Android traffic analysis & eMMC performance

Elixir Flash Technology

Proposal by 2012 . 10 . 15

EFTech

Contents

- Traffic patterns of two android verions
- Emmc performance comparison
- eMMC 4.5 features

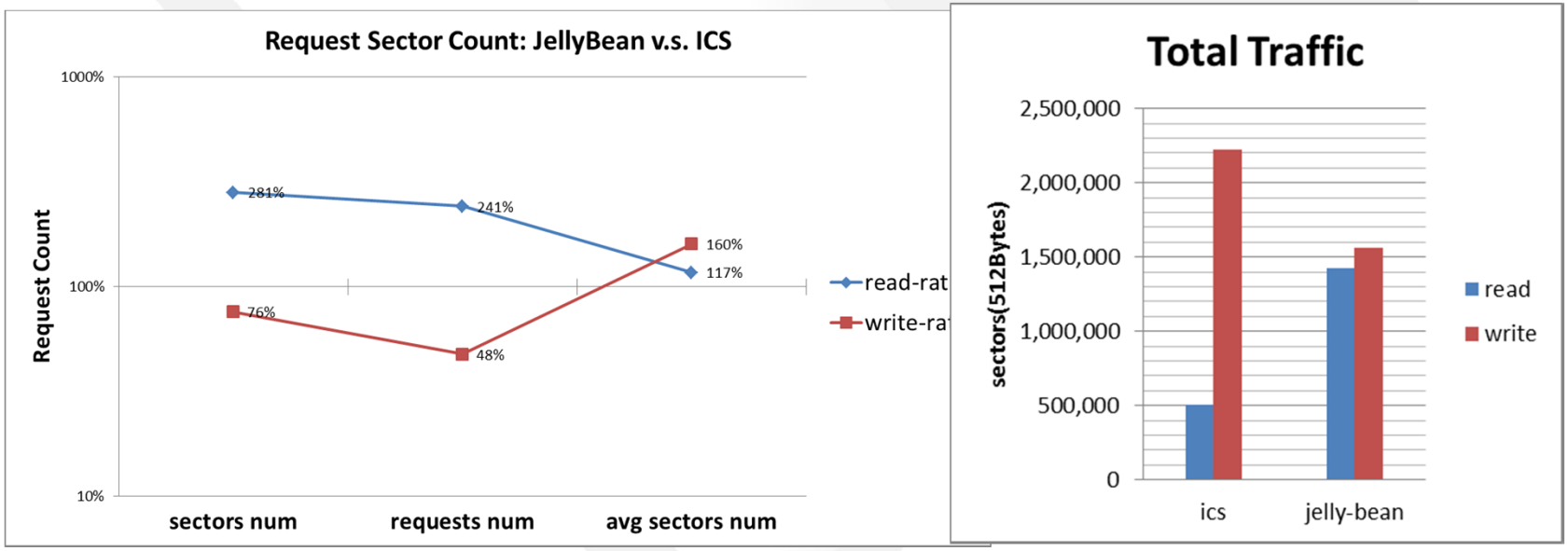
Traffic Analysis

Access pattern

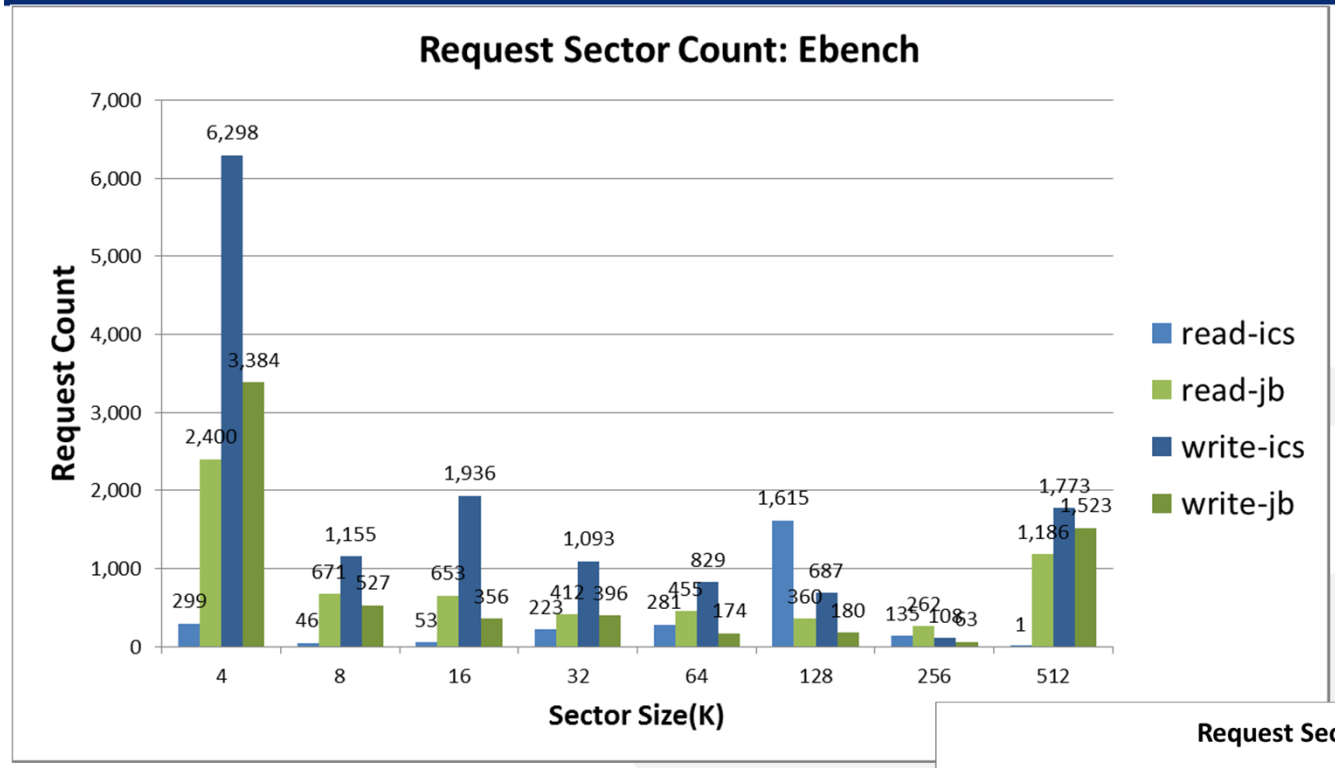
- Ebench : app-based bench
- Two Anroid versions : Jelly bean v.s. ICS
  - Jelly bean on Nexus 7
  - ICS on Mango board

