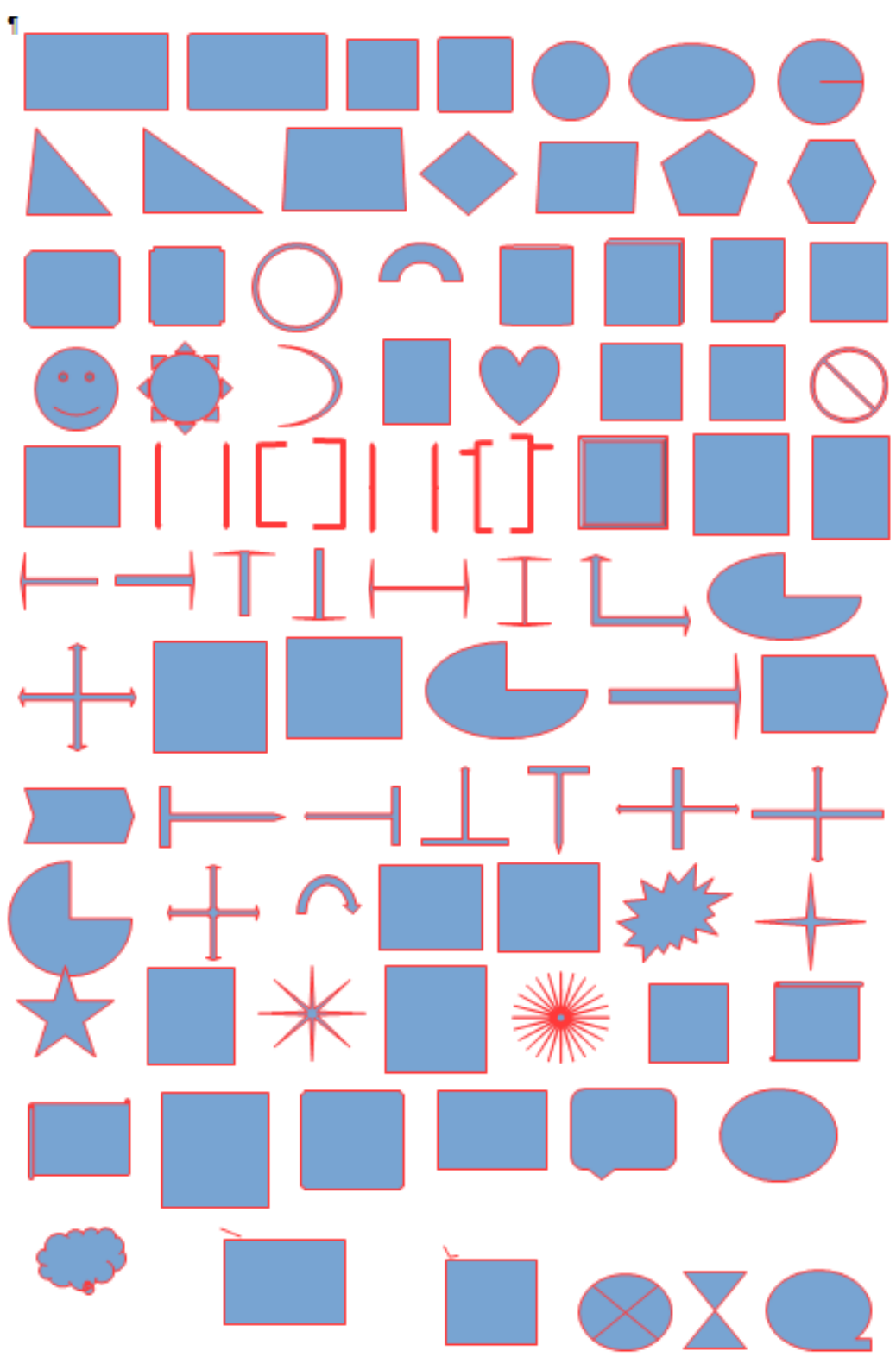
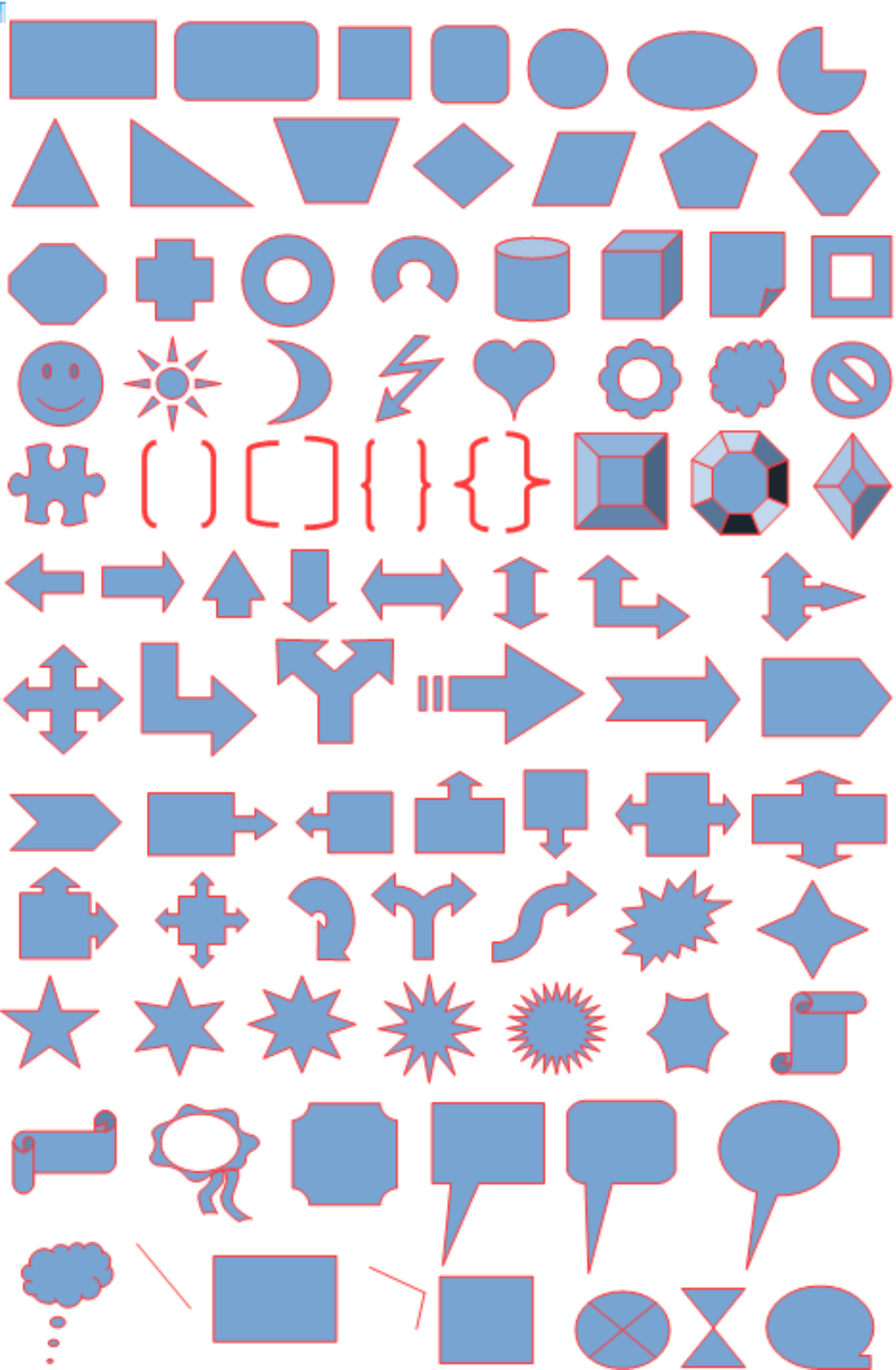
- Legacy VML (Vector Markup Language)
 - XML serialization of binary DOC format
 - MS Office 2007
 - deprecated
- New DrawingML standard
 - from MS Office 2010
 - LibreOffice 4.3 and newer exports DrawingML
 - not as good as VML and binary DOC support



