Anatomy of an Apache OpenOffice Extension

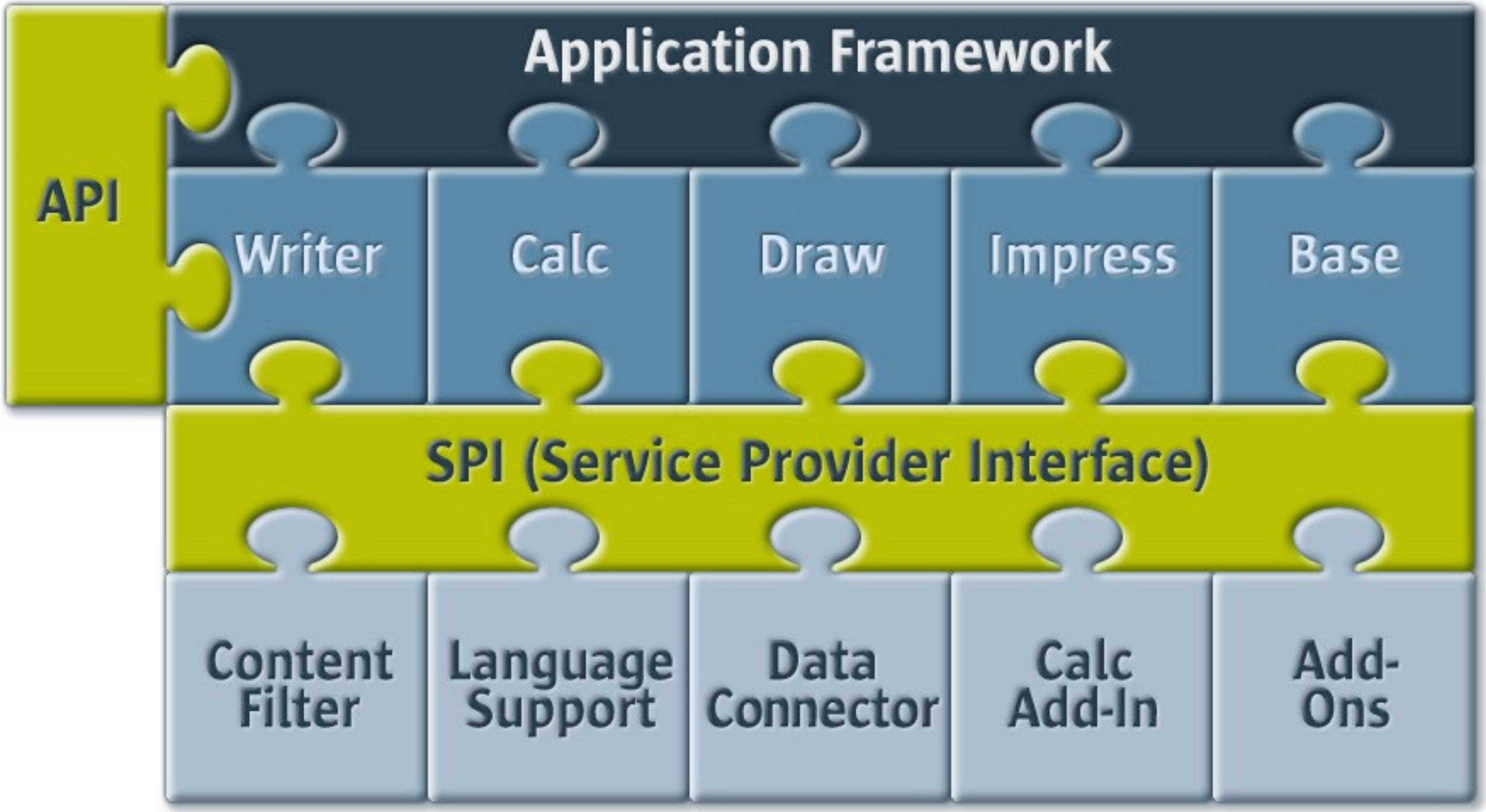
Pedro Giffuni

pfg@apache.org

APACHECON EUROPE

CORINTHIA HOTEL
BUDAPEST, HUNGARY
NOVEMBER 17-21, 2014

AOO Architecture: Jigsaw Puzzle



So how do I develop for Apache OpenOffice

- Develop within the main OpenOffice platform (yes, it is open):
  - Mostly C++ but also some Java.
  - Bugzilla
  - Find and pet a committer.
  - Lot’s of patience
- Extend it:
  - Scripting Framework
    - OpenOffice Basic
  - Templates
  - Extensions
    - C++, Java, Python

