

Open Source | Open Possibilities



Hardware for Android App Testing & Tuning

Brian Spencer
Senior Engineer, Qualcomm Innovation
Center, Inc.



Topics

Intro to different HW platforms

Tune for Battery Power

Get more out of the GPU

Pre optimized SDK's

Requirements

In picking a developer board look at your needs.

- Open board for HW customization
- Commercial looking form factor
- Purely Application development
- Modify the Android software
- Board features

Today I will discuss the Dragonboard by Qualcomm

- Similar thoughts can be applied to other boards
- Similar build concepts

Dragonboard™

- Through the **exposed connectors, adapters, and expansion headers**, app developers, component suppliers, and universities can innovate within the growing open source development community
- Available from Bsquare



APQ8060-based Dragonboard Specs

- APQ8060 dual core processor
- Adreno 220 Graphics
- 1500mAH battery
- 3.61" WVGA Display
 - Cap Sense Multi-touch screen
- 5MP main camera
- 2MP camera for video telephony
- BT/WiFi Expansion Card
- Sensors Expansion Card
 - Pressure and Temp
 - 3-axis accelerometer
 - 3-axis Gyro
 - Proximity and Ambient Light
 - 3-axis Compass
- Keypad
- Connectors
 - JTAG
 - RS-232
 - USB
 - Ethernet
 - Audio
 - 2x SIM
 - 2x SDIO
- Test points
 - MIPI-CSI (camera)
 - MIPI-DSI (display)
 - Parallel bus (EBI2)
 - GPIOs
- Wolfson Audio Codec

Open Source | *Open Possibilities*



Using the board

Live Demonstration

How to find battery bottlenecks

- Various types of hardware exist that can be purchased to connect to a device to provide overall battery drain.
- Connectors can be added to an open board to measure draw from components.
- How do you use this information?
- What if you are an app developer and have no Hardware knowledge?

Today I will discuss the Trepn Profiler tool

- Similar thoughts can be applied to power profiling equipment
- End goal is to find area's to improve application performance

Snapdragon Mobile Development Platform (MDP)

- A full-featured, fully integrated development and demonstration platform for mobile developers to **create**, **test**, and **optimize** applications and user experiences for Snapdragon-powered devices
- Available from Bsquare



MSM8660-based Snapdragon MDP/S Specs

- 3.6" WVGA Multi-touch display
- WiFi, BT, GPS, FM
- Adreno 220 graphics
- 1080p video encode/decode
- 13MP main camera w/ flash
 - 1MP front facing camera
- Audio
 - Stereo 15x11mm loudspeakers
 - 5 microphones (3 front, 2 back)
- Sensors
 - 3D accelerometer
 - 3-axis gyro and compass
 - Ambient light / proximity
 - Temperature and pressure
- Memory
 - 1GB LPDDR2
 - 16GB eMMC
- Haptics
 - Dual, independently controlled linear vibrator motors
- Hard keys
 - Dual stage camera shutter
 - +/- Switches (volume / zoom)
 - Power switch
 - HW reset
- External connectors
 - HDMI Type D
 - MicroUSB v2.0 HS OTG
 - 3.5mm audio jack
 - Micro SD

Open Source | *Open Possibilities*



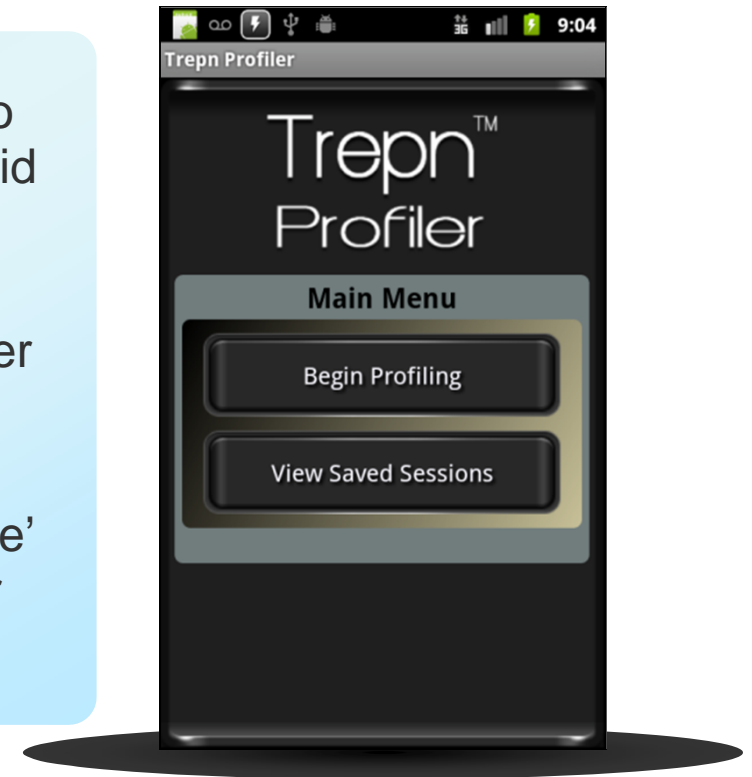
Trepan Profiler

The Need For Application Profiling

- Poor choices by app developers can significantly impact battery life of mobile devices
- How apps drain the battery
 - Inefficient or inappropriate use of resources
 - Using the network too much
 - Taking too many GPS location fixes
 - Preventing the processor from going to sleep

