

C++BUILDER FOR IOS

預覽發表會

Agenda

- ◎ 簡介
- ◎ C/C++：跨平台開發的霸主
- ◎ 萬眾矚目：C++Builder For iOS
- ◎ 一份程式碼同時開發移動和桌面應用程式
- ◎ C/C++程式碼和函式庫
- ◎ 結論

C/C++：跨平台開發的霸主

Bjarne Stroustrup



- ◎ C with Objects (1979)
 - Modeled OO after Simula and Ada
 - But syntax and RTL based on C
 - Classes
 - Inheritance
 - Inlining
 - Default arguments
 - Type checking
 - CFront compiler

AT&T C++

- ◎ Name changed to C++ in 1983
 - Virtual functions
 - Operator overloading
 - References
 - Constants
 - // Single line comments
- ◎ AT&T commercial c-front compiler

New Turbo C++ Professional

When the object is programming



(OOP) is programming in the '90s. It's the next step after structured programming and is the best way to write applications. So Borland combined the power of OOP with the efficiency of C to produce new Turbo C++ Professional.

And Turbo C++ Professional is the first Turbo-charged native code C++ compiler that brings Object-Oriented Programming to your PC. Since Turbo C++ Professional also compiles ANSI C code, you can be productive with C now, and move to C++ at your own pace.

Environment ++

The best compiler deserves the best environment, and our new Programmer's Platform™ environment makes you more productive. It features overlapping windows and mouse support. And sports a new multi-file editor, an integrated debugger, and a smart project manager. Its advanced open architecture lets you integrate the tools you need to feel right at home.

VROOMM adds room

VROOMM™ (Virtual Runtime Object-Oriented Memory Manager) lets you break

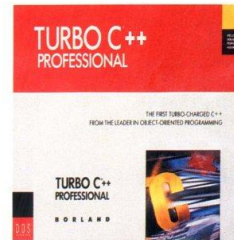
the 640K barrier. Just select the application code you want to overlay, and VROOMM does the rest—swapping modules on demand. It's fast, easy, automatic.

Another +

Turbo C++ Professional gives you all the tools you need to build fast, reliable C++ programs.

Turbo Debugger® 2.0 debugs your object-oriented programs. This powerful new version is the first and only debugger to support *reverse execution*. Letting you step backwards through your code to find the bugs you might have missed.

New Turbo Profiler™, the world's first interactive profiler, displays histograms of your program's performance. With it, you



can easily spot execution bottlenecks, and see where improvements or redesign of your code will yield maximum performance gains.

And Turbo Assembler® 2.0 lets you replace time-critical segments of your code using the world's fastest MASM-compatible assembler.

Turbo C++ Professional Compiler

- C++ conforming to AT&T's 2.0 specification
- C++ class libraries
- Full ANSI C compiler
- VROOMM overlay manager
- Complete documentation and tutorials

Programmer's Platform

- Open architecture for integration of your own tools
- Overlapping windows with mouse support
- Multifile, macro-based editor
- Smart project manager provides visual MAKE
- Integrated debugging and hypertext help

Turbo Debugger 2.0

- Class hierarchy browser and inspectors
- Reverse execution provides "true" undo
- 286 protected-mode and 386 virtual-mode debugging
- Keystroke record and playback

NEW Turbo Profiler

- Displays histograms of program execution
- Tracks call history, overlays, interrupts, file I/O

Turbo Assembler 2.0

- Multipass assembler with NOP squishing and 486 support

Special Introductory Offer

The suggested retail price for Turbo C++ Professional is \$299.95 (\$199.95 for Turbo C++). For a limited time, Borland is offering its dealers and distributors special introductory discounts.* So be objective, and SEE YOUR DEALER or call Borland** at 1-800-331-0877 now!

B O R L A N D

Code: MC66

*Offer expires July 31, 1990 or while supplies last. Offer good in United States and Canada only. **Special discounts for registered Turbo C++ owners are available from Borland. Mail orders to: Borland P.O. Box 660001, Scotts Valley, CA 95067-0001. For orders outside the U.S., call (408) 438-5300. Turbo C++, Turbo Debugger, Turbo Profiler and Turbo Assembler are trademarks or registered trademarks of Borland International, Inc. Copyright © 1990, Borland International, Inc. All rights reserved. B-1333

C98

⦿ Language

- Multiple Inheritance
- Templates
- Abstract Classes
- Exceptions
- Namespace
- Typecasts
- bool

⦿ Library

- Strings
- Streams
- Algorithms
- Containers
- Iterators

C++03 & TR1

- ◎ C++03 was a bug fix release for implementers
 - No change to the specification
- ◎ TR1 was library focused
 - Much of the standard library began its life in Boost
- ◎ Regular expressions
- ◎ Smart pointers
- ◎ Hash tables
- ◎ Random number generators
- ◎ Reference wrapper
- ◎ Function objects
- ◎ Tuple type
- ◎ Fixed size array

C++11

- ◉ Rvalue references and move constructors
- ◉ constexpr - Generalized constant expressions
- ◉ Core language usability enhancements
- ◉ Initializer lists
- ◉ Uniform initialization
- ◉ Type inference
- ◉ Range-based for-loop
- ◉ Lambda functions and expressions
- ◉ Alternative function syntax
- ◉ Object construction improvement
- ◉ Explicit overrides and final
- ◉ Null pointer constant
- ◉ Strongly typed enumerations
- ◉ Right angle bracket
- ◉ Explicit conversion operators
- ◉ Alias templates
- ◉ Unrestricted unions
- ◉ Variadic templates
- ◉ New string literals
- ◉ User-defined literals
- ◉ Multithreading memory model
- ◉ Thread-local storage
- ◉ Explicitly defaulted and deleted special member functions
- ◉ Type long long int
- ◉ Static assertions
- ◉ Allow sizeof to work on members of classes without an explicit object
- ◉ Control and query object alignment
- ◉ Allow garbage collected implementations
- ◉ Threading facilities
- ◉ Tuple types
- ◉ Hash tables
- ◉ Regular expressions
- ◉ General-purpose smart pointers
- ◉ Extensible random number facility
- ◉ Wrapper reference
- ◉ Polymorphic wrappers for function objects
- ◉ Type traits for metaprogramming

C/C++應用程式開發框架

Application Frameworks

◎ C++ (OWL)

- Windows Encapsulation
- Inheritance
 - Multiple-Inheritance
- v-tables
- Fields
- Pure-code generation

◎ Delphi/C++ (VCL)