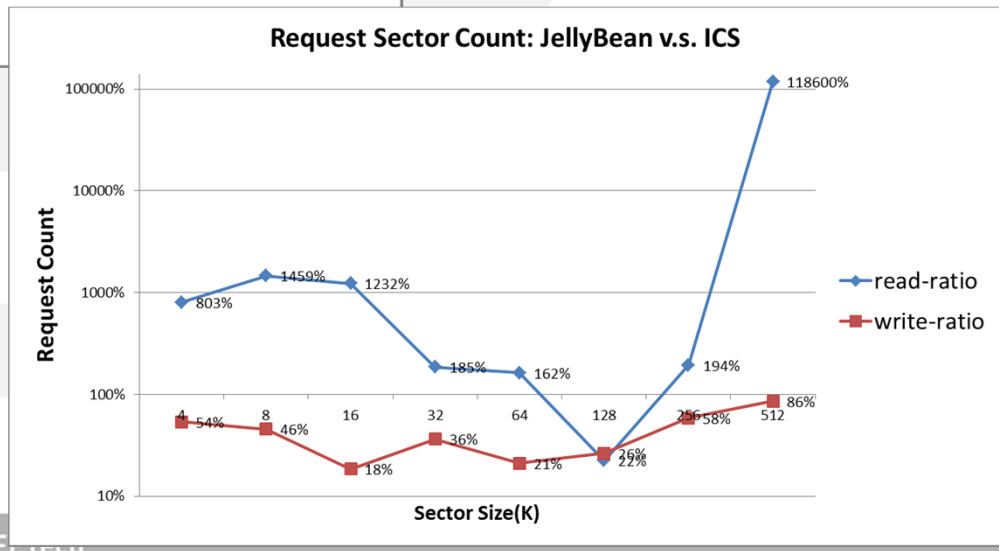
Pattern comparison

- Jelly-bean to ICS
  - More reads, less writes, Bigger chunk



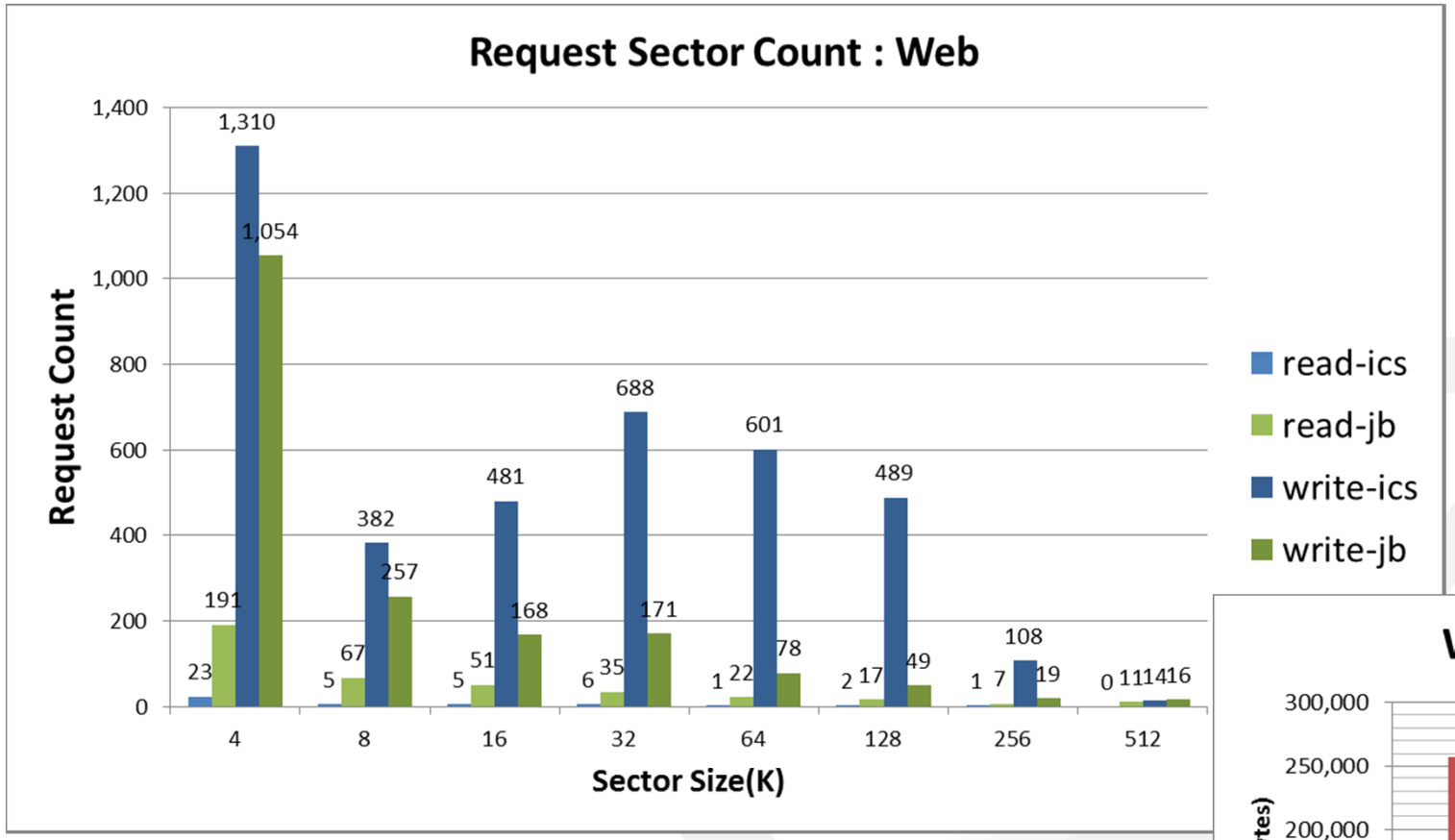
Pattern comparison : sector_size 별

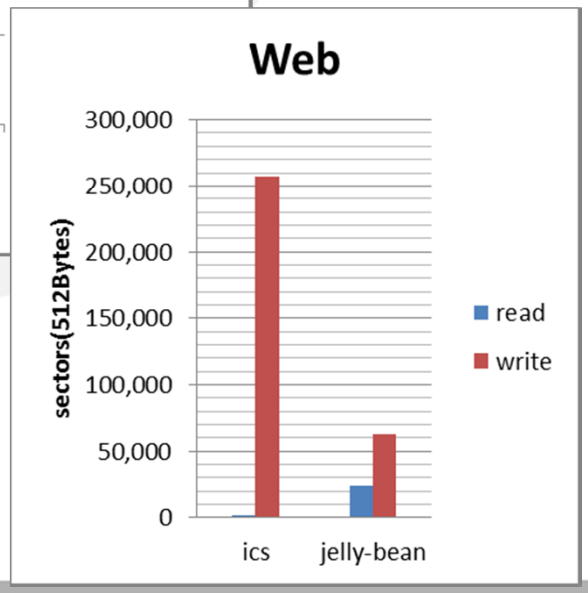




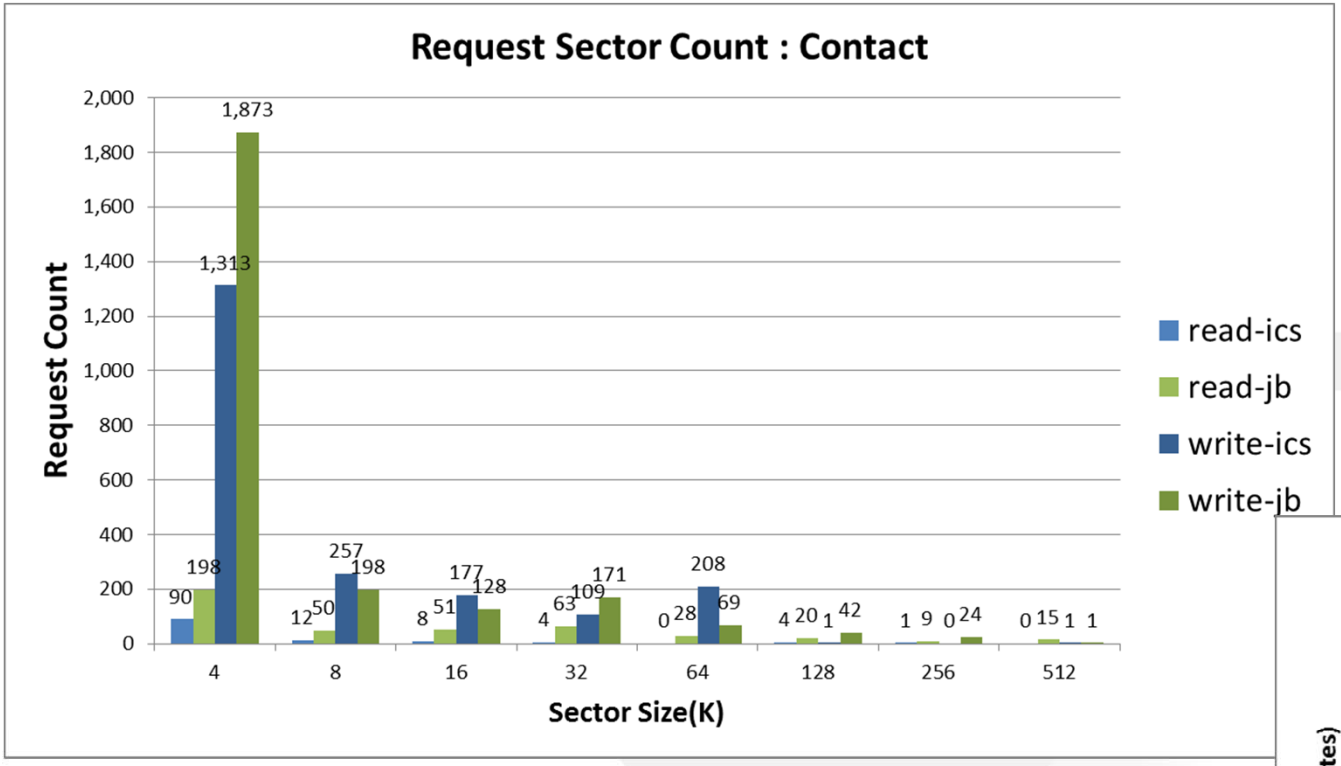
