



Python Scripts made simple w/ IDE_utils

2020



Hi

- Alain Romedenne
- alain.romedenne@libreoffice.org
- LibreOfficiant
- LOfficiant

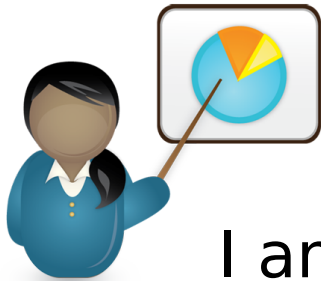


Agenda

- Intro
 - APSO, MRI/xRay extensions
 - IDE's Benefits
 - Steps : Start, ..., Stop
 - IDE_utils features
- IDEs
 - IDE's docstrings
 - Geany, PyZo, PyCharm
- Demos
 - Lorem Ipsum
 - Writer
 - @retry decorator
 - logging
- Links

but wait ...

- must-have APSO extension
- MRI / xRay object browsers



I am your guide to demos

Demo – APSO extension

The screenshot shows the APSO_DEBUG IDE window. The title bar reads "APSO_DEBUG". The toolbar includes buttons for NEXT, EXIT LOOP, STEP IN (highlighted with a mouse cursor), RETURN, RESTART, EDIT FILE, CONSOLE, and QUIT. The file path is `\appdata\roaming\step into\4\user\scripts\python\oslo2020\demo_1.py`. The main editor contains Python code with line numbers 1 through 33. Line 10, `sm = ctx.ServiceManager`, is highlighted in blue. The bottom of the window features a Stack and Local scope panel. The Stack panel shows the current function call: `create_instance(name='com.sun.star.awt.Toolkit', with_context=False)`. The Local scope panel shows variables: `ctx : pyuno object (com.sun.star.uno.XComponentContext...sun.star.uno.XW instance : name : 'com.sun.star.awt.Toolkit' sm : with_context : False`.

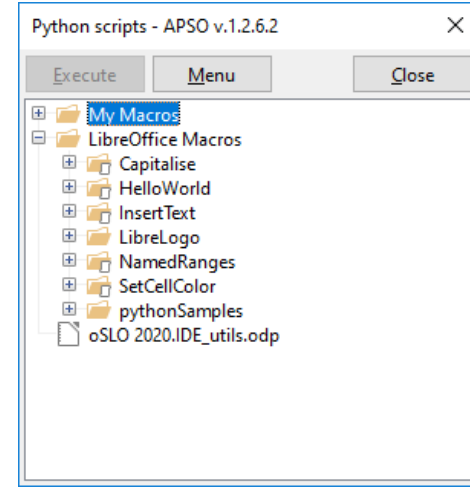
```
1. # coding: utf-8
2. from __future__ import unicode_literals
3.
4.
5. import uno
6. from com.sun.star.awt import MessageBoxButtons as MSG_BUTTONS
7.
8. def create_instance(name, with_context=False):
9.     ctx = XSCRIPTCONTEXT.getComponentContext()
10.    sm = ctx.ServiceManager
11.    if with_context:
12.        instance = sm.createInstanceWithContext(name, ctx)
13.    else:
14.        instance = sm.createInstance(name)
15.    return instance
16.
17. def msgbox(message, title='LibreOffice', buttons=MSG_BUTTONS.BUTTONS_OK, type_msg='infobox'):
18.    """ Create message box
19.        type_msg: infobox, warningbox, errorbox, querybox, messbox
20.        https://api.libreoffice.org/docs/idl/ref/interfacecom_1_1sun_1_1star_1_1lawt_1_1XMessageBoxFactory.html
21.    """
22.    toolkit = create_instance('com.sun.star.awt.Toolkit')
23.    parent = toolkit.getDesktopWindow()
24.    mb = toolkit.createMessageBox(parent, type_msg, buttons, title, str(message))
25.    return mb.execute()
26.
27. def Lorem_Ipsum():
28.    """
29.    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut semper mi sit amet turpis sollicitudin blandit. Aliquam ligula ex, maximus at sapien
30.
31.    Integer aliquam, arcu condimentum venenatis lobortis, felis nisl ullamcorper elit, eu luctus ligula mi sit amet ante. Donec eu ante facilisis, ten
32.    msgbox(Lorem_Ipsum.__doc__)
33.
```

Stack (double click a frame to display its state)

```
Lorem_Ipsum()
└─ msgbox(message='\n Lorem ipsum dolor sit amet, consectetur a.... Ma
└─ create_instance(name='com.sun.star.awt.Toolkit', with_context=False)
```

Local scope (double click to inspect a variable with MRI or Xray)

```
ctx : pyuno object (com.sun.star.uno.XComponentContext...sun.star.uno.XW
instance :
name : 'com.sun.star.awt.Toolkit'
sm :
with_context : False
```



IDE Benefits

- Objects explorer/colorizer
 - Code completion
 - Debugging
 - PEP compliance
 - Test Driven Dev (TDD)
 - Version Control (Git ...)
- and many more ...