- Windows Encapsulation
- Delegation
- Method Pointers (closures)
- Properties
- Text file form
- Streaming-system
- RTTI

C/C++市場應用



C and C++ are everywhere



C++ is The Choice for Clients Devices

- ② #2 and #4 languages for SourceForge open source projects
- ② Top commercial software is written in C++
 - Microsoft Word, Adobe Photoshop, Google Chrome, Firefox, Mac OS X, iOS, Oracle database, Microsoft SQL Server, MySQL, Games
- ② Top web companies use C++ – Facebook, PayPal, Amazon

C/C++ Consistent Top Languages

C++	9.698%	+1.61%	A
PHP	5.732%	-0.36%	A
C#	5.591%	-1.73%	A
(Visual) Basic	5.032%	-0.01%	A
Python	4.062%	+0.45%	A
Perl	2.182%	+0.10%	A
Ruby	1.739%	+0.24%	A

Language of choice for software development

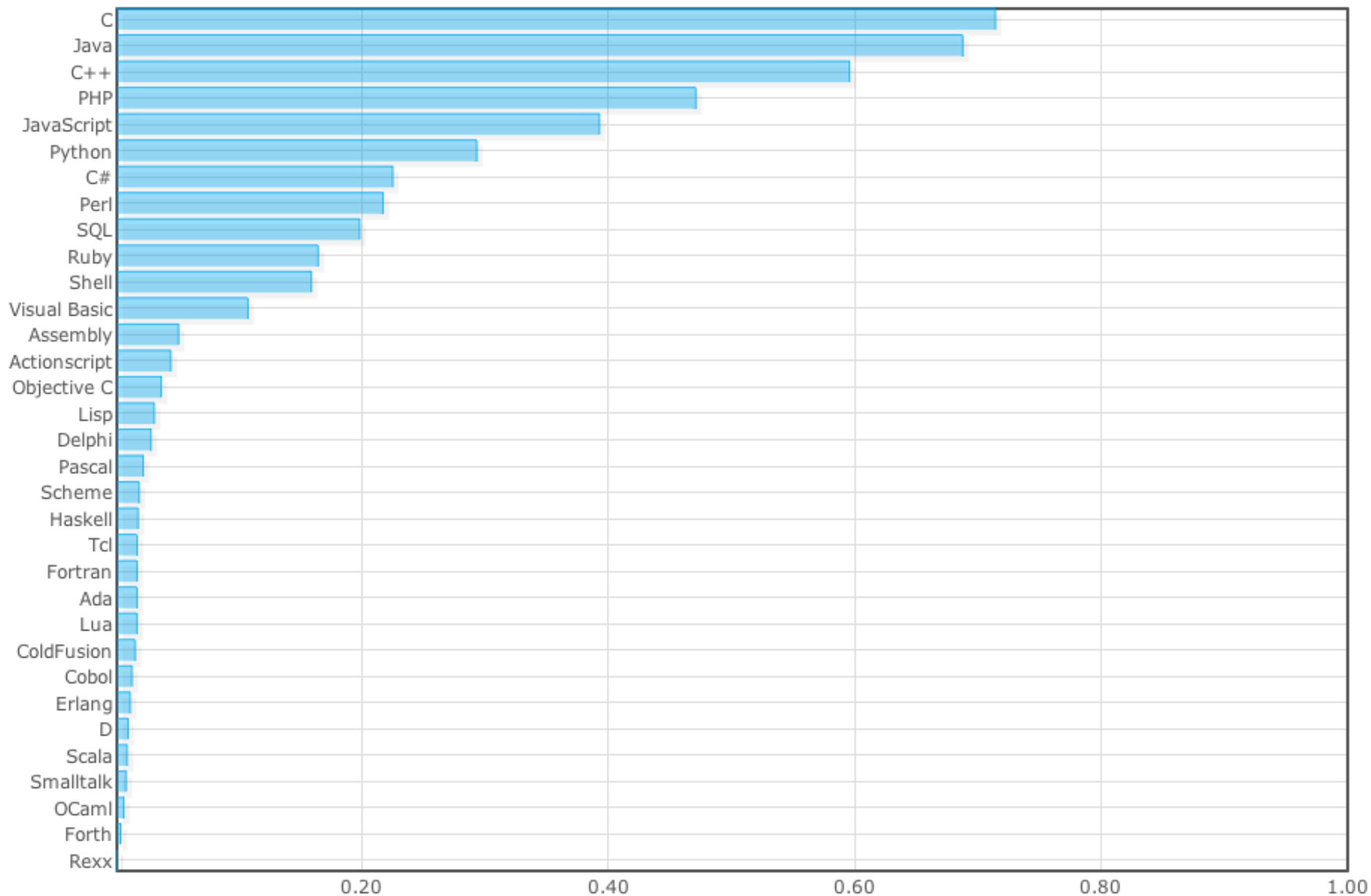
- ② #2 and #4 languages for SourceForge open source projects
- ② Top commercial software is written in C++
 - Microsoft Word, Adobe Photoshop, Google Chrome, Firefox, Mac OS X, iOS, Oracle database, Microsoft SQL Server, MySQL, games
- ② Top web companies use C++ – Facebook, PayPal, Amazon

C and C++ - popular as ever

C++	9.698%	+1.61%	A
PHP	5.732%	-0.36%	A
C#	5.591%	-1.73%	A
(Visual) Basic	5.032%	-0.01%	A
Python	4.062%	+0.45%	A
Perl	2.182%	+0.10%	A
Ruby	1.739%	+0.24%	A

Normalized Comparison

This is a chart showing combined results from all data sets, listed individually below.



Client / Server

◎ 1990s

- Large growth in desktop applications
- Client-server IP architecture nascent
- Most server side apps were basic web

◎ 2000

- Server side development increase
- Java growth with EJB

◎ 2005

- Device development kick-started again by Apple success with iPod, iPhone, and iPad

◎ 2010

- Tremendous device client growth through smartphones driven by Android
- Server side still important but leveraged

Native is Key for Client Device UX

Servers

Code Safety & Protection is Paramount
Ability to Scale Performance via HW
Web UX typically network bound



Virtual Code: Java, .NET

Client Devices

UX Performance is Paramount
Fixed/Deployed HW Profiles
UX typically HW bound



Native Code: C++, Obj-C, Delphi

再一次...

