Basic in OpenOffice / LibreOffice

Waikato Linux Users Group
Ian Stewart
2021 August 23

History – StarOffice / OpenOffice / LibreOffice

- 1985: Star Division (Germany) release StarOffice which included StarWriter. C++
- 1999: Acquired by Sun Microsystems.
- 2000: Sun release the source code as OpenOffice.
- 2000-2002: OpenOffice.org XML. Developed as an open community effort by Sun Microsystems.
- 2005: OpenOffice supports OpenOffice XML document formats.
- 2006: OpenDocument (ISO/IEC 26300:2006) is based on OpenOffice.org XML.
- 2010: Sun acquired by Oracle. StarOffice renamed “Oracle Open Office”.
- 2010: OpenOffice forked to “LibreOffice”.
- 2011: Oracle donate project to Apache Foundation. Renamed the software “Apache OpenOffice”.
- 2020: OpenDocument V1.3 Approved.
- 2021: LibreOffice now has more development / Users than OpenOffice.

History – StarBasic

- 1990's: StarBASIC added to product suite.
- VBA Mode. Visual Basic for Applications compatibility with MS Office suite.
- Code storage choices (Linux):
  - System: `/usr/lib/libreoffice/share/basic/`
  - Application: `/home/ian/.config/libreoffice/4/user/basic/Standard`
  - Document (Embedded): `Basic/Standard/Module1`

- Supports other scripting languages: JavaScript, BeanShell and Python
- APSO – Alternative Python Script Oganizer.

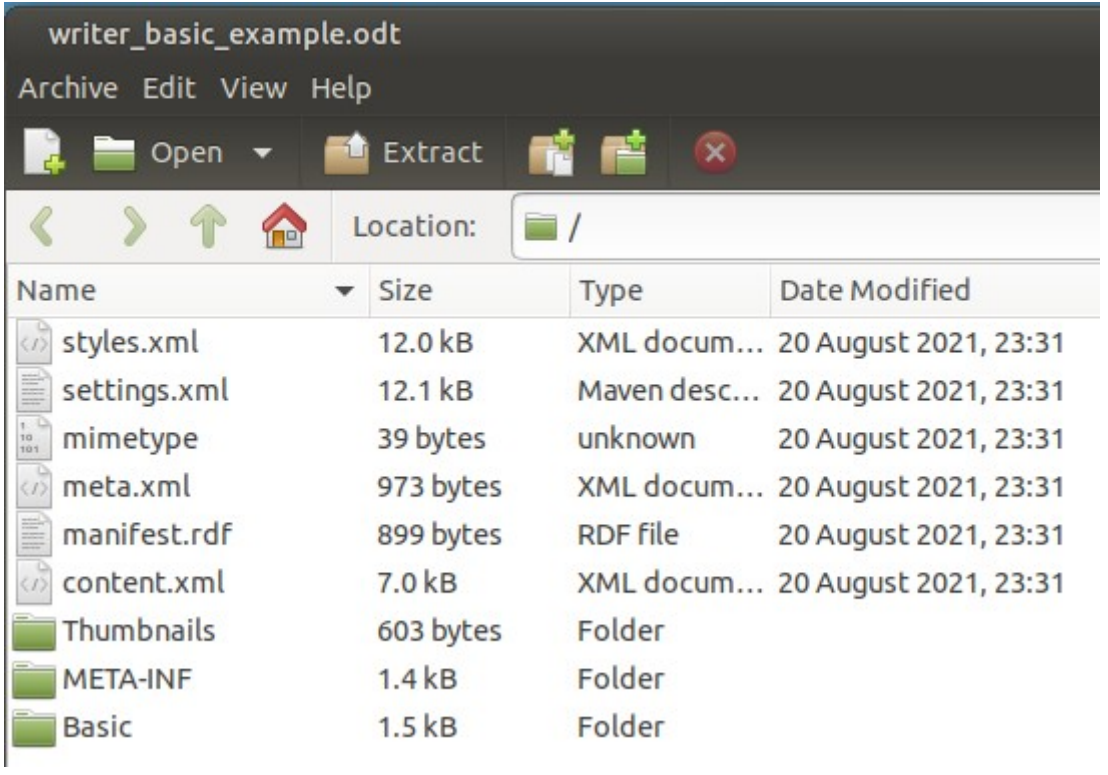
https://extensions.libreoffice.org/en/extensions/show/apso-alternative-script-organizer-for-python

- APSO Development: https://gitlab.com/jmzambon/apso
- Python Storage choices (Linux):
  - System: `/usr/lib/libreoffice/share/Scripts/python`
  - Application: `/home/ian/.config/libreoffice/4/user/Scripts/python`
  - Document (Embedded): `Scripts/python/module.py`

Embedded Basic Demo

Demo manually embedding Basic in a Writer document.
Provide a pushbutton.

