FastTrack: Foreground App-Aware I/O Management for Improving User Experience of Android Smartphones

Jihong Kim

Seoul National University

NVRAMOS 2018

October 25, 2018

USENIX ATC '18

2018 USENIX
Annual Technical Conference

JULY 11–13, 2018
BOSTON, MA

www.usenix.org/atc18

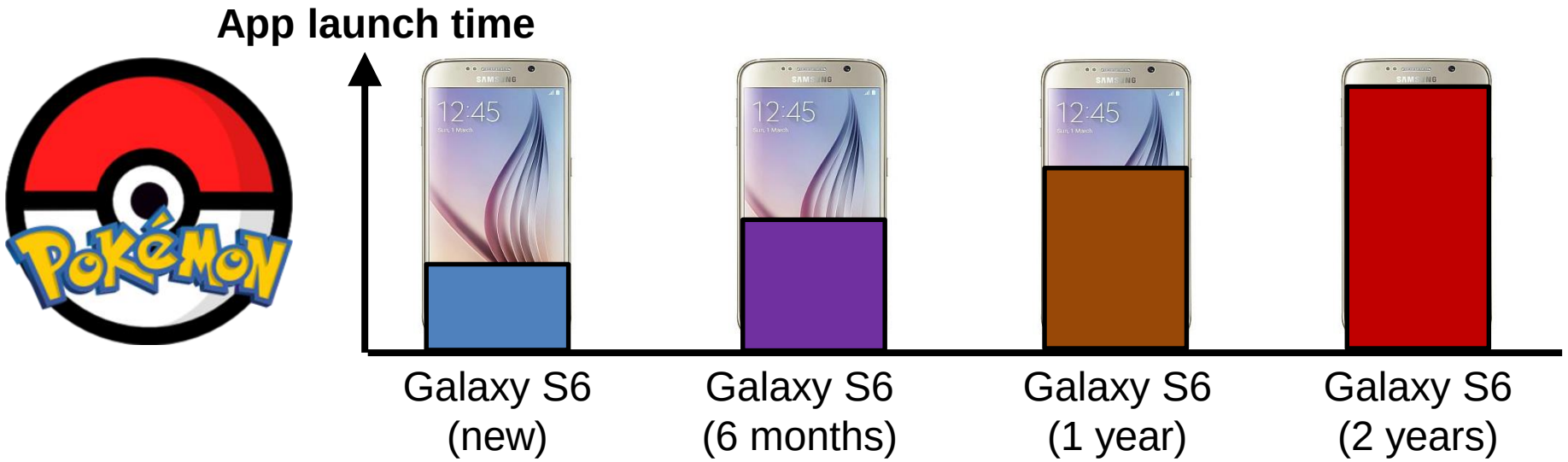
FastTrack: Foreground App-Aware I/O Management for Improving User Experience of Android Smartphones

Sangwook Shane Hahn, *Seoul National University;* Sungjin Lee, *DGIST;*
Inhyuk Yee, *AIBrain Asia;* Donguk Ryu, *Samsung Electronics;*
Jihong Kim, *Seoul National University*

Mobile Storage Optimization Research

Why does my aging smartphone get slower?

App Launch Time over Smartphone Age



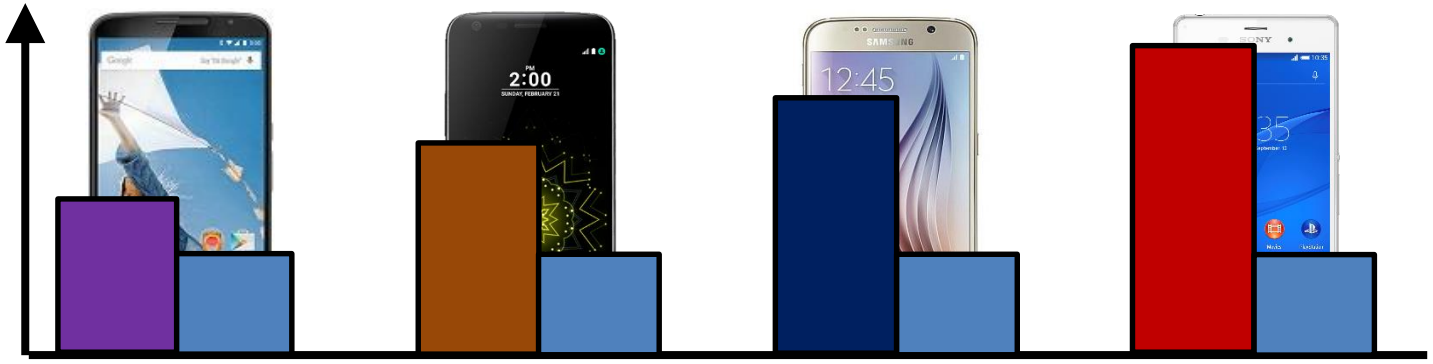
Application launch times increase up to <u>3 times</u> on 2-year old smartphones

Main Cause: File Fragmentation

Q: Why does performance degrade ?

A: File fragmentation

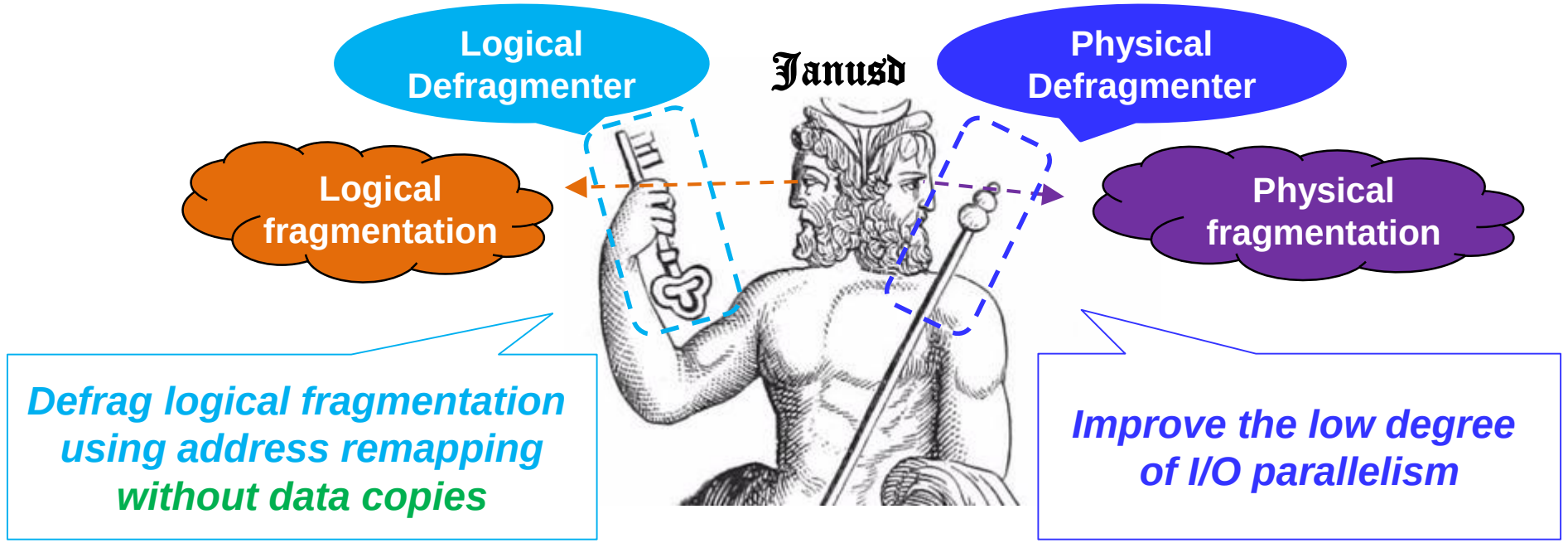
Application launching time



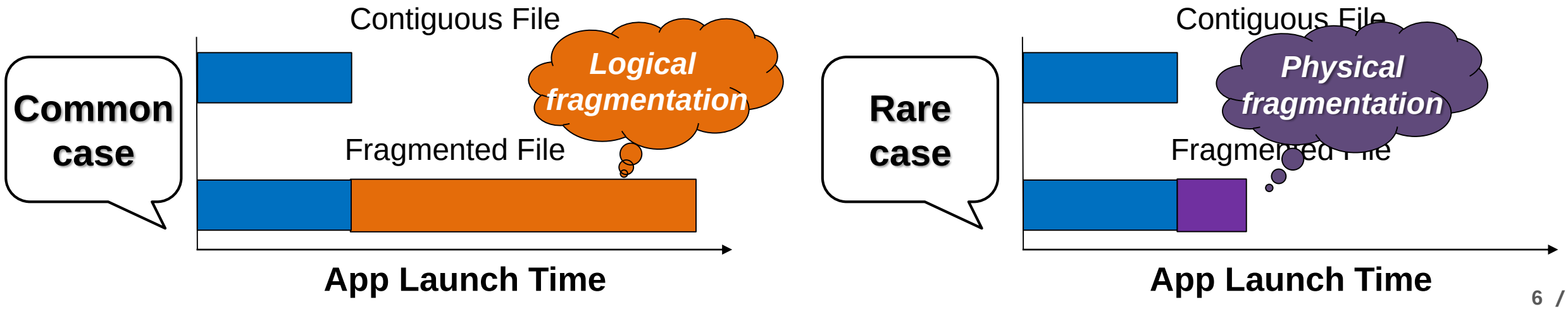
Nexus 6	G5	Galaxy S6	Xperia Z3
0% files fragmented	0% files fragmented	0% files fragmented	0% files fragmented

***Defragmentation* can *improve* the *degraded performance* by fragmentation**

Solution for Decoupled Fragmentation [ATC 2017]



