



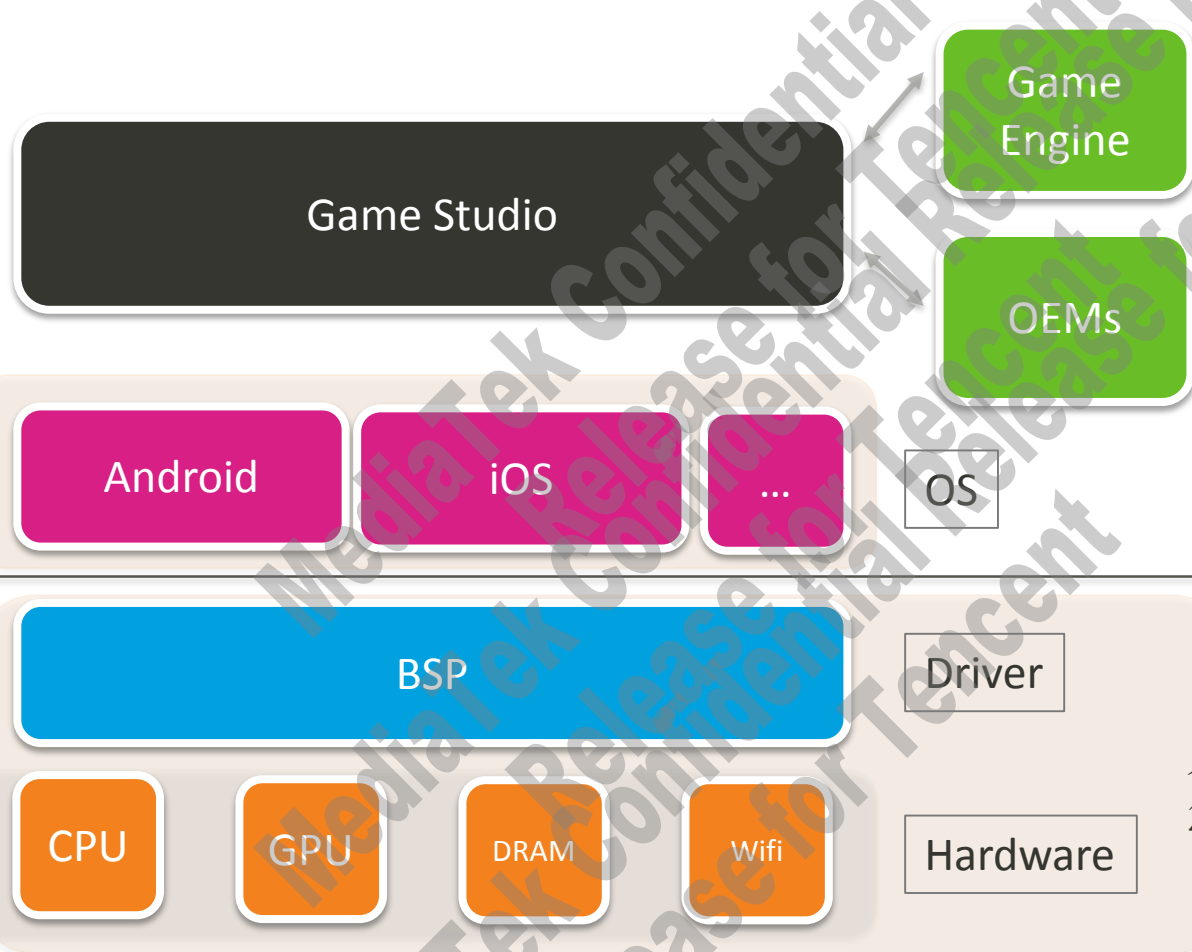
游戏性能分析

May.05.2018
Annu Wang

Outline

- 游戏性能分析
 - 流畅度
- 性能分析的流程
- 实例分享
- 总结

芯片系统架构