IDE_utils steps

- Start libO
- Connect to libO service
- XSCRIPTCONTEXT Adaptor
- Run macro
- Stop libO

+ No compulsory step

Runner context mgr
connect()
surrogate object
...
Runner context mgr

IDE_utils features

- Optional steps
- x-platforms, x-products
- On-demand startup options
- pipe/socket
- Service / context pooling
- KISS

IDE_utils docstrings

```
APSO console
APSO python console [LibreOffice]
3.7.7 (default, May 13 2020, 18:46:31) [MSC v.1916 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license" for more information.
>>> import IDE_utils as ide
>>> help(ide)
Help on module IDE_utils:

NAME
  IDE_utils - Run/Debug LibreOffice Python macros in IDE [#IDE_]

DESCRIPTION
  This module provides:
  i. a substitute to `XSCRIPTCONTEXT` (Libre|Open)Office built-in,
     to be used within `Python IDE`_ such as Anaconda, Geany,
     KDevelop, PyCharm, etc..
  ii. a (Libre|Open)`Runner` context manager with
      - `start`, `stop` paradigms to launch *Office instances
      and to facilitate `setup`, `tearDown` unit testing steps

Instructions:
  1. Copy this module into your <OFFICE>/program/ directory
     OR Include it into your IDE project directory
  2. Include one of the below examples into your Python macro
  3. Run your (Libre|Open) macro from your preferred IDE

Examples:

import uno
def my_1st_macro(): pass # Your code goes here
def my_2nd_macro(): pass # Your code goes here
def my_own_macro(): pass # Your code goes here

g_exportedScripts = my_1st_macro, my_2nd_macro, my_own_macro

# i. Runners.json argument file /OR/ (Libre|Open)Office pipe
```

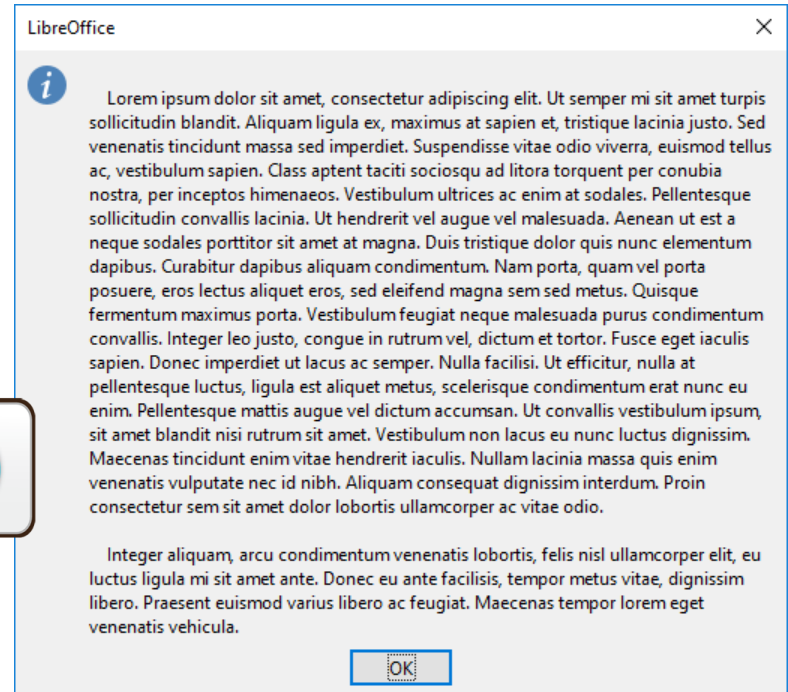
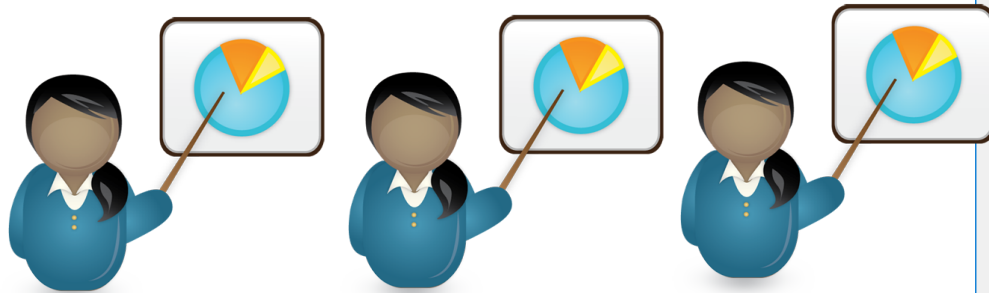
```
C:\Program Files\LibreOffice\program\python.exe
Python 3.7.7 (default, May 13 2020, 18:46:31) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import IDE_utils as ide
>>> dir(ide)
['BootstrapException', 'CONNECT_DELAYS', 'CONNECT_EXCEPTIONS', 'CONNECT_REPORT', 'Dis
posedException', 'NoConnectException', 'RUNNERS', 'Runner', 'ScriptContext', 'XSCRIPT
CONTEXT', 'XScriptContext', '_DEBUG', '_DELAYS', '_EMULATE_OFFICEHELPER', '_INFO', '_
SECONDS', '_builtins', '_cached_', '_doc_', '_file_', '_loader_', '_name_
_', '_package_', '_spec_', '_bootstrap', '_ctx', '_terminate_desktops', 'atexit',
'connect', 'create_service', 'itertools', 'json', 'killall_soffice', 'logging', 'off
icehelper', 'os', 'print_function', 're', 'retry', 'start', 'stop', 'subprocess', 'sy
s', 'time', 'traceback', 'uno']
>>>
```