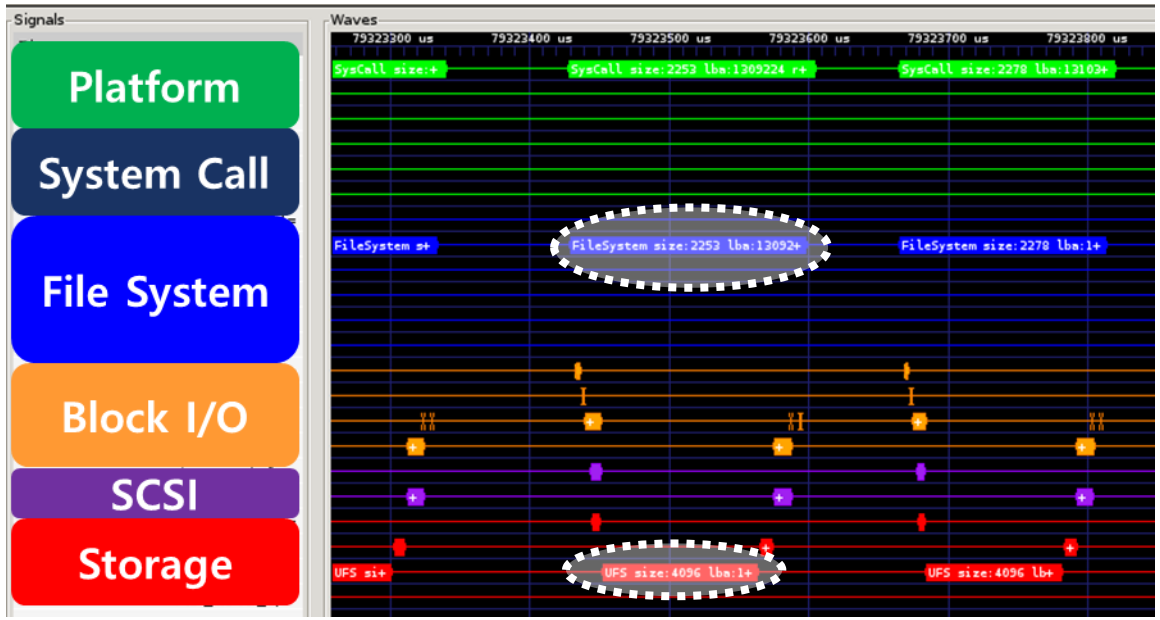
Janus Defragmenter



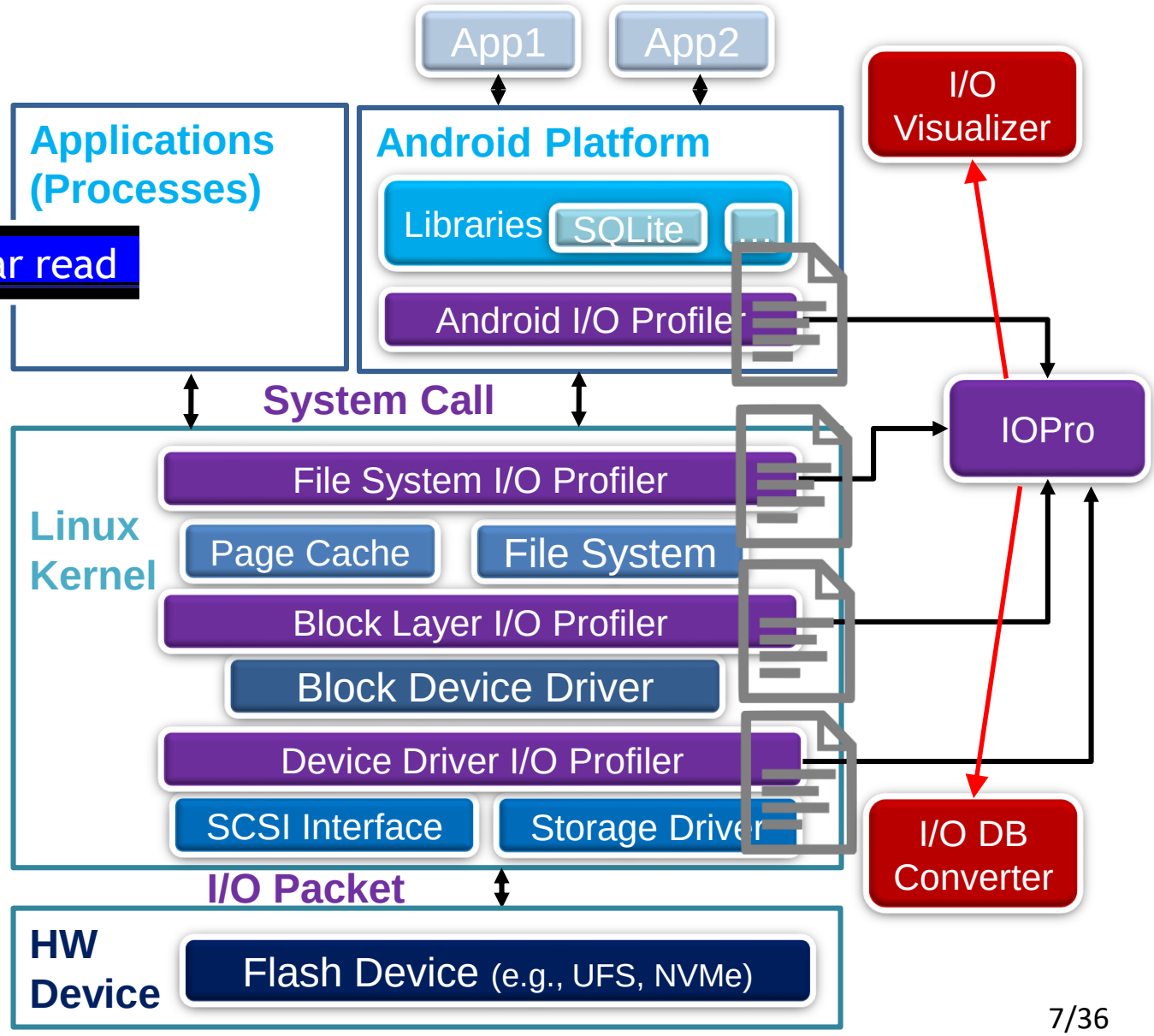
Storage I/O Stack Analysis for Performance Bottlenecks

- Storage I/O Profiler (IOPro)
  - Monitor Linux kernel I/O stack

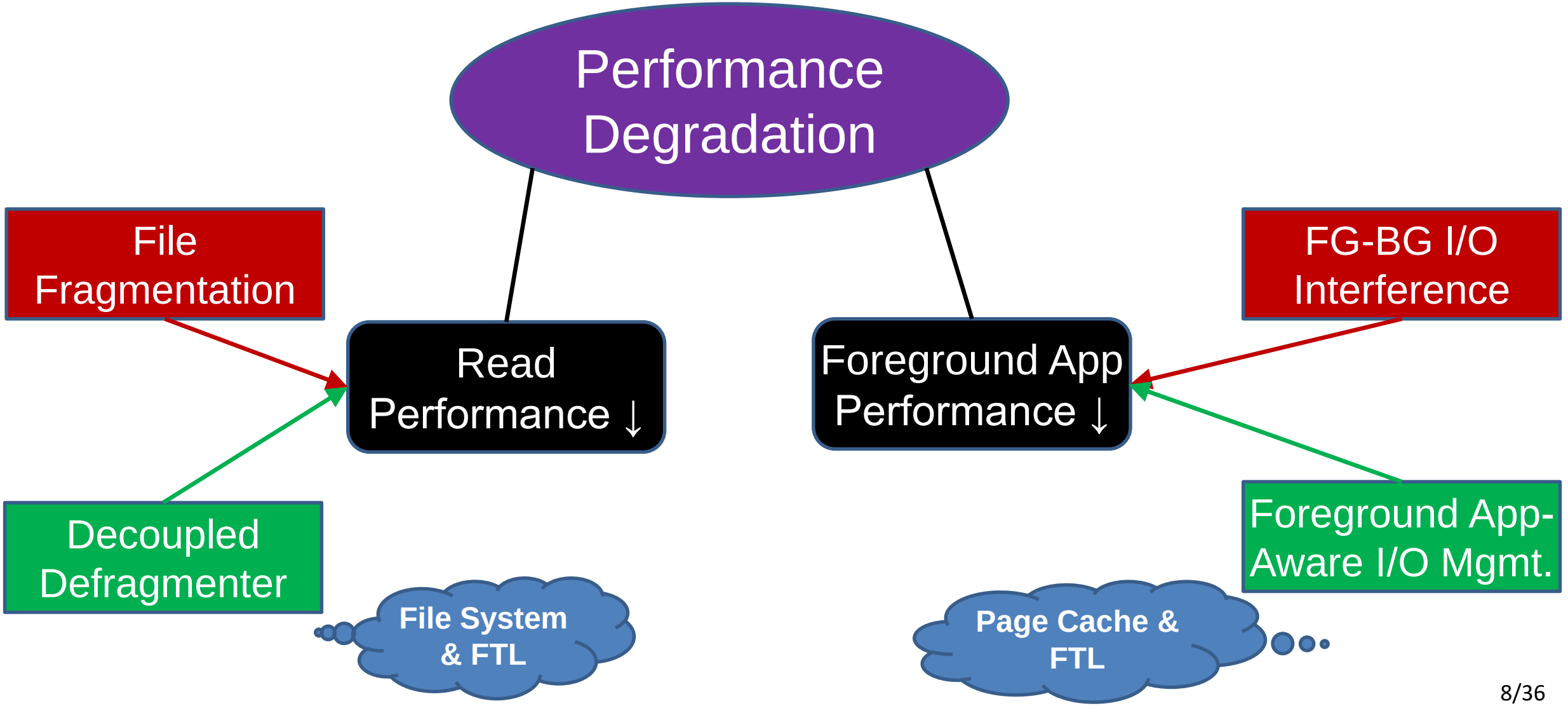
FileSystem size:2253 lba:1675816 file:scrollpause.jar read



UFS size:4096 lba:1675816 file:scrollpause.jar read



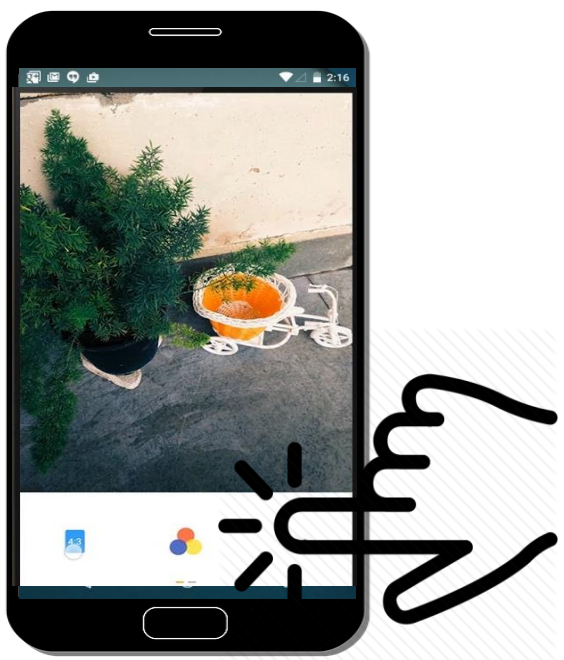
Focus on Foreground App Performance

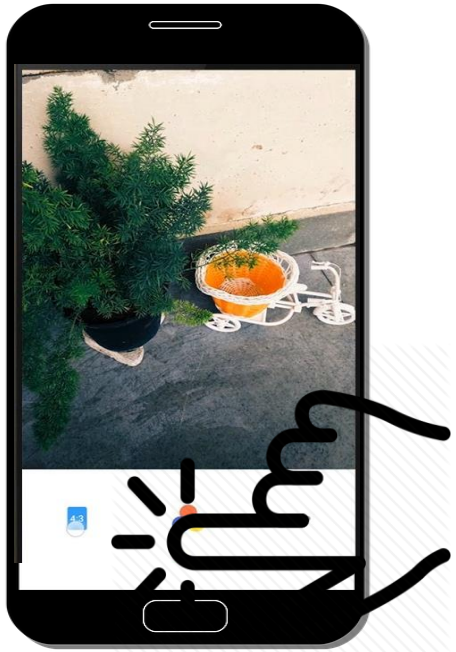


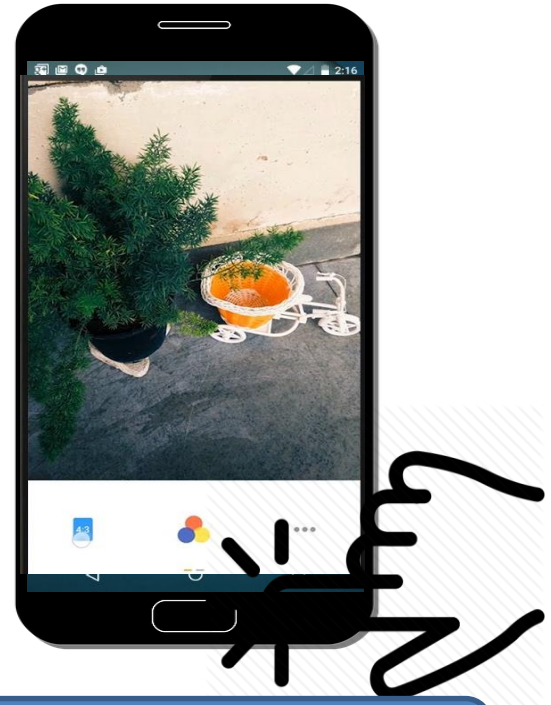
Outline

- **Impact of Background I/O on User Experience**
- **Foreground-Background Interference Analysis**
- **FastTrack: Foreground App-Aware I/O Management**
- **Experimental Results**
- **Conclusions**