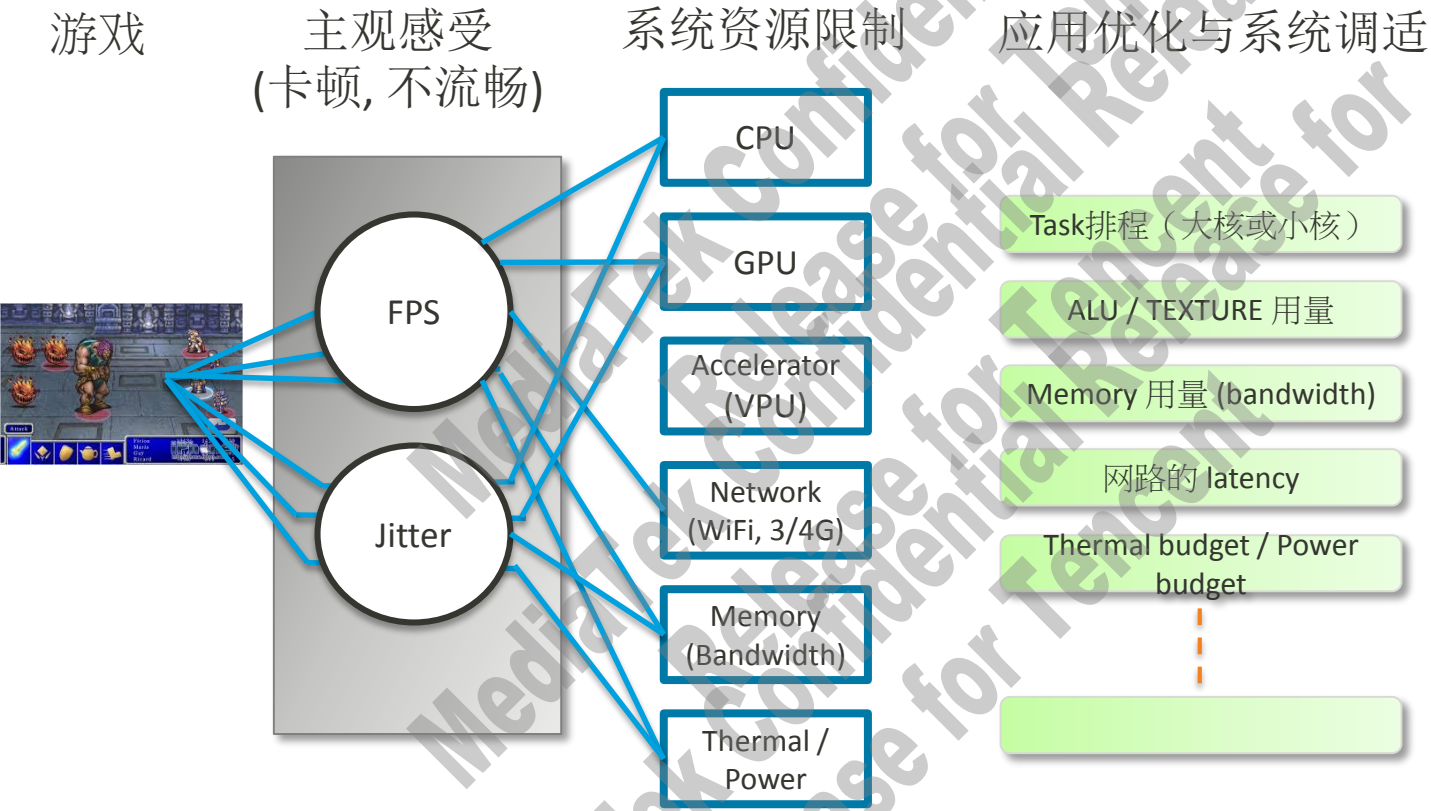
? 为什么一样的行为在平台间体验有差异

? 我要怎么分析性能问题



系统怎么解读卡顿的现象
怎么让游戏更流畅

芯片性能指标分析



游戏开发者

芯片平台的分析

性能分析流程 (以芯片角度)