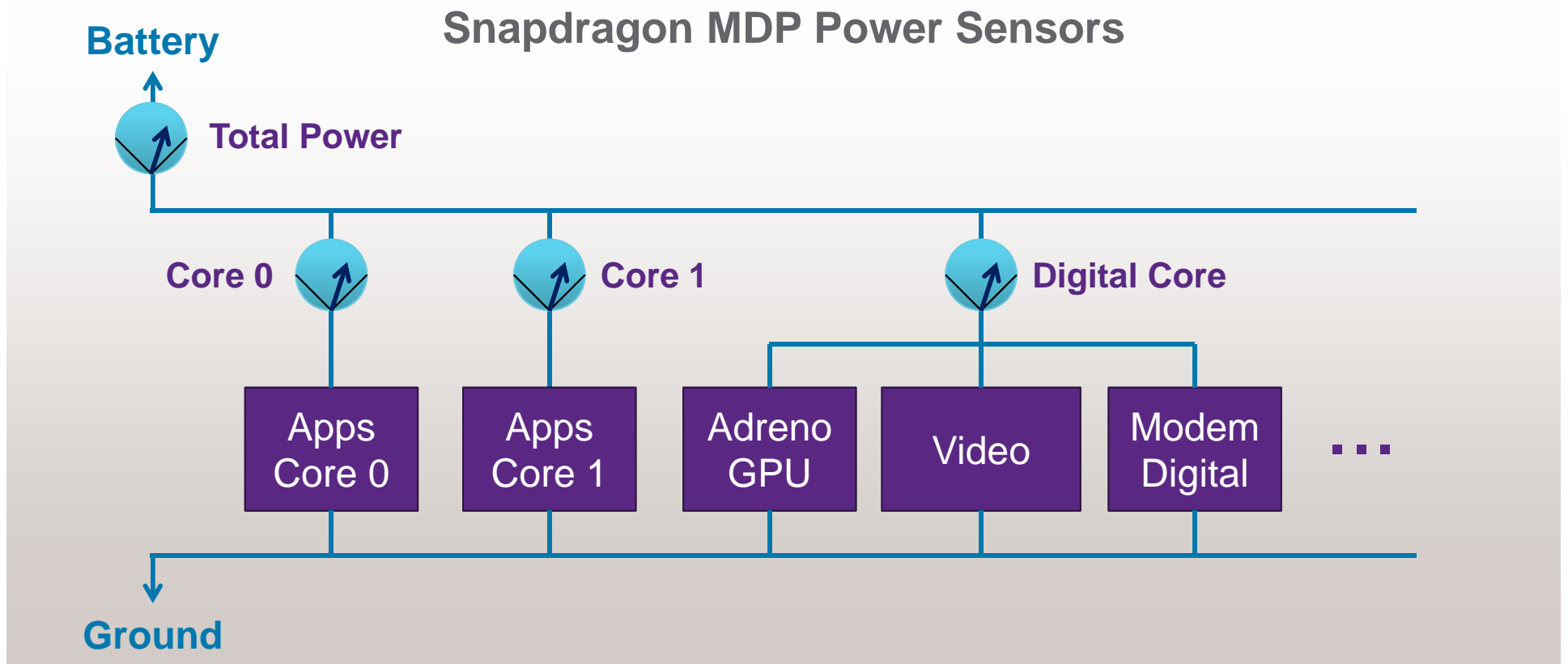
Trepn Profiler Overview

- Trepn Profiler is an app profiling tool designed to measure power, data, and CPU impact of Android apps run on the Snapdragon™ Mobile Development Platform (MDP)
- Trepn allows developers to understand the power impact of programming choices, and make intelligent decisions
- Trepn leads to 'power aware' and 'network aware' programming practices, resulting in lower power use and longer battery life



How Trepro Profiler Measures Power

- Snapdragon MDP has **embedded power sensors** to monitor power of various 'rails' available



Key Power Sensors And Other Log Points

In addition to power, Trepn also tracks CPU load, network usage, and per-app stats

Power Rails

- Battery power
- CPU core 0/1
- Digital core
- Camera
- Audio Codec
- Display IO
- HDMI

System Stats

- CPU 0/1 frequency
- CPU 0/1 user load
- CPU 0/1 system load

Network Stats

- Data state
- WiFi state
- WiFi RSSI

Per-App stats

- CPU usage
- Memory usage
- Application state

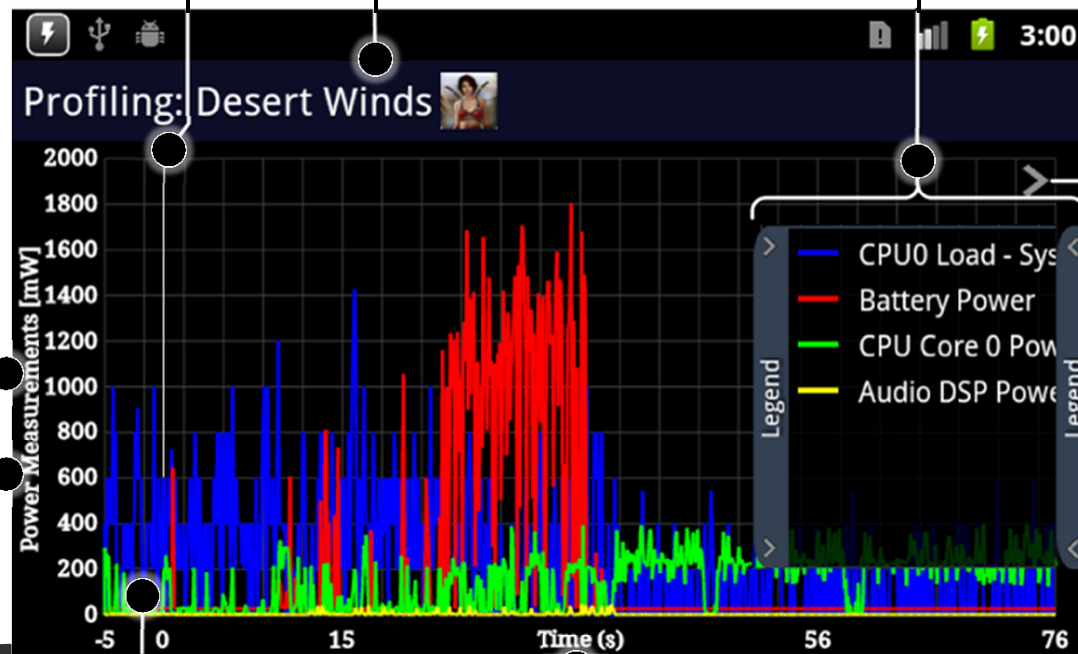
Treppn Profiler Graph Display

Vertical line separates end of Baseline (Reference Collection Time) and start of profile session.

Name of App being profiled

Graph legend is hidden in slide-out door; drag drawer to open/close or double-tap bar. Also tap name of item to hide/show graph line.