重點回到客戶端

The 1999 Enterprise



Windows

The 2005 Enterprise



Windows



Web

The 2012 Enterprise



Windows



Mac



Mobile



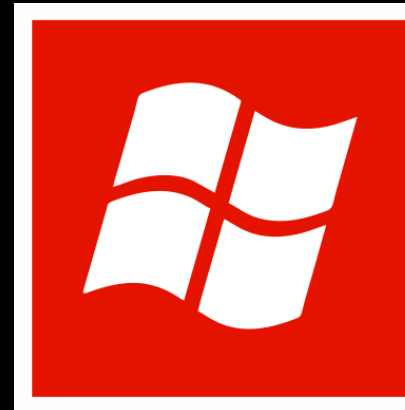
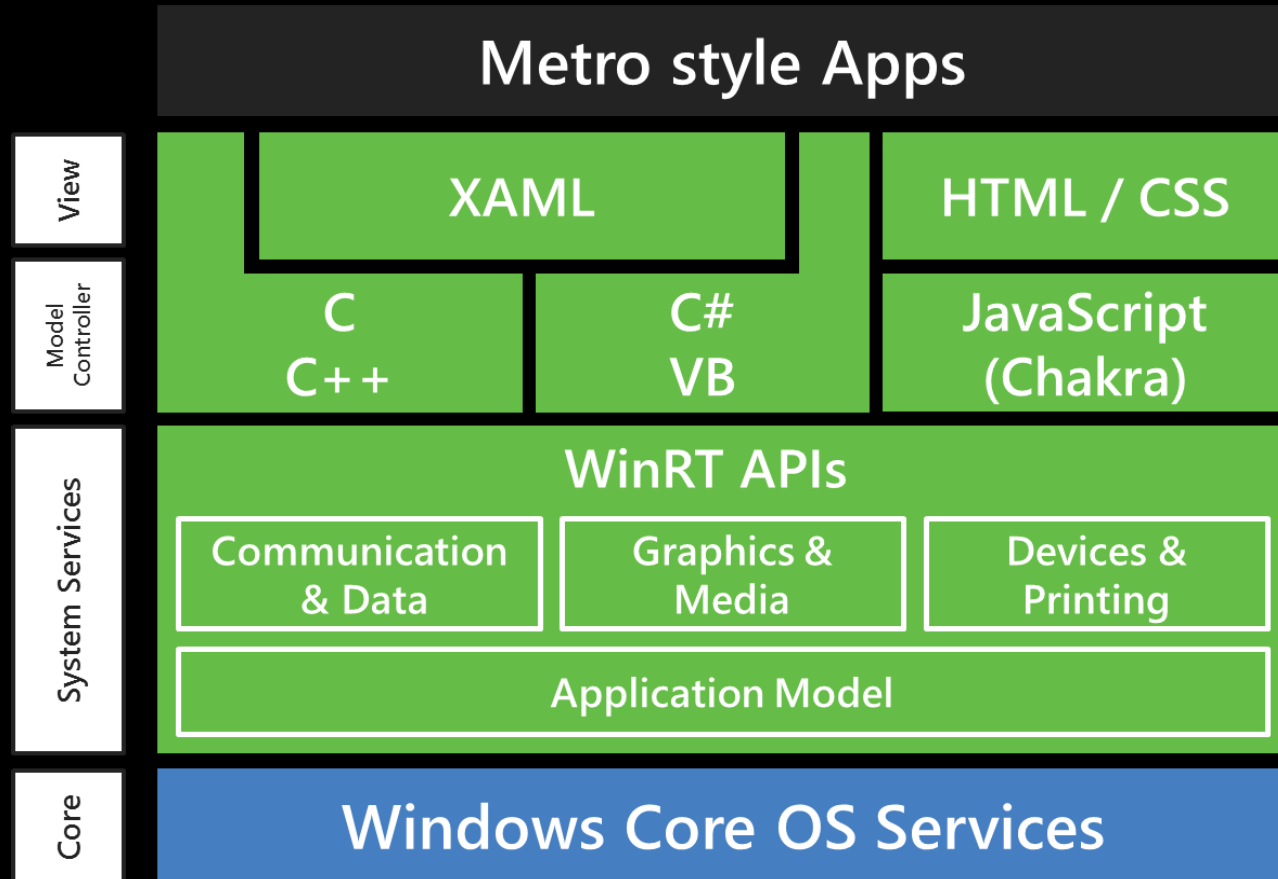
Web

