HUGHES SYSTIQUE™

Android Vs iOS

Shilpa Mahna Bhatnagar

Agenda

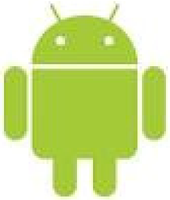
- Introduction
- Development Environment
- Tools & Frameworks
    - UI Design Tools
    - Testing Tools
    - Debugging Tools
- Infrastructure Support
- Ease of Injecting Legacy Code
- Way Forward

Introduction

- iOS was the first popular touch phone
  - Allowed user to build applications
- Blackberry was popular as a business phone
  - Allowed user to build applications
- Then came a good mix option
  - Allowed user to customize entire OS
  - Paradigm Shift
- The Advent of Smartphone
  - Market got Divided
    - iPhone
    - Android

Development Environment

- Operating System
  - Windows
  - Linux
  - MAX OS X
  - IDE
  - Eclipse
  - IntelliJ



- Development environment is not restricted to a specific platform
- ADT(Android Development Tool) plug-in for Eclipse provides a powerful, integrated environment to build Android applications.

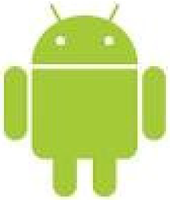
Development Environment

- Operating System
  - MAC OS X
  - IDE
  - XCode
- Development environment is restricted to Mac OS X platform only
- Development package from Apple is available for free. This package include
  - Interface Builder
  - Instruments
- Tools are very powerful and their Java equivalent costs hundreds or even thousands of dollars.

Tools and Frameworks

- Variety of Tools and Framework to aid
  - Development
  - Testing
  - Debugging

UI Design Tools



- Eclipse plug-in for Layout Editor
- Droiddraw
  - WYSIWYG UI designer for Android
- Draw 9-patch
  - Allows you to easily create a nine patch graphic using a WYSIWYG editor
- Layoutopt
  - Allows you to analyze and optimize application's layouts for efficiency
- Hierarchy Viewer
  - Allows to debug and optimize application‘s UI

UI Design Tools

- Interface Builder
  - Software development application for Apple's Mac OS X operating system
  - It is part of Xcode
  - Allows Cocoa and Carbon developers to create UI using a graphical user interface

Testing Tools

- Android Emulator
  - A QEMU-based emulator which can be used to test a debug applications in Android run-time environment
  - Traceview
  - Provides a graphical viewer for execution logs saved application
- Monkey
  - Generates pseudo-random events such as clicks, touches, or gestures, as well as a number of system-level events.
  - Can be used to stress-test applications in a random yet repeatable manner.
- Monkeyrunner
  - Provides an API for writing programs that control an Android device or emulator from outside of Android code.

Testing Tools

- iOS Simulator
  - iOS Simulator runs application in much the same way as an actual iOS device.
  - It is quick to launch and debug
  - iOS Simulator makes a perfect test bed to make sure user interface works the way it is intended
- Instruments
  - Application that helps you track down performance bottlenecks in Mac OS X and iOS apps.
  - Instruments collects data such as disk, memory, or CPU usage in real time, either on Mac or remotely from a connected iPhone.
  - The collected data is graphically displayed as tracks over time, making it easy to pinpoint problem areas, then drill down to the offending lines of code.

Debugging Tools

- hprof-conv
- Converts HPROF file to a format that can be viewed in pr tools
- Dalvik Debug Monitor Server (ddms)
- Allows to debug Android Application
- Android Debug Bridge (adb)
- Provides a command line interface to access Android device
- GDB
- Allows to debug Android Applications
- NDK Debug tool
- NDK provide tools similar to GNU tools to debug applications
    - Gdb
    - Gcov
    - Addr2line
    - nm

Debugging Tools

- LLVM Compiler
- The same parser used to build C/C++ and Objective-C powers Xcode’s indexing engine, providing incredibly accurate code completions.
- As you work, Apple LLVM constantly evaluates what you type, identifying coding mistakes that Xcode shows as Live Issues, and thinking ahead for ways to Fix-it for you.
- Other compilers can tell you what is wrong — Apple LLVM can make it right.
- Breakpoints
- You can use the debugger to pause execution of your program at any time and view the state of the running code, it's usually helpful to set breakpoints before running your executable so you can stop at known points and view the values of variables in your source code.

Infrastructure Support

- Garbage Collection
  - Periodically claim memory used by object that are no referenced
- Telephony
  - Allows to initiate/terminate calls from Application
  - Allows to trap/send SMS from Application
- Programming Language Supported
  - Java
  - C
  - C++
  - Bash Scripting
- OpenGL ES
  - Supported through both SDK API and NDK
- Open nature of Android provide flexibility to exploit H/W specific features

Infrastructure Support

- Garbage Collection
  - Automatic Reference Counting (ARC) for Objective-C makes memory management the job of the compiler.
  - Enabling ARC with the Apple LLVM, dramatically simplifying the development process, while reducing crashes and memory leaks.
- Telephony
  - Allows to initiate calls, monitor state change from Application
  - More restricted as compared to android
- Programming Language Supported
  - Obj C , C/C++,HTML5, JS, CSS3

Infrastructure Support

- OpenGL ES
  - GLKit is a high-level framework combining the best practices of advanced rendering and texture techniques with the latest OpenGL ES 2.0 features.
  - It’s optimized to take advantage of hardware accelerated math operations, to get the best performance without all the work.
  - iOS 5 SDK includes Apple-developed OpenGL ES extensions designed specifically for advanced game developers.
  - OpenGL ES debugger in Xcode allows to track down issues specific to OpenGL ES in your code.

Injecting legacy code

- Legacy code can be reused in Android application using NDK interface
- JNI bridge allows C/C++ functions to be invoked from Java
- Reduces Application development time as existing code can be used in Android Applications
- C/C++ Opensource software available for Linux can be ported to Android and included in Android Application

Injecting legacy code

- Legacy code can be reused in iOS application directly (.m, .mm files)
- Reduces Application development time as compared to Android as the code can be directly drag & drop and can be merged in same the Obj C file.

Way forward …..

- Reuse across Android and iOS
  - HTML5, JS, CSS3. Ex. PhoneGap
  - Reusable component developed in C,C++ can be used on both sides

And the Winner is …..

	Android	iOS
Development Environment	★★★	★★★★
UI Design Tool	★★★	★★★★
Testing Tools	★★★★	★★★★
Debugging Tools	★★★★	★★★★
Infra. Support	★★★★	★★★
Legacy Code	★★★★	★★★★
Language Support	★★★★	★★★

References

- http://developer.android.com
- http://omappedia.org/wiki/Android_Debugging
- http://developer.apple.com

Thanks