design to bind

Nick Porcino
Pixar Animation Studios
13 Jan 2022
OpenAssetIO wg meeting presentation

OpenTimelineIO and other languages

or how I spent my summer vacation

OpenTimelineIO-Swift

OpenTimelineIO-Java

OpenTimelineIO-C

OpenTimelineIO-Cpp

OpenTimelineIO-Python

Boy howdy did we learn a lot

Also, rustaceous crab people, we're ready for you ;)

abi

the interface to your interface

- things to consider
  - when is foo not a foo?
    - c++
    - gcc simd returns
    - ambiguity
  - dlopen
    - mystic nonsense? (C++/Rust name mangling)
    - type free for all? (C/everything ambiguous names)
  - structured interfaces
    - grandpa had COM
    - handshaked interfaces not bad, actually

data types

plain old data FTW

- cardinal types ⭐⭐⭐⭐
- recursive structs with cardinals ⭐⭐⭐
- types with initialization ⭐⭐
- compound types with initialization ⭐

Ranking scale:
⭐⭐⭐⭐ The world is your oyster
⭐⭐⭐ cruisin' down the highway
⭐⭐ sigh, if we must
⭐ wait, is this a talk about language bindings, or a new special hell?

lifespan

data and objects

- wither malloc?
  - override an allocator
    - via linker tricks, common but scary
  - be explicit
    - allocator as a function

- ownership
  - cart before the horse?
    - python
  - intrusive?
  - borrowed/released?
  - ref counted?
  - good intentions?
  - managers?

- grouping
  - I have a hammer
    - Rust, C++, Python
  - what does the *problem domain* demand?

design to bind

by lifespan
by dependency

making it

idl vs bindgen vs hand crafted

- idl -> generators -> target
  - high up front effort
  - low maintenance
- bindgen -> heuristic translation -> target
  - low up front effort
  - high maintenance
- handcrafted -> what's best -> target
  - high up front effort
  - high maintenance

versioning and distribution

is this what I say it is?

soversion
namespacing
side x side

how is the library presented to the binder?
how is the binder presented to the consumer?

marshalling

how do I get data into my interface?

- functionally?
    - api
- struct layouts?
    - like protobuf
- named response?
    - like json

reflection

what can you tell me about yourself?

- these are the interfaces you can talk to me with
  - this is what you can give these interfaces
- these are the objects you can ask me about
  - these are the interfaces that work with that object
  - these are the properties of this object
    - these are the interfaces that work with that property
- these are some bytes you can copy

reification

I can put that in a data structure for you

- M? MV? MVC? MVVC? MC?
- also save and load
- see also reflection

fragility and safety

weak, or a rugpull?

- two sides of the same coin
  - here's a docker environment
    - because no one can figure out how to build the build system
  - I bundled everything
    - compiler, musl, every shim
    - cargo vendor
  - some dev in Nebraska (xkcd)
- abuse/misuse of the api
- test suites
- continuous integration
- coverage
- fuzzing

environment

are there drop bears in banyan trees?

- where will the library function?
  - dashboard
  - feature app
  - native host in dcc
  - cli piping
  - web host
  - cloud host

- scripting
  - sure python
  - lua
  - dsl
  - my own wacky thing
  - more than one of the above AT THE SAME TIME

- wasm
  - the high frontier
  - the new docker
    - node -wasi foo.wat
  - dependencies
    - can explode the payload

api

the fun bit

don't even think about it until you've hit everything else
starting with the api and figuring out everything else after will cost 10x as much