User Experience in Android Smartphones







App Launch → App Runtime → App Switch

App response time
is crucial to the user experience

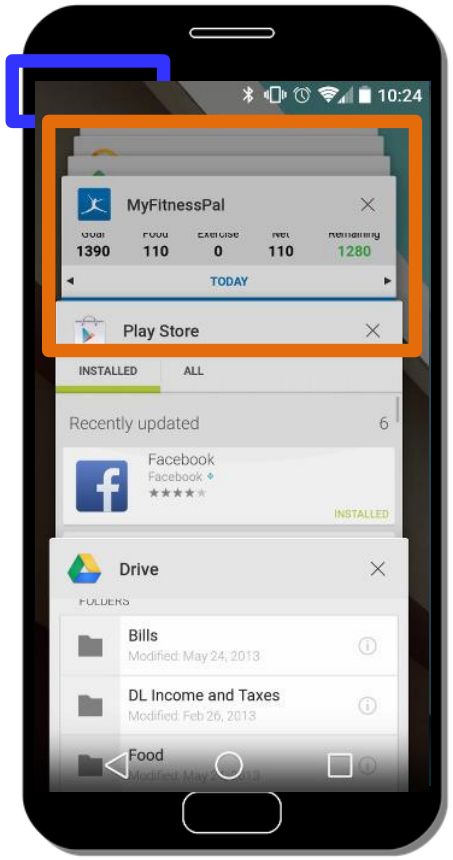
Number of Background Apps Increasing

Apps running in background

Updating Apps

Sync files

Sync photos



User can allow apps to run background

8 cores 8 GB RAM

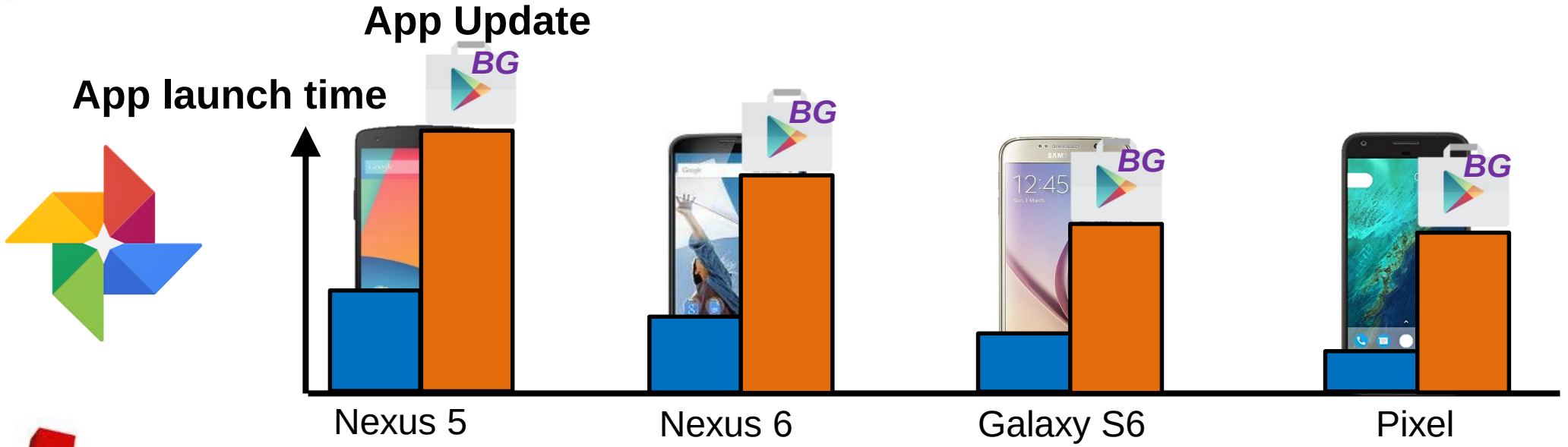
8 cores 6 GB RAM

8 cores 4 GB RAM

Number of background processes has been increased with larger memory and more CPU cores

Background Apps Degrade the Quality of User Experience

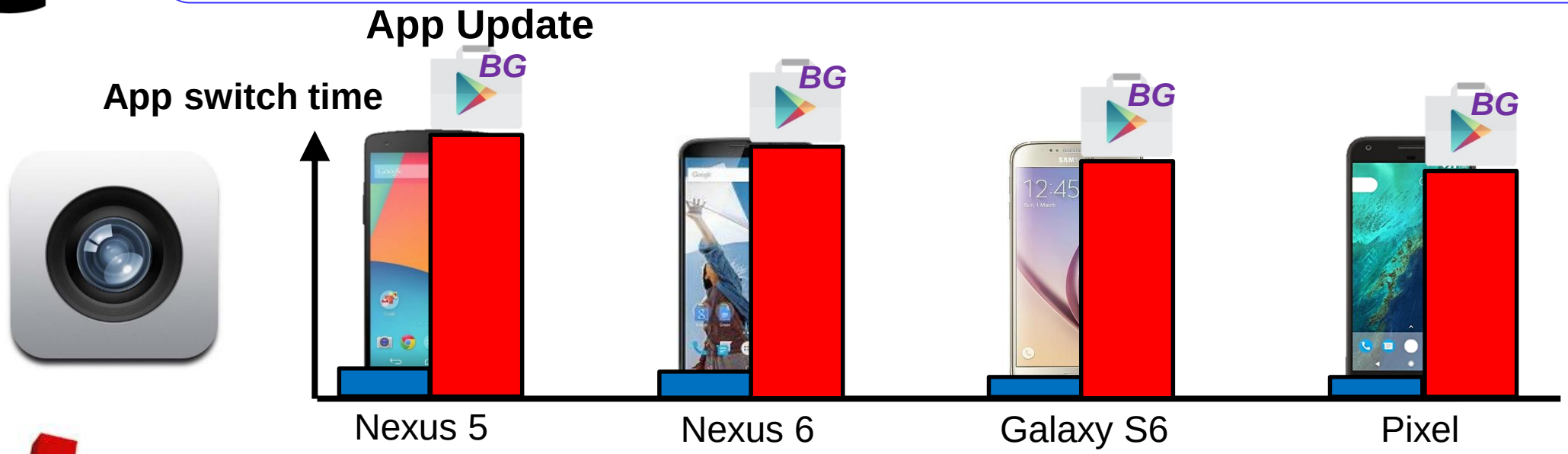
App launch time increases due to background apps



Background apps increase foreground app launch time by up to 2.6 times

Background Apps Degrade the Quality of User Experience

App switch time increases due to background apps



Background apps increase foreground app switch time by up to 19.5 times

Main Cause of Performance Degradation

How background apps degrade the quality of user experience?

Background (storage) I/Os

No rendering for BG apps

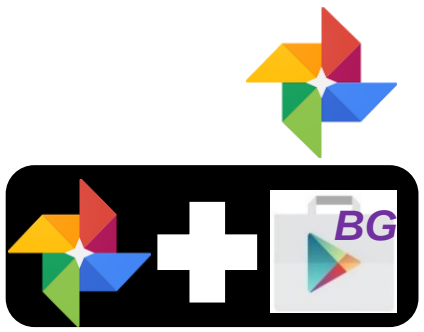
GPU?

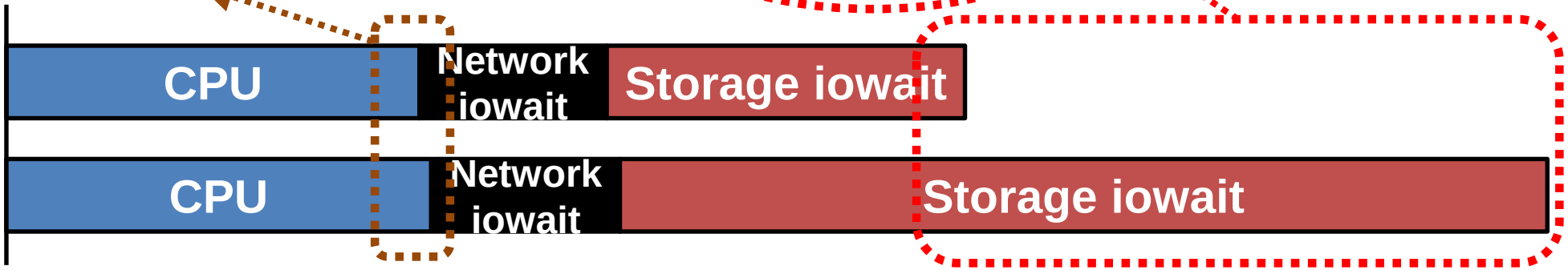
CPU?

Network?

Performance ↓ even in offline

Storage?

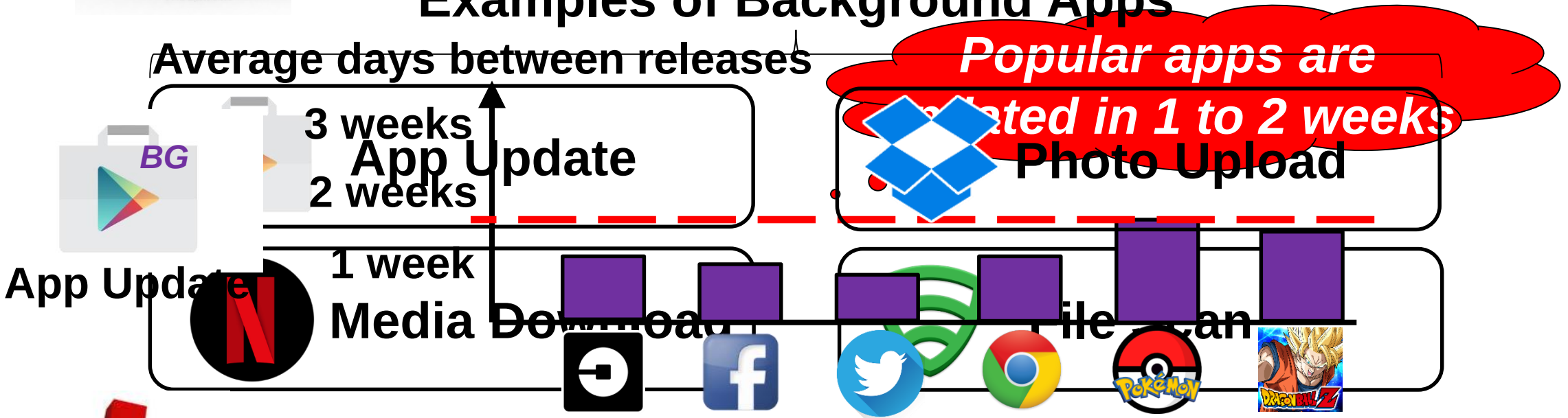




Background I/O Occurrence Frequency

Q: Do background I/Os (BG I/Os) occur frequently ?