An Unprecedented Multi-platform Opportunity for Developers

Devices, Devices, Devices



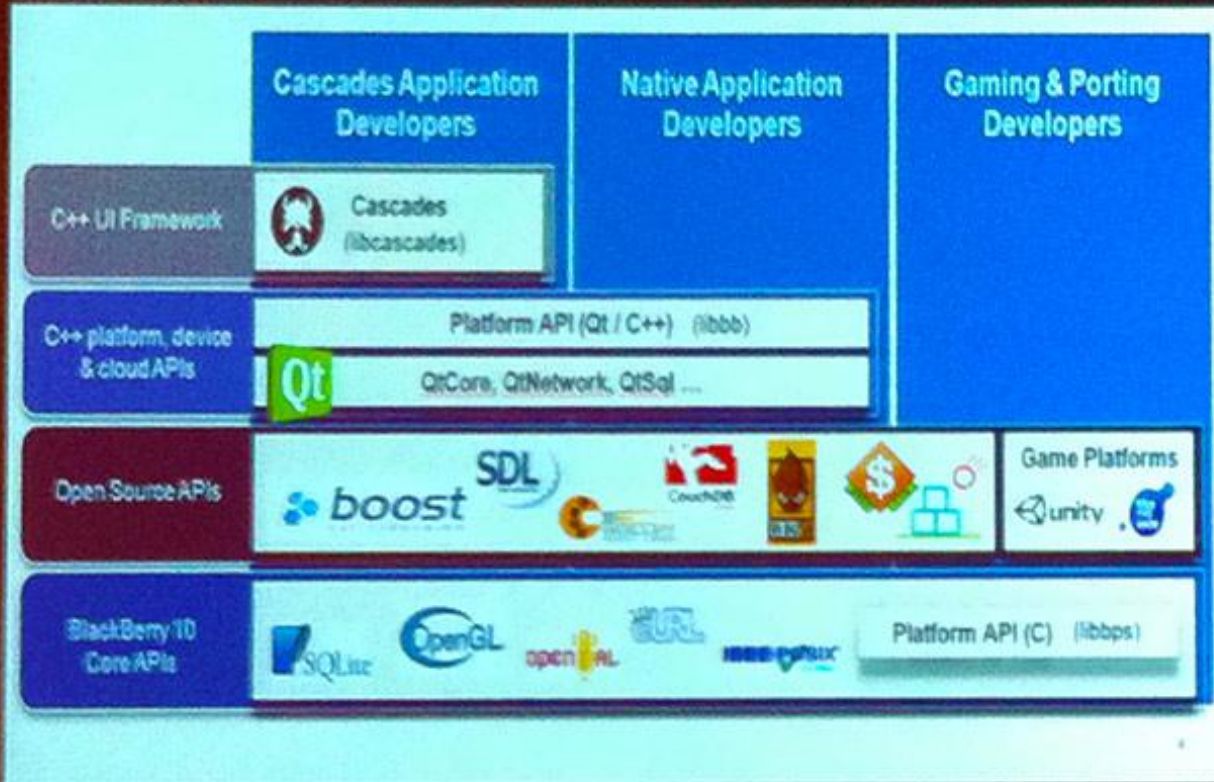
Windows 8 WinRT



BlackBerry 10

Zooming in on the NDK...

BlackBerry 10 Jam



Android NDK

Android NDK app

Android
manifest

Dalvik classes

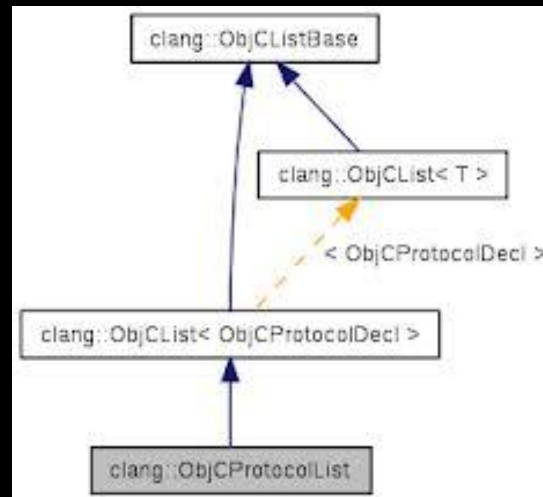
Resource
bundle

Libraries
& JNI



iOS

- Objective-C
 - C-based OO language modeled after SmallTalk
- Now shares a common front-end and architecture with a major C++ compiler
CLANG



