

# Android Tools for Developers

Paul Marois - GenYmobile



# Introduction

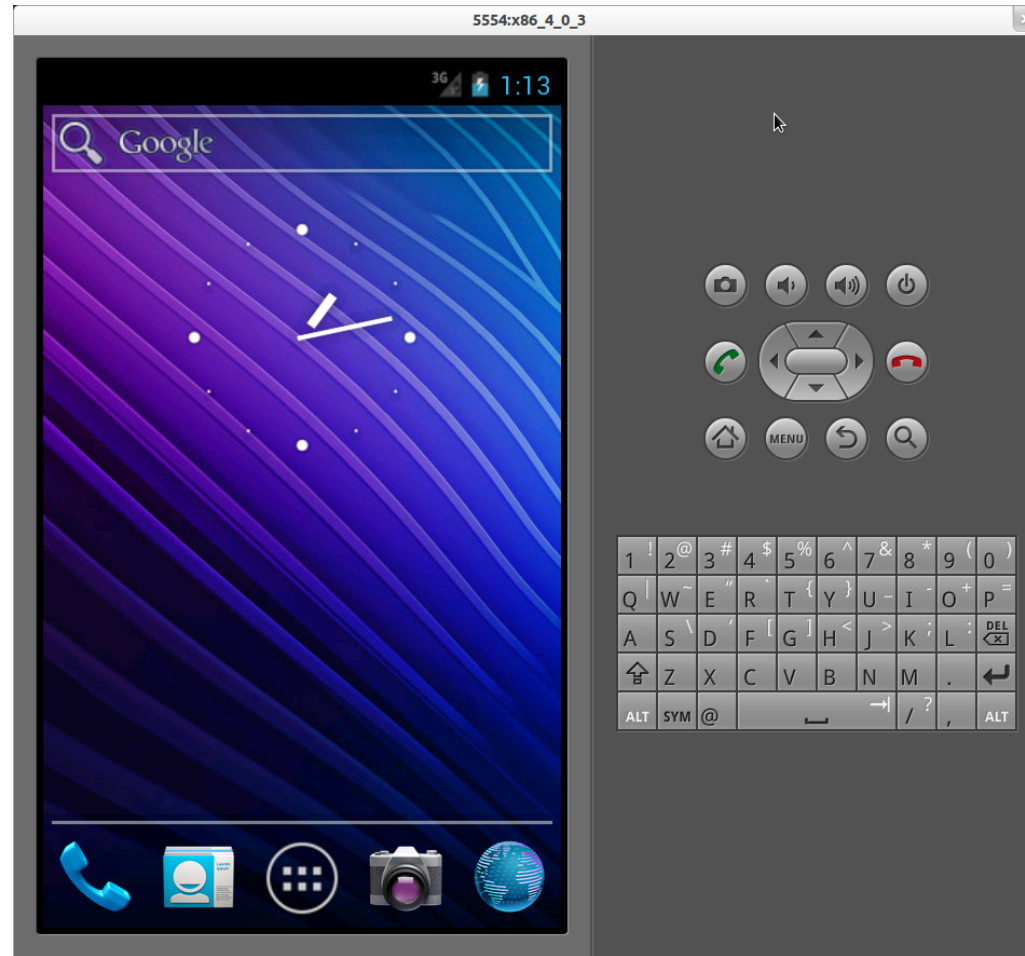
## Who Am I ?

- Android Developer.
- Android GENYMOBILE employee.

## What this presentation will be ?

- A totally subjective presentation about **official** tools I find interesting or useful.
- Not about tools presented elsewhere in OWF.

# Emulator



# Emulator

## Latest addition

### GPU Support

- SDK tools > 17
- emulator -avd <avd\_name> -gpu on

### Improved CPU performance

### More hardware feature emulation

- SDKController in tools/apps/
- adb forward tcp:1970 localabstract:android.sdk.controller
  - MultiTouch Input Emulation
  - Sensor Emulation

