

# Swift Working Group

Academy Software Foundation (ASWF)

Wabi Foundation. 2.7.24.

# Purpose

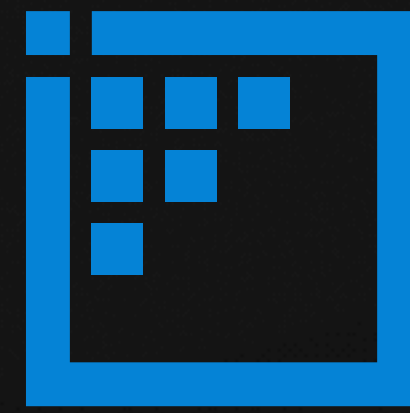
## Swift Working Group

- Providing both Swift and C/C++ interoperability between the many projects across the **ASWF** landscape while exploring the possibility of interoperability with other languages like **Rust** and **Python**.
- Allow existing libraries across the industry to be easily extensible, safer, and easier to use.
- Empowering software development both within and outside the industry through easy adoption of industry software through harnessing the capabilities of the Swift Package Manager (**SPM**).
- **SPM** makes adding a library from other projects as simple as copying a library's GitHub link and pasting it into its package dependency section.

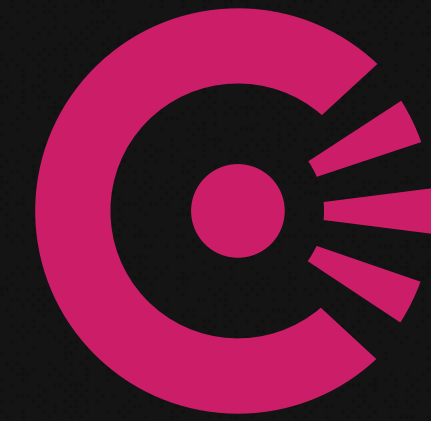
# Honorable Mentions

## Swift Working Group

- Built 13 projects across the ASWF without a single line of **CMake**, through the tooling of SPM, enabled by C++ interoperability in **Swift 5.9**.
- Required zero or very minor changes to existing libraries for this to work.
- All available as a Swift package dependency from a single monolithic SPM package.



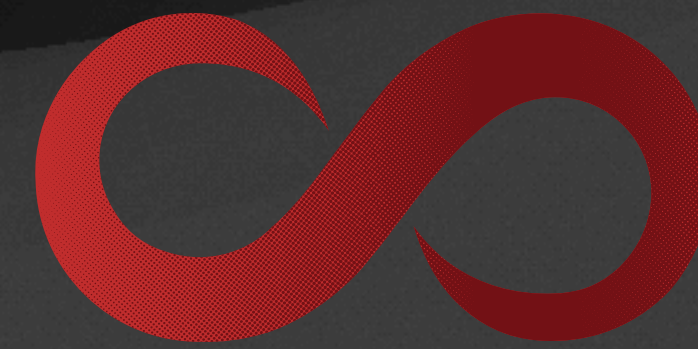
OpenImageIO



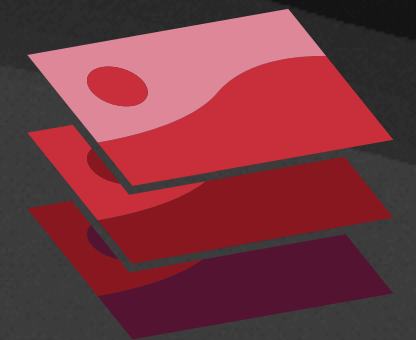
OpenColorIO



OpenSubdiv



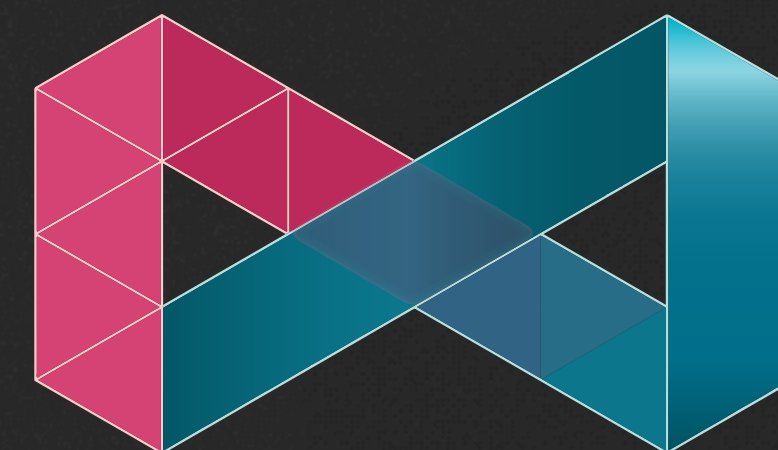
OpenVDB



OpenEXR



ALEMBIC



MATERIALX



Ptex

0%

CMake

**100%**

**Swift Package Configuration**

# Goals

## Swift Working Group

- Collaboration with the **ASWF** to allow existing projects and libraries to be consumed by the Swift Package Manager, in the form of Package.swift files at project roots.
- Availability of plugins, and other tooling to aid in bringing existing libraries to **SPM**.
- Creating safe, easy to use Swift APIs of these libraries, or allowing for interesting paradigms such as declarative APIs.
- Cross platform support across **macOS**, **iOS**, **visionOS**, **Linux**, **Wasm**, and **Microsoft Windows**.



# Non-Goals

## Swift Working Group

- Existing C/C++ libraries should not be modified, or modified very little (acceptable minor change would be in the form of moving C++ includes out of namespace blocks) or adding clang attributes to existing classes and types.
- Rewriting existing C/C++ or Python code into Swift.
- Supporting versions of Swift prior to Swift 5.9 for CXX libraries.

# Deliverables

## Swift Working Group

- Universal Scene Description (**USD**) for the Swift programming language, **SwiftUSD**.
- Monolithic Swift Package dependency (**MetaverseKit**) which successfully builds 13 existing ASWF projects, including dependencies of these libraries such as **TBB** and **Python**.
- Make **MetaverseKit** monolithic Swift package obsolete through industry adoption of Swift Package.swift files at project roots, or via submodules if project maintainers do not wish to add Package.swift files to the root of their projects.





# Support

## Swift Working Group

- **Wabi Foundation (@wabiverse)**, an American nonprofit organization dedicated to fostering the evolution of Swift and the languages in which it interoperates with across the industry and an open-source Metaverse.
- Dedicated engineers at the direction of the industry, to aid in the adoption of the Swift programming language, to aid in solving challenges within production settings, and the development of tooling and library support at the benefit to both the Swift programming language and to the languages it interoperates with (**C/C++**, potential for **Python** and **Rust** interoperability with Swift in the future).