

# Interoperability SIG Webinar 2017:

## Developer Guide for CAPE-OPEN Type Libraries and Primary Interop Assemblies

Michael Halloran,

CO-LaN contractor / UK

# Agenda

- ◆ Review of recent Interoperability SIG work on Type Libraries and PIAs.
- ◆ Requirements
- ◆ Installer Use Cases
- ◆ Installer Design
- ◆ Demonstrations
  - Using the .msm
  - Using the .msi with NSIS
- ◆ Links
- ◆ Q&A

# Review of recent work on TLB installers

- ◆ CO-LaN has delivered a TLB installer since 2007
- ◆ Interoperability SIG restructured the CAPE-OPEN IDL to make it easier to extend with new versions.
- ◆ M&T raised the issue of .NET interoperability
  - developers requiring a PIA for .NET implementations were creating their own
- ◆ In 2014 the Interoperability SIG began working on a standard PIAs
- ◆ How to ensure use of a single CO-LaN defined PIA?
- ◆ How to ensure use of a single CO-LaN defined TLB (consistent with PIA)?
- ◆ How to ensure that CAPE-OPEN definitions survived the software lifecycle on an end-user's machine?

# Requirements

- ◆ Ensure the foundation for interoperability by providing a single shared component to distribute and install TLBs, PIAs and CAPE-OPEN-specific Registry entries
- ◆ Ensure continued operation of the shared component through the software lifecycle on a user's machine
- ◆ Support 64- and 32-bit implementations
- ◆ Deliver the shared component to software vendors for inclusion in their CAPE-OPEN software
  - Not just an example
  - Reduces developer's part to install packaging.

# TLB Installer Use Cases

- ◆ Software vendor developing CAPE-OPEN compliant software
- ◆ Software vendor with CAPE-OPEN compliant software already installed
- ◆ Software vendor delivering CAPE-OPEN software using Windows Installer
- ◆ Software vendor delivering CAPE-OPEN software using other installation technologies
- ◆ Manual installation by an end-user

# Software vendor developing CAPE-OPEN compliant software

- ◆ **As a Software Vendor developing CAPE-OPEN compliant software using COM and/or .NET, I need access to the Type Libraries and PIAs containing the interface definitions of the CAPE-OPEN standard.**
- ◆ **I want to use the correct Type Libraries and PIAs when developing CAPE-OPEN software.**
- ◆ **I want the CAPE-OPEN definitions to be automatically included and easily identifiable in the lists of COM and .NET components that can be referenced from my software projects in MS Visual Studio.**

- ◆ **As a Software Vendor with CAPE-OPEN compliant software already installed using earlier Type Libraries installers from CO-LaN, or using a Type Library installer developed independently, I want to be sure that uninstalling the earlier packages and installing the current one will not break any CAPE-OPEN software from any vendor installed using the new Type Library and PIA installers.**

- ◆ As a software vendor who has developed CAPE-OPEN compliant software using the CO-LaN Type Libraries and PIAs, I want to include the same Type Libraries and PIAs as part of the installation of my software.
- ◆ Since I'm already using a tool for creating Windows Installer packages, I want to use a mechanism that will easily integrate with that tool. My installer may allow per-user and per-machine installations so the CAPE-OPEN installer must allow the same choice at installation time.
- ◆ When my installation package is uninstalled, I want the CAPE-OPEN Type Libraries and PIAs and all their registry entries to be uninstalled as well, unless they are being referenced by other software packages.



## Software vendor delivering CAPE-OPEN software using other installation technologies

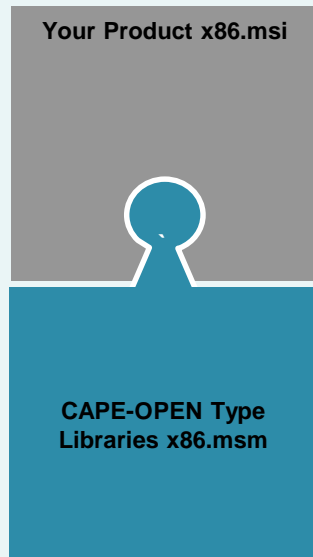
- ◆ As a software vendor who has developed CAPE-OPEN compliant software using the CO-LaN Type Libraries and PIAs, I want to include the same Type Libraries and PIAs as part of the installation of my software.
- ◆ I cannot use Windows Installer Merge Modules so I need a standalone installation package which can be invoked by my installer and installed silently. My installer may allow per-User and per-Machine installations so the CAPE-OPEN installer must allow the same choice at install time.
- ◆ When my installation package is uninstalled, I want the CAPE-OPEN Type Libraries and PIAs and all their registry entries to be uninstalled as well unless they are being referenced by other software packages.