# Emulator

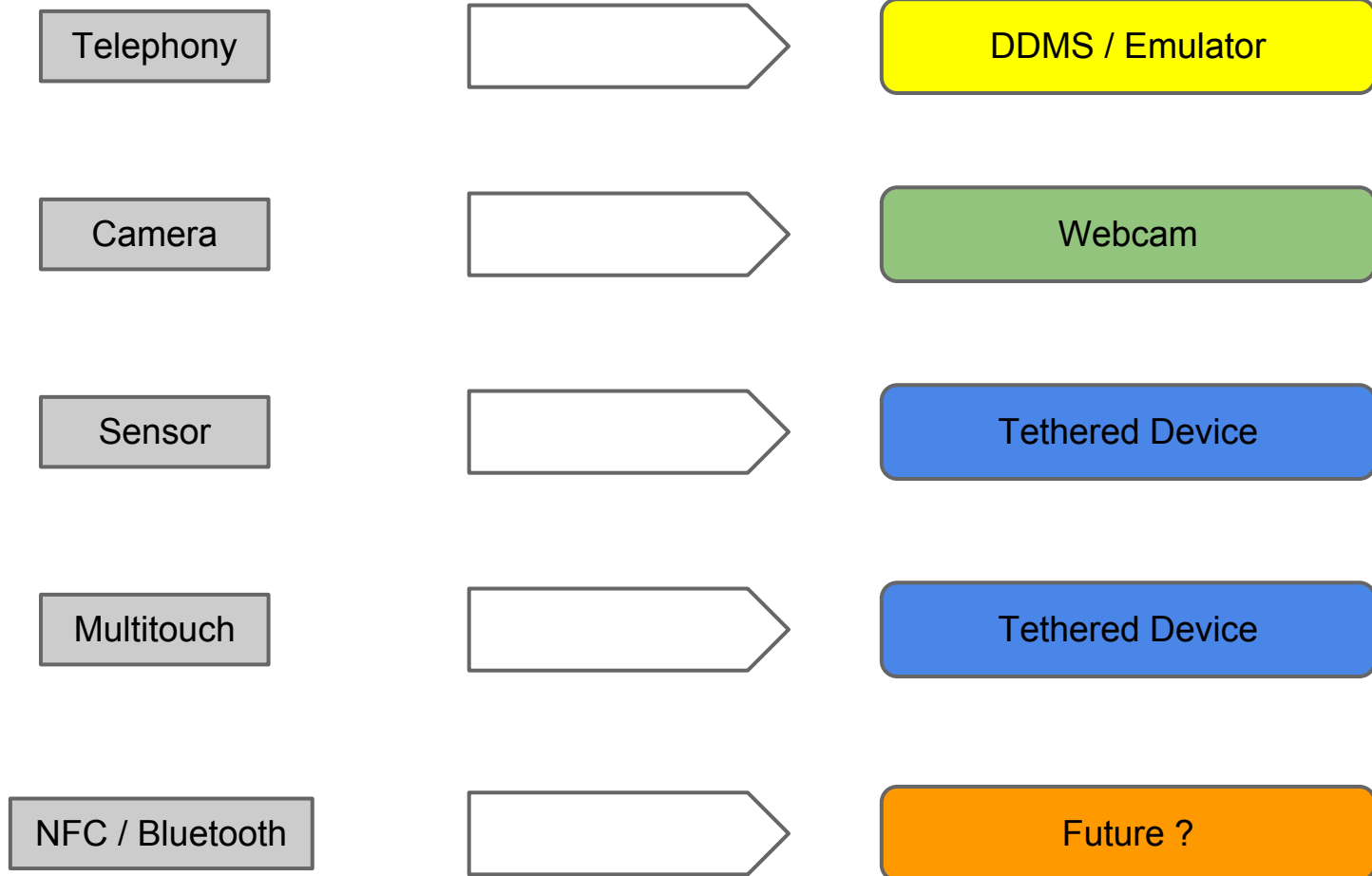
## Emulator Telnet Interface

```
paul@paul-Studio-XPS-1645:~$ telnet localhost 5554
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Android Console: type 'help' for a list of commands
OK
help
Android console command help:

  help|h|?      print a list of commands
  event         simulate hardware events
  geo           Geo-location commands
  gsm           GSM related commands
  cdma          CDMA related commands
  kill          kill the emulator instance
  network       manage network settings
  power         power related commands
  quit|exit     quit control session
  redir         manage port redirections
  sms           SMS related commands
  avd           control virtual device execution
  window       manage emulator window
  qemu         QEMU-specific commands
  sensor       manage emulator sensors

try 'help <command>' for command-specific help
OK
```

# Emulator





# New Wizard & Layout Configuration


New Android App


## New Blank Activity



Creates a new blank activity, with optional inner navigation.