Inside a Writer document with embedded Basic



Manifest.xml file

Contains references to three files off the Basic folder.
Module1.xml contains the Basic script.

manifest.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<manifest:manifest xmlns:manifest="urn:oasis:names:tc:opendocument:xmlns:manifest:1.0" manifest:version="1.2"
xmlns:loext="urn:org:documentfoundation:names:experimental:office:xmlns:loext:1.0">
 <manifest:file-entry manifest:full-path="/" manifest:version="1.2" manifest:media-type="application/vnd.oasis.opendocument.text"/>
 <manifest:file-entry manifest:full-path="Basic/Standard/Module1.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="Basic/Standard/script-lb.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="Basic/script-lc.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="Configurations2/" manifest:media-type="application/vnd.sun.xml.ui.configuration"/>
 <manifest:file-entry manifest:full-path="manifest.rdf" manifest:media-type="application/rdf+xml"/>
 <manifest:file-entry manifest:full-path="meta.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="settings.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="Thumbnails/thumbnail.png" manifest:media-type="image/png"/>
 <manifest:file-entry manifest:full-path="styles.xml" manifest:media-type="text/xml"/>
 <manifest:file-entry manifest:full-path="content.xml" manifest:media-type="text/xml"/>
</manifest:manifest>
```

Embedded Basic

Module1.xml

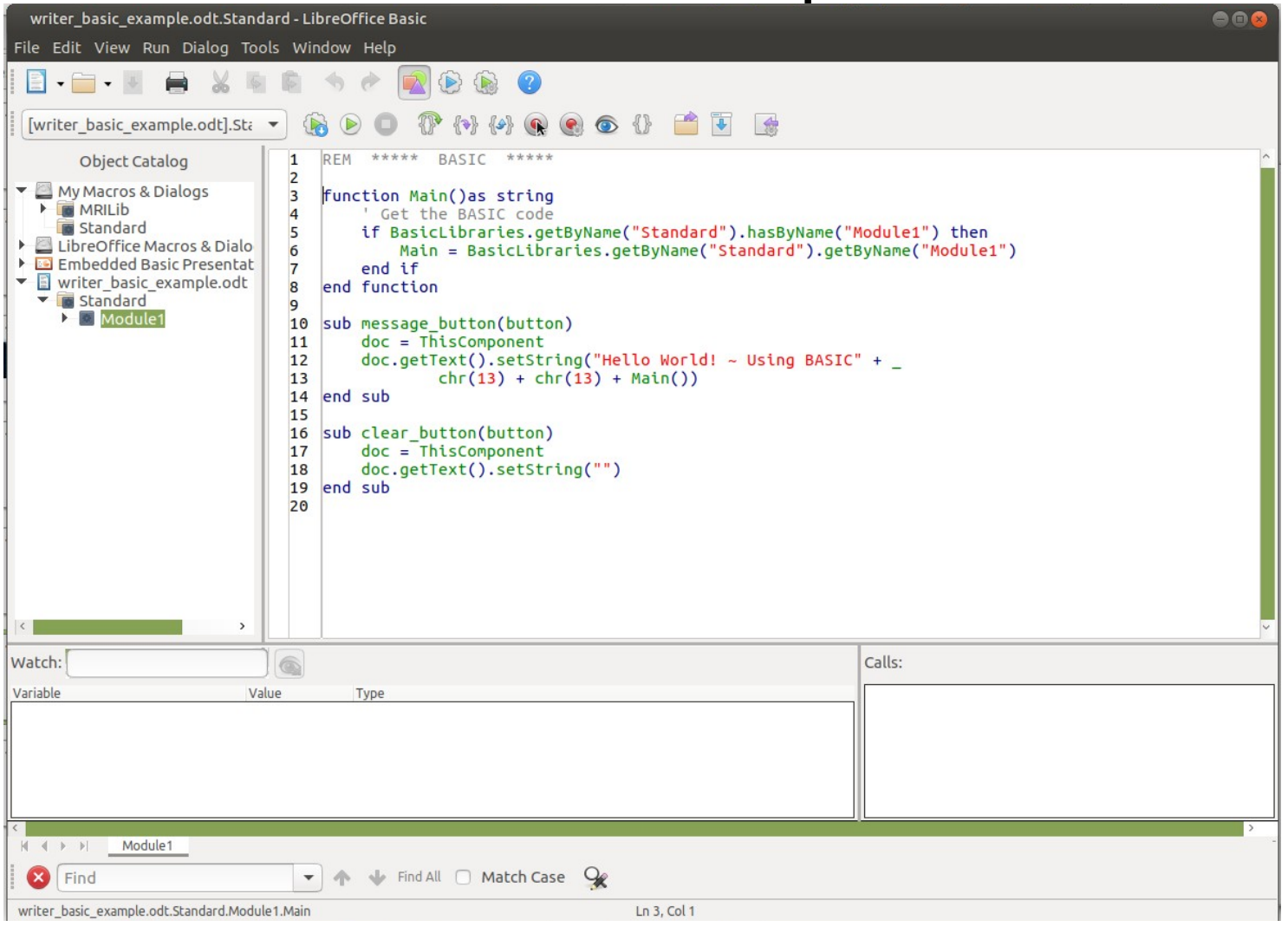
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE script:module PUBLIC "-//OpenOffice.org//DTD OfficeDocument 1.0//EN" "module.dtd">
<script:module xmlns:script="http://openoffice.org/2000/script" script:name="Module1"
script:language="StarBasic" script:moduleType="normal">REM  *****  BASIC  *****

function Main()as string
    &apos; Get the BASIC code
    if BasicLibraries.getByName(&quot;Standard&quot;).hasByName(&quot;Module1&quot;) then
        Main = BasicLibraries.getByName(&quot;Standard&quot;).getByName(&quot;Module1&quot;)
    end if
end function

sub message_button(button)
    doc = ThisComponent
    doc.getText().setString(&quot;Hello World! ~ Using BASIC&quot; + _
            chr(13) + chr(13) + Main())
end sub

sub clear_button(button)
    doc = ThisComponent
    doc.getText().setString(&quot;&quot;)
end sub
</script:module>
```