Demo 1 – Lorem Ipsum API ‘MsgBox’

The IDE starts a named pipe - as a LibreOffice background service -

- connects to it,
- sets XSCRIPTCONTEXT substitute,
- executes the macro and
- stops LibreOffice service



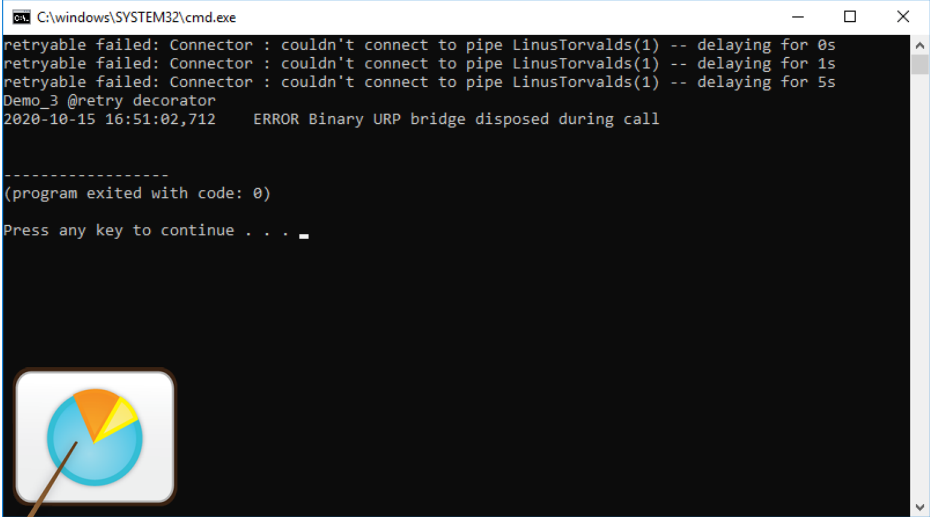
Demo 2 – ‘Hello Writer’

- LibreOffice starts first as a ‘socket’ service
- The IDE connects to port #2017, sets a XSCRIPTCONTEXT substitute and run the script:
 - which creates a Writer doc and updates it
- LibreOffice remains active



Demo 3 – @retry decorator

- The IDE waits for 'LinusTorvalds' named pipe, attempts to connect are reported in console ...
- LibreOffice starts as 'LinusT...' piped service
- The IDE connects, the script executes



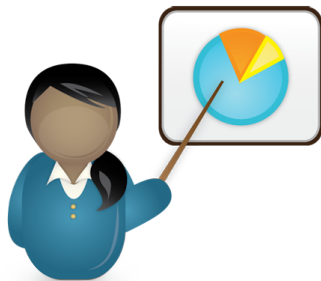
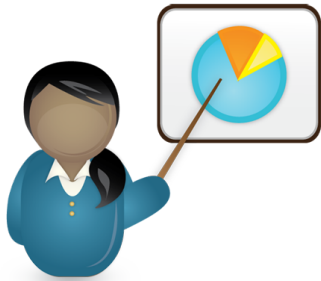
```
C:\windows\SYSTEM32\cmd.exe
retryable failed: Connector : couldn't connect to pipe LinusTorvalds(1) -- delaying for 0s
retryable failed: Connector : couldn't connect to pipe LinusTorvalds(1) -- delaying for 1s
retryable failed: Connector : couldn't connect to pipe LinusTorvalds(1) -- delaying for 5s
Demo_3 @retry decorator
2020-10-15 16:51:02,712 ERROR Binary URP bridge disposed during call

-----
(program exited with code: 0)
Press any key to continue . . .
```



Demo 4 – Failed session

- The IDE looks for LibreOffice as a service
- The IDE attempts to connect twice
- The IDE throws an exception