Activity Name  MainActivity


Layout Name  activity\_main


Navigation Type  Tabs + Swipe

Hierarchical Parent  

Title  MainActivity

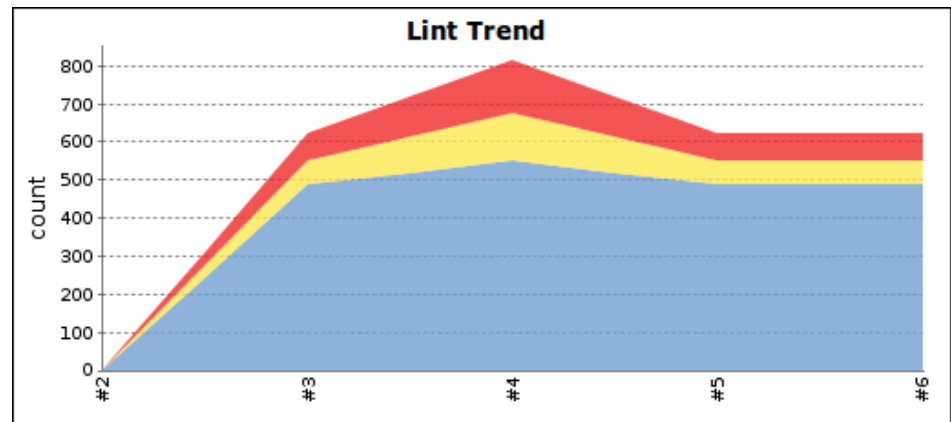


 The name of the activity class to create

 < Back Next > Cancel Finish

# Lint

- Available in ADT > 16.
- Bug detector.
- Multiple Inputs file types : xml, java, ...
- Multiple Output file types : text, html, xml, ...
- Multiple Front-Ends : Eclipse, IntelliJ, command line.
- Multiple severity warnings.
- Jenkins Plugin.
  - <https://wiki.jenkins-ci.org/display/JENKINS/Android+Lint+Plugin>
- API to write additional checks.





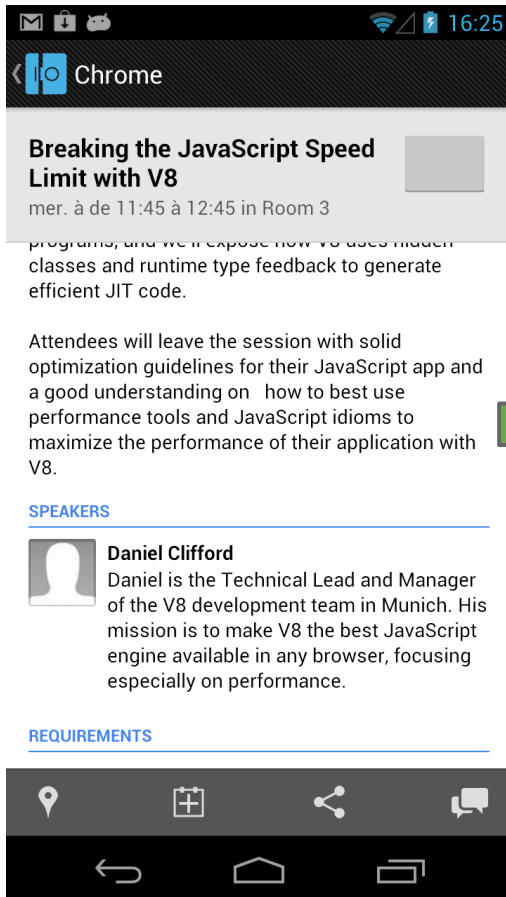
# Lint

## Create Your Own Rules

- Import lint\_api.jar
- Create a detector which can identify one or more different types of "issues" that you'll create too.
  - *public class MyDetector extends ResourceXmlDetector*
- Provide your own registry.
  - *public class MyIssueRegistry extends IssueRegistry*
- Create your manifest
  - *Lint-Registry: owf.demo.MyIssueRegistry*
- Put the jar into ~/.android/lint folder