A common strategy

iOS

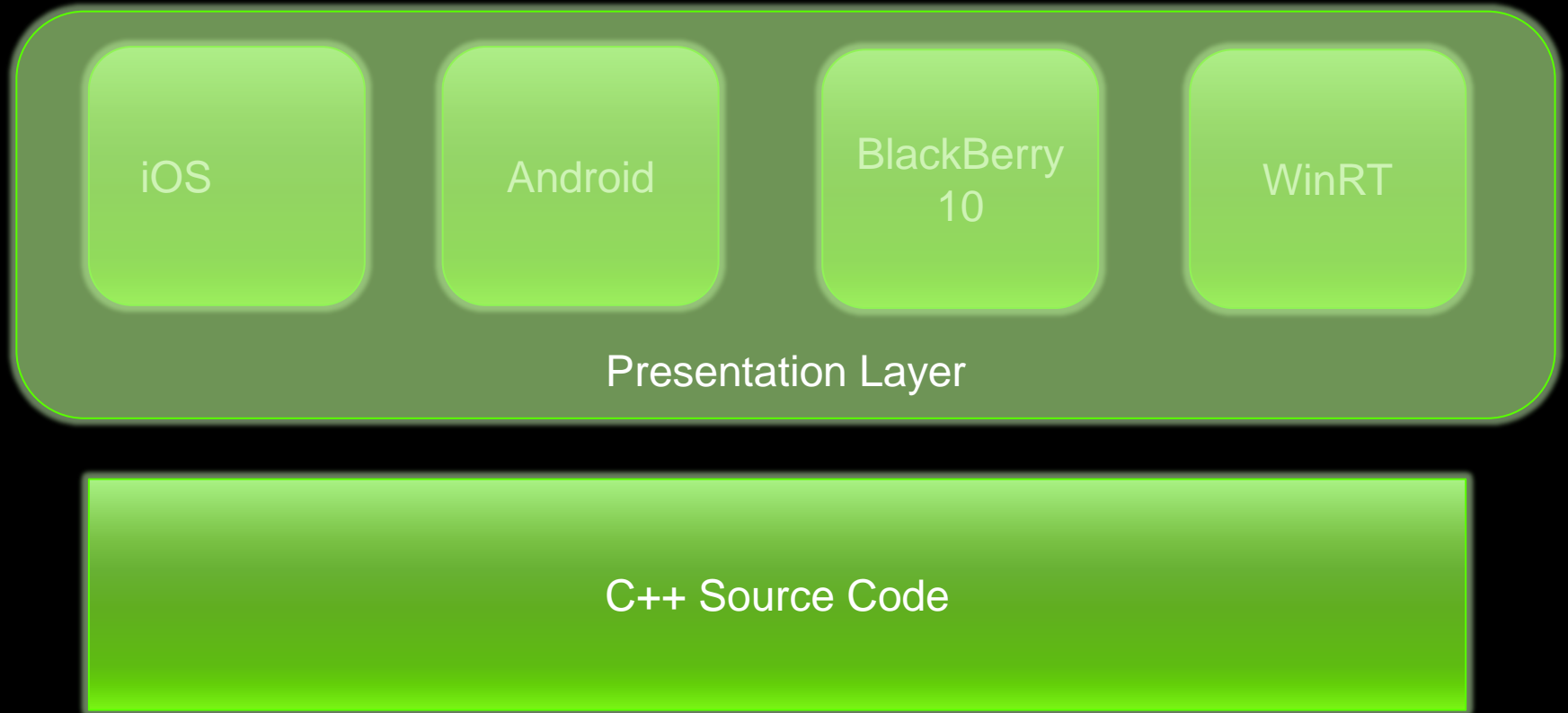
Android

BlackBerry
10

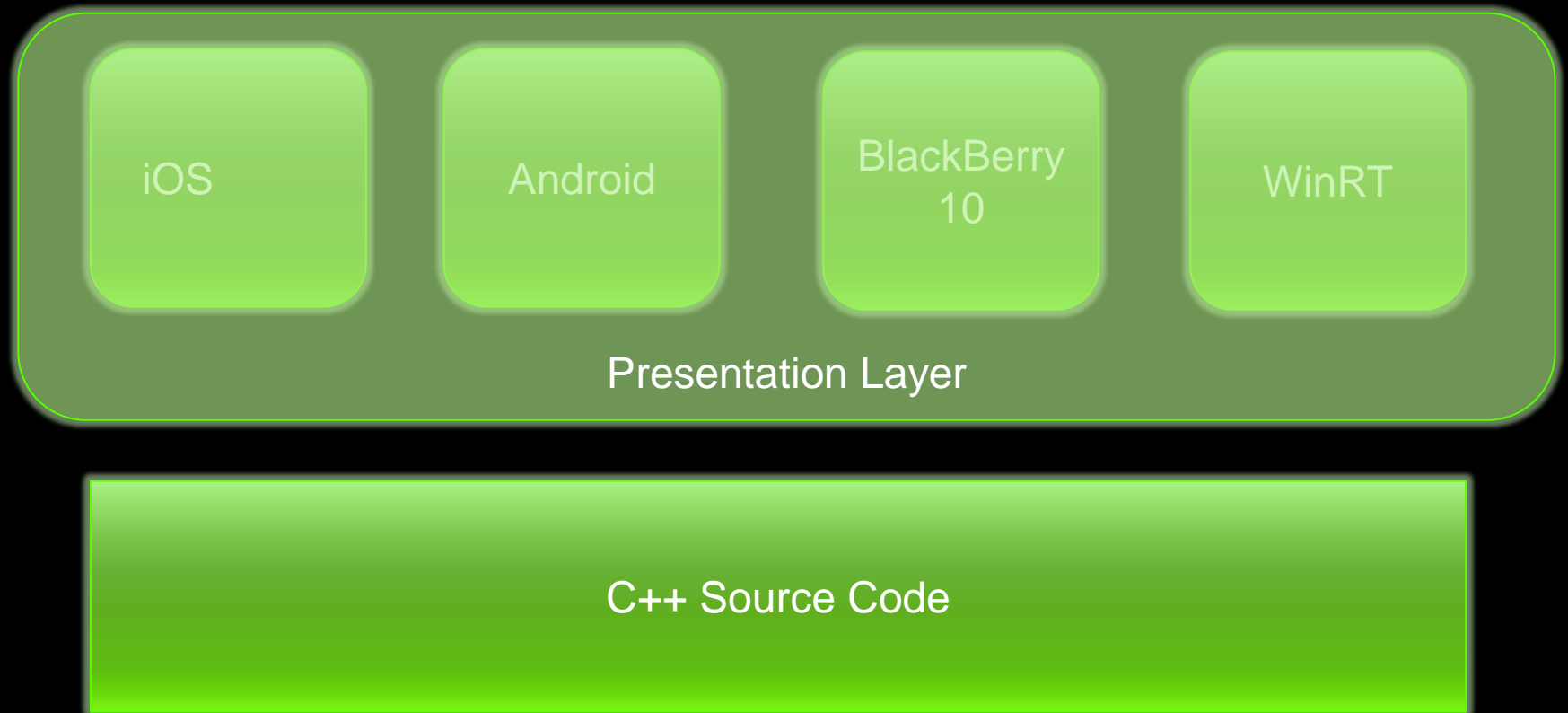
WinRT

C++ Source Code

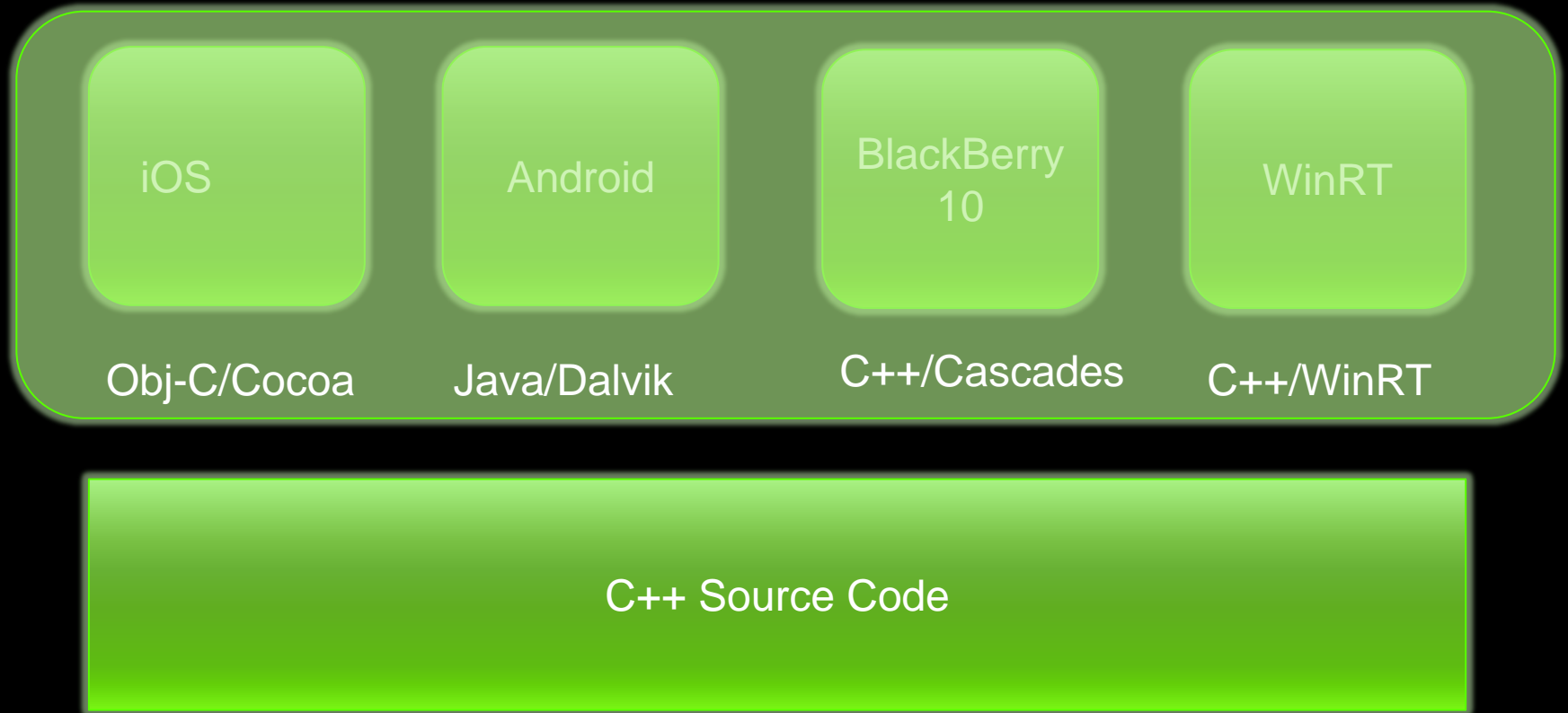
A common strategy