Scale of Y-axis
Double-tap in Y-axis scale to change; can also be changed via Menu



Arrow indicates that additional information is offscreen in that direction. Swiping with the finder left and right scrolls

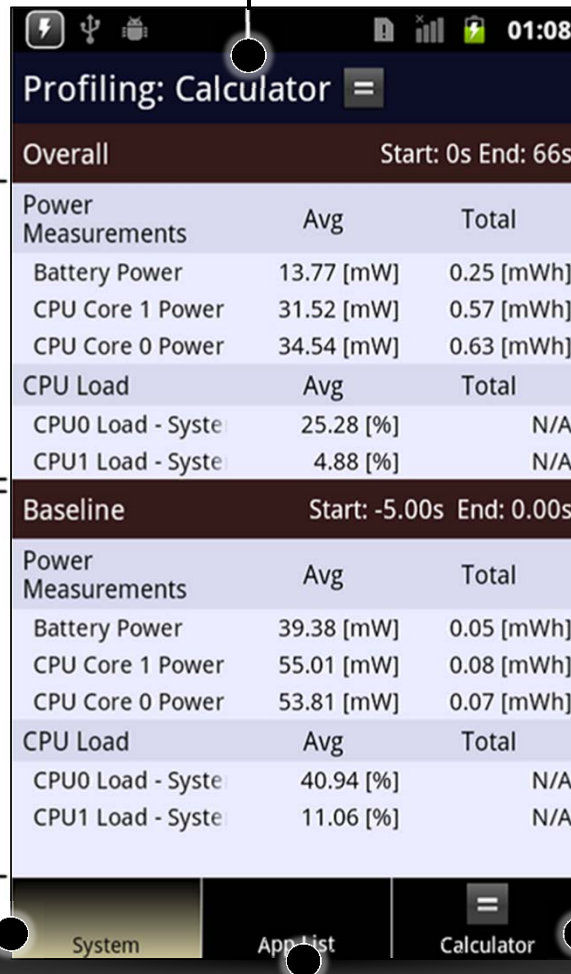
Graph line

Scale of X-axis (always expressed as time in seconds)

Pinch-zooming is available in both Profiling and Analyzing modes.

Trepp Profiler Stats Display

Name of App being profiled



Values are grouped in categories (Power Measurements and CPU Load in this example).

Baseline provides reference for each data point before application is launched.

System tab gives stats on items selected via Preferences

App List tab gives CPU stats on all apps running on device.

This section refers to the power consumed and CPU usage by the application from launch across the logged test points until the end of the session. Current values are given in mW and total in mWh.

CPU Load is given as average of total CPU in percentage.

Right tab(s) (Calculator in this example) gives CPU and Memory Stats on profiled app (and other Apps running on device).

Trepn Profiler Overlay Graphs

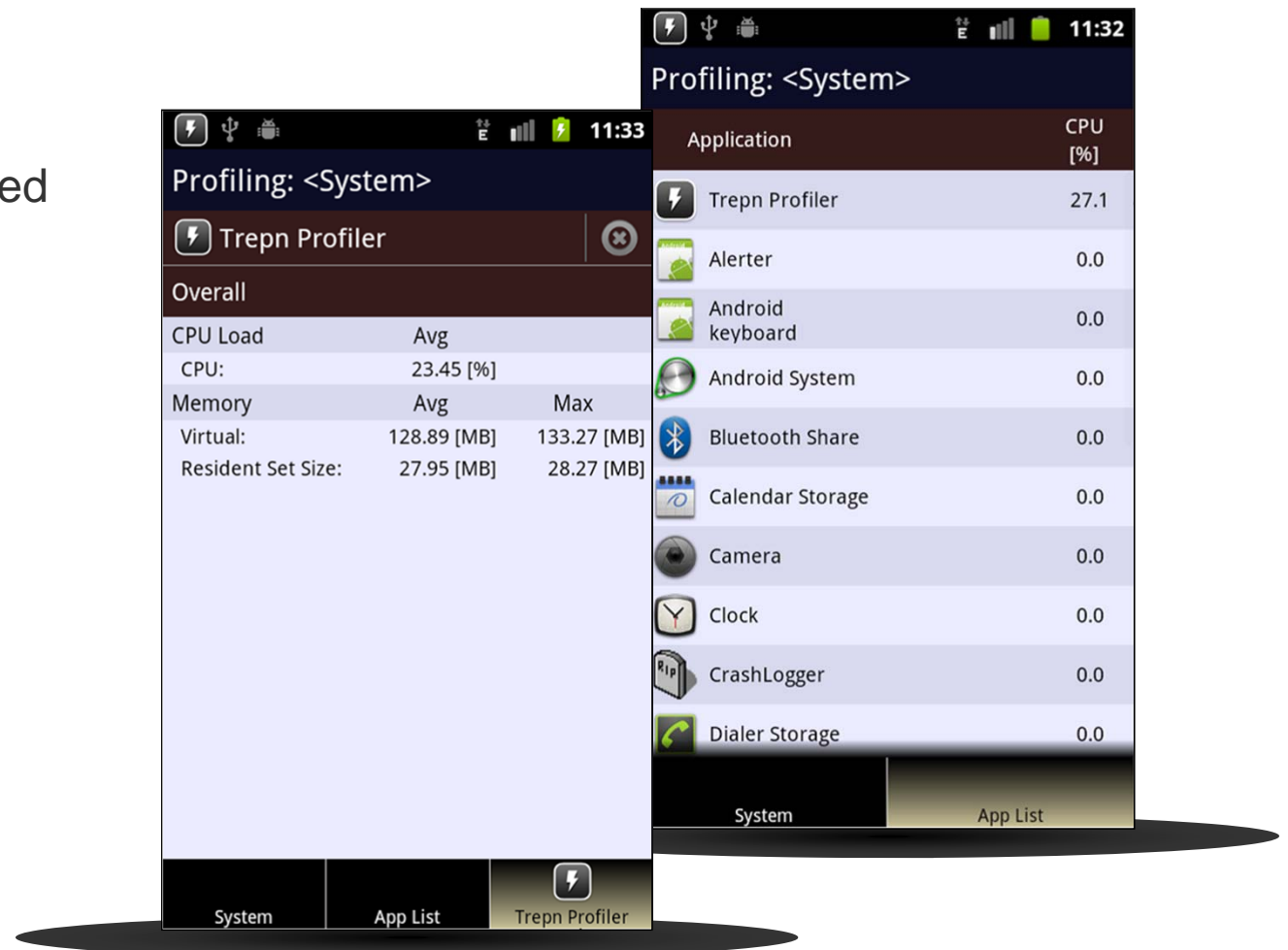
- Overlays allow developers to monitor log points in real-time
- Overlays are setup via the *preferences* menu



Desert Winds by SouthEnd Interactive

Trepro Profiler Per-App Statistics

- Trepro tracks some items on a per-app basis
 - CPU load
 - WiFi data sent/received
 - Dormant-to-active transitions
- Stats on each app 'wake-up' state are also provided



Using Trepn Profiler

■ Standard Operation

- Install app on Snapdragon MDP using *adb install*
- Use Trepn Profiler to 'begin profiling', select your app, and launch
- Run your app through its paces
- Return to Trepn Profiler to 'stop profiling' and save data
- Use 'view results' to explore the captured profiling data