# Device Monitor

## HierarchyViewer




**Breaking the JavaScript Speed Limit with V8**  
mer. à de 11:45 à 12:45 in Room 3

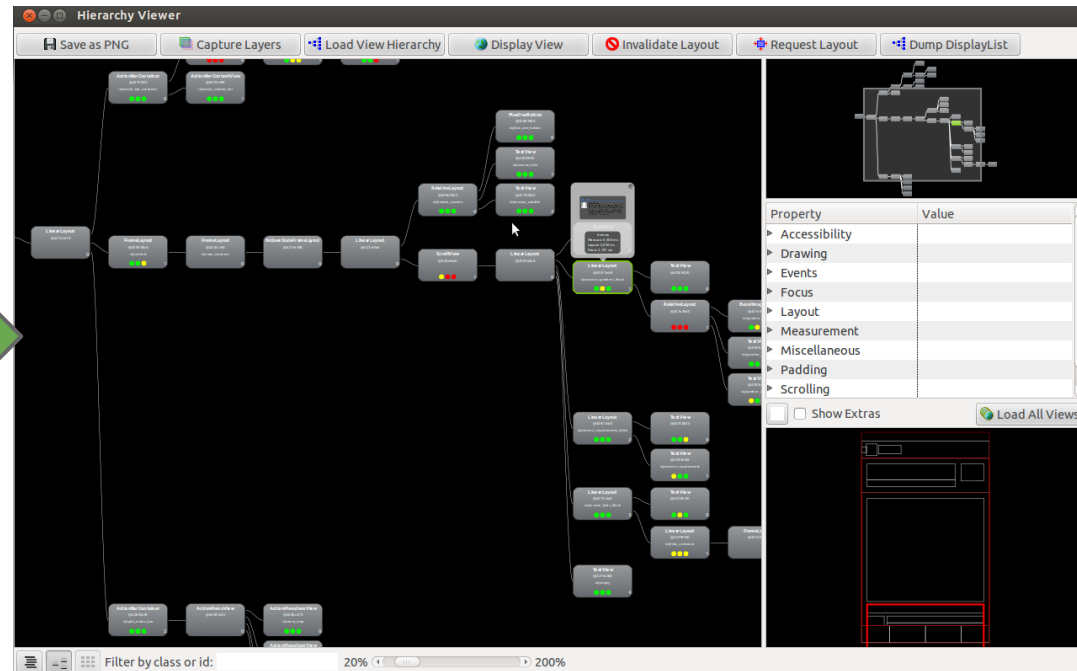
programs, and we'll expose how V8 uses hidden classes and runtime type feedback to generate efficient JIT code.

Attendees will leave the session with solid optimization guidelines for their JavaScript app and a good understanding on how to best use performance tools and JavaScript idioms to maximize the performance of their application with V8.

**SPEAKERS**

 **Daniel Clifford**  
Daniel is the Technical Lead and Manager of the V8 development team in Munich. His mission is to make V8 the best JavaScript engine available in any browser, focusing especially on performance.

**REQUIREMENTS**



# Device Monitor

## HierarchyViewer

- Do not work on devices for security reason
  - <https://github.com/romainguy/ViewServer>
  - Requires Internet Permission
  - Mode Debug only
- onCreate :
  - `ViewServer.get(this).addWindow(this);`
- onDestroy
  - `ViewServer.get(this).removeWindow(this);`
- onResume
  - `ViewServer.get(this).setFocusedWindow(this);`
- Customizable
  - `@ViewDebug.ExportedProperty`

# Device Monitor

## adb command lines

- *Adb bugreport*
  - Complete log of the system.
- *adb shell dumpsys* : log of services
  - ex : *adb shell dumpsys meminfo <pid>*

Exs :

SurfaceFlinger	permission
meminfo	batteryinfo
alarm	phone
network_management	bluetooth
appwidget	power
nfc	gfxinfo
audio	vibrator
notification	hardware
battery	wallpaper
	...