Examples of Background Apps

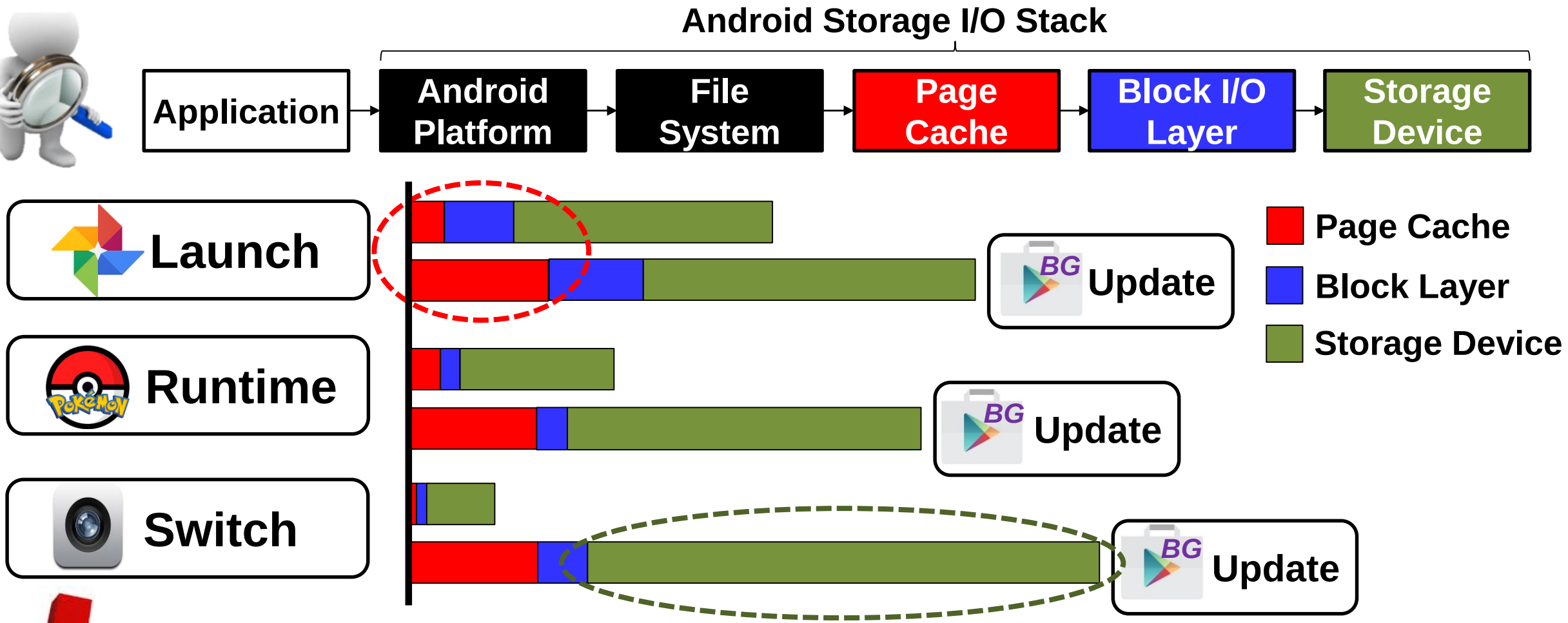


Frequent BG I/O occurrences have a critical impact on the user experience

Outline

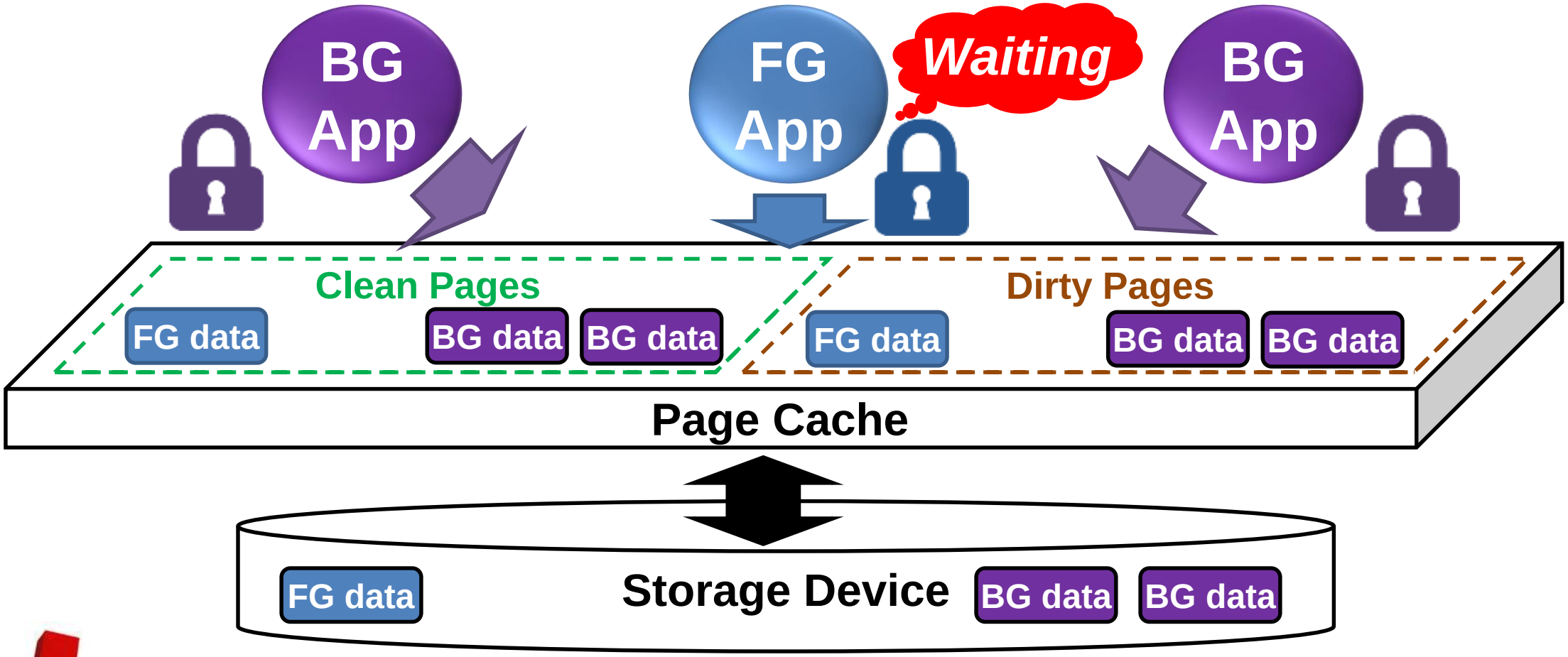
- Impact of Background I/O on User Experience
- **Foreground-Background Interference Analysis**
- FastTrack: Foreground App-Aware I/O Management
- Experimental Results
- Conclusions

Foreground-Background Interference Analysis



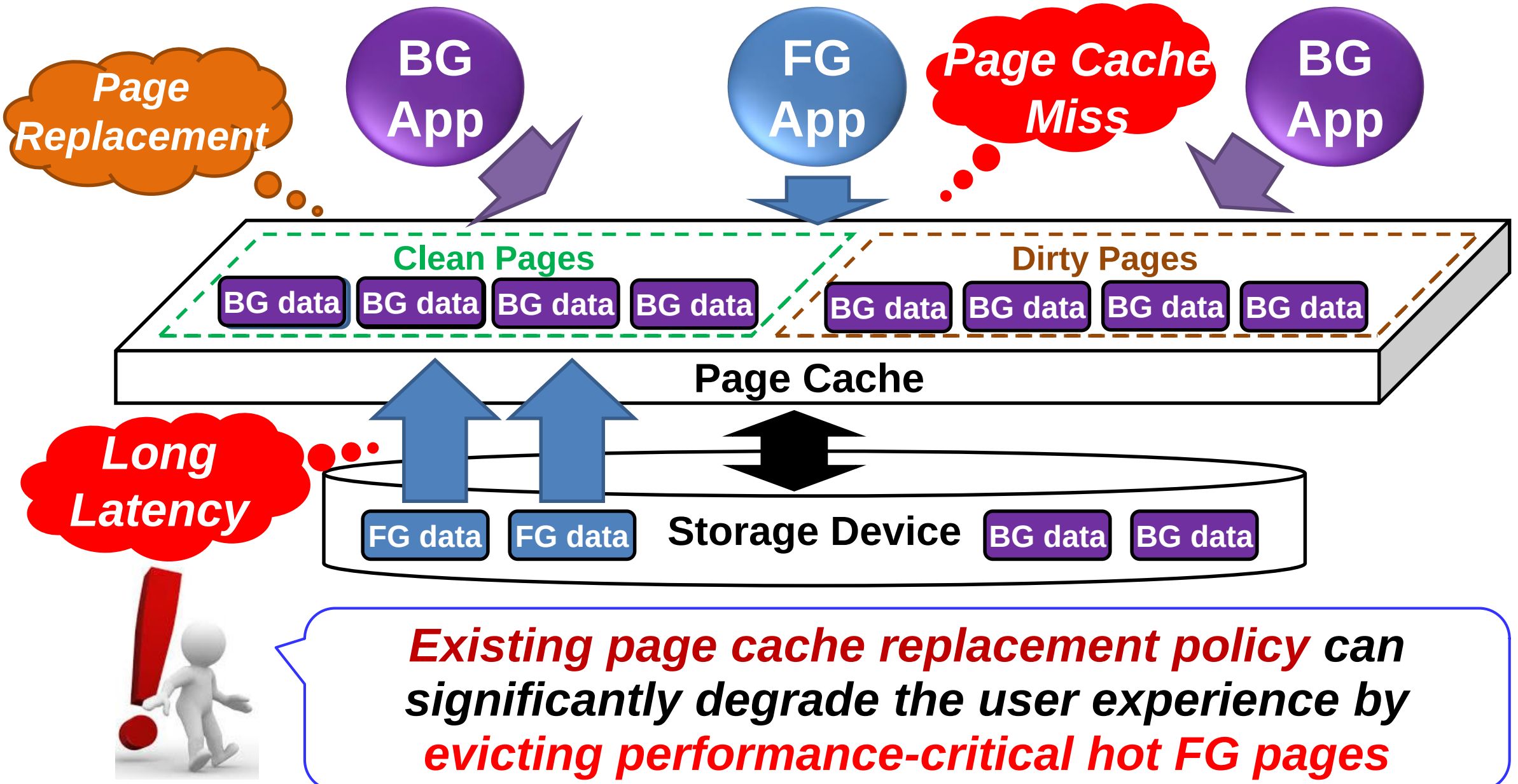
***Page cache* and *storage device* are affected most by FG-BG interference**

Impact on Page Cache: Lock Contention



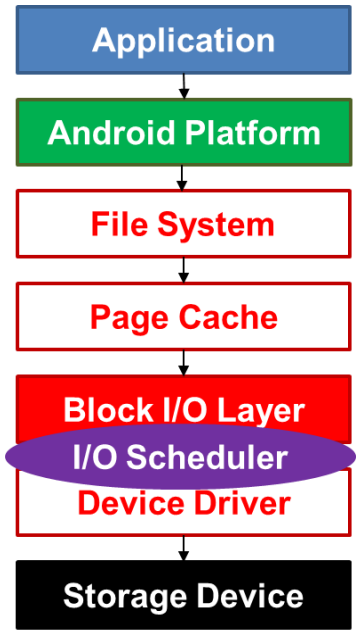
Lock contentions* in the page cache can significantly degrade the user experience by *waiting the completion of BG I/Os frequently

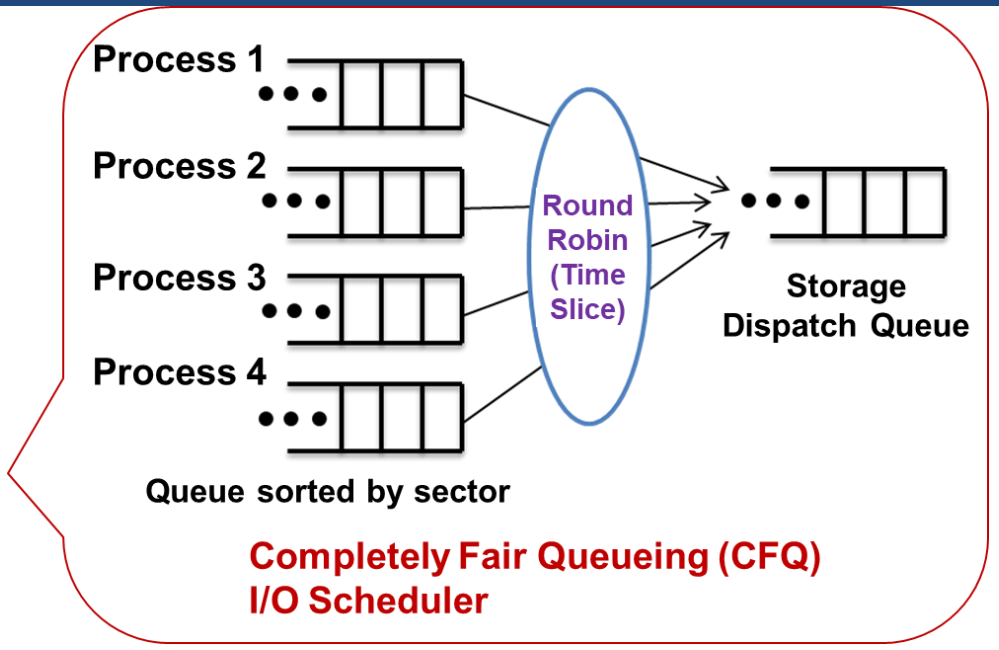
Impact on Page Cache: High Miss Rate



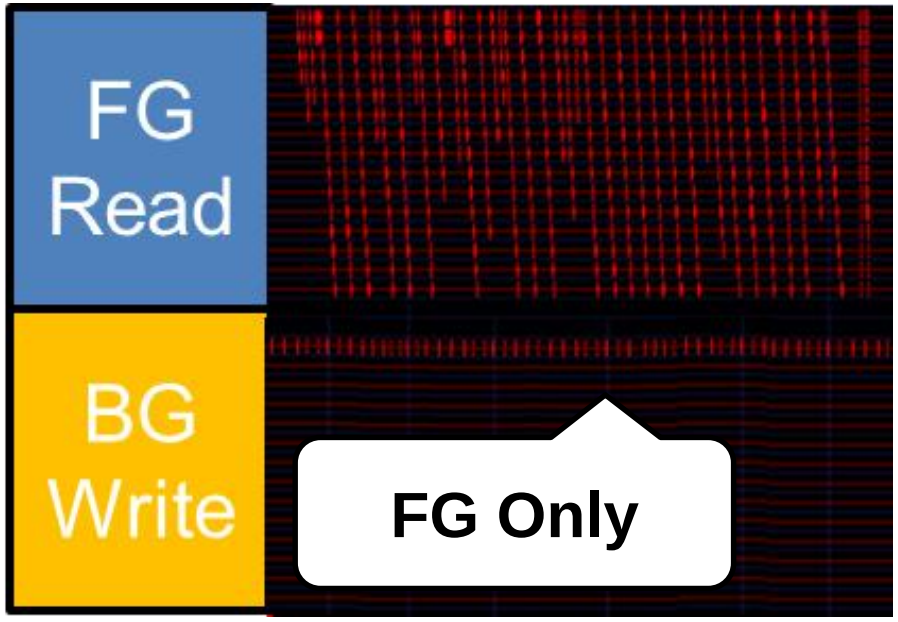
Existing page cache replacement policy* can significantly degrade the user experience by *evicting performance-critical hot FG pages