确认场景

确认游戏设定 & 场景
确认操作手法 & 在意问题

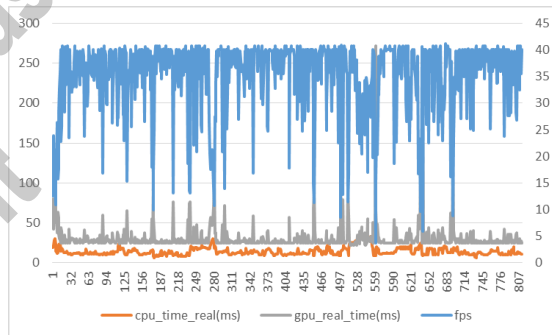
定位问题

初步定位问题 (CPU or GPU bound)
By MTK 内部工具
(Game_FPS_Tracker)

深入分析 & 结论

可能原因

CPU bound
GPU bound
Memory bound
Thermal
Wifi



性能分析流程 - CPU

分析threading 分布

确认task placement
确认task state
(runnable, running etc...)
By MTK internal Tool - MET

分析系统资源状态 &
结论

分析CPU频率
分析DRAM频率
分析BW影响 (影响 latency)
分析thermal影响
By MTK internal Tool - MET
分析wifi 状况
By Log



性能分析流程 - GPU

分析GPU bound原因

分析GPU bound 组成
(alu, tex, BW bound etc ..)

By

MALI – DS5

IMG – PVR Tune

分析系统资源状态 &
结论

分析CPU频率

分析DRAM频率

分析BW影响 (影响 latency)