Basic “Hello world” script viewed with IDE



Basic “Hello world” script viewed with IDE - Zoom-in

```
1   REM  *****  BASIC  *****
2
3   function Main()as string
4       ' Get the BASIC code
5       if BasicLibraries.getByName("Standard").hasByName("Module1") then
6           Main = BasicLibraries.getByName("Standard").getByName("Module1")
7       end if
8   end function
9
10  sub message_button(button)
11      doc = ThisComponent
12      doc.getText().setString("Hello World! ~ Using BASIC" + _
13              chr(13) + chr(13) + Main())
14  end sub
15
16  sub clear_button(button)
17      doc = ThisComponent
18      doc.getText().setString("")
19  end sub
20
```

Alternative to doc = ThisComponent

```
REM  *****  BASIC  *****

function Main()as string
    ' Get the BASIC code
    if BasicLibraries.getByName("Standard").hasByName("Module1") then
        Main = BasicLibraries.getByName("Standard").getByName("Module1")
    end if
end function

sub message_button(button)
    'msgbox button.Source.Model.Parent.Parent.Parent.Dbg_Methods

    button.Source.Model.Parent.Parent.Parent.getText().setString("Hello World! ~ Using BASIC" + _
            chr(13) + chr(13) + Main())
    'doc = ThisComponent
    'doc.getText().setString("Hello World! ~ Using BASIC" + _
    '        chr(13) + chr(13) + Main())
end sub

sub clear_button(button)
    doc = ThisComponent
    doc.getText().setString("")
end sub

```

Embedded Basic Demo

- Demo Writer example documents.
- Basic creates the document.