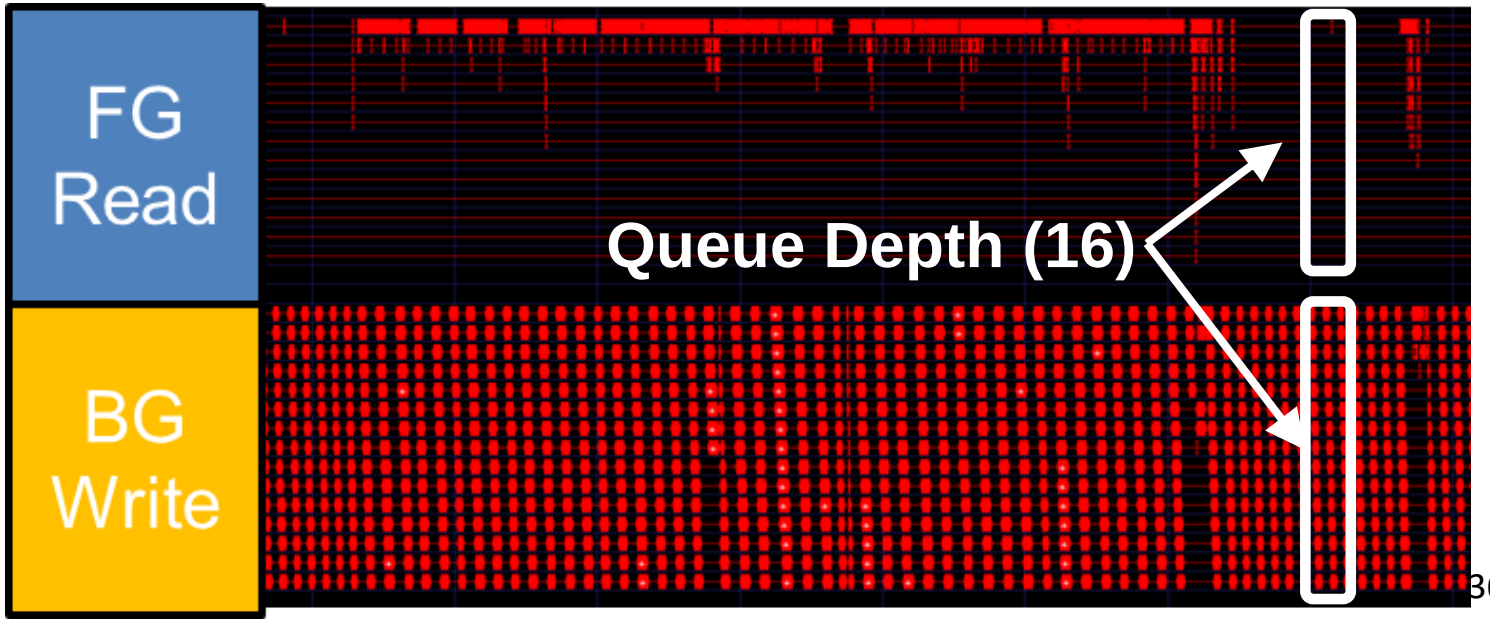
Impact on I/O Scheduler: Limited Bandwidth by CFQ



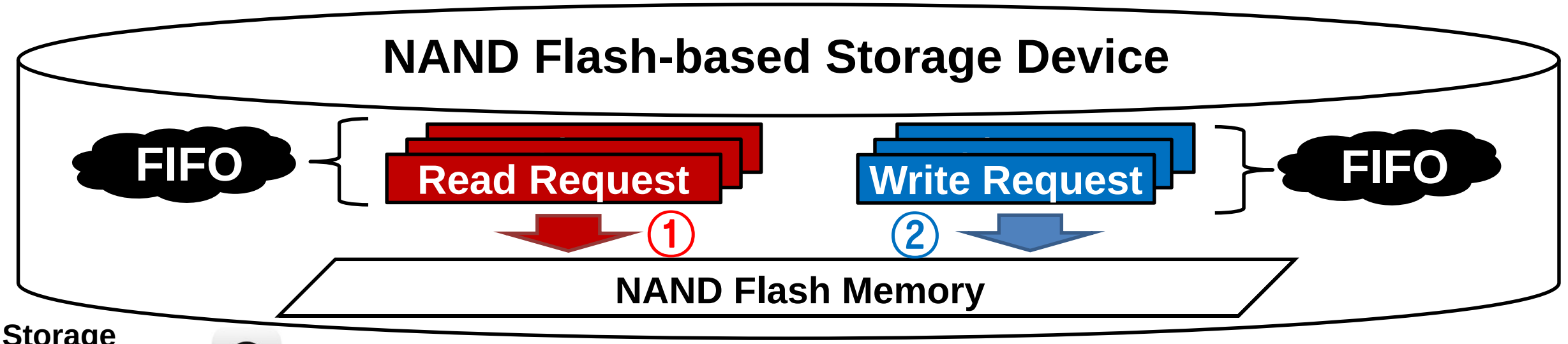


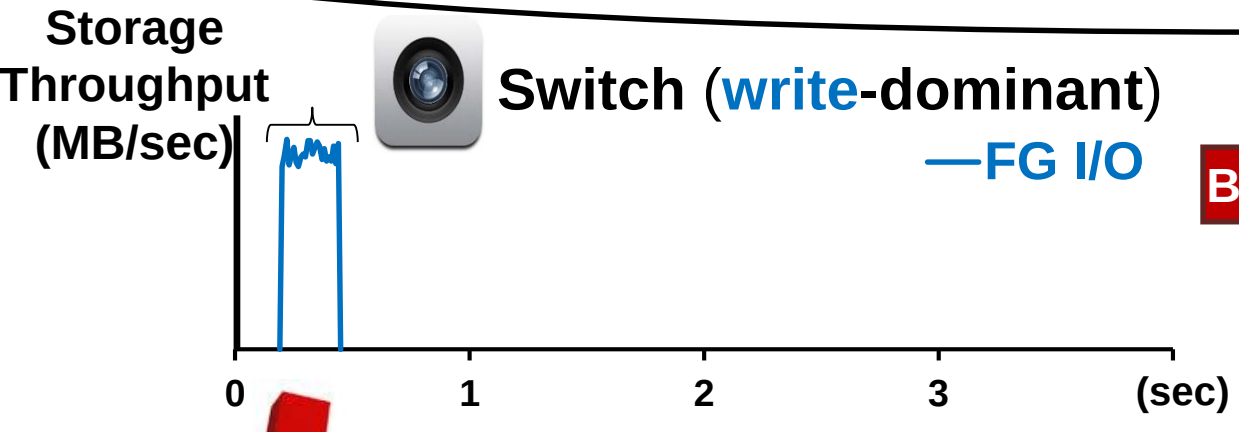
FG + BG

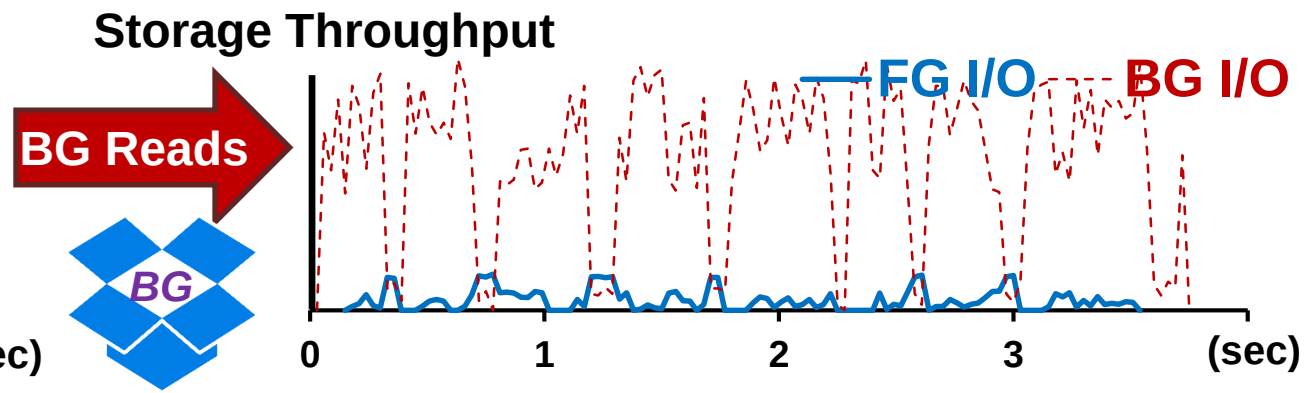




Impact on Storage Device: Internal Priority Inversion

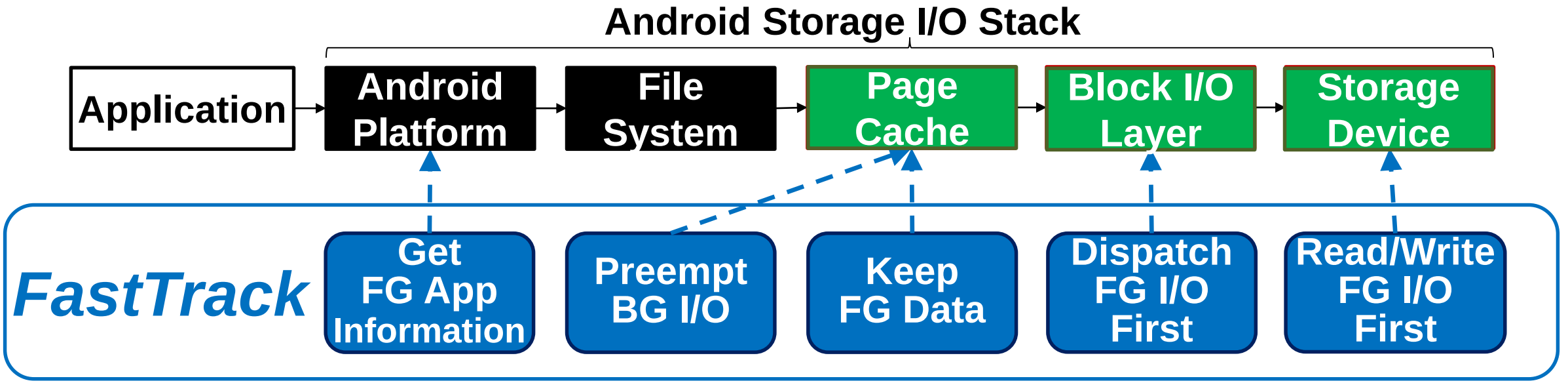


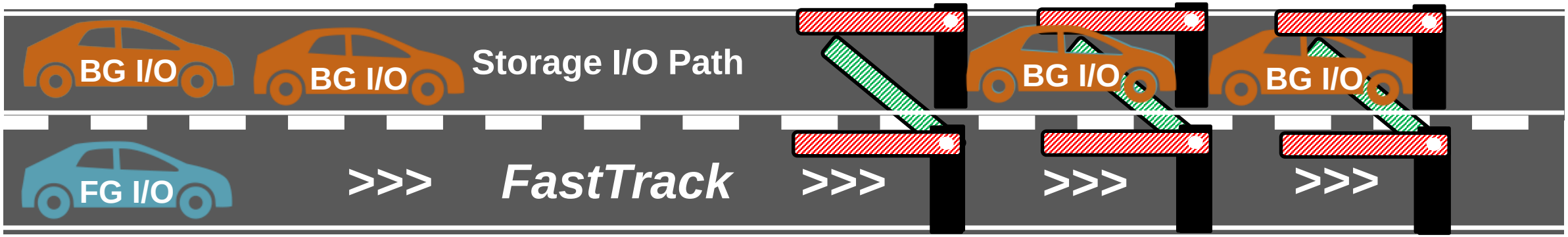




Existing device I/O scheduler's priority policy* can *significantly degrade the user experience by limiting the throughput of foreground I/Os