- ◆ **As a user of CAPE-OPEN-compliant software I need to be able to install CAPE-OPEN Type Libraries and PIAs to repair CAPE-OPEN software interoperability. In an enterprise environment, I need to be able to install on a per-User basis.**

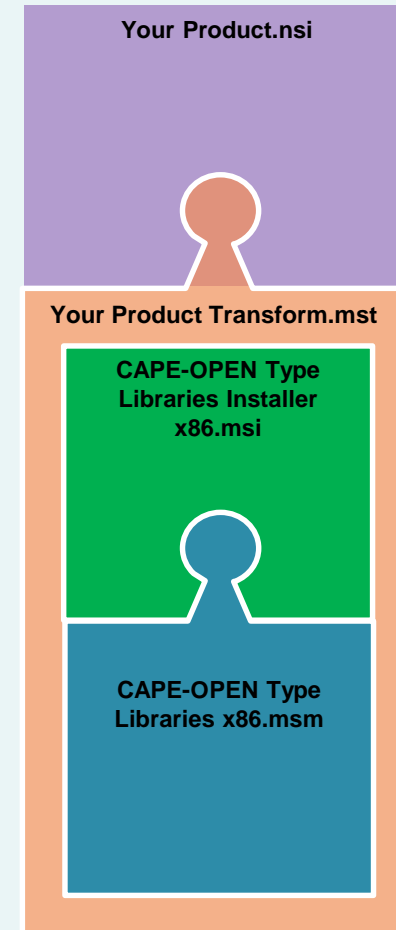
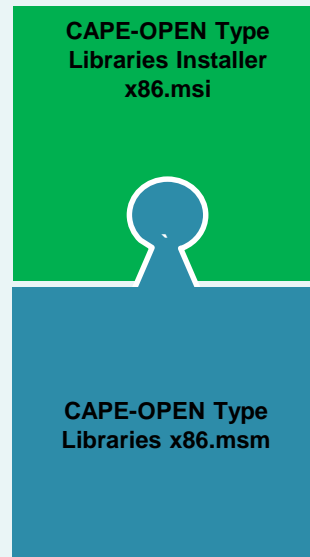
# TLB Installer Design

- ◆ **Uses Windows Installer**
- ◆ **Shared Component designed for multiple install and uninstall**
  - **Relies on Windows Installer Reference Counting**
- ◆ **Delivered as .msm for reuse with Windows Installer authoring tools**
- ◆ **Delivered as .msi for scripted installation and for end-user installation**
  - **Using a “Transform” to create a new “Product” ref count**
- ◆ **End-user installation supported as last resort repair option.**
- ◆ **Use “Single Package Authoring” to allow per-user and per-machine installs**

