Optimizing Storage Performance on Android

Doug Dumitru, CTO
EasyCo LLC

http://easyco.com
doug@easyco.com
+1 610 237-2000 x43

What is Android?

- An ARM-based Linux host
  - Java virtual machine for apps
- 1-4 CPU cores
  - 600MHz to 1.5GHz
- 256MB to 2GB ram
- Internal block storage device
  - 1GB to 64GB
  - MLC or TLC Flash based

Android Flash Storage

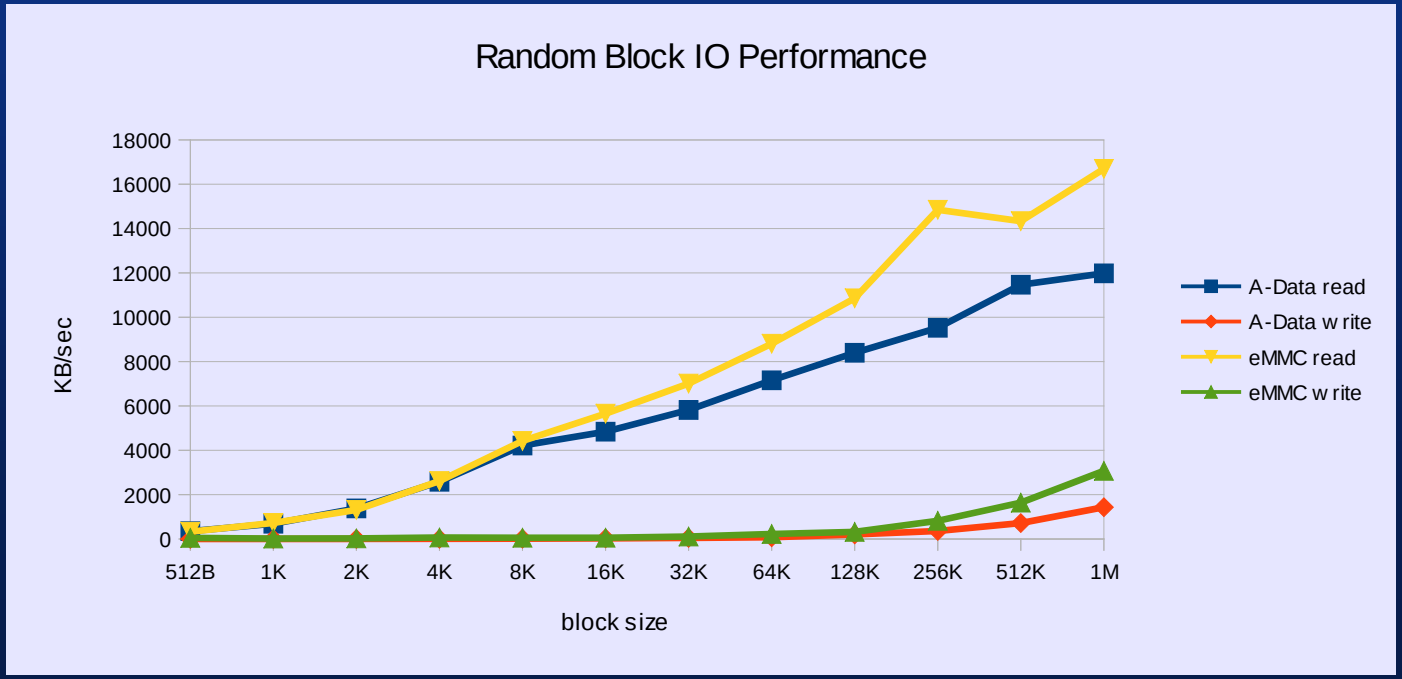
- Typically eMMC or eSD modules
- Basically an MMC or SD card as a chip
- Performance similar to many USB sticks
- Wide variability from part to part

Android Storage Performance

- **Linear reads at 10+ MB/sec**
- **Random reads at 700+ IOPS**
  - **Better than desktop hard disks**
- **Linear writes at 8+ MB/sec**
- **Random writes as low as 1.3 IOPS**
  - ***Ugh!***

Some Benchmarks

- **Synthetic IOPS vs Block Size**
  - **A-Data uSDHC card**
  - **Toshiba eMMC module**



… Filesystem Benchmarks

- PostMark v 1.50

	Total Run Time	Trans Time	Trans / sec	Read MB/sec	Write MB/sec
Bare A-Data card	15714 sec	15666 sec	1	0.27	0.28
Bare System eMMC internal storage	4008 sec	3961 sec	6	1.03	1.08

What This Does to System Responsiveness

- **It is not just that writes are slow, but that the system has to wait for them to complete.**
- **This introduces IO latency**
- **A handful of writes with the wrong pattern can "lock up" the IO channel for 5+ seconds.**

It Can't Be That Bad ....