Solution for Foreground-Background Interference





FastTrack is effective in **improving** **Android smartphone user experience**

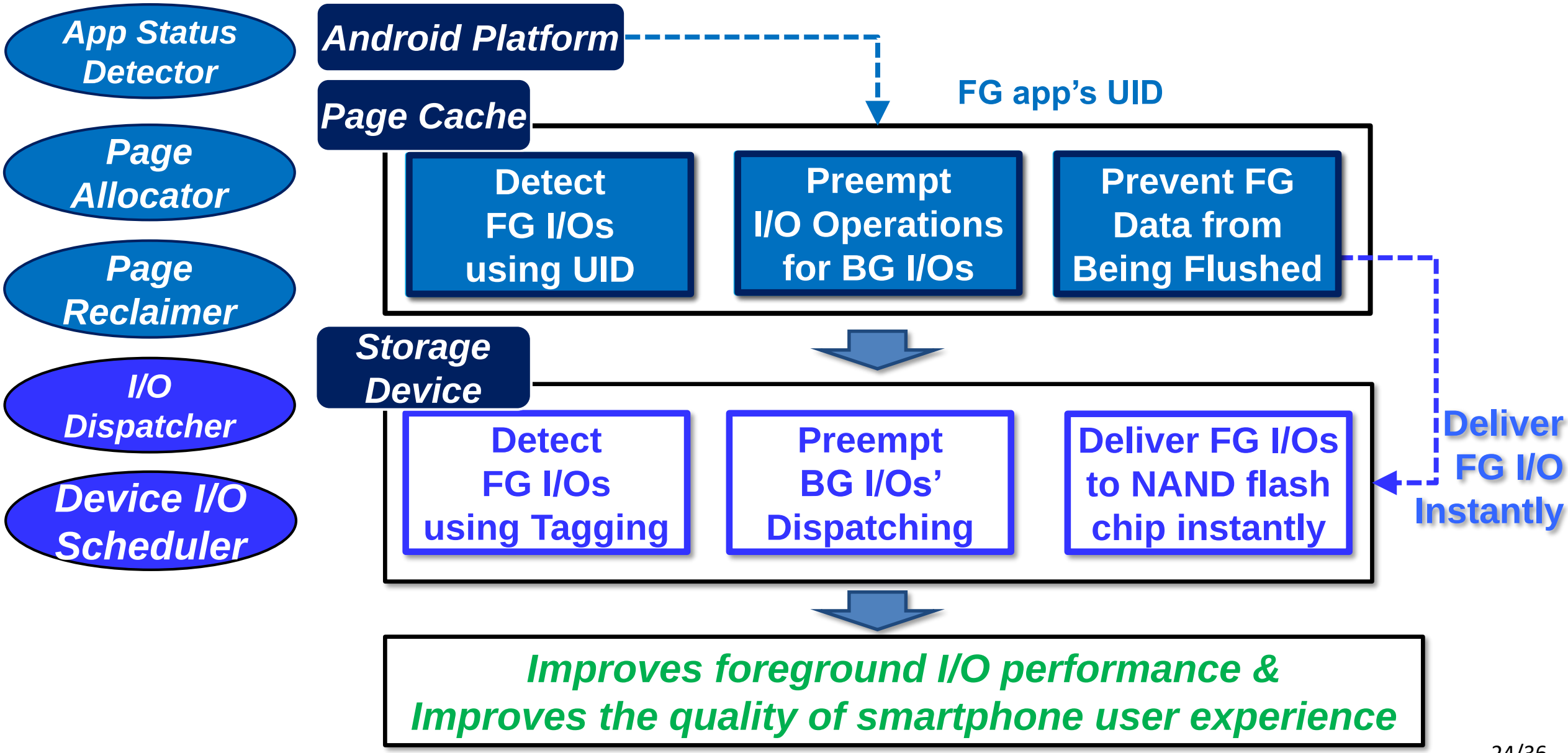
egrated

eeded

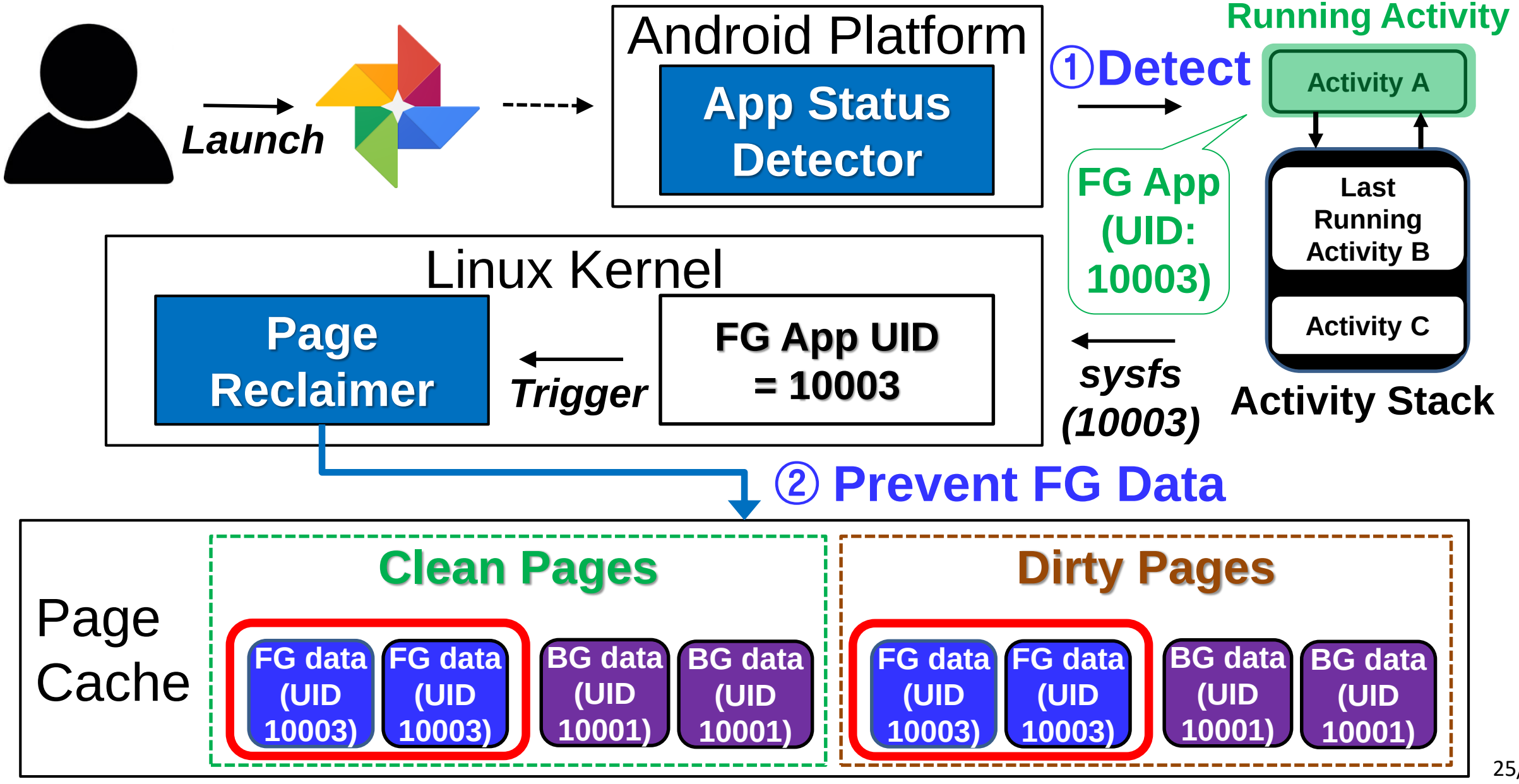
Outline

- Impact of Background I/O on User Experience
- Foreground-Background Interference Analysis
- **FastTrack: Foreground App-Aware I/O Management**
- Experimental Results
- Conclusions