# Combining Installer Components



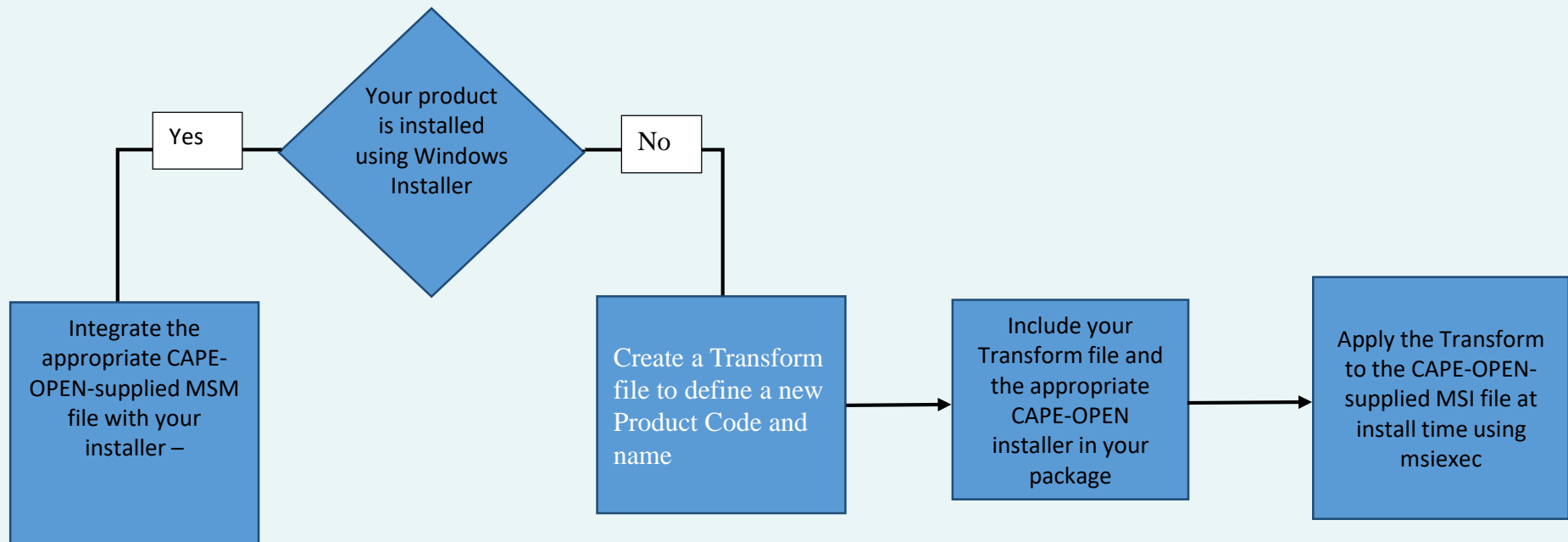
Windows Installer scenarios



Scripted Installer scenario

# TLB Installer Demonstrations.

## ◆ Type Libraries / Primary Interop Assemblies installers

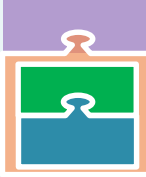


# Demonstration: Using the .msm

- ◆ **Objective:** Show how to include a reference to the .msm in an existing installer built using Windows Installer.
- ◆ **Example will use WiX but principles apply to Installshield and other Windows Installer tools equally.**
- ◆ **Steps:**
  - **Edit WiX scripts to add reference to .msm**
  - **Compile to .msi**
  - **Test**



# Demonstration: Using the .msi with NSIS



- ◆ **Objective: Show how to invoke the .msi in a scripted installer without using Windows Installer.**
- ◆ **Example will use NSIS but principles apply to other tools equally.**
- ◆ **Steps:**
  - **Identify .msi and options required.**
  - **Create Transform using Orca tool**
  - **Edit .nis script to add execution of msiexec command using the Transform for install and uninstall**
  - **Test**



HOME · CAPE-OPEN STANDARD · PARTICIPATE · ACTIVITIES · RESOURCES · NEWS & EVENTS · ABOUT US



# Experience CAPE-OPEN as a Software Developer

V1.0

[Click here](#)

V1.1

[Click here](#)

## Learn about CAPE-OPEN



**COLTT**  
COLTT is a tool designed to assist in the task of achieving interoperability between CO compliant software.



**TLB/PIA Installer**  
Provides the CAPE-OPEN Type Libraries and Primary Interop Assemblies.



**Training sessions**  
Learn about CAPE-OPEN experts



**Forum**  
Join and discuss CAPE-OPEN within a forum

## Specifications News

14 JUL 2016

**Persistence Errata & Clarifications approved for release**

Clarifies persistence related requirements for PMCs and PMEs

[+ More](#)

## Developer Tools

### News

27 JUL 2016

**COLTT 2.3.1 released**

Upgrade to Microsoft Visual Studio 2013 redistributables

[+ More](#)



# Links

## ◆ Repository Hosting area for the TLB/PIA Installers:

[https://colan.repositoryhosting.com/trac/colan\\_coidl/wiki](https://colan.repositoryhosting.com/trac/colan_coidl/wiki)

- Goto Downloads to find the Developer Guide and Installation kits.
- Goto New Ticket to raise an issue or ask a question about the installer (or email [support@colan.org](mailto:support@colan.org))

## ◆ Orca download:

[https://msdn.microsoft.com/en-us/library/windows/desktop/aa370557\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa370557(v=vs.85).aspx)

## ◆ Single Package Authoring:

[https://msdn.microsoft.com/en-us/library/windows/desktop/dd408068\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd408068(v=vs.85).aspx)

# Q & A