分析thermal影响

By MTK internal Tool - MET

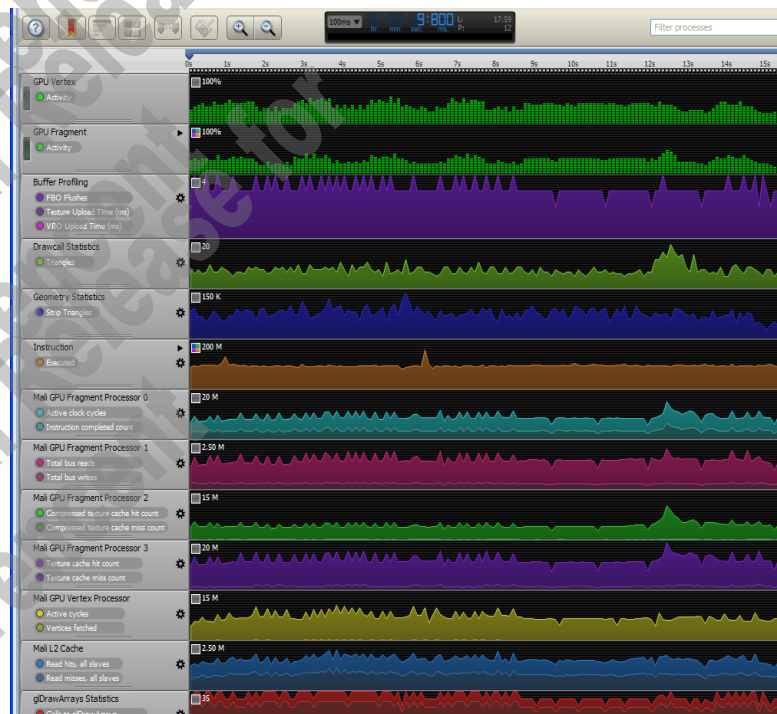
分析APP绘图行为

By

MALI – MGD

IMG – PVR Trace

MALI DS 5



实例分享 1

- P60 上, 吃鸡游戏(超清 + 自定义画质全开下) 画面卡顿



实例分享 1

确认场景

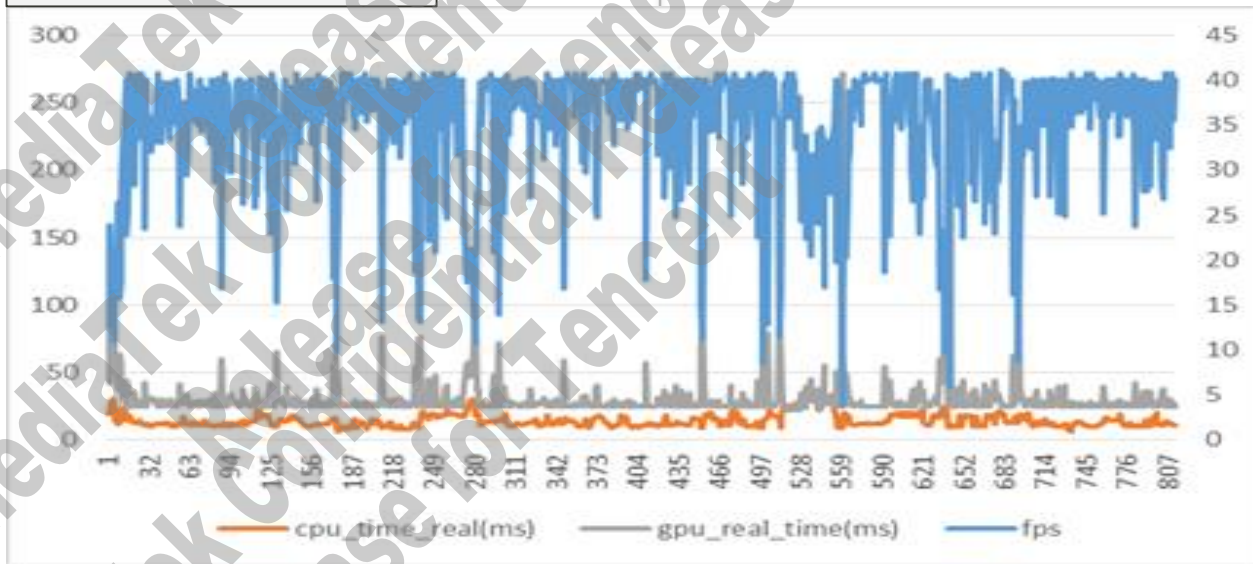
P60上, 长草区人员在转动时候
会卡顿

发现GPU时间表较长



定位问题

初步定位问题



实例分享 1

分析GPU bound原因

MALI DS5上观察到
GPU cycles已经接近硬件能力上限
从counter得到为Texture bound

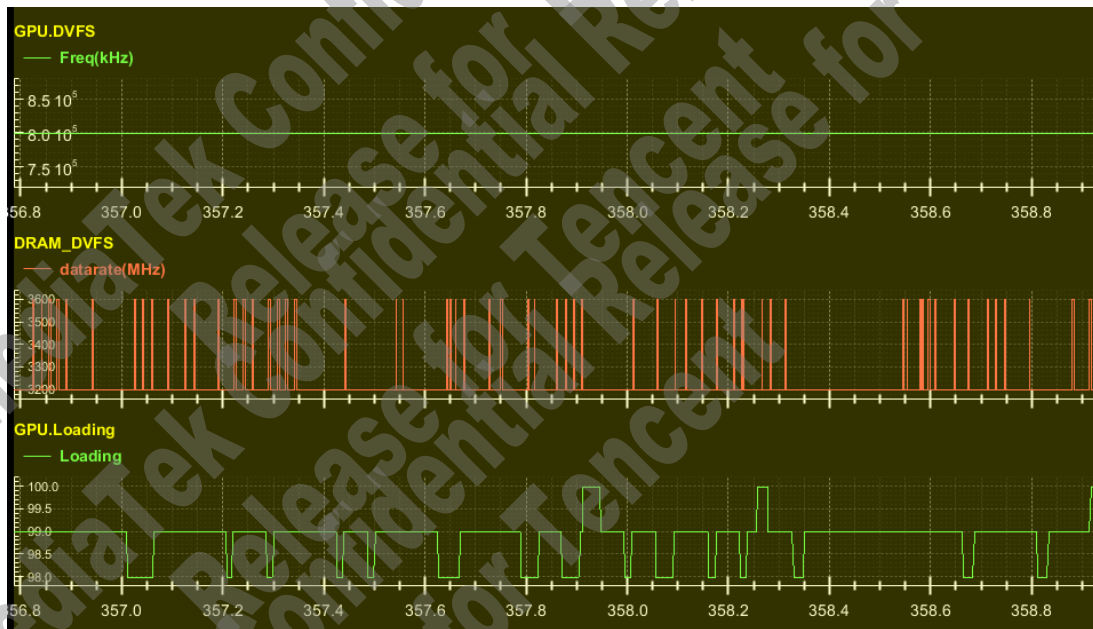


实例分享 1

分析系统资源状态 &
结论

分析CPU频率
分析DRAM频率
分析BW影响
(影响 latency)
分析thermal影响
(By Mediatek
MET)