■ Advanced Usage

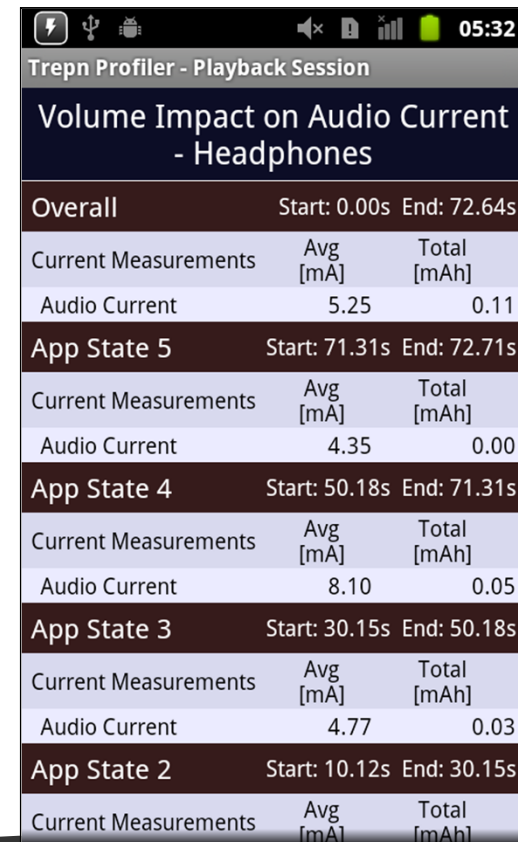
- Align your logs with app behavior
 - Use Android Intents to instrument your code to set the 'app state' log point
- Select additional logging points through preferences
- Enable overlaying of particular data points on top of running application
- Import saved data (stored on SD card) into Excel for detailed analysis

Instrumenting Your Android Code

- Use Android 'Intents' to change the 'app state' log point within Trepn. This will allow you to correlate the Trepn output with your app
- Trepn calculates stats for each state
- Use the following code to add an auto-incrementing application state:

```
Intent stateUpdate = new  
    Intent("com.quicinc.Trepn.UpdateAppState");  
sendBroadcast(stateUpdate);
```

- Also capable of adding description information to the state intent. Refer to the Trepn User Guide for example Android code



The screenshot shows the Trepn Profiler app interface. At the top, it says 'Trepn Profiler - Playback Session'. Below that, the title is 'Volume Impact on Audio Current - Headphones'. The data is presented in a table with columns for 'Overall', 'App State', 'Current Measurements', 'Avg [mA]', and 'Total [mAh]'. The 'Overall' row shows a start time of 0.00s and an end time of 72.64s, with an average audio current of 5.25 mA and a total of 0.11 mAh. The 'App State' rows show data for states 5, 4, 3, and 2, with their respective start and end times and average audio current values.

Overall	Start: 0.00s	End: 72.64s
Current Measurements	Avg [mA]	Total [mAh]
Audio Current	5.25	0.11
App State 5	Start: 71.31s	End: 72.71s
Current Measurements	Avg [mA]	Total [mAh]
Audio Current	4.35	0.00
App State 4	Start: 50.18s	End: 71.31s
Current Measurements	Avg [mA]	Total [mAh]
Audio Current	8.10	0.05
App State 3	Start: 30.15s	End: 50.18s
Current Measurements	Avg [mA]	Total [mAh]
Audio Current	4.77	0.03
App State 2	Start: 10.12s	End: 30.15s
Current Measurements	Avg [mA]	Total [mAh]

Open Source | *Open Possibilities*



Live Trepn Demonstration



Open Source | *Open Possibilities*



Adreno Profiler

Adreno Profiler

- Graphics performance analysis tool for 2D and 3D content
- Enables application profiling and optimization in real time while connected to the device



Using the Adreno Profiler

Who should use it?

- Developers creating applications for commercial devices with Snapdragon and its Adreno GPU

Why use Adreno Profiler?

- Provides quick insight into GPU and hardware performance bottlenecks
- Simulation and override features enable application modifications with the click of a button
- Easy to use: multiple, iterative app changes can otherwise be time-consuming and expensive