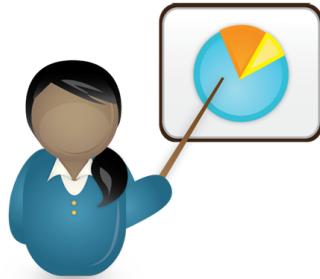
```
C:\windows\SYSTEM32\cmd.exe
retryable failed: Connector : couldn't connect to socket (WSAECONNREFUSED, Connection refused) -- delaying for 1s
retryable failed: Connector : couldn't connect to socket (WSAECONNREFUSED, Connection refused) -- delaying for 3s

retryable failed definitely: [Connector : couldn't connect to socket (WSAECONNREFUSED, Connection refused), Connector :
couldn't connect to socket (WSAECONNREFUSED, Connection refused), Connector : couldn't connect to socket (WSAECONNREFUS
ED, Connection refused)]
Traceback (most recent call last):
  File "Demo_4.py", line 26, in <module>
    with Runner(soffice-pgm) as jesse_owens: # Start/Stop, Connect/Adapt
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 249, in __enter__
    return self._start()
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 330, in _start
    self.pool[key] = self.connect(uno_url=key)
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 317, in connect
    port=None, pipe=None, uno_url=uno_url, flush=True)
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 451, in _connect
    ctx = resolve() # Raises NoConnectException
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 366, in wrapped
    return function(*args, **kwargs)
  File "C:\Program Files\LibreOffice\program\IDE_utils.py", line 450, in resolve
    return resolver.resolve(uno_url)
uno.NoConnectException: Connector : couldn't connect to socket (WSAECONNREFUSED, Connection refused)

-----
(program exited with code: 1)
Press any key to continue . . .
```

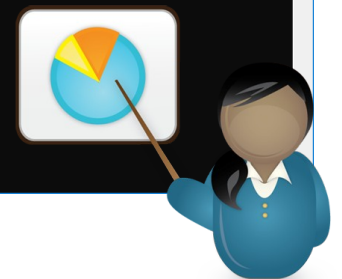
Demo 5 – Logging execution

- The IDE starts LibreOffice as a service
- The IDE connects to it
- The IDE sets XSCRIPTCONTEXT
- The IDE executes the macro
- The IDE stops LibreOffice as a service



```
C:\windows\SYSTEM32\cmd.exe
2020-10-15 15:18:19,004 DEBUG C:\Program Files\LibreOffice\program\
2020-10-15 15:18:19,004 INFO BOOTSTRAPPING (Libre|Open)Office instance..
2020-10-15 15:18:19,004 DEBUG READING.. JSON argument services' list
2020-10-15 15:18:19,004 INFO STARTING (Libre|Open)Office instances..
2020-10-15 15:18:19,019 DEBUG STARTed.. C:\Program Files\LibreOffice\program\soffice.exe--accept=pipe,name=OfficeHelp
er;urp; --nodefault --nologo
2020-10-15 15:18:22,025 INFO CONNECTing to uno:pipe,name=OfficeHelper;urp;StarOffice.ComponentContext
2020-10-15 15:18:22,025 DEBUG {'uno:pipe,name=OfficeHelper;urp;StarOffice.ComponentContext': 13076}
2020-10-15 15:18:22,025 DEBUG STARTed (Libre|Open)Office instances..
2020-10-15 15:18:22,025 INFO CONNECTing to uno:pipe,name=OfficeHelper;urp;StarOffice.ComponentContext
2020-10-15 15:18:22,025 DEBUG READING.. default JSON file services' list
2020-10-15 15:18:22,025 DEBUG ENTERing.. Runner context manager
2020-10-15 15:18:22,025 INFO STARTING (Libre|Open)Office instances..
2020-10-15 15:18:22,025 DEBUG {}
2020-10-15 15:18:22,025 DEBUG STARTed (Libre|Open)Office instances..
Chester Gould was here
2020-10-15 15:18:22,025 DEBUG EXITing.. Runner context manager
2020-10-15 15:18:22,025 INFO STOPping (Libre|Open)Office instances..
2020-10-15 15:18:22,025 INFO EXITing IDE_utils
2020-10-15 15:18:22,025 DEBUG dict_keys(['officehelper'])

-----
(program exited with code: 0)
Press any key to continue . . .
```






e-References

- [LibO Python Wiki](#), [LibO start options](#)
- [LibO API Documentation](#)
- [Geany](#), [Pyzo](#), [PyCharm](#)
- [IDE_utils](#)

IDE demos recap.



			
docstrings	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Lorem ipsum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Hello Writer			<input checked="" type="checkbox"/>
Report connections			<input checked="" type="checkbox"/>
Throwing exception	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Logging execution	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

2020



Thank You



All text and image content in this document is licensed under the Creative Commons Attribution-Share Alike 4.0 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 thereof is subject to trademark policy.