# Device Monitor

Ex : adb shell dumpsys meminfo <pid>

\*\* MEMINFO in pid 15697 [com.google.android.apps.iosched] \*\*

	Pss	Shared Dirty	Private Dirty	Heap Size	Heap Alloc	Heap Free
Native	16	24	16	16952	6971	468
Dalvik	10433	11836	9976	18823	14637	4186
Cursor	16	0	16			
Ashmem	0	0	0			
Other dev	9884	1100	8616			
.so mmap	3613	1880	820			
.jar mmap	0	0	0			
.apk mmap	220	0	0			
.ttf mmap	327	0	0			
.dex mmap	2024	0	0			
Other mmap	1746	16	48			
Unknown	7575	380	7568			
TOTAL	35854	15236	27060	35775	21608	4654

Objects

Views:	199	ViewRootImpl:	1
AppContexts:	6	Activities:	2
Assets:	2	AssetManagers:	2
Local Binders:	43	Proxy Binders:	21
Death Recipients:	0		
OpenSSL Sockets:	3		

SQL

MEMORY_USED:	549		
PAGECACHE_OVERFLOW:	331	MALLOC_SIZE:	62

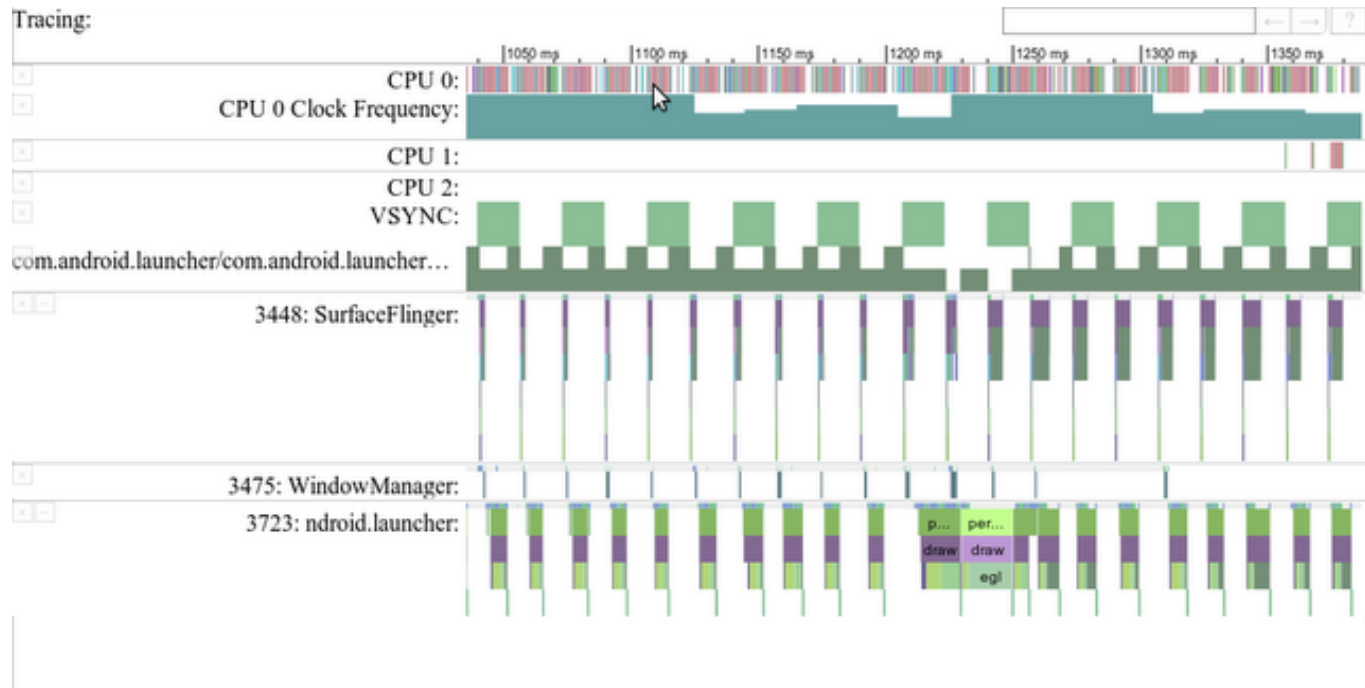
DATABASES

pgsz	dbsz	Lookaside(b)	cache	Dbname
4	40	18	0/33/1	/data/data/com.google.android.apps.iosched/databases/webview.db
4	552	280	4052/103/17	/data/data/com.google.android.apps.iosched/databases/schedule.db