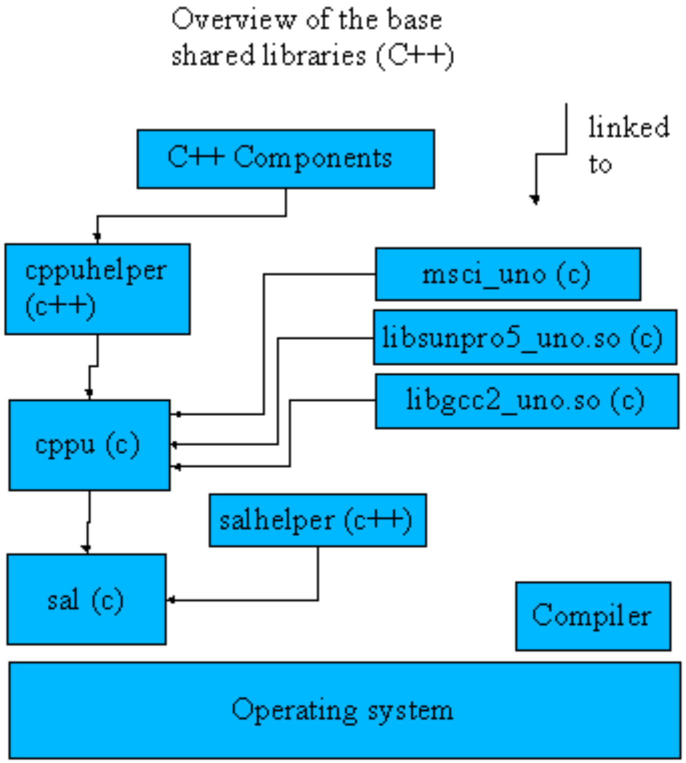
Why do my stuff as an AOO Extension?

- You don’t always want bundle your code into OpenOffice.
- Extensions are more flexible.
- More tooling choices.
- Mostly compatible with AOO derivatives: AOO is the standard for +100 M users.
- Licensing issues: commercial uses welcome.
- Nice Website to promote/share them http://extensions.services.openoffice.org/

Broken Promises

- Like CORBA or DCOM.
- Language Independent.
- Portable (?)
- Can be used without OpenOffice(?).

Universal Network Objects (UNO)



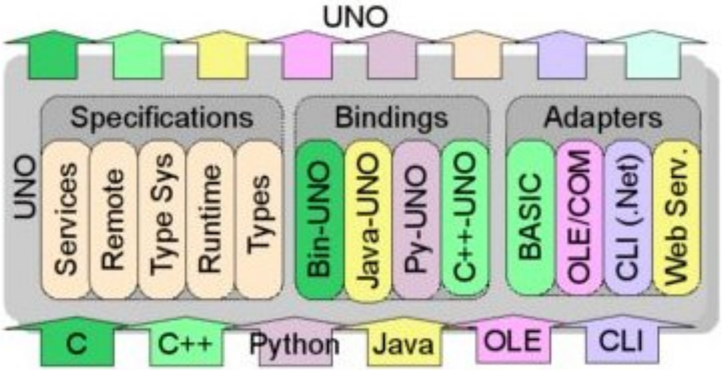
- UNO is a component model that offers inter-operability between different programming languages, different objects models, different machine architectures, and different processes; either in a LAN or via the Internet.
- Each component lives in a **U**no **R**untime **E**nvironment (**URE**). A URE is identified by the implementation language (e.g., C++, Java, Perl, ...) and the current process. There is no performance overhead for components, that are instantiated within the same URE, e.g., in C++, a call from component A to B is just a virtual call. The calls between components from different UREs are bridged by UNO.
- UNO-interfaces are specified in Interface Definition Language (IDL). All UNO-interfaces must be derived from a superinterface, that offers acquire, release, and a queryInterface method.

Type of Extensions

- Apache OpenOffice Add-On
  - It is an external UNO component providing one or more functions through the user interface of Apache OpenOffice.
- Apache OpenOffice Calc Add-in
  - Specialized office extensions that can be used directly in the normal user interface of the Spreadsheet application.
- Apache OpenOffice Component
  - Implementation of at least one service specified in the UNO IDL.
- Apache OpenOffice Client Application
  - Standard client application project with integrated support of the Apache OpenOffice API.