- JB starts generating 512bytes read

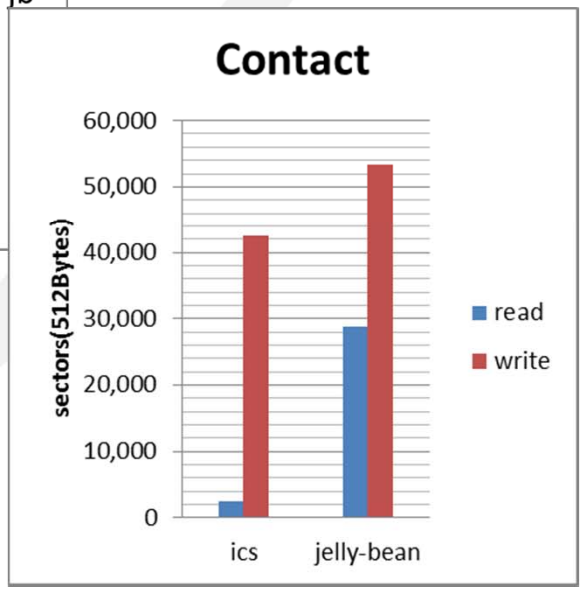
Pattern comparison: Web



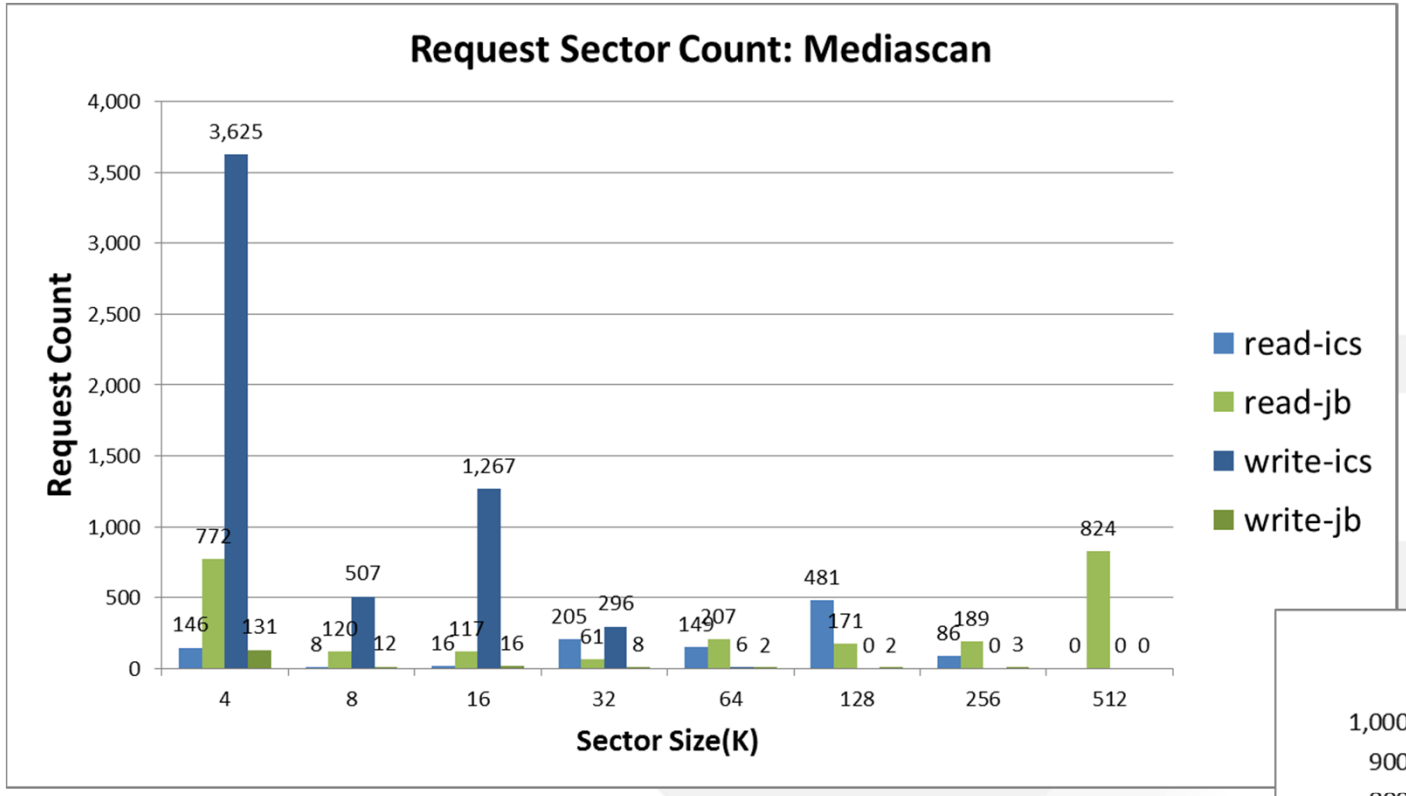


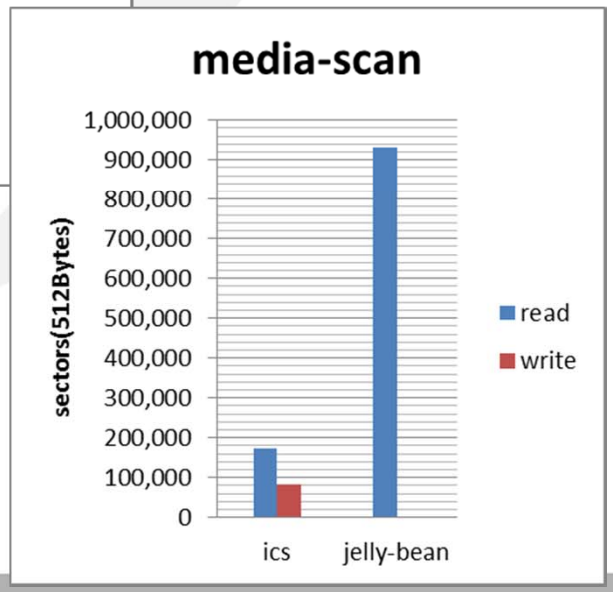
Pattern comparison: Contact





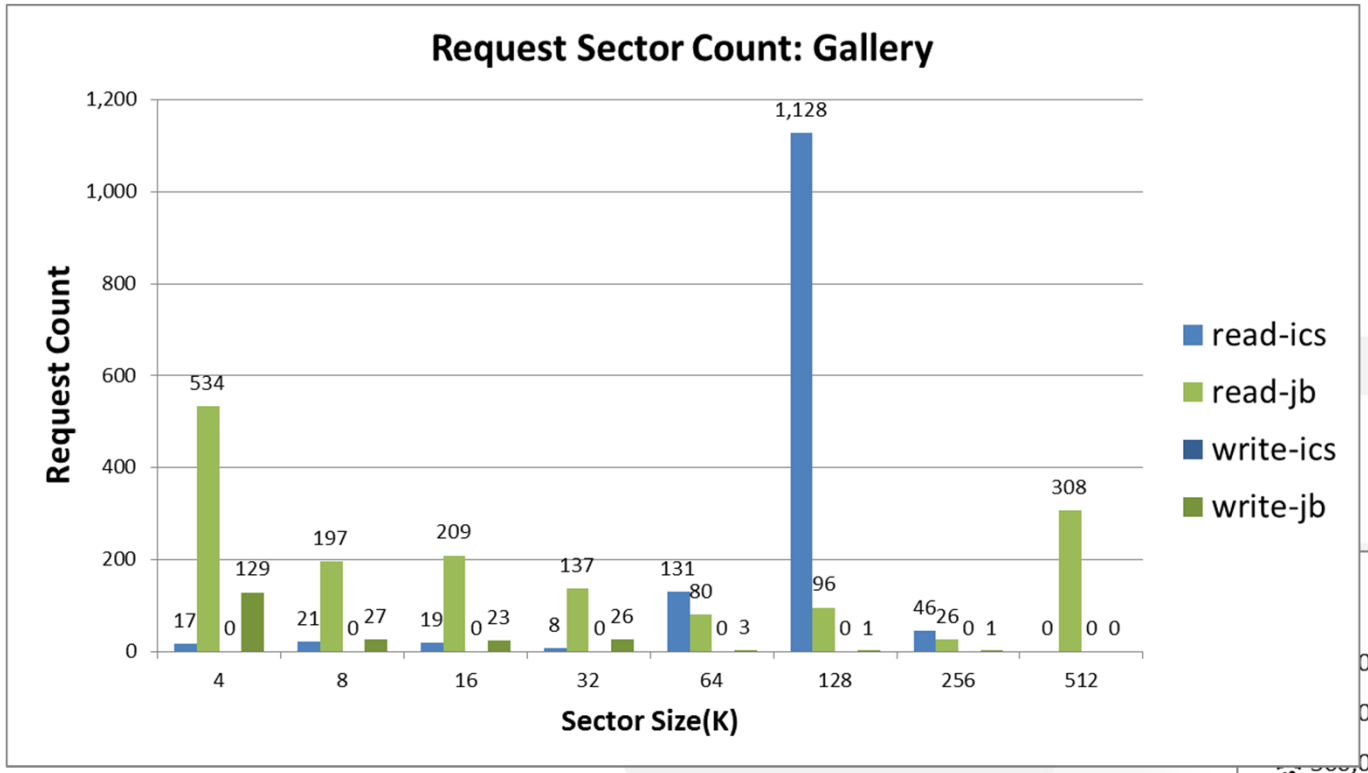
Pattern comparison: Media-scan