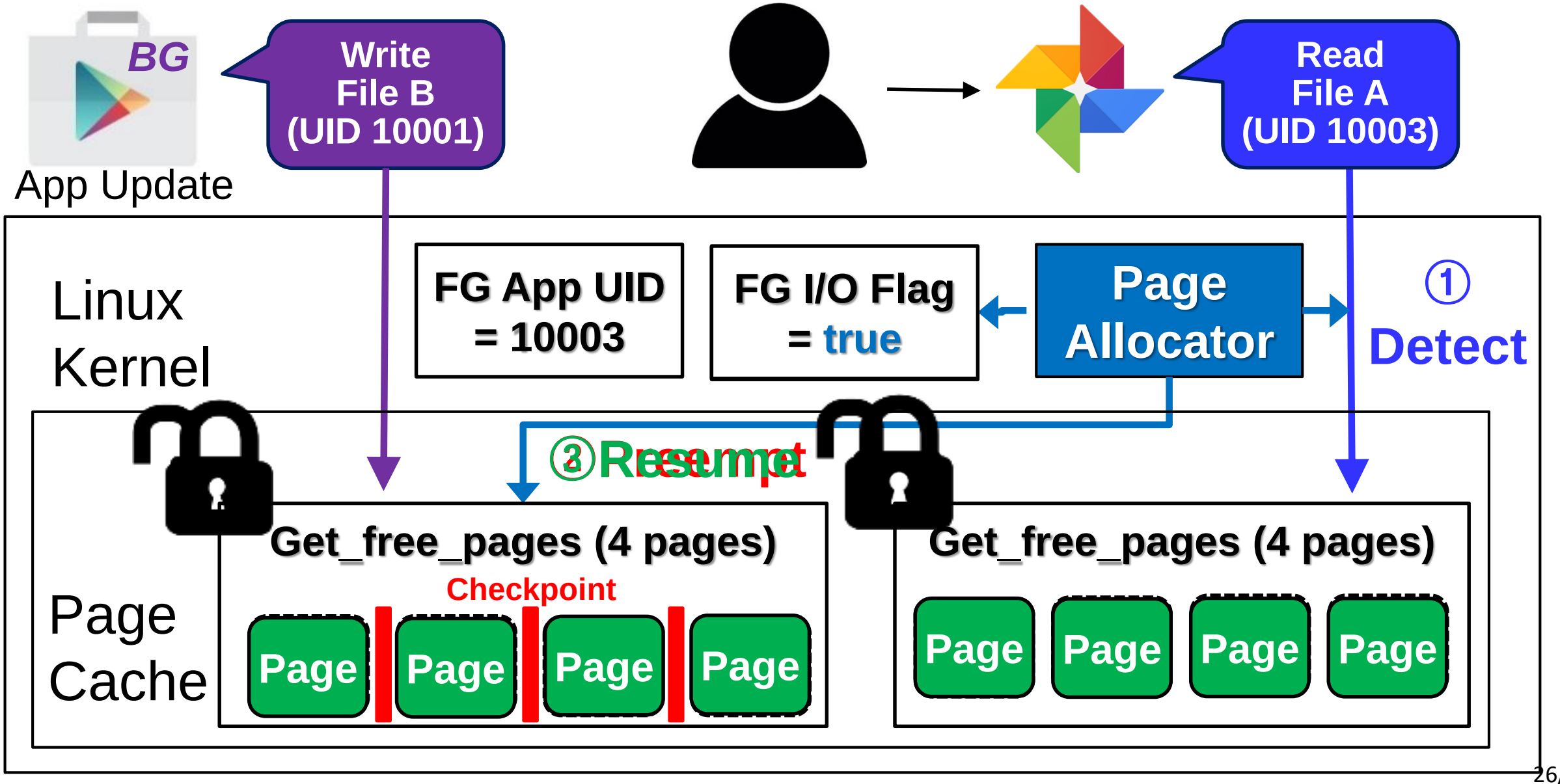
Overview of FastTrack



App Status Detector & Page Reclaimer

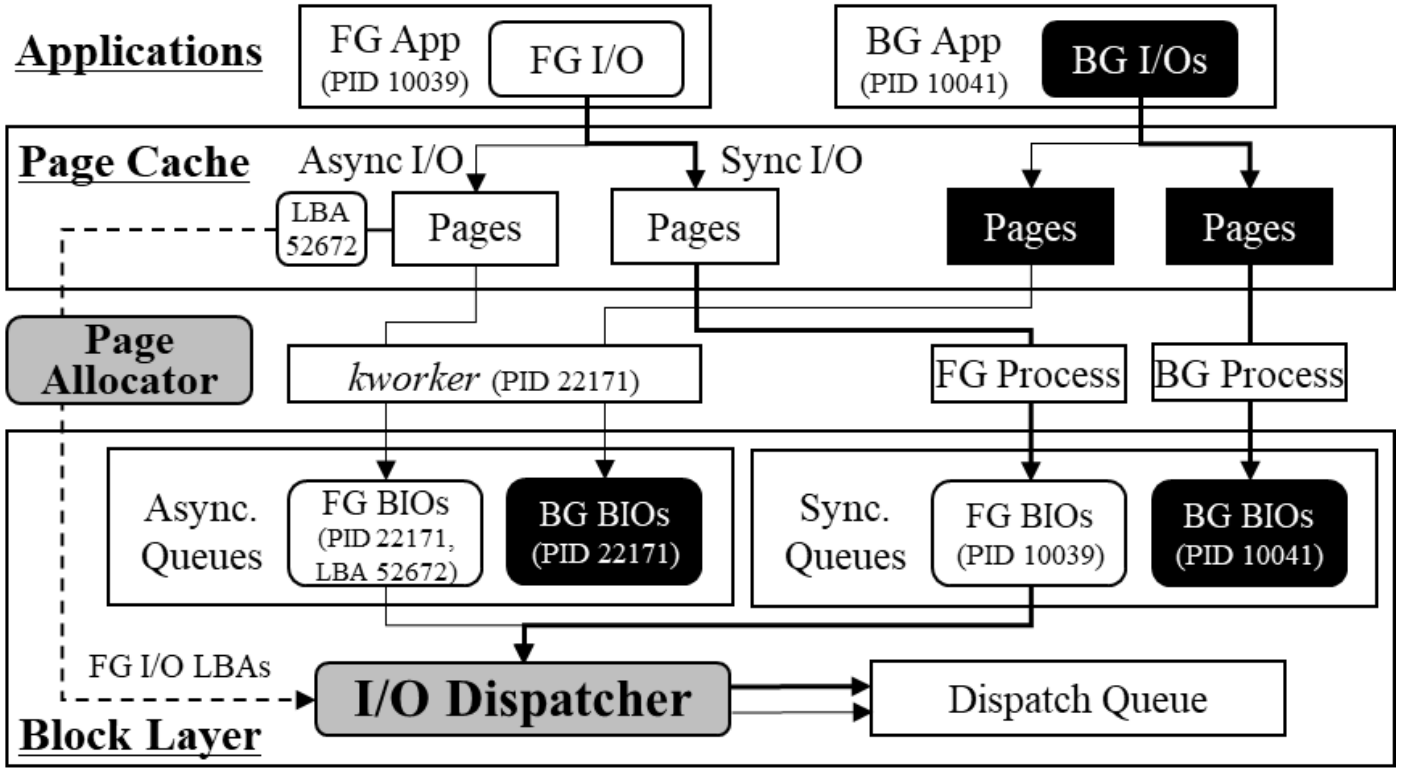


Page Allocator



I/O Dispatcher

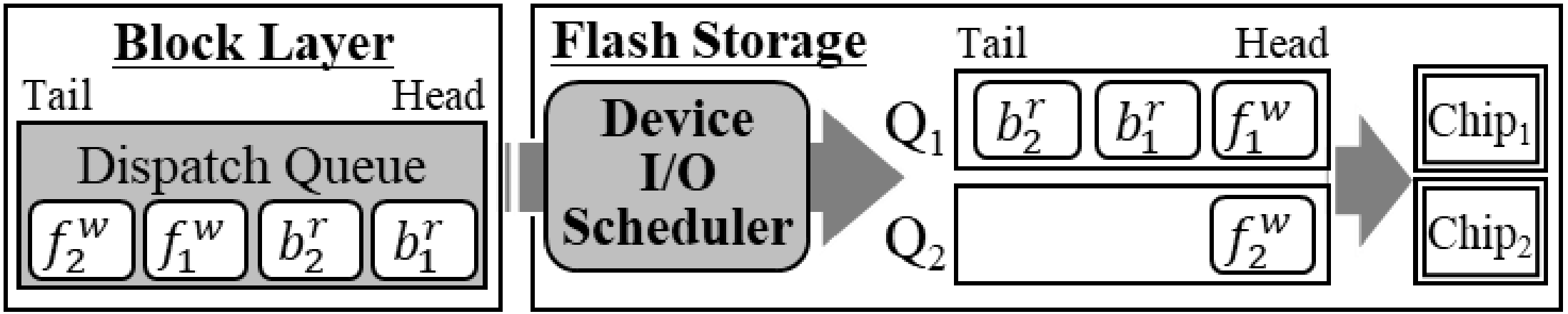
- **Moves FG block I/O requests to the dispatch queue immediately**
  - 1. Detect new block I/O request enters the sync/async queues
  - 2. Prevalidate whether new block I/O request is FG I/O
  - 3. Directly deliver FG block I/O request to the dispatch queue



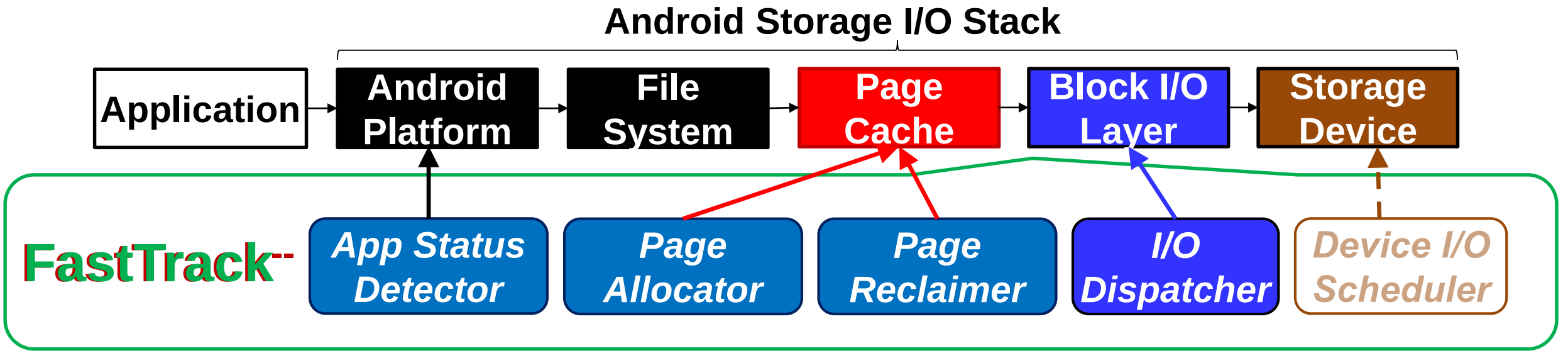
Device I/O Scheduler

Execute FG I/O commands with high priority

- New priorities for I/O command execution
  - ➢FG reads >FG writes > BG reads > BG writes
- Modify an SCSI command set to carries an FG I/O flag in a reserved opcode

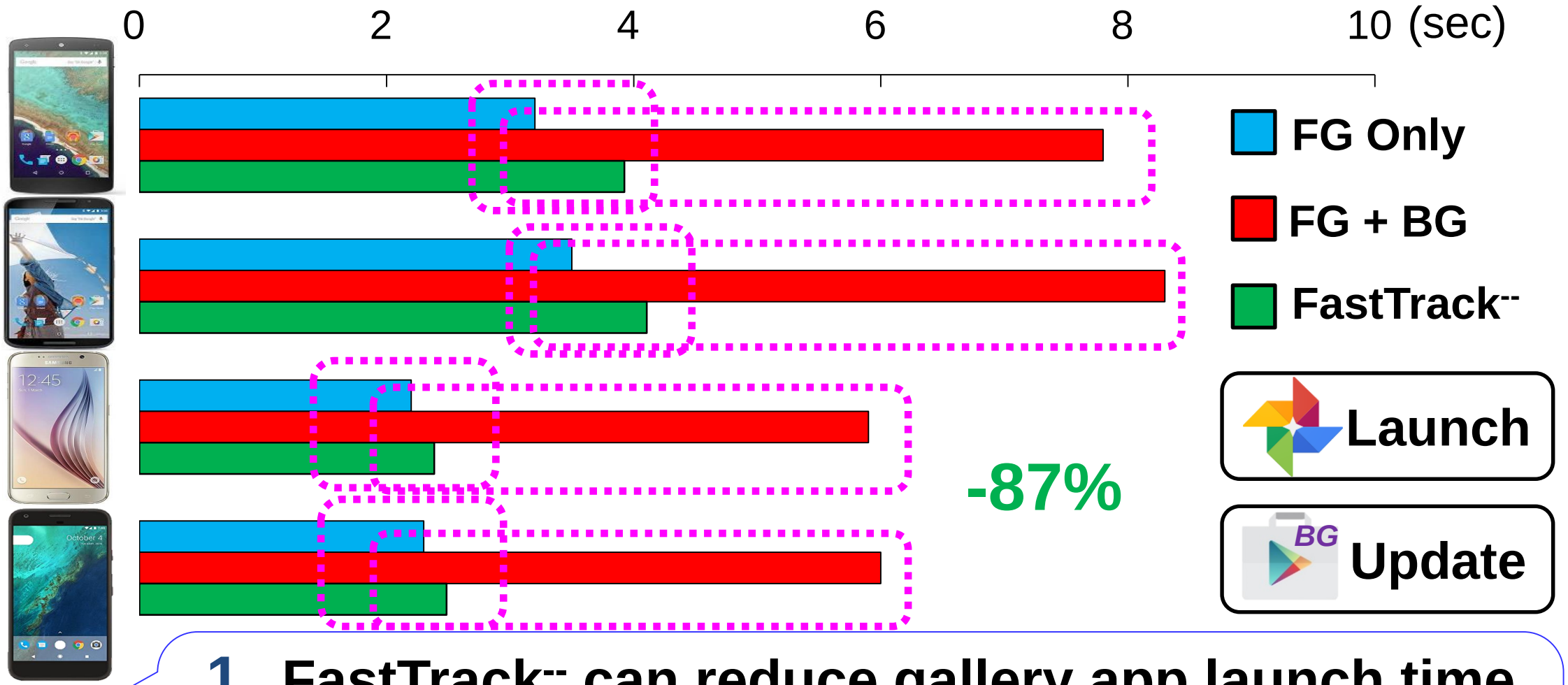


Experimental Settings for Android Smartphones



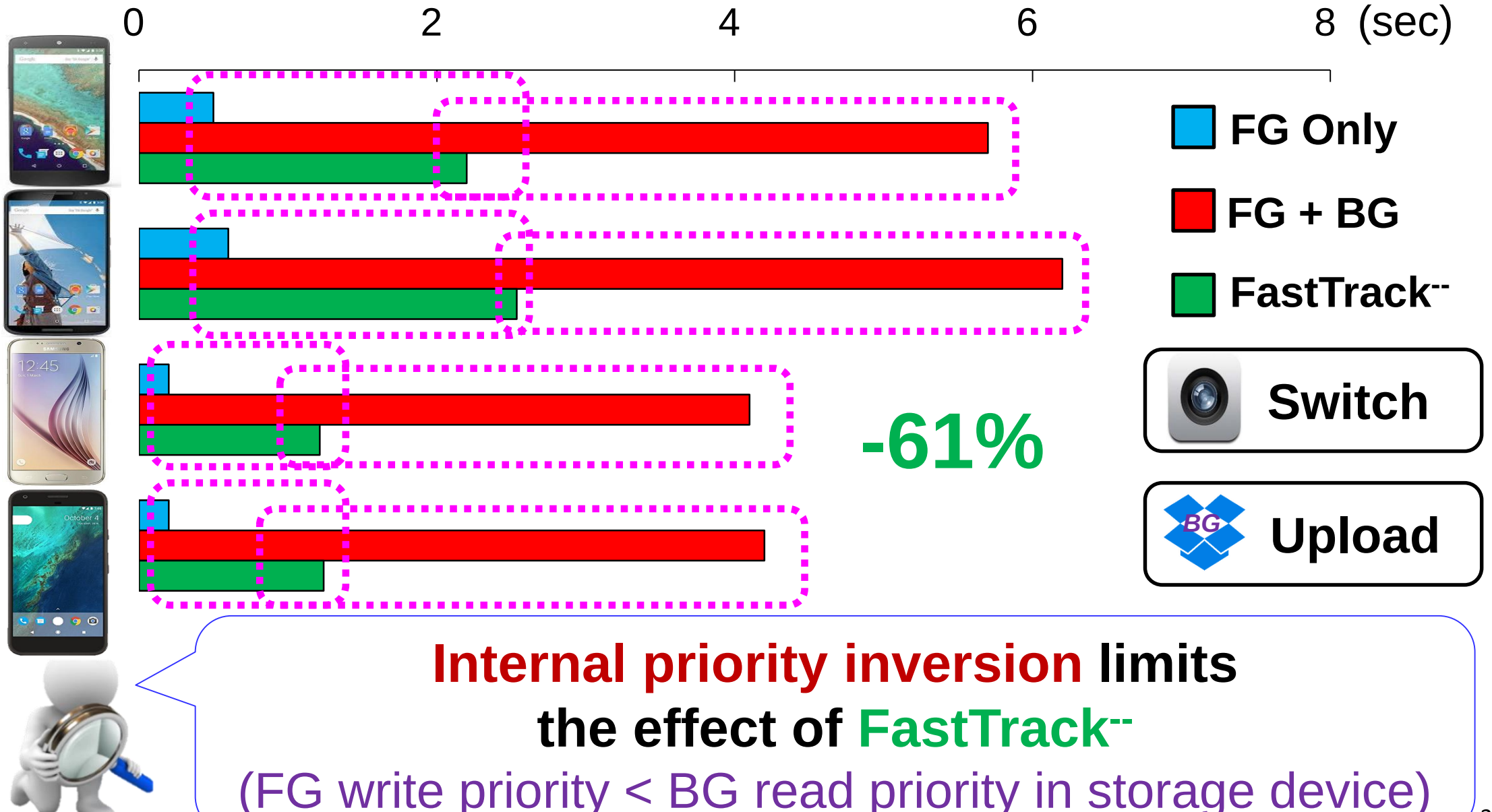
	FG App Usage Scenario		BG App Usage Scenario
①	Launch (Read-dominant)	+	Update (Write-dominant)
②	Switch (Write-dominant)	+	Upload (Read-dominant)