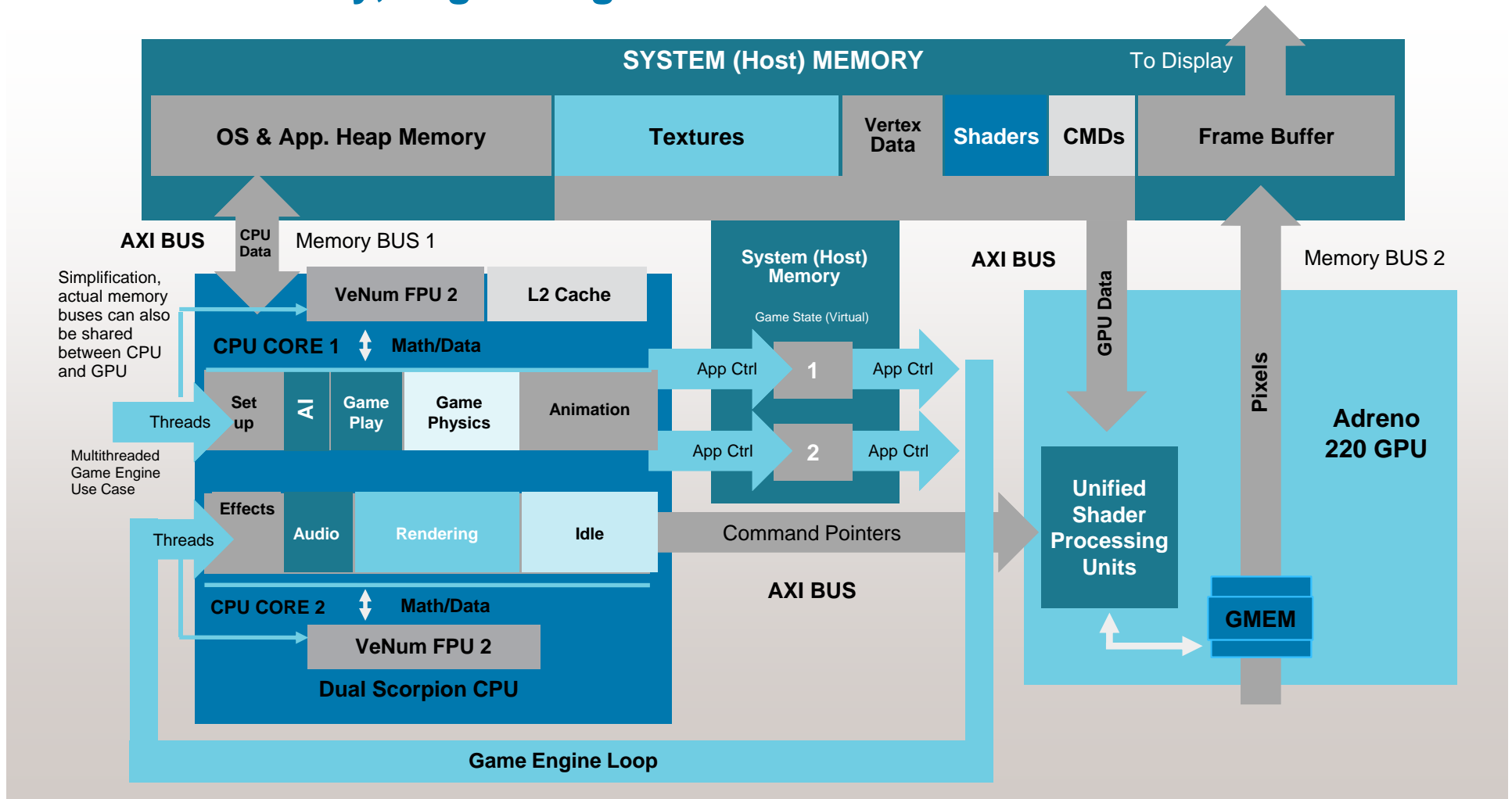
Adreno Graphics Advantages

- Best In Class GPU Design
 - Designed and built from the ground up for Snapdragon SOC
 - Architectural heritage from game console GPU
- Advanced User Experience
 - Console quality and smooth graphics on smart-phones and tablets for games, UIs and Web content
 - Hours of game play between charges
- Large Ecosystem Drive
 - More Android and Windows Phone 7 games running on Snapdragon than any other mobile chipset
 - Great momentum for Snapdragon™ GamePack
 - Extensive Developer Resources
 - Leading mobile 3D content optimization tools
 - Thorough documentation and support from industry experts



System Level Flow for Graphics Apps

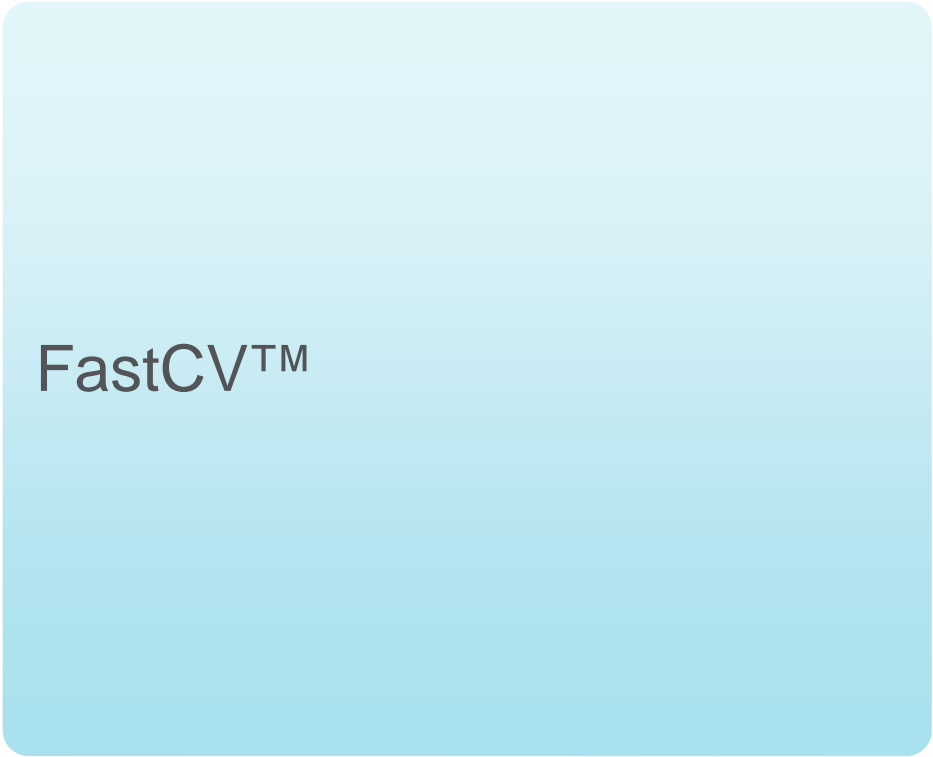
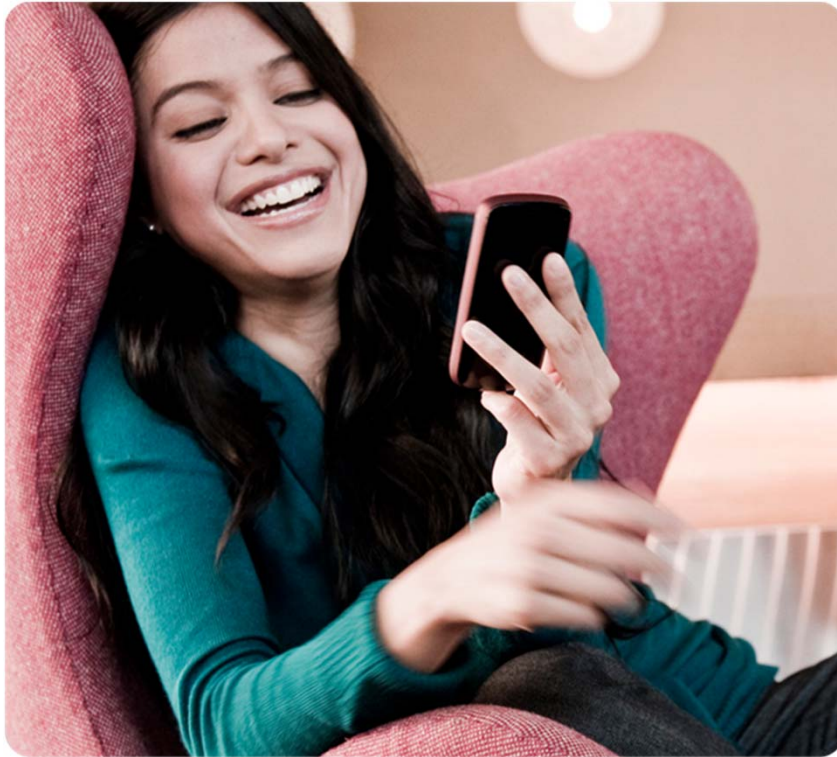
Game Discovery, Organizing & News



What is VeNum?