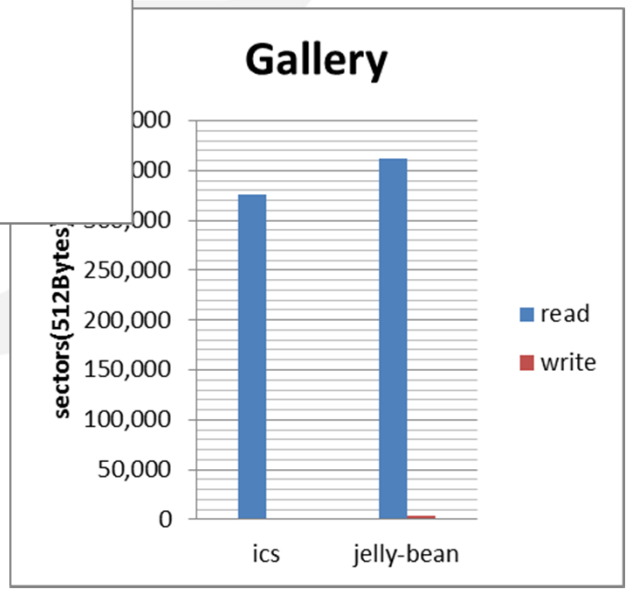




- Read 양 급격증가

Pattern comparison: Gallery

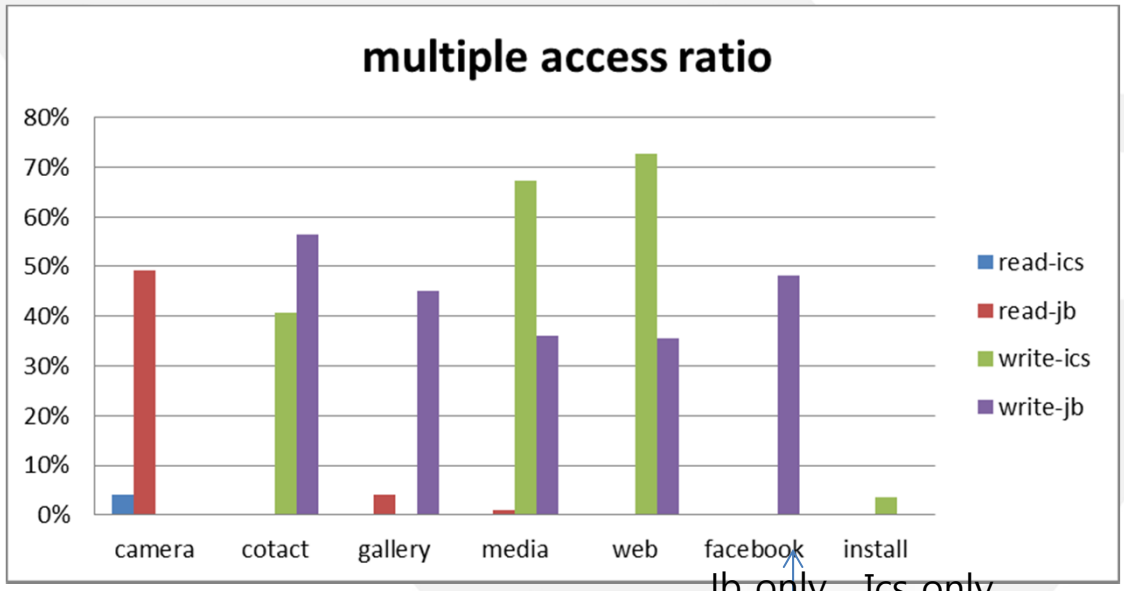




- Sequential read pattern변화

Multiple access ratio

- Multiple access ratio
  - (requested sectors - access area)/requested sectors



Jb only Ics only

ICS	read-ics	write-ics	read-area	write-area
total	506,200	2,219,448	503,568	1,956,856
camera	576	1,836,976	552	1,836,792
cotact	2,480	42,536	2,480	25,256
gallery	326,624	0	326,624	0
media	175,064	83,112	175,064	27,185
web	1,456	256,824	1,456	69,912
install	615,696	822,880	615,696	794,504

