Hybrid application

Making C# and C++ work together

C++ benefits

- Ability to write performance efficient code
- Low-level manipulation (Assembler, GPU)
- Full memory management
- Common and rare platforms support (via rare drivers)
- Easy to make reverse engineering painful
- Cross platform

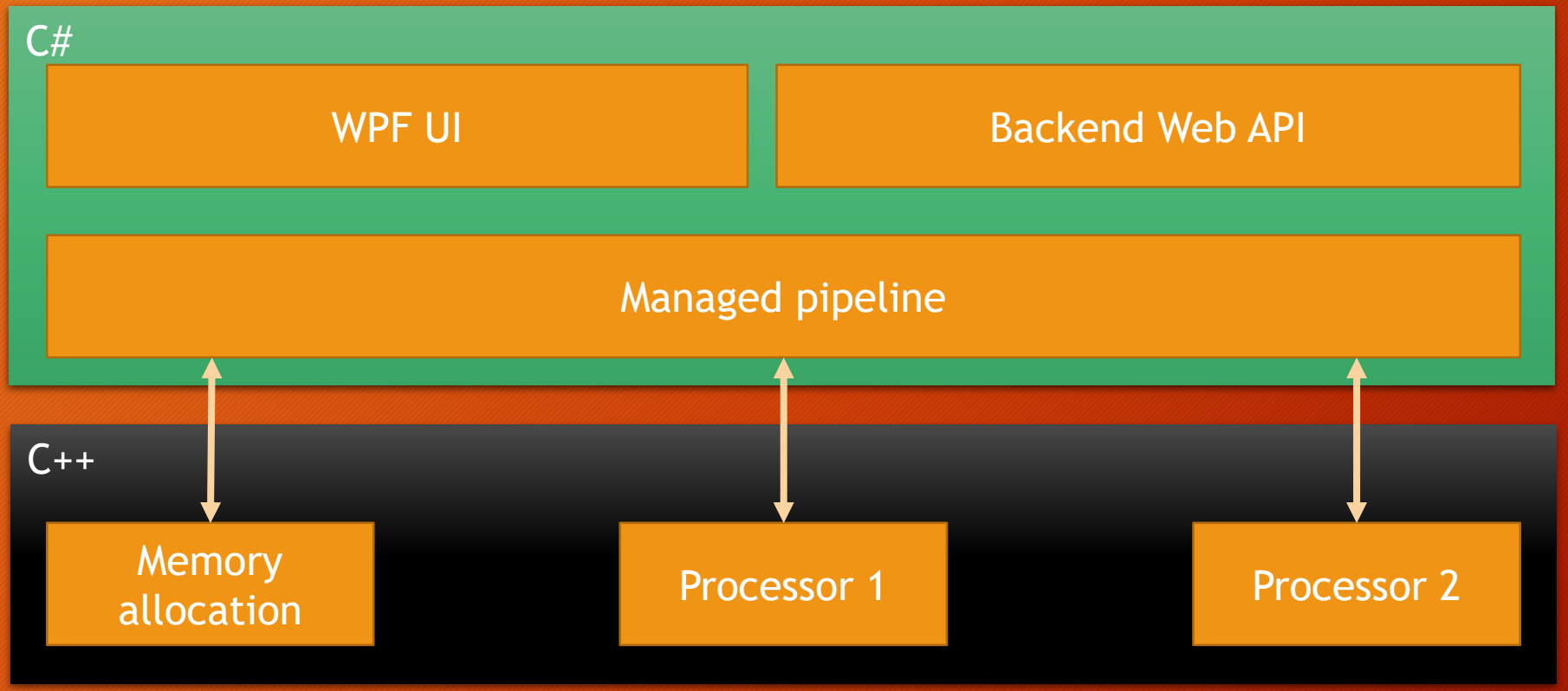
C# benefits

- Easy and pleasantly developing (with existing tools)
- Application high level developing (Communication, UI)
- Automatic garbage collection
- Cross platform (somehow)

Hybrid approach

- Every task can be done easy with proper tool so
  - Try to get most of language advantages
  - Collect smallest possible amount of disadvantages
- Leave development process controllable

Common hybrid example



Code example

Neural link imitation with managed pipeline and unmanaged coefficient calculation.

Ability to visualize impulse route.

References

- Microsoft Interface Definition Language
- Active template library (ATL)
- Exporting from a DLL Using __declspec(dllexport)
- Interop Marshaling
- Graphviz – graph visualization
- QuickGraph – graph manipulation library
- How to have a Project Reference without referencing the actual binary
- What is a TLB file?