# Device Monitor

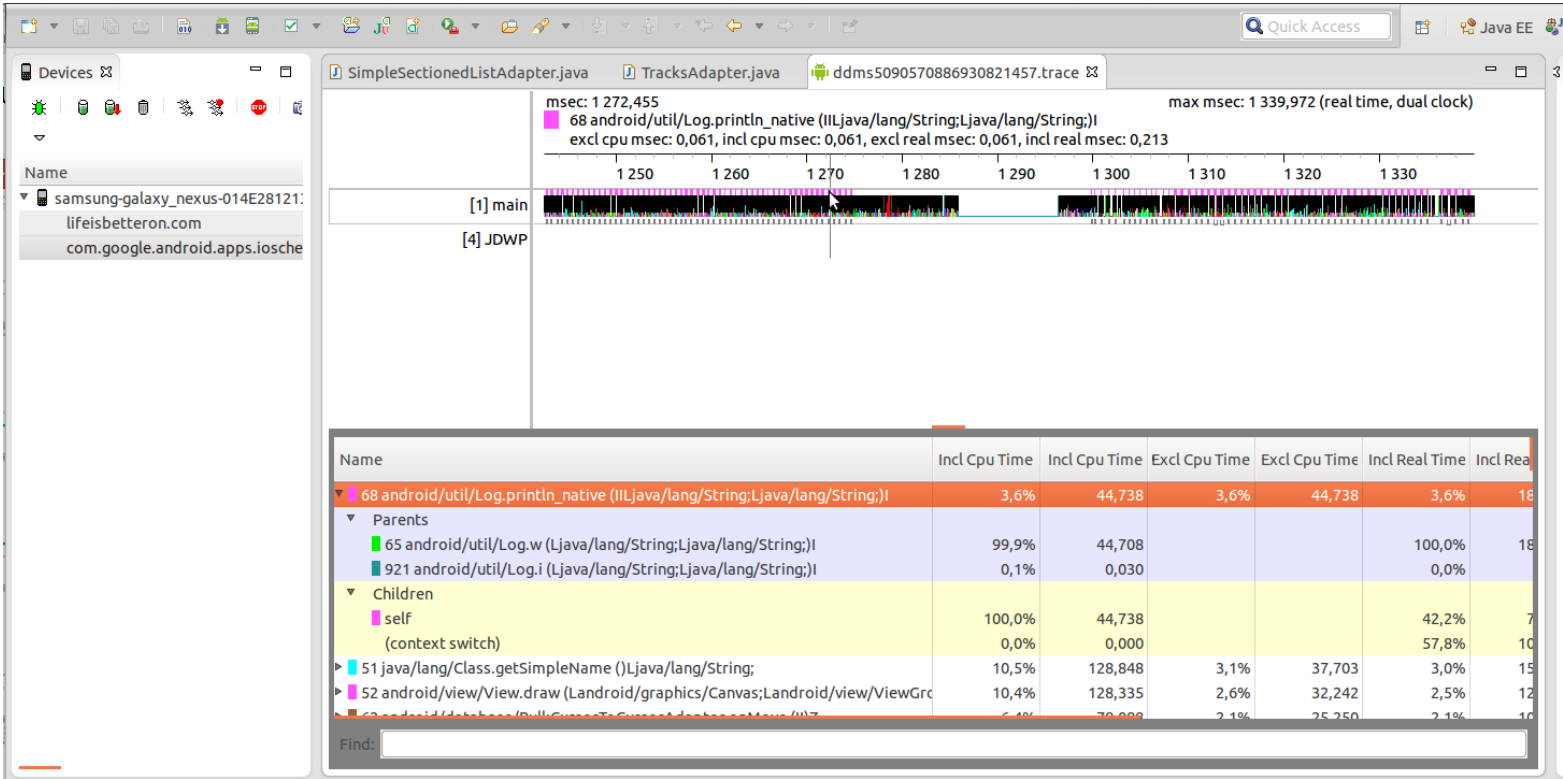
## Systrace

- Settings > Developer Options > Enable Traces
- `sdk/tools/systrace`
- `python ./systrace.py`



# Device Monitor

Traceview



# Device Monitor

## Monkey

- Generate pseudo-random streams of user events on an emulator or device
- *adb shell monkey [options] <event-count>*
  - ex : *adb shell monkey -p com.google.android.apps.iosched -v 500*



# And more...

- Draw9patch
- OpenGLTracer
- Procrank
- DexGuard
- AndroVM
- AndroidAnnotations
- RenderScript
- **ETC...**

Thanks

Q & A

# Bonus

## MAT