The problem and scope

- Originally DrawingML export code preserved only OOXML custom shapes (no data loss on roundtrip)
- Preset shapes, like the arrow and many others can be adjusted (shape is defined by equations)
- ODF and OOXML standards define different equations for the same preset shapes
- Adjustment values in OOXML and modifier values in ODF are not equivalent





Lot of bugs...

- Adjustment values mismatch
- Shape exported as plain rectangle
- Wrong mapping between ODF and OOXML preset shapes
- Code location:
 - `oox/source/export`



First cut

- Shape is OOXML shape → export as is
- Shape is non-OOXML shape with adjustments → export as polypolygon
 - We loose the adjustment handle but the view is OK
- Shape is an ODF preset shape without adjustment handles
 - Blacklist
 - ODF preset shapes which have good OOXML equivalents, so we export them as such and not as polypolygon
 - Whitelist
 - These ODF preset shapes have OOXML equivalents, but it looks a bit different, so we export them as polypolygons



First cut

- Export non-OOXML shapes with adjustments as polypolygons
 - We loose the adjustment handle but the view is OK
- Blacklist
 - ODF preset shapes which have good OOXML equivalents, so we export them as such and not as polypolygon
- Whitelist
 - These ODF preset shapes have OOXML equivalents, but it looks a bit different, so we export them as polypolygons

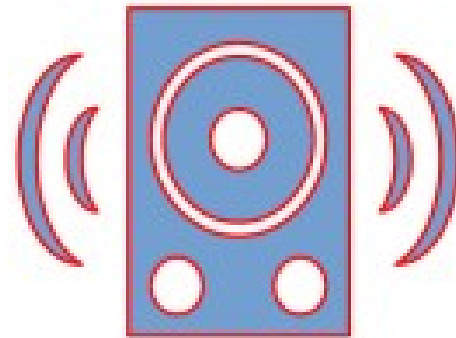
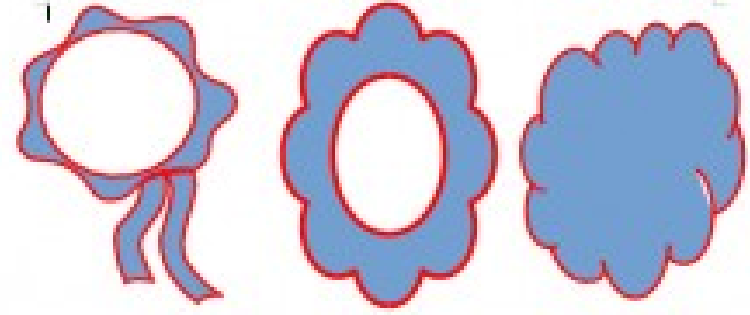
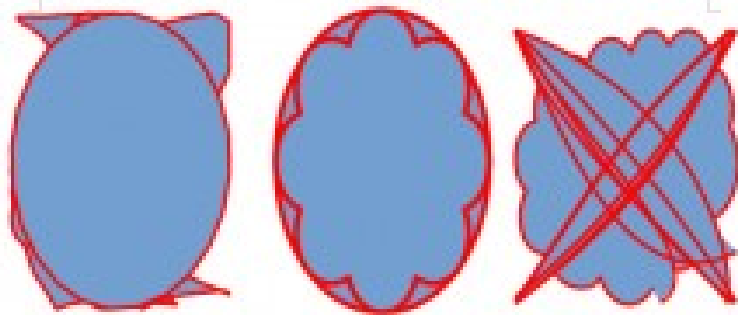


Second cut

- Cambridge Hackfest 2015
- Instead of converting the shape to polypolygon and exporting it, I queried CustomShapeGeometry property via UNO and created the DrawingML export based on that.



Result (before – after)



Tools



VIM

- .vimrc

```
au BufReadCmd *.odt,*.ott,*.ods,*.ots,*.odp,*.otp,*.odg,*.otg,*.oxt call zip#Browse(expand("<amatch>"))
au BufReadCmd *.docx,*.docm,*.dotx,*.xlsx,*.pptx call zip#Browse(expand("<amatch>"))

map <F12> :% !prettyprint<CR>
```

- prettyprint

- <https://gerrit.libreoffice.org/gitweb?p=dev-tools.git;a=tree;f=prettyprint>
- Unlike 'xmllint --format', this works even if the XML is corrupt
- It does not alter the XML itself in any way



XRayTool

- Download
 - <http://bernard.marcelly.perso.sfr.fr/index2.html>
- Open XrayTool60_en.odt
- Install

2 Installation

Click on this button to run the installation of Xray. After confirmation, it will suppress any existing installation of Xray and install the new version. If your version of OpenOffice is incompatible with this version of Xray, a previous version will be installed.



