

Android (Marshmallow or Nougat) I/O Management and Disk Scheduling

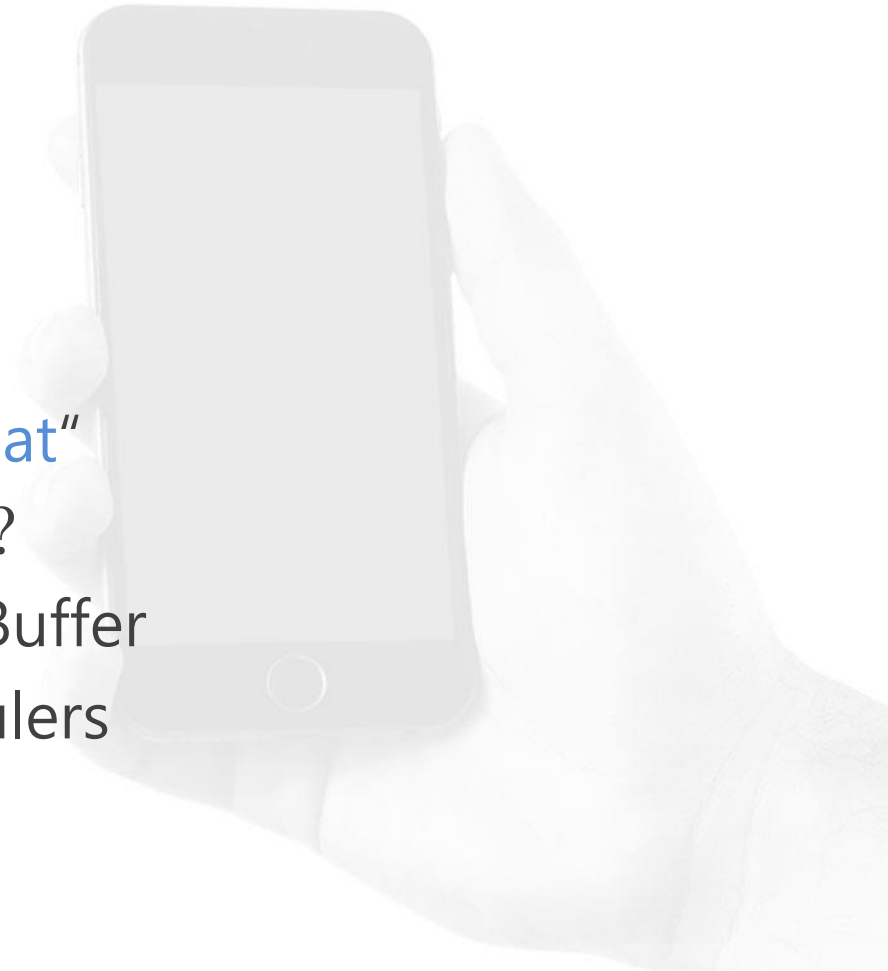
Prepared by
Murtadha Mohammed ali



Name: Murtadha Mohammed Ali
Student NO: 163104062
Email: mmrtdha234@gmail.com

❖ Outline:

- ❖ Introduction
- ❖ What is **Android**?
- ❖ "**Marshmallow**" and "**Nougat**"
- ❖ What is an I/O Scheduler ?
- ❖ I/O Manage Read-Ahead Buffer
- ❖ Recommended I/O Schedulers
- ❖ Conclusion



❖ INTRODUCTION

In this project , we present android (Marshmallow and Nougat) I/O performance evaluation on Android devices and demonstrate that I/O scheduling has effect on I/O performance in these devices .The rest of the project is organized as follows explains Android operating system and its I/O schedulers. presents I/O performance evaluation in Android devices and discussion on I/O schedulers' effect on performance .



What is Android ?

- **Android** is the name of the mobile operating system owned by American company, Google and based on the Linux kernel



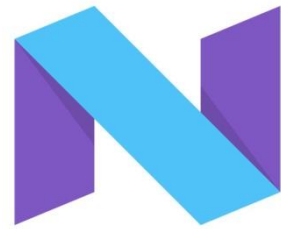
❖ Android " Marshmallow "



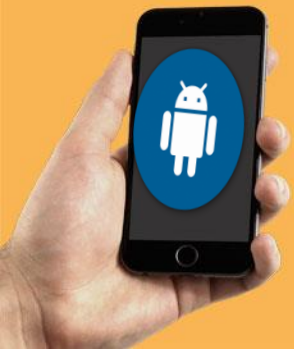
Android "marshmallow" (the so-called Android M during development) is the sixth major version of the Android operating system. First released as a beta building on May 28, 2015, it was officially released on October 5, 2015



❖ Android "Nougat"



Android "Nougat" (codenamed Android N during development) is the seventh major version of the Android operating system. First released as a (beta) build on March 9, 2016, it was officially released on August 22, 2016



❖ What is an I/O Scheduler :

Input/output (I/O) scheduling is a term used to describe the method computer operating systems decide the order that block I/O operations will be submitted to storage volumes. I/O Scheduling is sometimes called 'disk scheduling'.