UNO Language Bindings

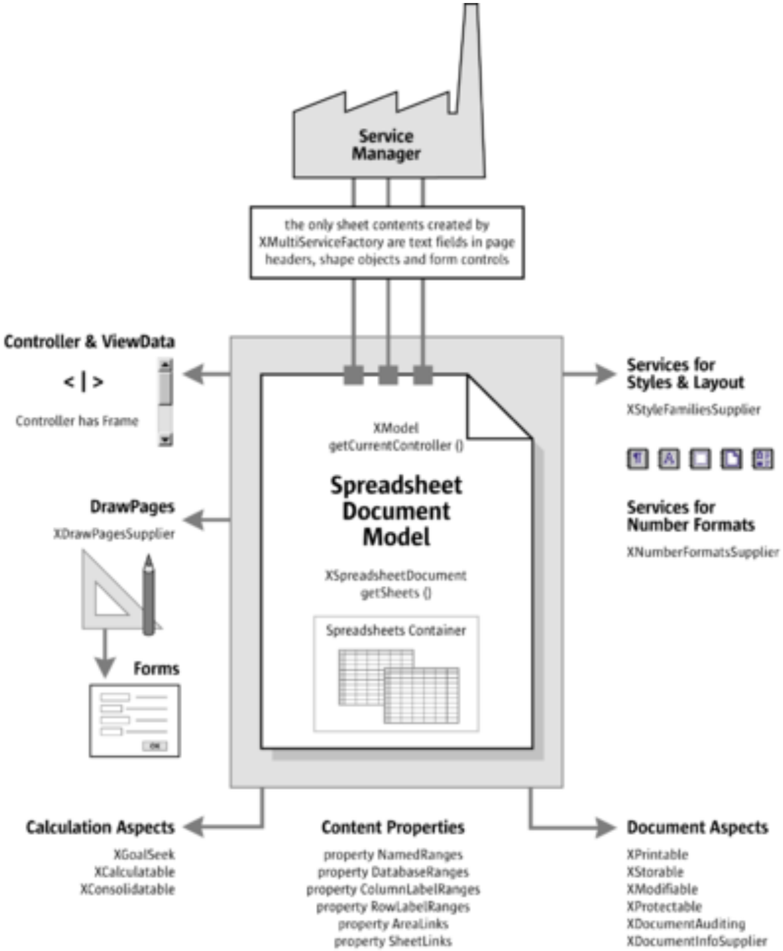
- C++
- Python
- Java - NetBeans



Example – Calc Add-On

NetBeans wizard creates:

src/com/example/AddInImpl.javasrc/com/example/AddIn.idl
src/com/example/XAddIn.idl
src/uno-extension-manifest.xml
registry/data/org/openoffice/Office/CalcAddins.xcu

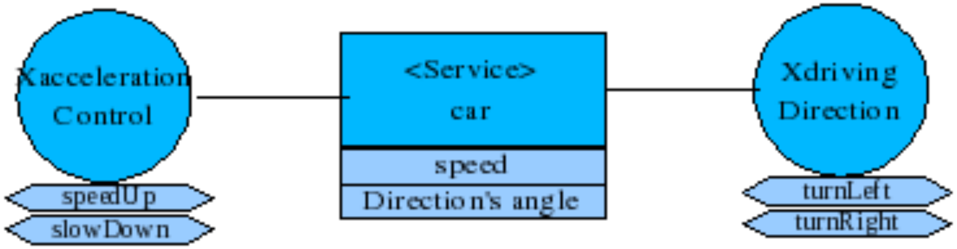


IDL – Interface Data Language

```
// IDL
module my_module
        {
        interface Xsomething
                {
                        void methodone();
                };
        service my_service1
                {
                        // exported interfaces:
                        interface Xsomething;
                };
        interface XsomethingElse
                {
                        void methodTwo();
                        void methodThree();
                };
        service my_service2
                {
                        // exported interfaces:
                        interface XsomethingElse;
                };
};
```

The Interface Definition Language (IDL) is a descriptive language (not a programming language) to describe the interfaces being implemented by the objects. Within IDL, you define the name of the interface, the names of each of the attributes and methods, and so forth. Once you've created the IDL file, you can use an IDL compiler to generate the header files in the C++ programming language.

http://www.openoffice.org/udk/cpp/man/component_tutorial.html



OXT – Office ExTension packages

- Zip container:
  - META-INF/manifest.xml
  - Description.xml
  - Payload (Content): jars or .xcu

Thank You!!