- VeNum is Qualcomm's implementation of the ARM7 "NEON" SIMD coprocessor in the Scorpion SOC. It is an acronym of "**Vector Numerics**".
- VeNum is automatically used in "VFP mode" by default on the Scorpion by the vanilla GCC compiler flag settings from the Android NDK.
 - Everyone compiling has already been tuning their app for the HW without possibly realizing
 - Ability to modify assembly code for greater performance

Open Source | *Open Possibilities*

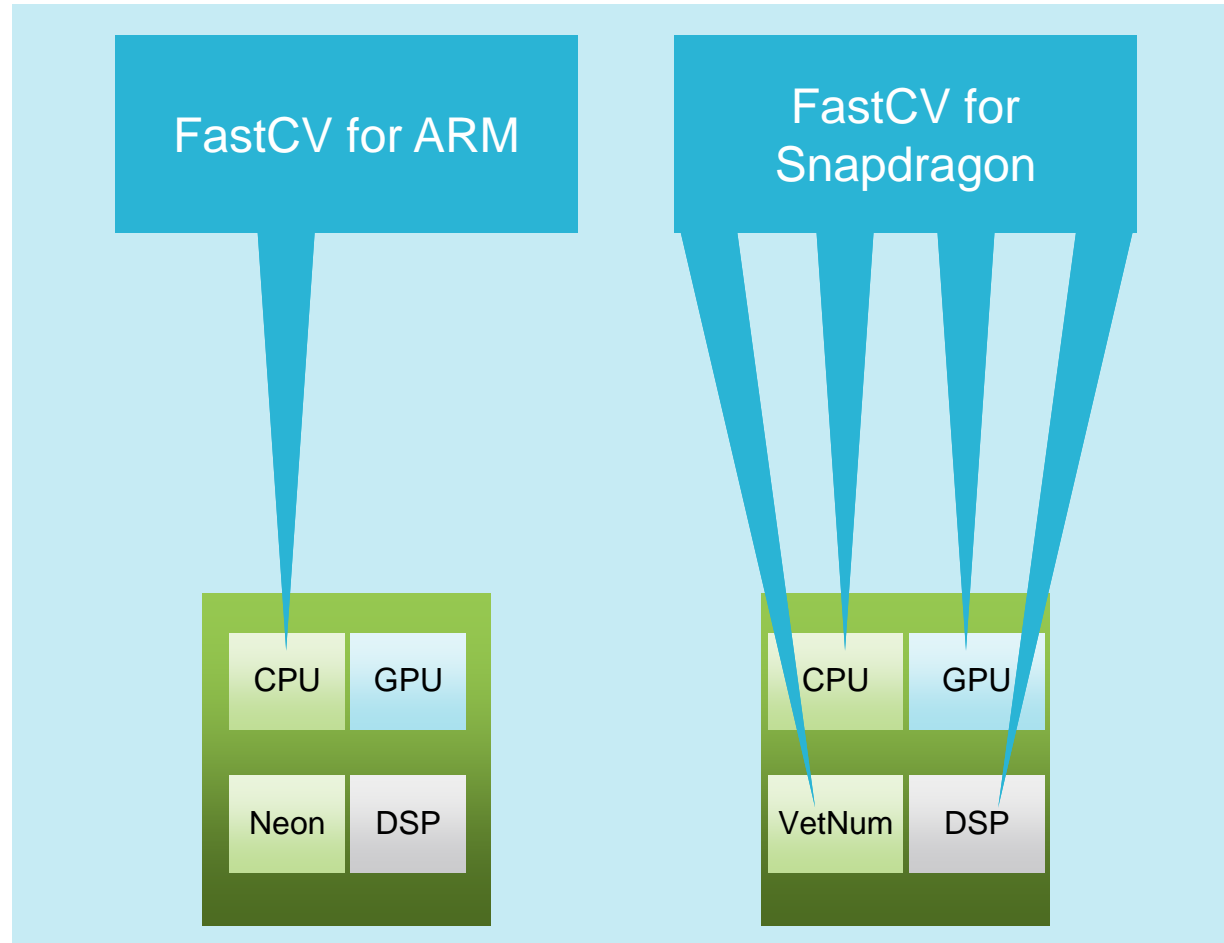


FastCV™ Overview

- An API & library which enables Real-Time Computer Vision (CV) applications.
- Enables mobile devices to run CV applications efficiently.
- Allows developers to HW accelerate their CV application.
- Analogous to OpenGL ES in the rendering domain
- Clean modular library.

