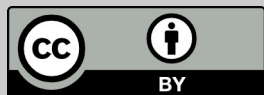




OpenAccess Scripting Language Interfaces Architecture: oaScript

John McGehee Voom, Inc.
June 6, 2011



Copyright © 2011 John McGehee, Voom, Inc. This work is licensed under the [Creative Commons Attribution 3.0 Unported License](http://creativecommons.org/licenses/by/3.0/).

<http://oascript.si2.org>

Scripting Language Pioneers



- oaTcl, included with **OpenAccess**
 - Documented, widely available
 - Manually created wrappers
- **LSI Logic Pyoa**, download from Si2.org
 - Good performance
 - Wrapper code generated from OpenAccess API Doxygen documentation
 - 11,000 lines of C++ and 4000 lines of Python
 - Has become unmaintainable
- **CiraNova** Python
 - Emphasis on PyCells

Design Objectives



- No modification of OpenAccess
- Fidelity to OpenAccess C++ API
- Use standard software development tools
- Unified architecture for every language
- OpenAccess version independent
- Apache license
- Industry-wide development and adoption

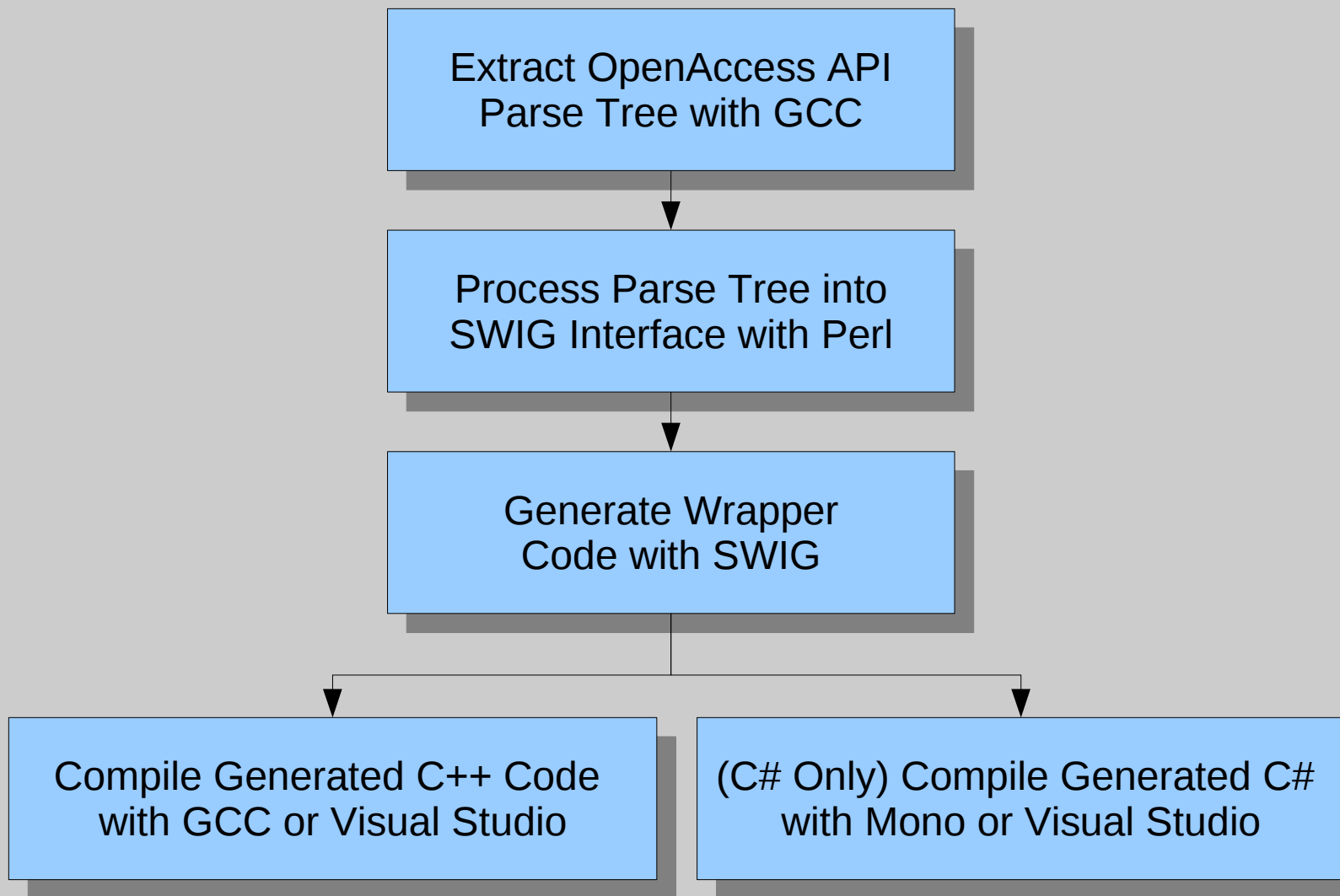


The oaScript Approach

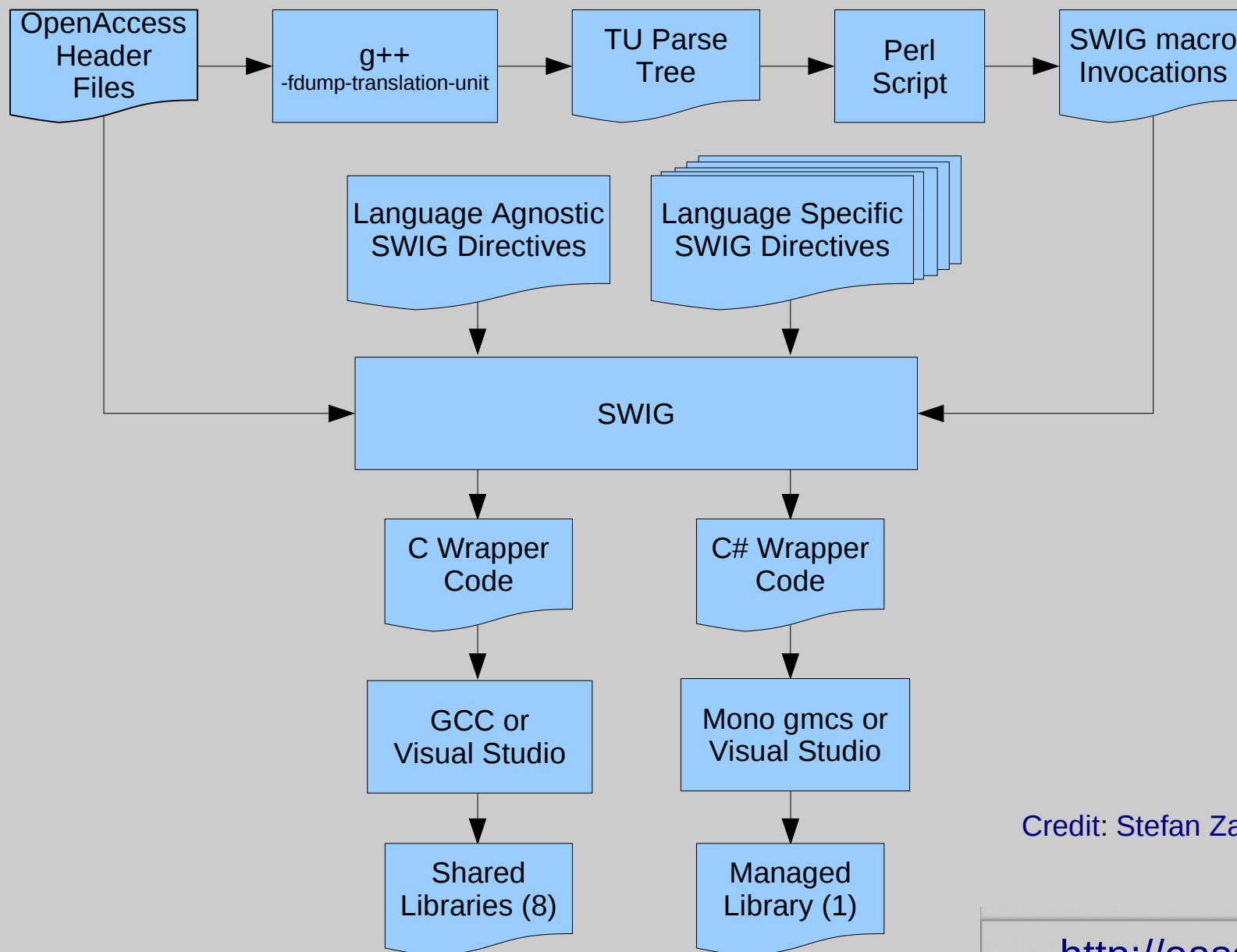
- Standard software tools
 - SWIG
 - GCC
 - Perl script
 - GNU Make, CMake
- Partitioned for expansion
 - Language agnostic code
 - Separate directories for language dependent code
- Perl, Python, Ruby, Tcl, and now C#



Build Flow



Data Flow



Credit: Stefan Zager, AMD

<http://oascript.si2.org>

TU Extractor



- `gcc -fdump-translation-unit` outputs the parse tree in a TU file (GCC 4.1, 4.2 only)
- `gen_macros.pl` processes the TU file into SWIG interface files that identify the usage of classes:

```
COLLCLASS(OpenAccess_4, oaInst, OpenAccess_4, oaBlock)