- **It is actually worse**
- **During filesystem benchmarks:**
  - **eMMC reported 'await' > 24 seconds**
  - **uSDHC card reported 'await' > 72 seconds**
    - **Applications were crashing because of timeouts**

Proposed Fixes

- **Tune out writes**
  - **No swap**
  - **'noatime' mount option**
  - **'ext4' with journal file system**
- **Pick “better” storage devices**
- **Exotic solutions**
  - **One paper proposed hybrid phase/change solution**
- **The bottom line - Avoid Writes ;)**

EasyCo's Solution

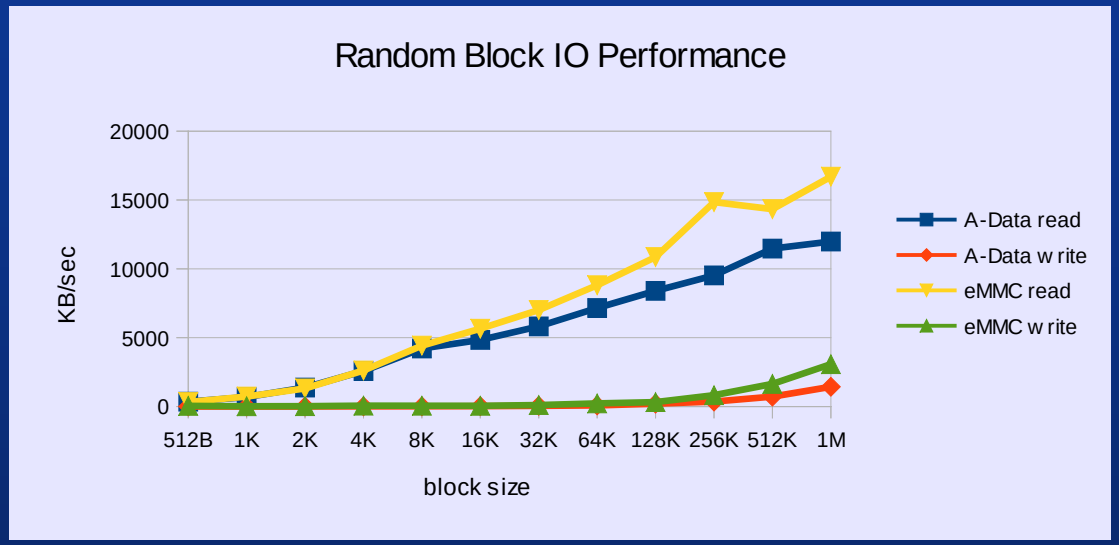
"Flash SuperCharger"

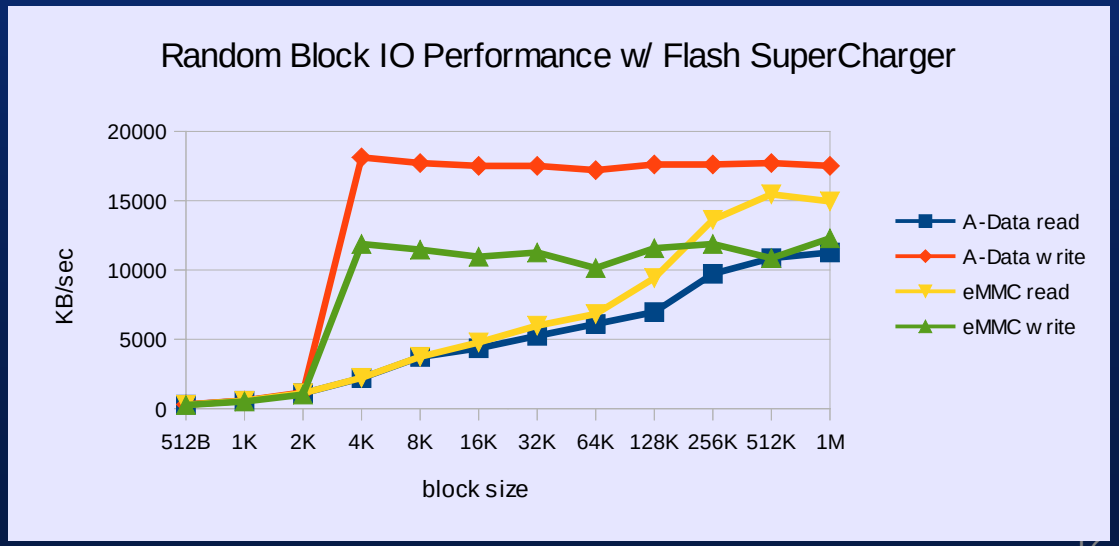
- **Virtual flash controller**
  - **Runs on host CPU/RAM**
  - **Uses stock block storage**
  - **Transparent block filter**
- **No hardware changes**
- **No application changes**
- **Only block/partition layout changes**

Flash SuperCharger Random Write Performance

- All writes become linear writes to the media
- Random writes at linear speed
  - 8MB/sec device yields 2000 4K random write IOPS
- Random writes are much faster than random reads.
  - System flush daemon has trouble actually building up a queue
  - 'await' during file system benchmark < 50ms

Benchmarks Again





FileSystem Benchmarks

	Total Run Time	Trans Time	Trans / sec	Read MB/sec	Write MB/sec
Bare A-Data card	15714 sec	15666 sec	1	0.27	0.28
Bare System eMMC internal storage	4008 sec	3961 sec	6	1.03	1.08
A-Data card w/ Flash SuperCharger	178 sec	172 sec	145	23.27	24.29
System eMMC storage w/ Flash SuperCharger	278 sec	272 sec	91	14.78	15.60

How this changes Android

- Swap is fine
- 'noatime' can still be used but really does not matter
- Local storage is no longer toxic
- Go ahead and write ;)

Benefits

For ODMs

- More flexible hardware choices
- Lower total BOM costs
- Lower DRAM requirements
- Lower Flash specs
- Longer Flash Life

For Developers

- Robust Local Storage
- More opportunities for off-net devices
- Less reliance on the cloud

Benefits

For Carriers

- Lower network bandwidth
  - Caching local proxy

For Users

- More responsive devices
- Lower cost
- Longer life
- New classes of data centric applications

For More Information:

Doug Dumitru, CTO
EasyCo LLC

http://easyco.com

doug@easyco.com

+1 610 237-2000 x 43