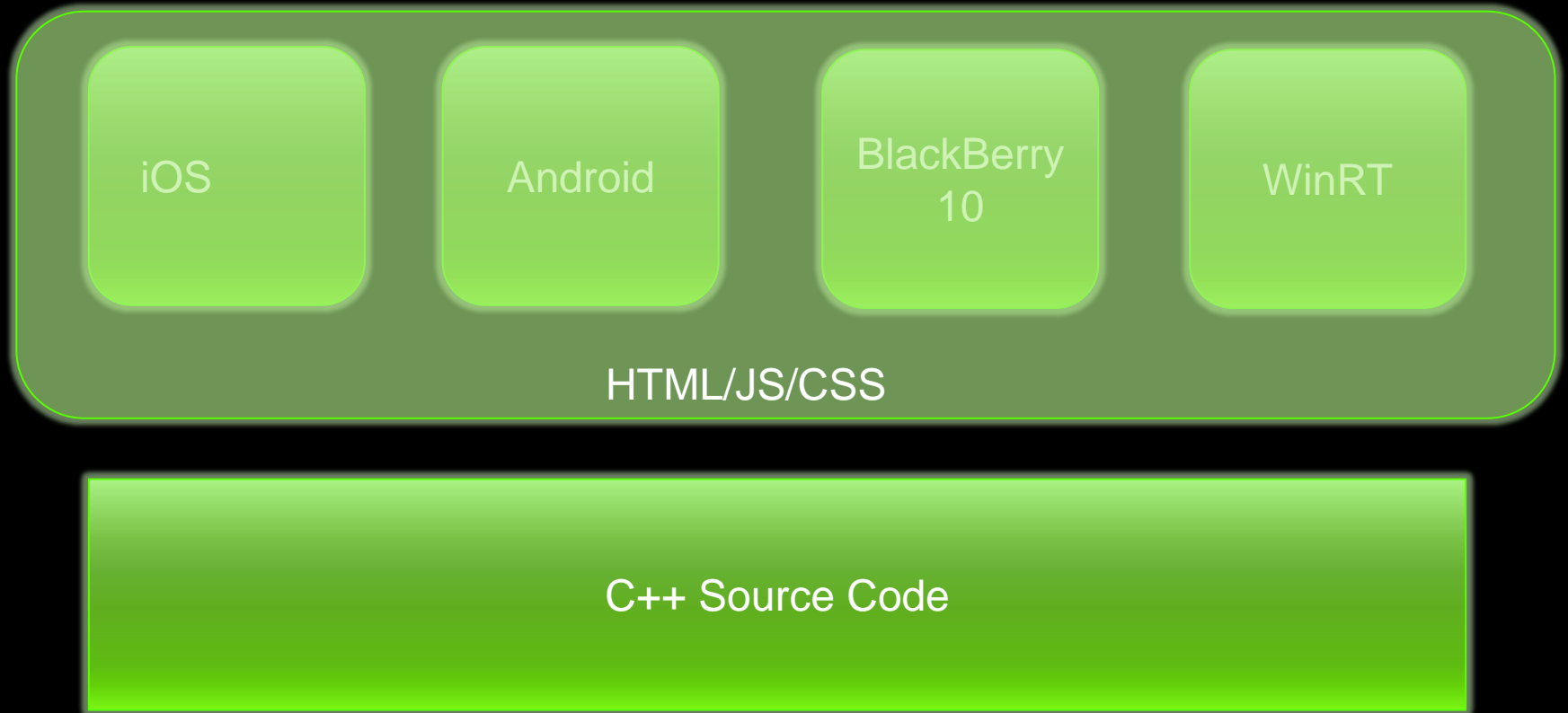
A common strategy



A common strategy



A common strategy



A common strategy

iOS

Android

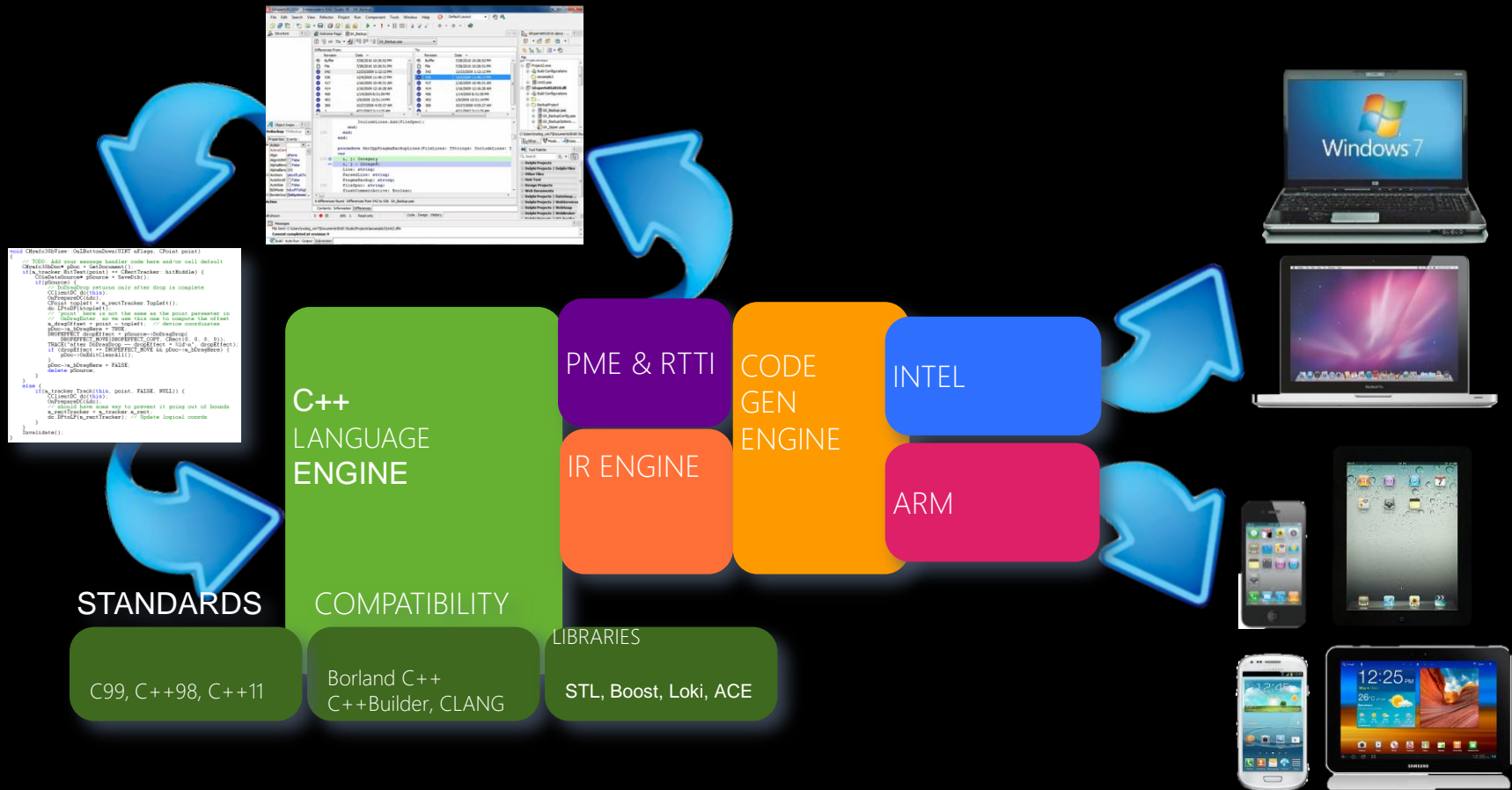
BlackBerry
10

WinRT

FireMonkey (2013)