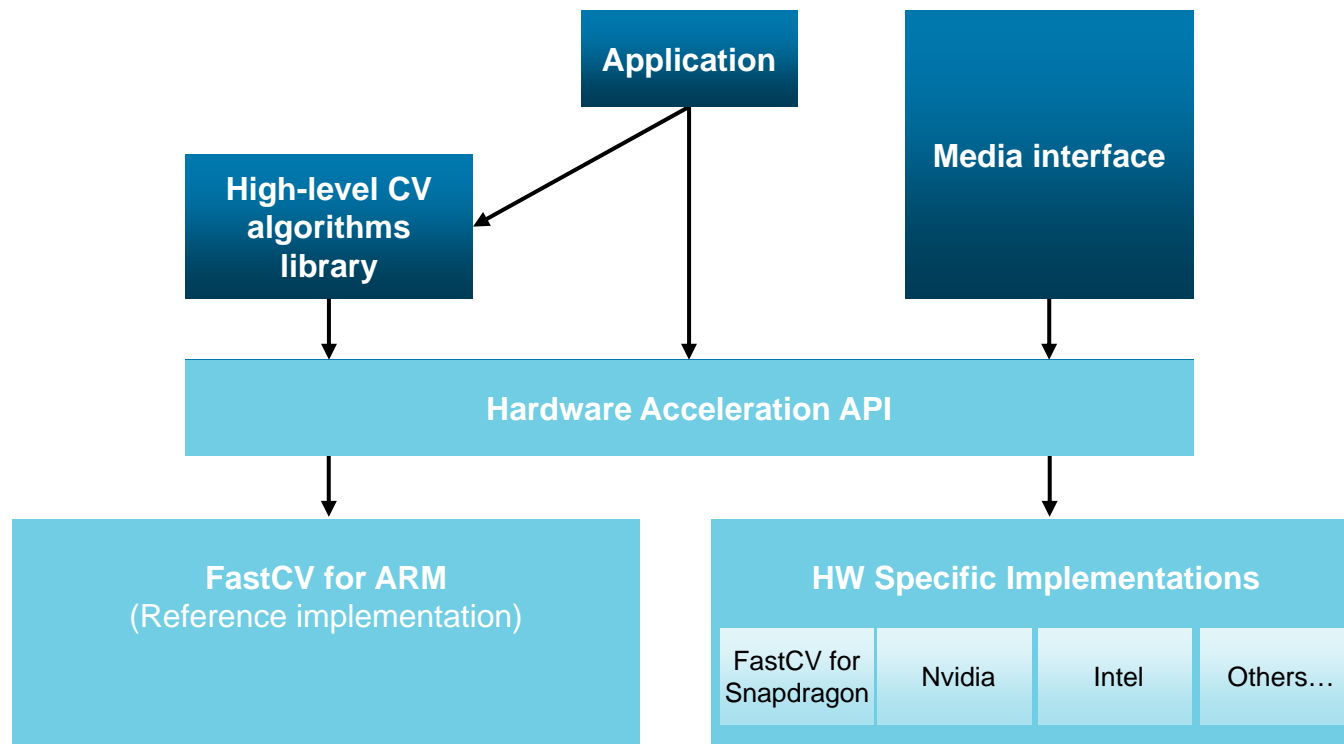
FastCV Value	Specifics
Smaller Scope	<ul style="list-style-type: none">▪ API's most widely used▪ API's most computationally intense▪ API's suitable for heterogeneous core optimization
Optimize for embedded/mobile	<ul style="list-style-type: none">▪ More granular API▪ Better power/performance

One Development Toolkit – Two Implementations



Industry Computer Vision Solutions

- FastCV is a processor-core agnostic acceleration API
- Khronos is looking to provide a standard CV API
 - Potentially utilizing portions of OpenCV
- FastCV will evolve as Khronos standard is defined



Open Source | Open Possibilities



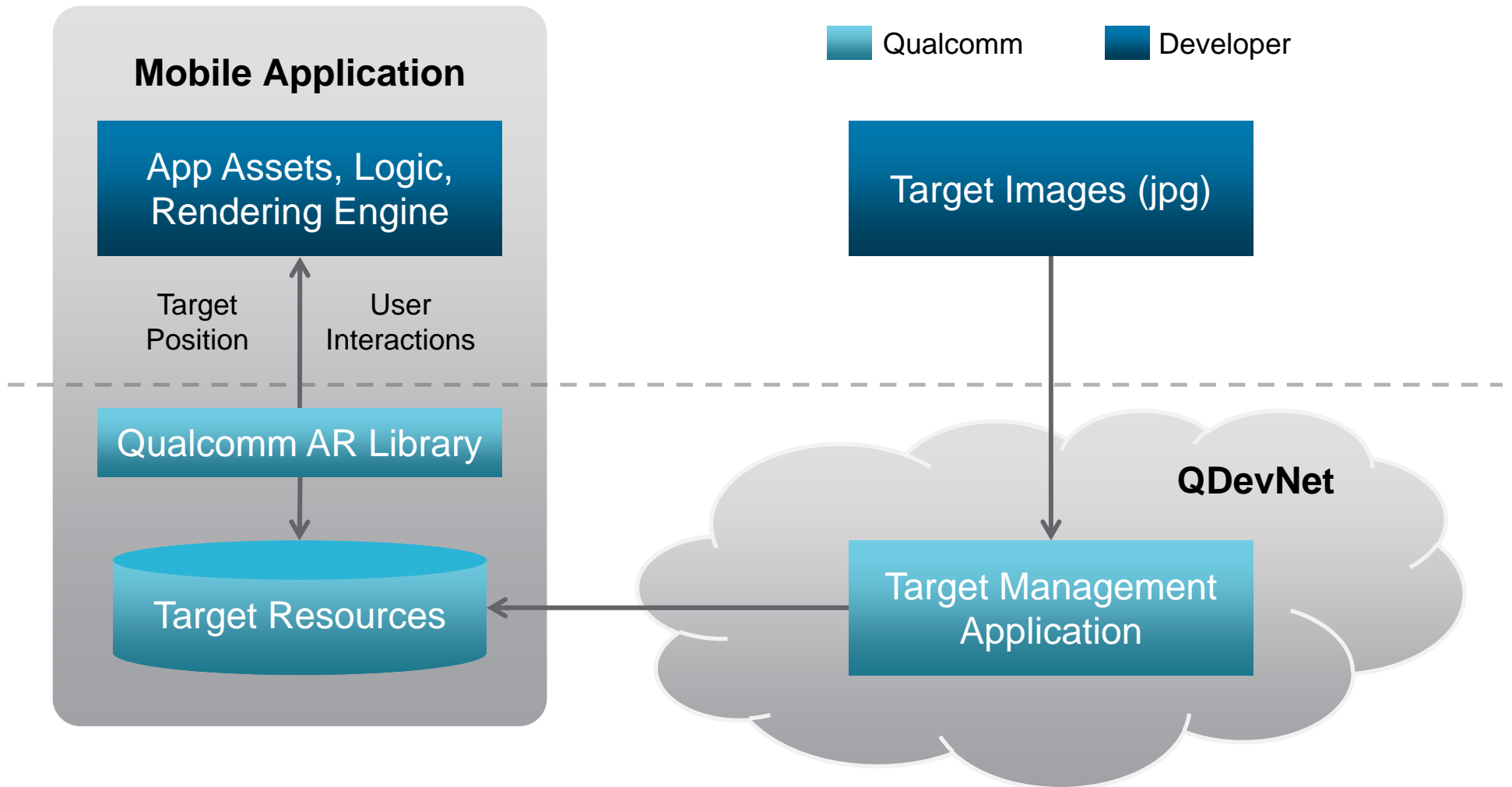
Augmented Reality Platform

Qualcomm AR Platform

- Empowers developers to create compelling mobile AR experiences based on Qualcomm's industry leading R&D.
- Enables a new generation of interactive 3D experiences which take place on top of everyday objects
- Why use the Qualcomm AR SDK?
 - Computer vision technology to tightly align graphics with printed surfaces and simple 3D objects
 - Support for multiple development tools including Eclipse, Xcode, and Unity
 - **Optimized for the various cores in the snapdragon chipset including the NEON co-processor.**



Qualcomm's AR Platform



Development Options

	Eclipse	Xcode	Unity 3
Supported OS	Android	iOS	Android / iOS
Advantages	<ul style="list-style-type: none"> ▪ Low level APIs provide enhanced flexibility and performance ▪ Supports integration with 3rd party game/rendering engines 		<ul style="list-style-type: none"> ▪ Fully integrated game engine reduces development time and cost ▪ Single app supports both iOS and Android
Devices	All Android 2.1 & above	iPhone 3GS iPhone 4 iPad2 iPod Touch (4th Generation)	N/A
Availability	Now	Now	Now

Download Qualcomm AR SDKs <http://developer.qualcomm.com/ar>

Open Source | *Open Possibilities*





Why Can't All Our Smartphones Discover & Connect in this Conference Room?