Install Xray

You should read carefully the description of Xray in the next chapters.

If you want to uninstall Xray libraries from your OpenOffice installation, click [here](#):



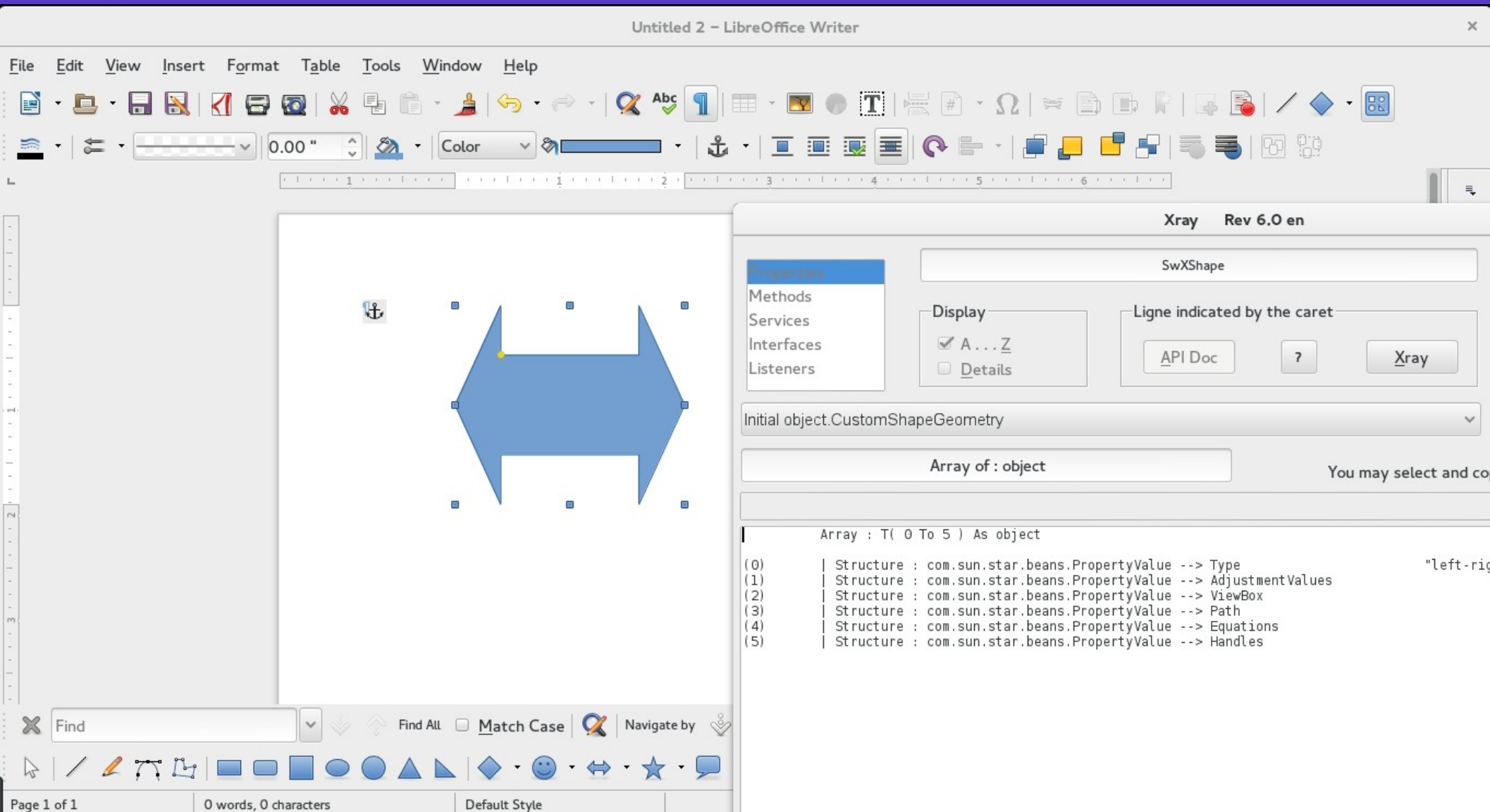
XRayTool

- Alt + F11 (open LibreOffice Basic Macros)

```
Sub XrayShape
  GlobalScope.BasicLibraries.LoadLibrary("XrayTool")
  Document = ThisComponent
  Xray Document.getDrawPage.getByIndex(0)
End Sub
```