COLLCLASS(OpenAccess_4, oaShape, OpenAccess_4, oaBlock)
...
```

- The language specific SWIG interface files give definitions to these macros:

```
%define COLLCLASS(ns1, paramtype1, ns2, paramtype2)
...
    oaCollection_##paramtype1##_##paramtype2.__iter__ = ...
...
%enddef
```

SWIG



- Widely used, stable, open source tool that supports 20 scripting languages
- Enhancements in **SWIG 2.0.4** support improved oasPython performance
- SWIG reads an extended C/C++ syntax
 - SWIG directives tell how to wrap the C++ types and methods (.i files)
 - Declarations provided by OpenAccess header files (.h files)
- Creates wrappers needed to access the declarations from the scripting language
 - Lots of C
 - Assisted by a bit of Perl and Python, lots of C#

Compile



- After SWIG, source code to target a particular OpenAccess release is ready
- But it is still platform independent. This is released as the *wrappers* package.
- Compile C code to create libraries that the scripting language interpreter can load
- (C# only) Compile C# proxy classes into a managed DLL
- Fully compiled code to target Red Hat Enterprise Linux 4 and Windows XP is released as the *bin* package

Future Architectural Goals



- SWIG directors for cross-language polymorphism to support oaObserver and oaOccTraverser
- Better OpenAccess version independence
- Lower barrier to entry for new users and additional languages
- Want added features, more influence, new languages?
Join us as a contributor!

For More Information



- See the *Getting Started with oaScript* pamphlet available here and at the Si2 booth
- All oaScript documentation is hosted on <http://oascript.si2.org>