The core LAN technologies are already ubiquitous and in all our phones (Bluetooth, Wi-Fi)

.....
But...

Can your phone find me?

Can you share the music or pictures currently on your phone with me?

.....
Why not?

What Is AllJoyn™?

Open Source Framework to Enable Ad Hoc, Proximity-Based, Peer-to-Peer Networking



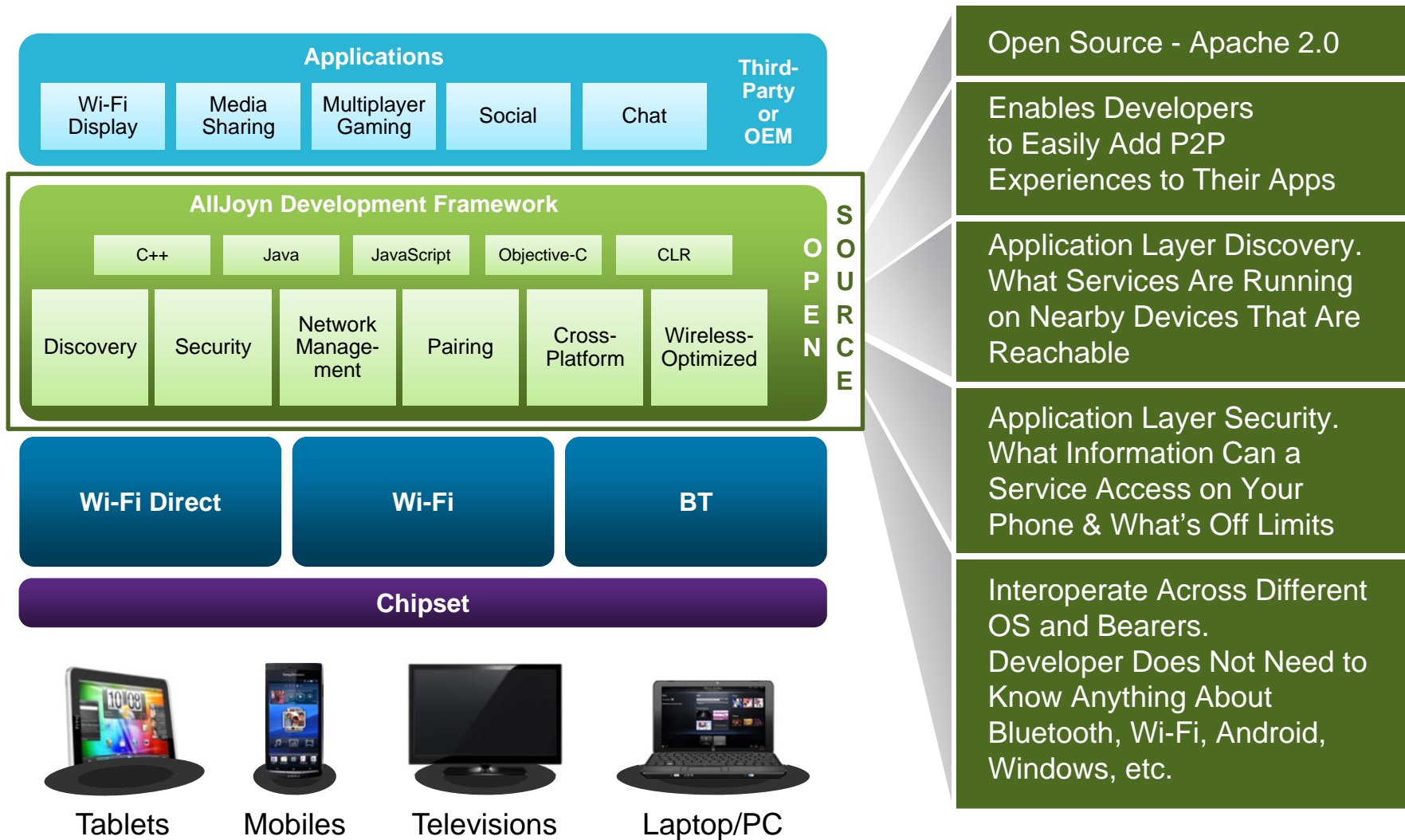
AllJoyn unleashes a whole new set of user experiences that span across mobile, tablets, PCs, TVs and more by abstracting all the complexity of P2P networking from developers

Why Peer-to-Peer (P2P) Is Hard

P2P Friction Developers Face Today



AllJoyn Is a Software Framework



Links for More Info

- Order Snapdragon developer platforms from BSQUARE
 - www.bsquare.com/qualcomm
- Learn more about Trepn Profiler at
 - developer.qualcomm.com/Trepn
- Learn more about Adreno Profiler at
 - developer.qualcomm.com/adreno
- Develop AllJoyn-Enabled Applications
 - alljoyn.org
- Build Your Vision with Qualcomm's Augmented Reality SDK
 - developer.qualcomm.com/ar

質問は？

Questions??

问题？

Questions?

Questions?

Fragen??

Domande??

Вопросы??

Spørsmål?

Questions?

Questions?

질문이 있습니까?

Spørgsmål?

Vragen?

問題？

Ερωτήσεις?

Cwestiynau?

¿Preguntas?

Kysymyksiä?

Questions?

FRÅGOR?

Questions?

Disclaimer

Nothing in these materials is an offer to sell any of the components or devices referenced herein. Certain components for use in the U.S. are available only through licensed suppliers. Some components are not available for use in the U.S.

Copyright © 2012 Qualcomm Innovation Center, Inc. All rights reserved. AllJoyn is a trademark of Qualcomm Innovation Center, Inc. Other product and brand names may be trademarks or registered trademarks of their respective owners.