检查其他系统资源状态是否会影响GPU行为? DRAM 频率,
BW & thermal 状况都没影响到GPU行为



实例分享 1

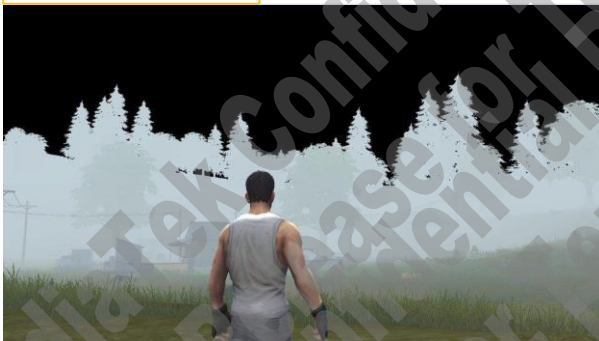
分析系统资源状态 &
结论

分析AP行为解释
MALI – MGD
IMG –PVR Trace

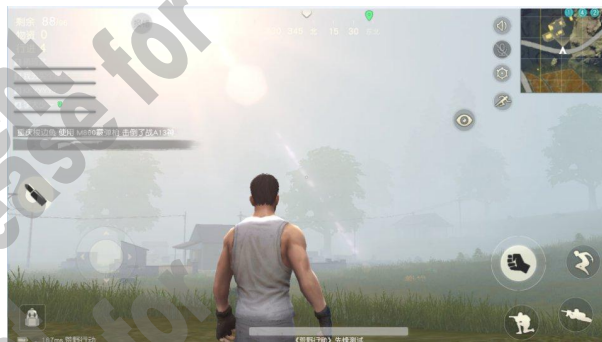
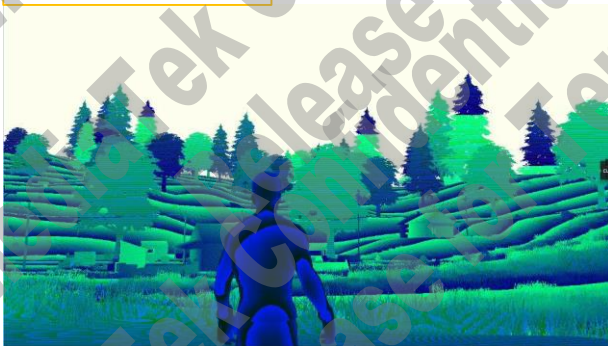
最終畫面

shader code用了很多texture运算
导致texture bound

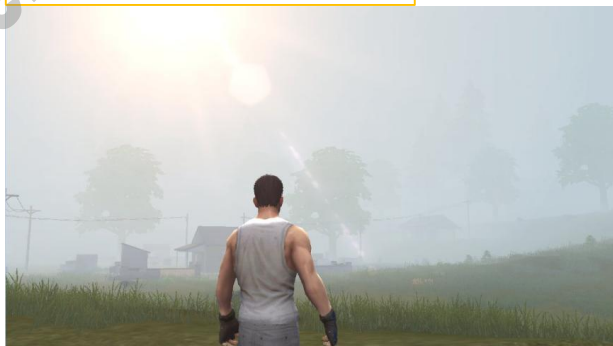
1. Original scene



2. Volumetric Fog



3. Volumetric Fog + Original scene



实例分享 2 - GLES APIs Analysis

地圖 - 60人 (with CA73@2GHz)	CPU time	# of GL calls
精細畫質	16.9	13290
普通畫質	16.5	12150
流暢畫質	13.9	9756