Jelly bean	read-jb	write-jb	read-area	write-area
total	1,422,984	1,682,160	1308345	1,624,904
camera	77,272	1,558,312	39,320	1,557,688
cotact	28,792	53,416	28,792	23,248
gallery	362,448	4,072	347,384	2,240
media	930,928	3,792	922,737	2,424
web	23,544	62,568	23,544	40,320
facebook	26,720	60,984	26,720	31,520

Summary

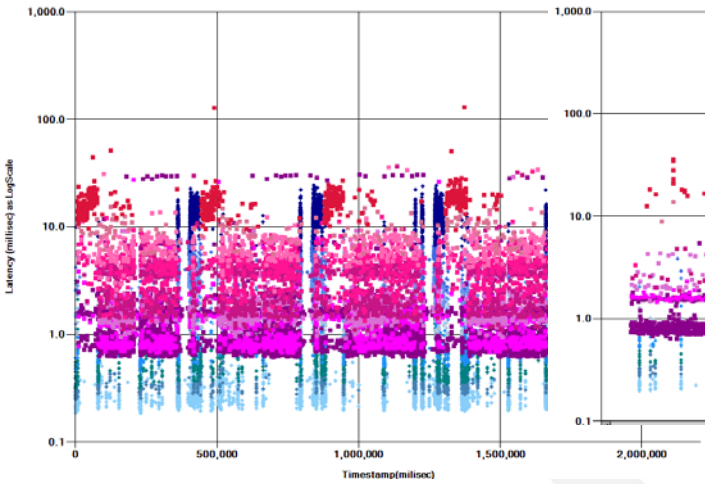
- Jelly-bean
  - More reads, less writes, Bigger chunk
  - DB의 write 양이 줄어든 듯
  - Media-scan의 읽기 양의 급격한 증가
  - Booting time의 증가 by media-scan 증가

- web, facebook, contact 등은 DB로 인한 write-overwrite 비율이 높다.