C++ Source Code

Embt C++ Architecture



萬眾矚目:C++Builder For iOS

FireMonkey Architecture

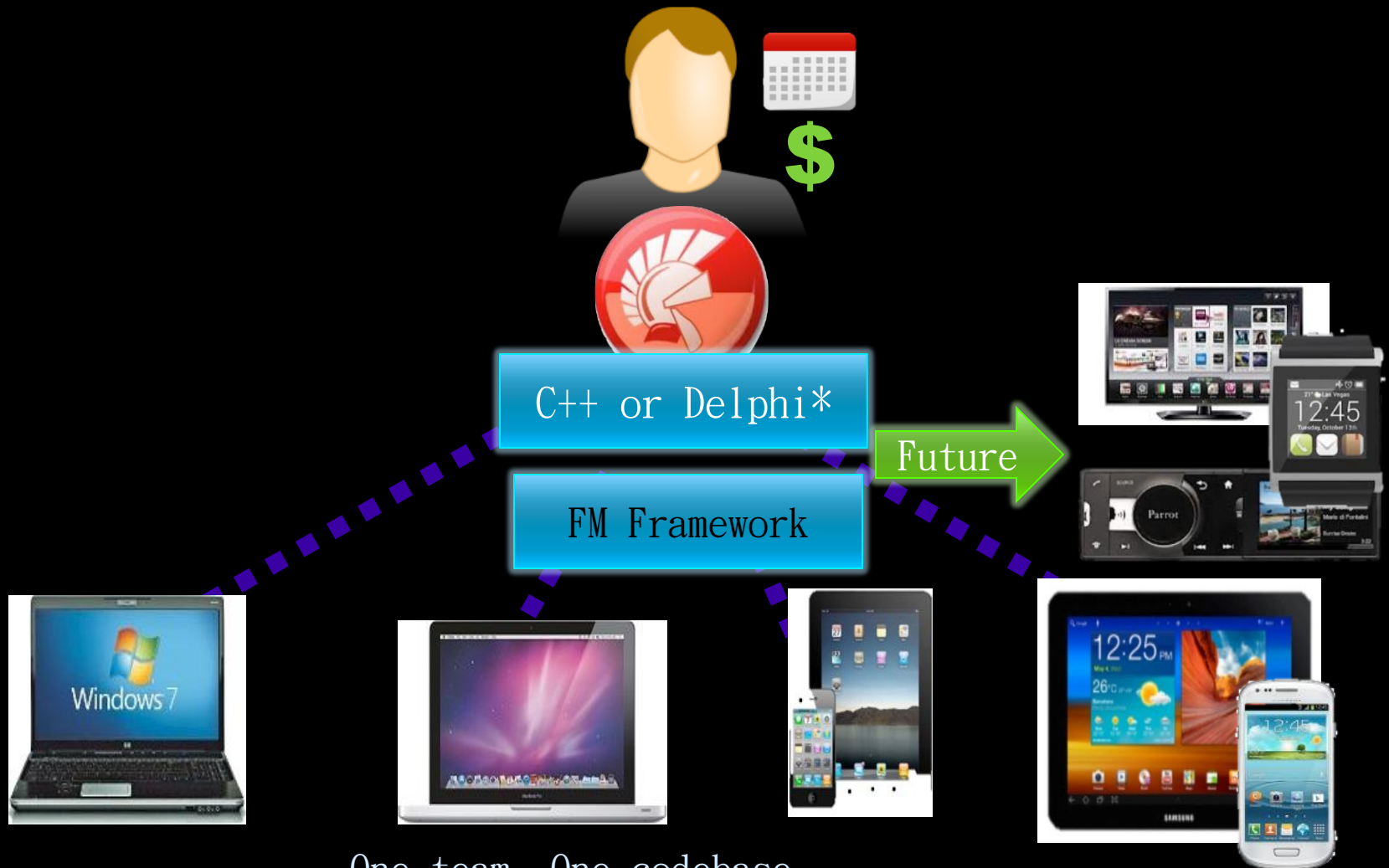
Language	 		
Framework			
3D API			
Platform			 

Truly Native or Scripted?