XrayTool session



The screenshot shows the LibreOffice Writer application window titled "Untitled 2 - LibreOffice Writer". The main editing area contains a blue double-headed arrow shape. An "Xray" tool window is overlaid on the right side of the document. The tool window has a title bar "Xray Rev 6.0 en" and a "Properties" tab selected. The "Properties" tab shows a list of categories: Properties, Methods, Services, Interfaces, and Listeners. The "Display" section has a checked box for "A...Z" and an unchecked box for "Details". The "Ligne indicated by the caret" section has three buttons: "API Doc", "?", and "Xray". Below this is a dropdown menu showing "Initial object.CustomShapeGeometry". The "Array of : object" section is empty. The bottom section shows a list of properties for the selected object:

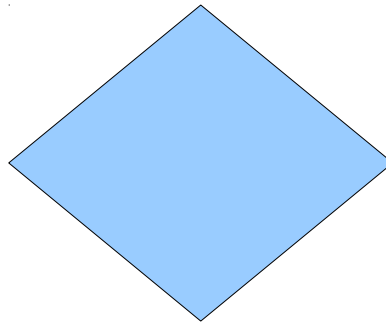
```
Array : T( 0 To 5 ) As object
(0) | Structure : com.sun.star.beans.PropertyValue --> Type
(1) | Structure : com.sun.star.beans.PropertyValue --> AdjustmentValues
(2) | Structure : com.sun.star.beans.PropertyValue --> ViewBox
(3) | Structure : com.sun.star.beans.PropertyValue --> Path
(4) | Structure : com.sun.star.beans.PropertyValue --> Equations
(5) | Structure : com.sun.star.beans.PropertyValue --> Handles
```

The bottom status bar of the LibreOffice Writer window shows "Page 1 of 1", "0 words, 0 characters", and "Default Style".



XrayTool benefits

- Easy to explore objects
- Helps in writing unit tests
- Helped me to find out that some shapes do not have Segments sequence after creation ([tdf#92527](#))



