# Binding

* Best: .framework or .aar as they contain resources
  + .a or .jar cannot hold resources
* [Verify] in iOS is compiler error, and only meant for devs
* Async can be done using TaskCompletionSource<T> in .NET
  + Used for event-based and callback-based
  + Take advantage of platform cancellation
* Additional members can be added via partial classes or extension methods
* Dependencies can be added as NuGet / assembly references

|  |  |  |
| --- | --- | --- |
|  | Android / Wear / TV / etc | iOS / tvOS / macOS / watchOS |
| Style | * XML transform on generated api.xml   + Just plain old XPath | * > sharpie bind -o OutDir -n NS -sdk SDK -scope . \*.h * > sharpie xcode -sdks * Sharpie and then mods   + Just plain old C#   + Frameworks     - delete the main header and then work as normal |
| Helps | * Remove internal packages to reduce * Java allows private classes with both public nested and public derived types * Not good generic support yet * Errors are usually missing/excluded dependencies | * NSArray can be NSObject[] * [DisableDefaultCtor] |
| Events | * Listeners * C# does not support anonymous class implementations * “eventName” * “argsType” | * Delegates * WeakDelegate / IDelegate * [Delegates=[“”], Events=[typeof]] * [EventArgs] |
| Async | * Usually callback types can be converted into delegate-based callback types | * [Async]   + Not on [Protocol], use extension methods * Void methods * Last param is Action   + One or both NSError and T |
| Rename | * <attr> * “managedName” * Namespaces * Xamarin.Android Bindings Generator | * Rename directly * More common in iOS due to the obj-c * Delegates usually have “sender” first so is actuall the second part * [BaseType(, Name = “”)] |
| Visibility | * “obfuscated” and “visibility” attributes | * [Internal], [Private], attribute-based |
| Properties | * “propertyName | * Directly changed |
| Coolness |  | * initWithString and initWithText   + Change from Constructor   + Make [Internal]/[Private]   + Create extra constructor     - : base(NSObjectFlag.Empty)     - base.InitializeHandle() |
| Native | * .jar/.aar | * Static+dynamic frameworks / .a |
| Dependencies | * Maven / gradle has the dependencies * Multiple .jar files, only bind what is needed | * Podspec has all the linker / dependency info * Multiple .a/.framework files, but only needed API |
| Decompiler | * JD-GUI |  |

# PCL

* [assembly: ReferenceAssembly]
* Type forwarding

# NuGet

* Bait-and-switch

# Component

* Samples / docs
* Sell
* Libraries
  + Shell with NuGet
  + Contains assemblies