Result 1: App Launch Time Comparisons



1. FastTrack-- can reduce gallery app launch time delay from BG I/Os by up to 87%
2. FG reads have higher priority in storage device

Result 2: App Switch Time Comparisons



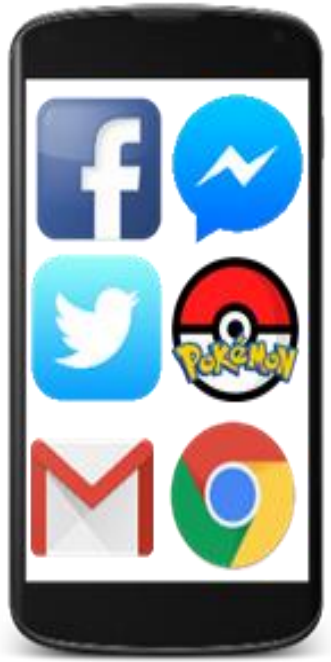
Internal priority inversion limits the effect of FastTrack--

(FG write priority < BG read priority in storage device)

Experimental Settings for Emulator

FastTrack--

- *App Status Detector*
- *Page Allocator*
- *Page Reclaimer*
- *I/O Dispatcher*
- *Device I/O Scheduler*

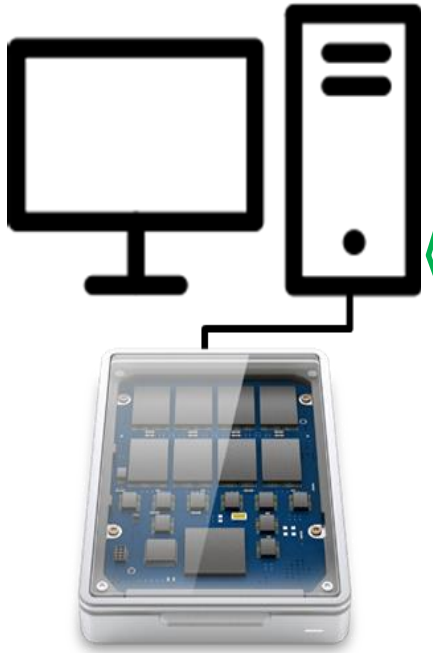


Android Smartphone

Application Launch/ Usage

System Call Trace

Trace Replayer



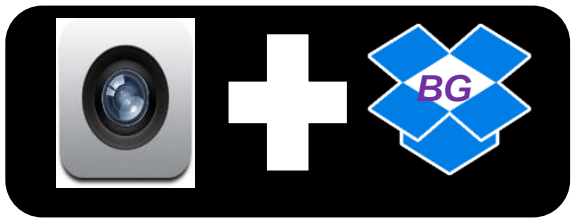
Emulation at Host-level FTL + Customized SSD

FastTrack

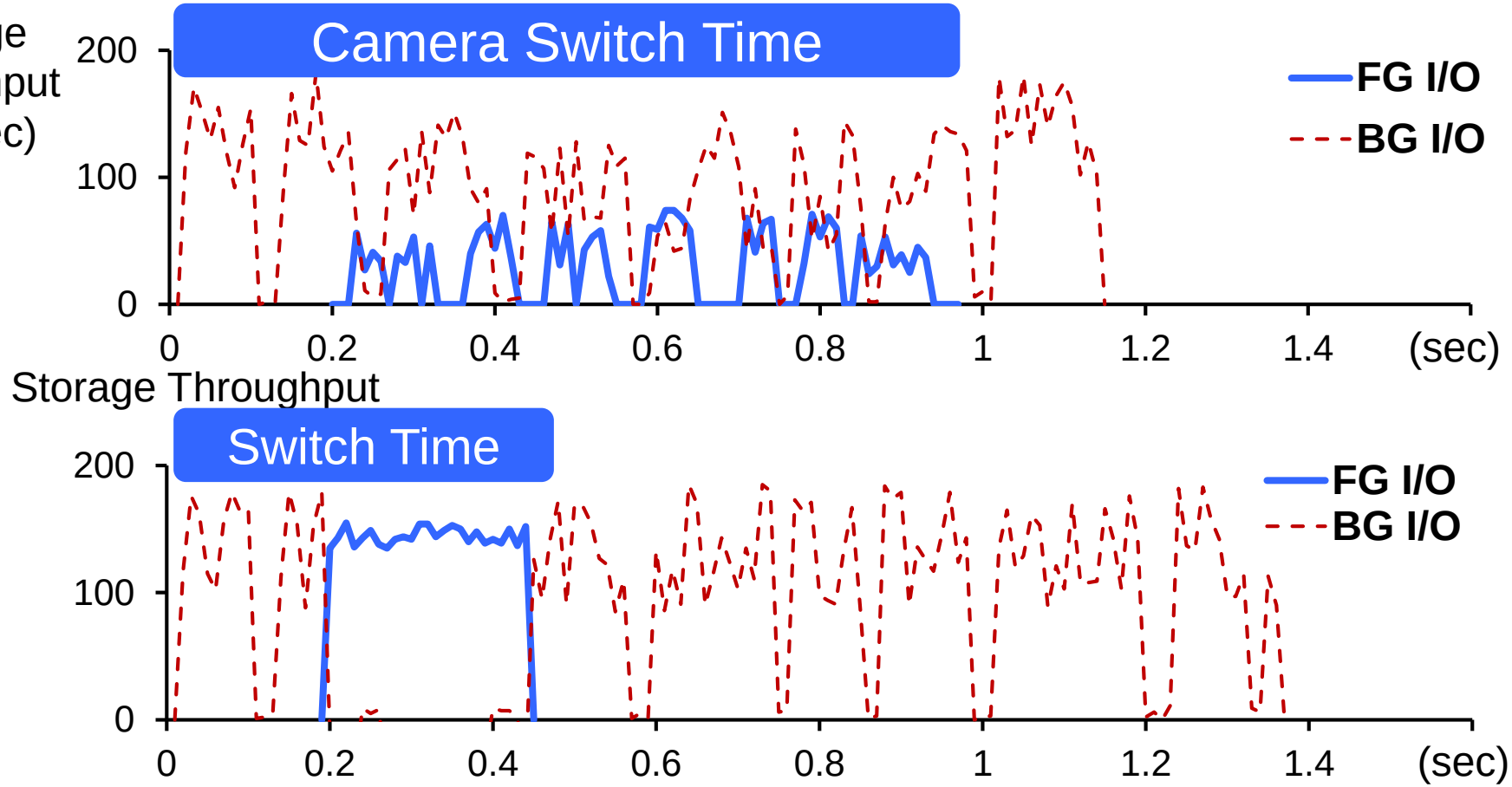
- *App Status Detector*
- *Page Allocator*
- *Page Reclaimer*
- *I/O Dispatcher*
- *Device I/O Scheduler*

Result 3: Storage-Level Snapshot

FastTrack--



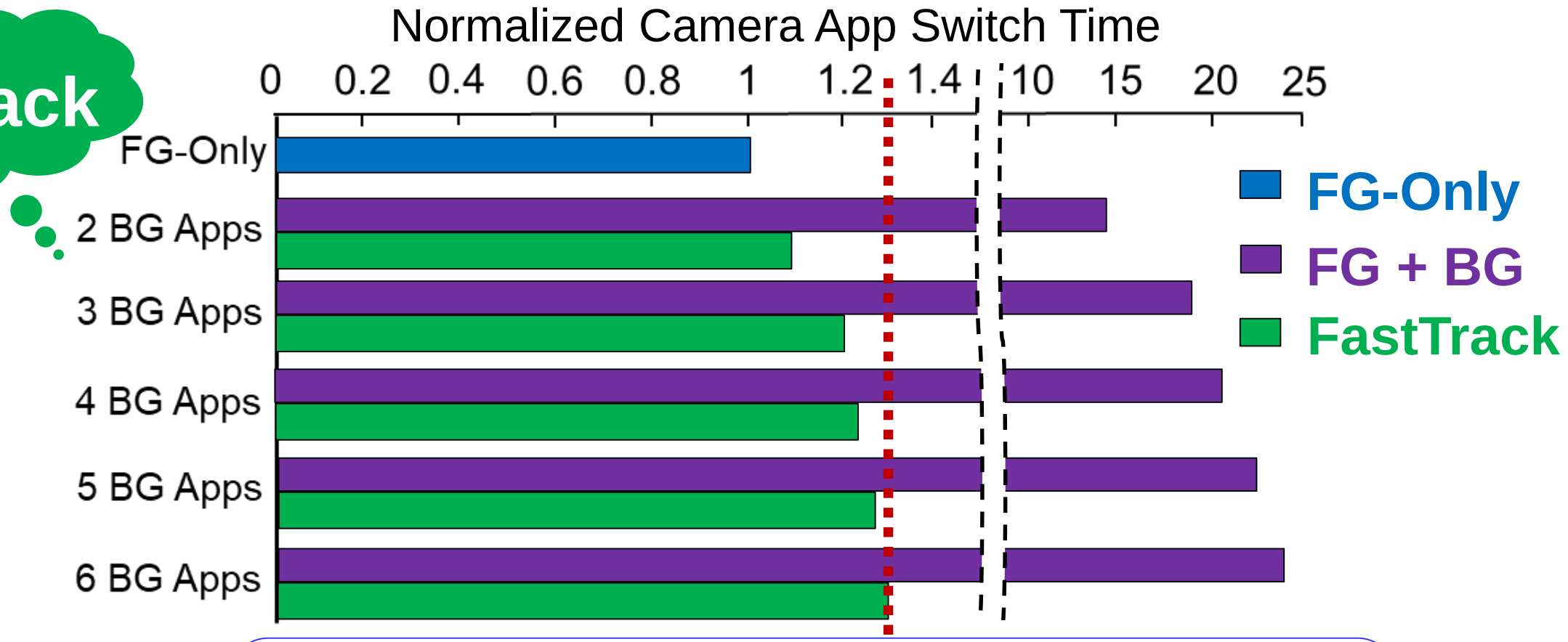
FastTrack



Device I/O scheduler **can provide**
a much higher throughput to FG I/Os
even when FG I/Os are write and BG I/Os are read

Result 4: Effectiveness of FastTrack over Varying BG Apps

FastTrack



FastTrack can provide the equivalent level of responsiveness to an FG app regardless of the number of BG apps

Conclusion

- **We have presented a foreground app-aware I/O management (FastTrack) for improving user experience**
  - **FastTrack preempts BG I/Os in the page cache**
  - **FastTrack prevents FG I/O's data from being flushed**
  - **FastTrack immediately delivers FG I/O to the NAND flash memory with minimum interference from inflight BG I/Os**
  - **FastTrack reduces the user-perceived response time delay by up to 95%**

- **Future work**
  - **Multiple foreground app usage environment (split view, multiple windows)**
  - **FastTrack for desktop/server computing system**


감사합니다
Natick
Grazie
Danke
Ευχαριστίες
Dalu
Obrigado
Thank You
Köszönöm
Tack
Спасибо
Dank
Gracias
谢谢
Merci
Seé
ありがとう