Summary

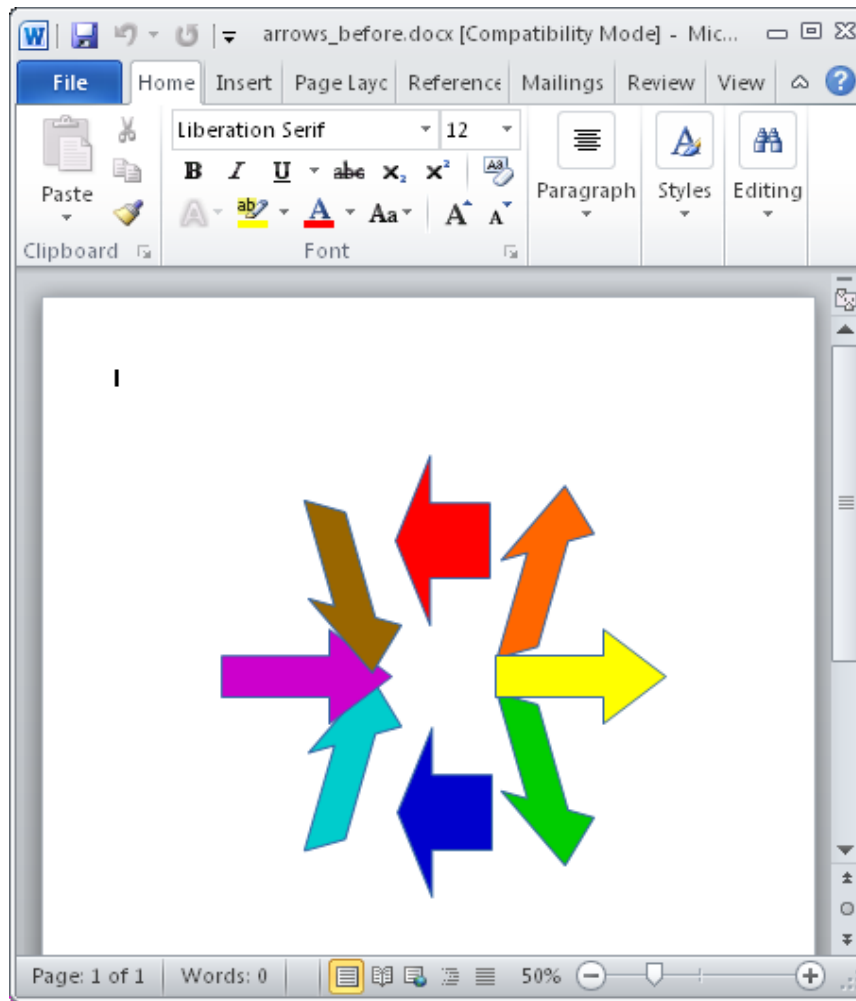


Final thoughts

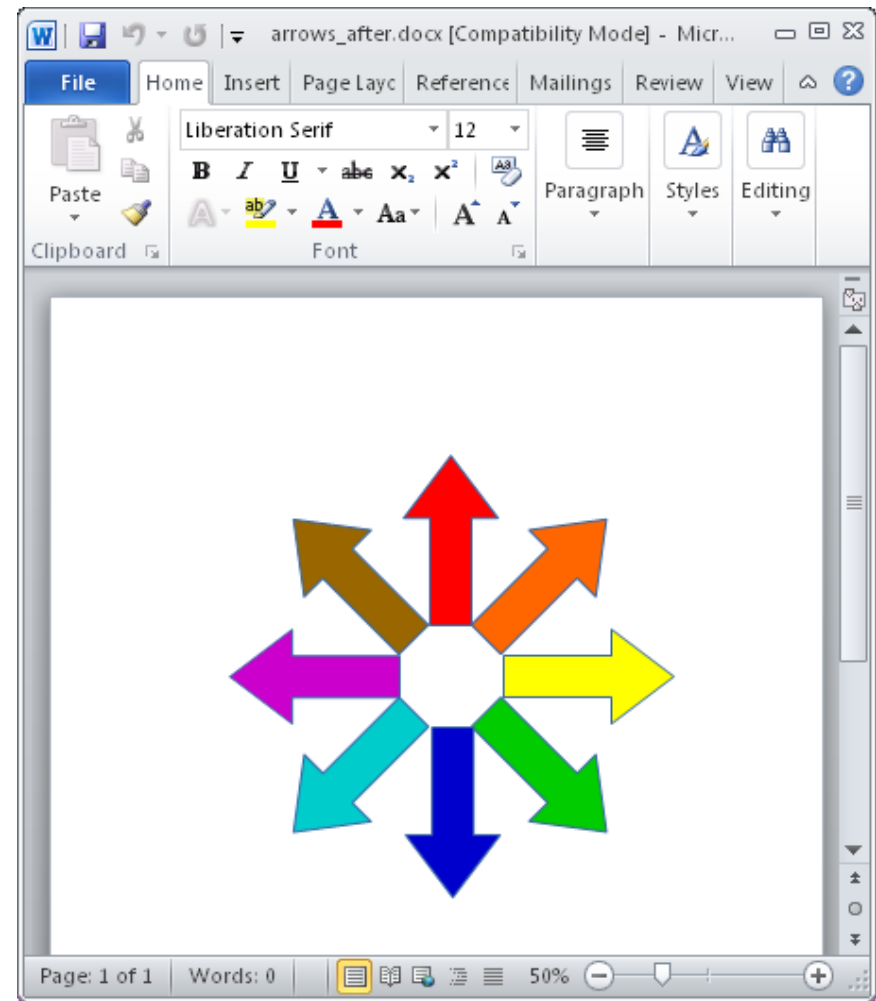
- 2 Hackfests, 4 days, big improvements
- It is rather a hack than a systematic approach
- Full implementation of DrawingML shape export was out of scope (many weeks of work estimated)
- Biggest missing feature: support of equations



Hackfest FOSDEM 2016



before



after