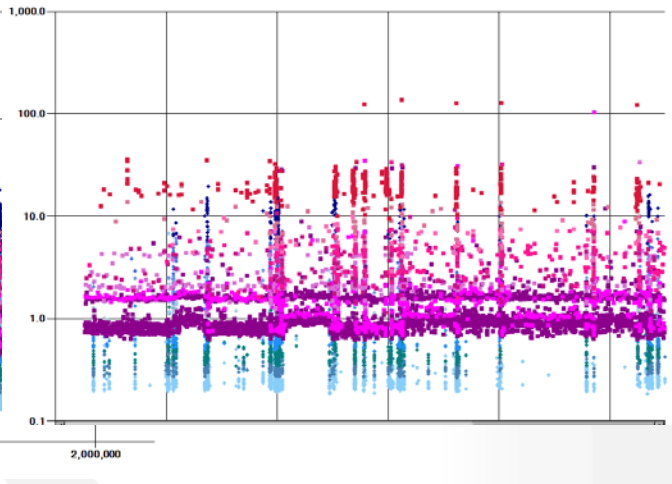
Emmc Performance

Performance degradation

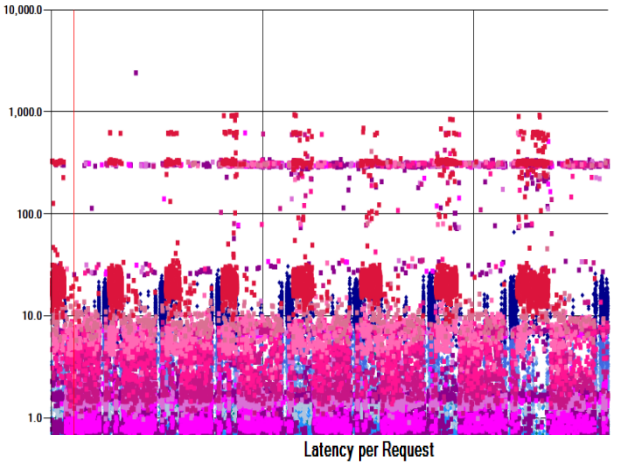
FS status : Under 40%

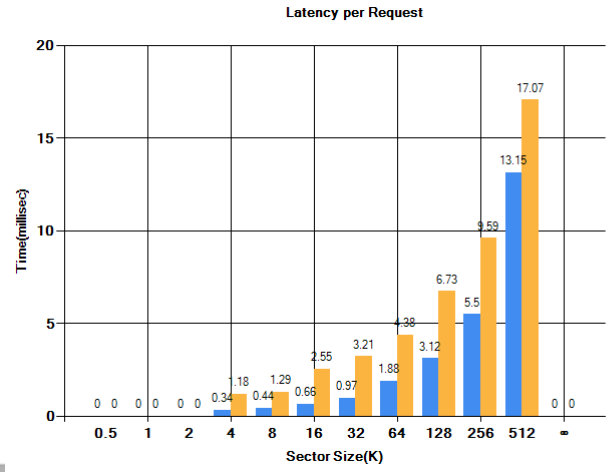


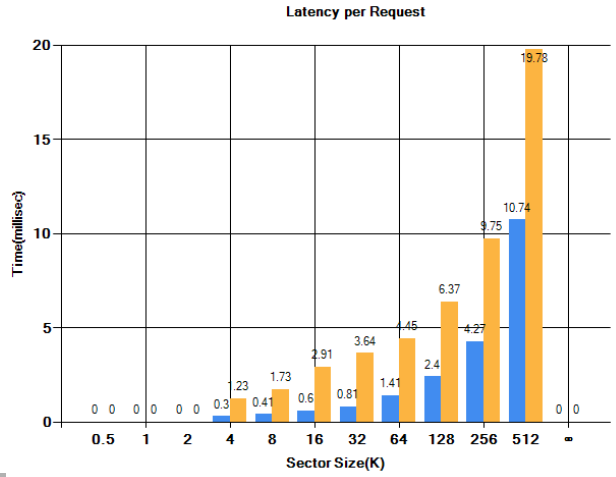
60%

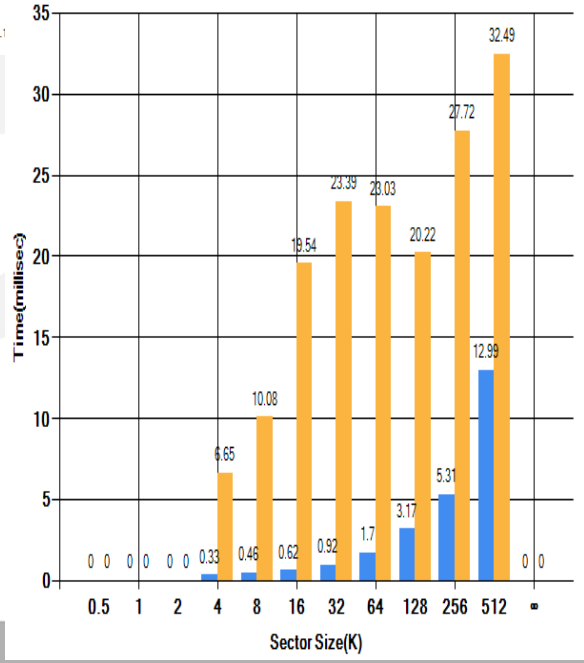


90%



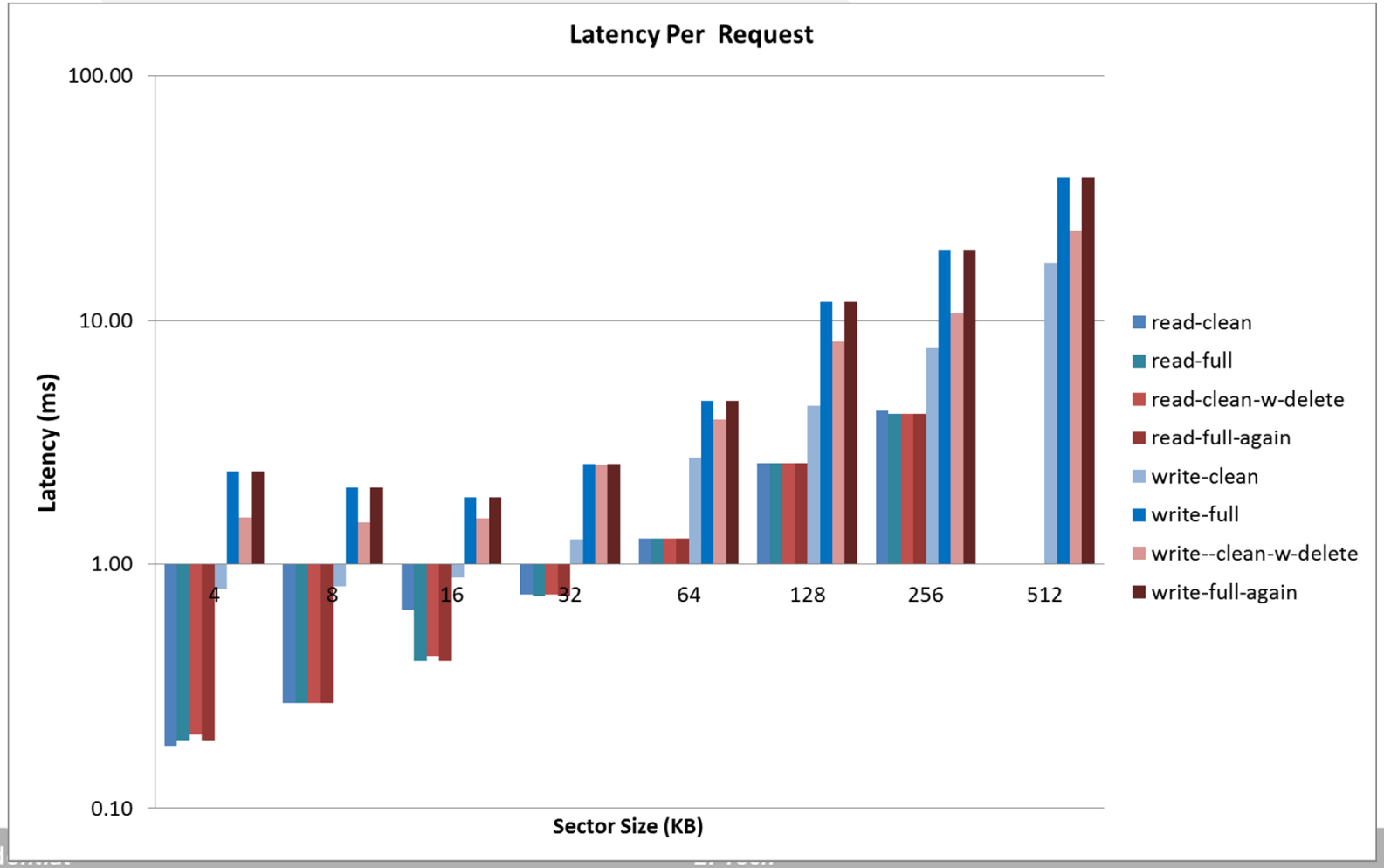




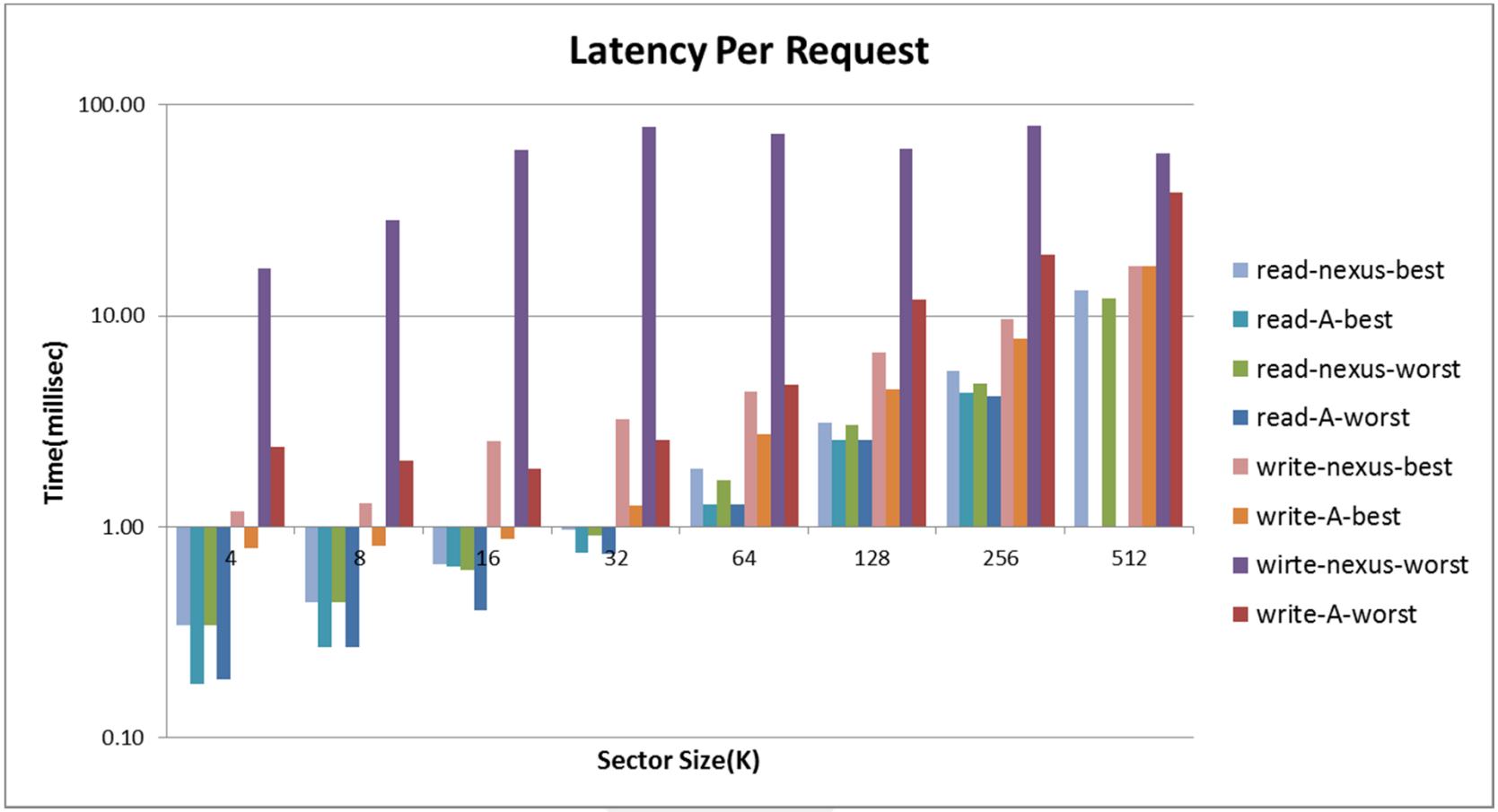


Sample A-device performance

- Ebench full test
  - Run to full, delete files, Run to full



Comparison : Nexus v.s. A-device



Read Latency variation (1)

- Read Latency variation
  - Nexus-device는 dirty상황에서 주요분포의 latency 떨어짐

	nexus-clean	nexus-dirty
평균	0.33	0.34
표준편차	0.26	0.28

millisec	nexus-clean	nexus-dirty
8이상	0.00%	0.03%
4	0.07%	0.71%
2	0.37%	2.31%
1	1.55%	3.94%
0.5	4.65%	29.76%
0.3	27.65%	22.77%
0.25	28.41%	38.48%
0.2	36.06%	2.00%
0.15	1.24%	0.00%
0.1	0.00%	0.00%
0	0.00%	0.00%

Read Latency variation (2)

- Comparison
  - A-device는 거의 변함없음

	nexus-clean	nexus-dirty	A-clean	A-dirty
평균	0.33	0.34	0.19	0.20
표준편차	0.26	0.28	0.43	0.47