	True Native		Scripted		VM
	Platform Vendor Tools	Embarcadero	HTML5 Cross Platform	Appcelerator	Xamarin
Tools	XCode, Visual Studio, Eclipse	RADStudio	Adobe, Sencha, Kendo, HTML5Builder	Titanium	Mono
Platforms	Win <u>or</u> Mac <u>or</u> iOS <u>or</u> Android*	Mac/Win/iOS & Android* (2013)	iOS/Android	iOS/Android	iOS/Android
Single Source / Multi-Device	No	Yes	Yes	Partial	Partial
Runtime	ARM	ARM	JavaScript	JavaScript	Bytecode VM

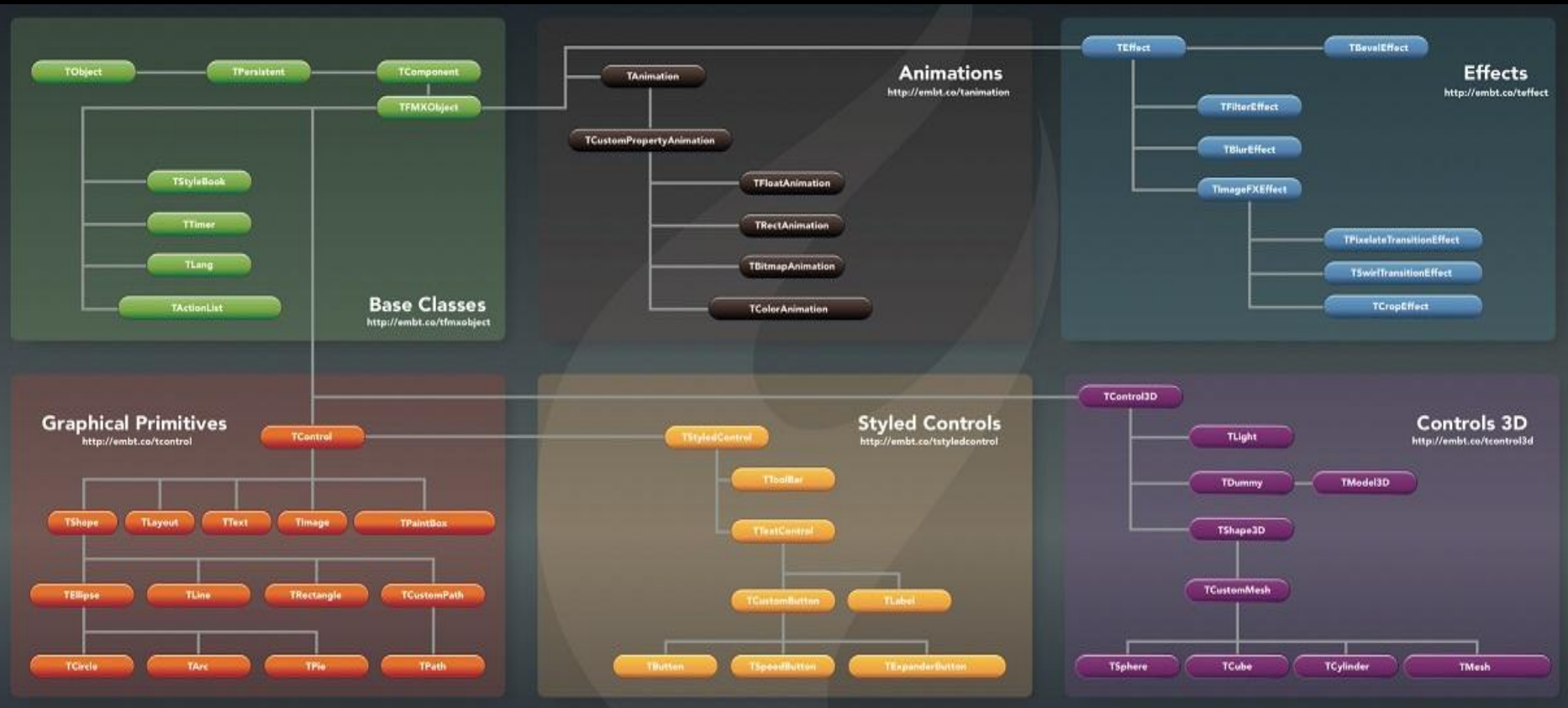
*Android NDK is True Native - Android ADK with Java/Davlik VM is not

RAD Studio XE5



One team, One codebase

Component-Based Framework



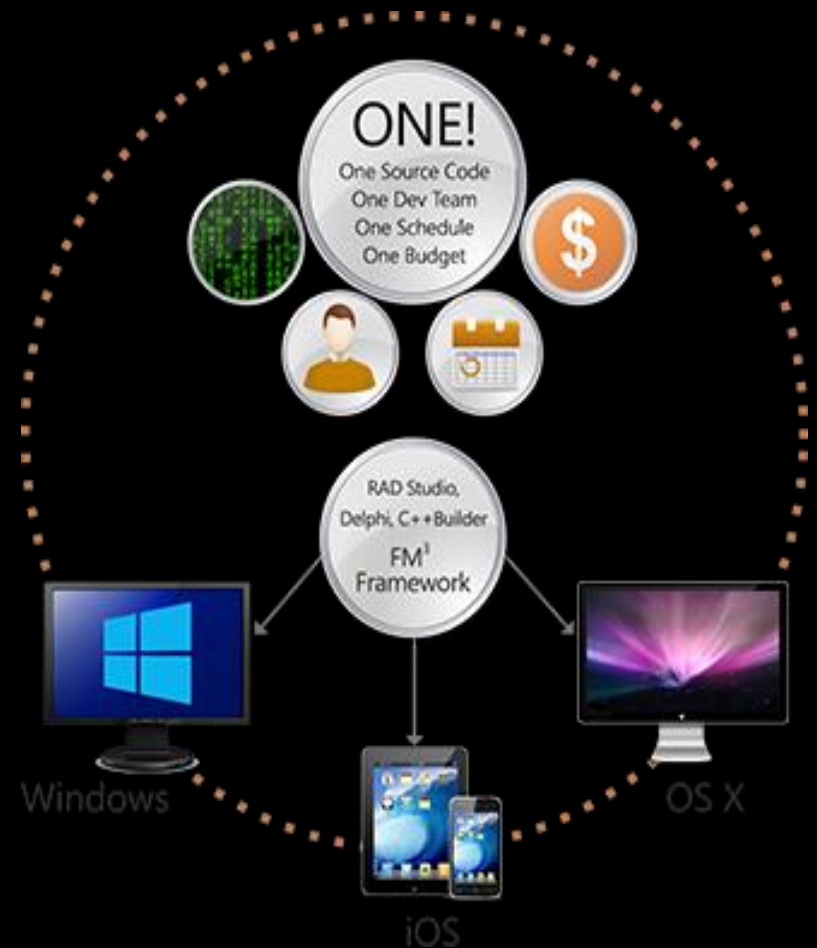
萬眾矚目:C++Builder For iOS

範例

一份程式碼同時開發移動和桌面應用
程式

One Codebase, One Team

- One set of development resources to create apps for multiple platforms
 - one codebase
 - one team
 - one schedule
- Reduced budgets and complexity



一份程式碼同時開發移動和桌面應用程式

範例

C/C++程式碼和函式庫

C/C++ 程式碼和函式庫

C++Builder
RTL

C++Builder
RTTI