I/O schedulers can have many purposes depending on the goal of the I/O scheduler, some common goals are:

- To minimize time wasted by hard disk seeks.
- To prioritize a certain processes' I/O requests.
- To give a share of the disk bandwidth to each running process.
- To guarantee that certain requests will be issued before a particular deadline.



❖ Which schedulers are available ?

- CFQ
- Deadline
- VR
- Noop
- BFQ
- FIOPS
- SIO (Simple)
- ROW
- ZEN
- SIOplus
- FIFO
- Tripndroid
- Test
- Maple

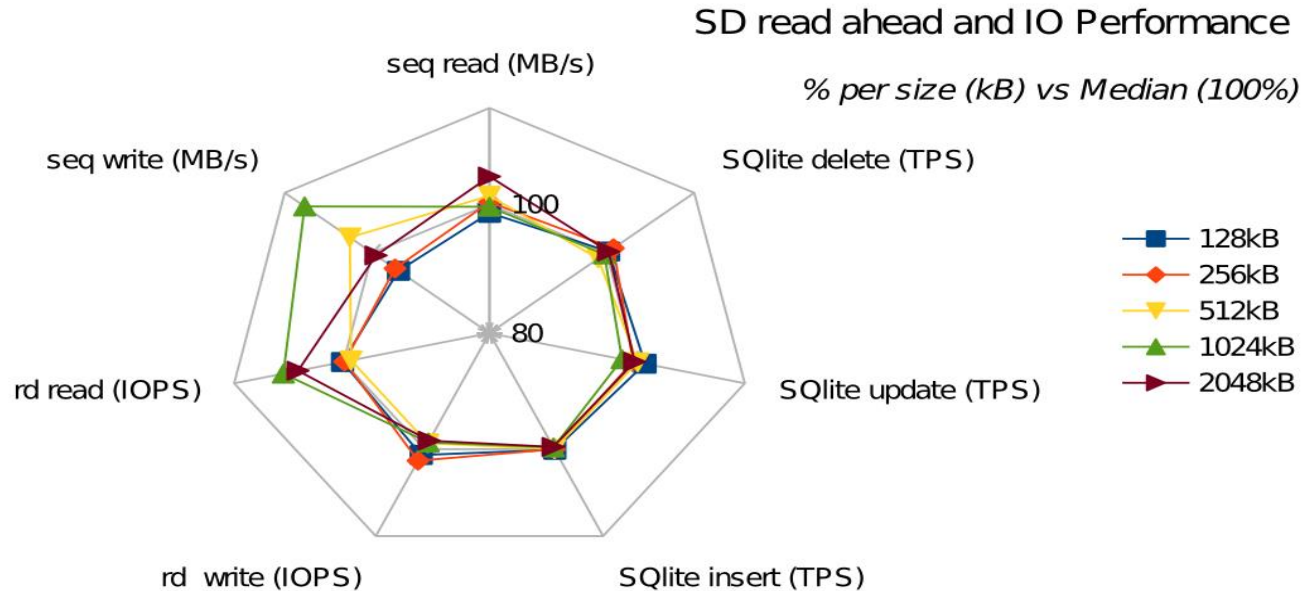


❖ I/O Manage Read-Ahead Buffer :

If you've used a custom kernel, you probably have heard of a term called Read Ahead Buffer or Cache. It's basically a cache for files that have been opened recently on your mobile device, so that they can be quickly accessed again if needed. By android default, this value has been set to 128kB. Usually having more buffer means that more files can be cached, this can mean higher read and write speeds, but also this can result in more I/O latency. There is a point where increasing the I/O read ahead will have no benefit to read/write speeds. Have a look at the graph below:



❖ I/O Manage Read-Ahead Buffer :



I/O Read Ahead Buffer is dependent on the size of your flash storage (internal/external) unlike I/O schedulers. Below is the recommended settings for the given size that will yield the best performance (differs between setups).

Less than 8GB - 128KB

8GB - 512KB

16GB - 1024KB

32GB or above - 2048KB

Any setting above what I have recommended may yield no extra performance!



❖ Recommended I/O schedulers:

For everyday usage:

- ZEN (First choice)
- ROW (Second choice)
- SIO (Third choice)
- Noop
- CFQ
- Deadline

For battery life:

- Noop (First choice)
- FIOPS (Second choice)
- SIO (Third choice)
- ROW (Forth choice)

For gaming:

- Deadline (First choice)
- ZEN (Second choice)
- ROW (Third choice)
- CFQ



❖ Recommended I/O schedulers:

For performance(Benchmarking):

- FIOPS (First choice)
- Deadline (Second choice)
- Noop
- ROW

For heavy multitasking:

- BFQ (First choice)
- CFQ (Second choice)
- Deadline (Third choice)



❖ Conclusion

we have focused on I/O scheduling in Android devices with flash memories. We have evaluated I/O performance and effect of I/O scheduling, and have demonstrated that scheduling has had effect on I/O performance. In addition, we have compared performance of the Android I/O schedulers and have shown that choosing Deadline has been able to improve I/O performance. After the evaluation, we have revealed the reason why Deadline scheduler has provided the better performance.



THANK YOU