millisec	nexus-clean	nexus-dirty	A-clean	A-dirty
8이상	0.00%	0.03%	0.03%	0.04%
4	0.07%	0.71%	0.00%	0.03%
2	0.37%	2.31%	0.03%	0.01%
1	1.55%	3.94%	0.03%	0.02%
0.5	4.65%	29.76%	0.01%	0.03%
0.3	27.65%	22.77%	0.09%	0.17%
0.25	28.41%	38.48%	2.51%	4.00%
0.2	36.06%	2.00%	18.62%	17.66%
0.15	1.24%	0.00%	78.05%	77.70%
0.1	0.00%	0.00%	0.64%	0.33%
0	0.00%	0.00%	0.00%	0.00%

Read Latency variation (3)

- Read latency variation seems to be caused by map-cache managements
  - Internal SRAM size and management algorithm

eMMC 4.5 Features

eMMC 4.5

- Packed commands
  - Reducing transfer time
  - Making effective map-info by packed random
  - Decide targets by real workload analysis
- Cache commands
  - Effective map-info
  - Flush-ratio of real workload
- Context id
  - Specifying sequential-data
- Data tag
  - Specifying hot-data
- Power-off notification
  - Do something before power-off

- 4.41 features, but not applied yet
  - Discard(trim)
  - Background op

- What of 4.5 features will be applied?


EFTech