Using Basic to Create a document

- Avoids manually performing repetative tasks.
- Precision in placement of objects. E.g. In Draw application, Lines are accurate to 1/100th of mm.

Using Basic to perform Modelling of a document

- Allows viewing alternative scenarios.
- May turn on and off layers.

Example Basic subroutines creating a Draw document

```
Sub Main
    global_instantiation
    clear_elements ' Except for Control Buttons
    a4_setup
    add_border_layer
    add_grid_layer
    border_line
    border_text_field
    compass
    grid
    grid_supplement
    ' Measurment lines:
    ' Pile centers
    ruler(3000, 4000, M1 *12, 0, 1500, False)
    ' Grid square of 1 meter
    ruler(3000, 4000, M1, 0, 800, False)
    add_pile_layer
End sub
```

Embedded Basic Demo of Draw document

- Demo Draw example documents.
- Basic Creates the Document and then allows modelling.
- Zoom in to show accuracy.

Embedded Basic to Create Calc Document

- Provide scrollbars for modelling parameters.
- Scrollbar programming:
  - Control section to set Position and Size.
  - Model section to set Min Value, Max Value, etc.
  - Two listeners. Separate subroutines
- Use of “with” statement for chart.

Example of setting the Chart properties...

```
Chart.Diagram.XAxisTitle.String = "Month"
Chart.Diagram.YAxisTitle.String = "Amount"
Chart.Diagram.HasXAxisGrid = True
Chart.Diagram.XMainGrid.LineColor = RGB(192, 192, 192)
Chart.Diagram.HasYAxisGrid = True
Chart.Diagram.YMainGrid.LineColor = RGB(192, 192, 192)
Chart.PageBackground set until over-ridden by Area.Fillcolor, etc.
Chart.PageBackground.FillBackGround = True
Chart.PageBackground.FillStyle = com.sun.star.drawing.FillStyle.SOLID
Chart.PageBackground.FillColor = RGB(0, 255, 0)
Chart.Diagram.Wall.FillStyle = com.sun.star.drawing.FillStyle.SOLID
Chart.Diagram.Wall.FillColor = RGB(100, 160, 255)
Chart.Diagram.Wall.LineColor = Rgb(80,80,255)
Chart.Diagram.Wall.LineWidth = 80
Chart.Diagram.Wall.LineStyle = com.sun.star.drawing.LineStyle.SOLID ' DASH NONE
Chart.HasMainTitle = True
Chart.Title.String = "Amortization"
Chart.HasSubTitle = True
Chart.Subtitle.String = "Monthly Interest and Principal"
Chart.HasLegend = True
Chart.Legend.Alignment = com.sun.star.chart.ChartLegendPosition.BOTTOM
Chart.Legend.FillStyle = com.sun.star.drawing.FillStyle.SOLID
Chart.Legend.FillColor = RGB(210, 210, 255)
Chart.Legend.CharHeight = 10
```

Example of setting the Chart properties, using “with” statement...

```
with Chart
     .HasMainTitle = True
     .Title.String = "Amortization"
     .HasSubTitle = True
     .Subtitle.String = "Monthly Interest and Principal"
     .HasLegend = True
     with .Legend
          .Alignment = com.sun.star.chart.ChartLegendPosition.BOTTOM
          .FillStyle = com.sun.star.drawing.FillStyle.SOLID
          .FillColor = RGB(210, 210, 255)
          .CharHeight = 10
     end with
     with .Diagram
          .XAxisTitle.String = "Month"
          .YAxisTitle.String = "Amount"
          .HasXAxisGrid = True
          .XMainGrid.LineColor = RGB(192, 192, 192)
          .HasYAxisGrid = True
          .YMainGrid.LineColor = RGB(192, 192, 192)
          with .Wall
               .FillStyle = com.sun.star.drawing.FillStyle.SOLID
               .FillColor = RGB(100, 160, 255)
               .LineColor = Rgb(80,80,255)
               .LineWidth = 80
               .LineStyle = com.sun.star.drawing.LineStyle.SOLID
          end with
     end with
end with
```

Embedded Basic Demo of Calc document

- Demo Calc example document.
- Basic Creates the Document and then allows modelling.

Questions?

Future Demos: The same, but using Python.

Files to be posted in:

https://github.com/WLUG/meetings/tree/master/2021/2021-08-23