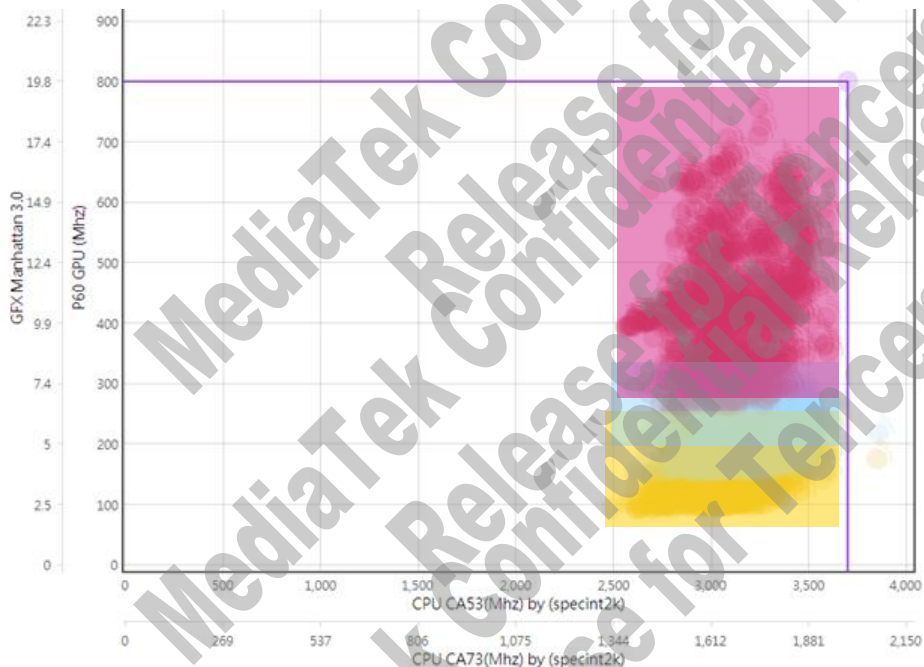
1. A73@2GHz，進行量測
2. glGetError 使用量過多，造成額外50% CPU loading
3. 一個人物，要使用40個gl calls (含4個draw calls)完成，地圖上人物從0人到60人，會多出額外3.2倍CPU loading

流暢畫質					
地圖 - 0人			地圖 - 60人		
Name ▾	CPU Duration (ms)	Occurrences	Name ▾	CPU Duration (ms)	Occurrences
glGetError	1.22	1680	glGetError	4.71	6504
glDrawElements	0.59	25	glBufferSubData	2.99	189
glClear	0.41	3	glDrawElements	2.50	269
glBindTexture	0.14	121	glVertexAttribPointer	0.79	834
glBufferSubData	0.13	7	glBindBuffer	0.69	567
glBindBuffer	0.11	71	glClear	0.64	3
glDrawElementsBaseVertex	0.11	13	glUniformMatrix4fv	0.60	717
exOES					
Total	3.42	2520	Total	13.89	9756
Total excluding glGetError	2.20	840	Total excluding glGetError	9.18	3252
Ratio	64%		Ratio	66%	

driver cost 417%成長

实例分享 3 特效设定的调适

- CPU 需求介於 CA73 1.3G-1.9 GHz，需要CPU大核才能流暢遊玩
- GPU 變化較大，但按不同設定可滿足不同手機的需求
 - 其中 Level 2:4:5 = 1 : 1.2 : 3，level 4-5 有很大的 gap



总结

- 遊戲效能優化

- 芯片效能分析 (CPU/GPU/Memory)

- CPU 相關的優化

- GPU 相關的優化 (ALU / TEXTURE 用量)

- GLES APIs / Vulkan APIs / Shader analysis

- 功耗優化

- 個別遊戲場景的分析

- 特效設定與幀率的調適

everyday genius