CAUSE: Critical Application Usage-Aware Memory System using Non-volatile Memory for Mobile Devices

Yeseong Kim, Mohsen Imani, Shruti Patil and Tajana S. Rosing

Mohsen Imani

UC San Diego
Department of Computer Science and Engineering

November 2015, ICCAD Conference

User Experience in Mobile

User experience

Performance

Power

Processor speed

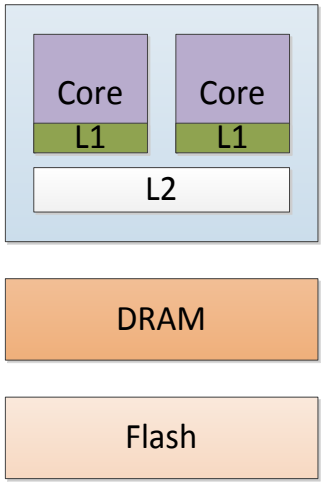
Memory bandwidth

Limited battery

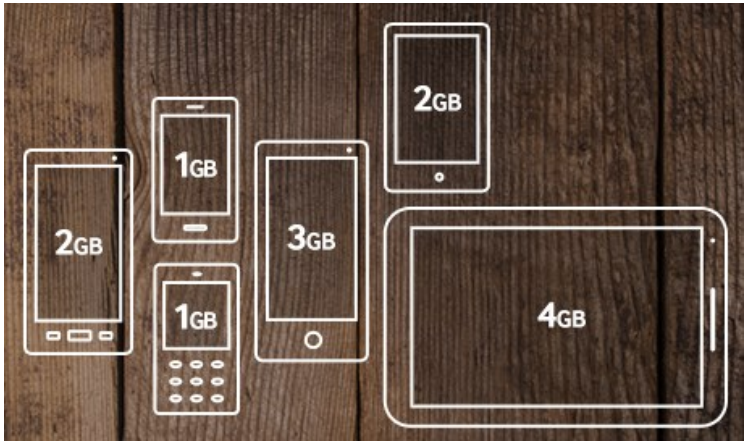
Launching Process (Angry Birds Rio)

Mobile, Applications and DRAM

- **Limited DRAM capacity**
  - State of the art mobile phones e.g 1G in iphone 6
  - 15% of applications close due to limited memory
  - Re-launching needs 10X slower [Wook et al. IEEE TECS14]



- **Problem caused by limited main memory**
  - Application termination
  - Launch time + energy overhead
  - Process service times

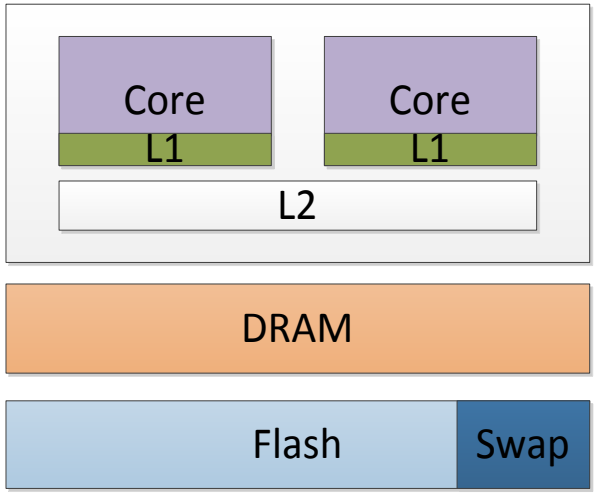


- **User experience degradation**

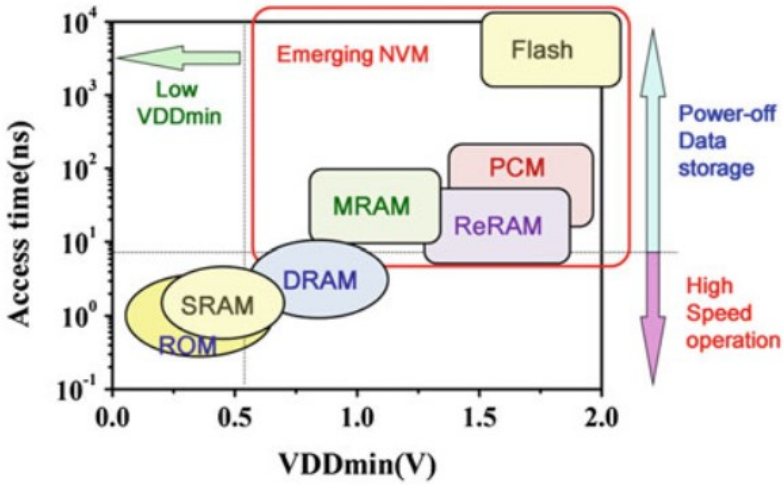
http://www.samsung.com/semiconductor/products/dram/mobile-dram/

Swap and Mobile Device

- **Swap Memory with Flash**
  - Energy
  - Latency
  - Endurance of eMMC flash



- **Emerging NVM technology**
  - Efficient read operation
  - Denser than DRAM (PCM ~2-4X)
  - **Low write performance!**

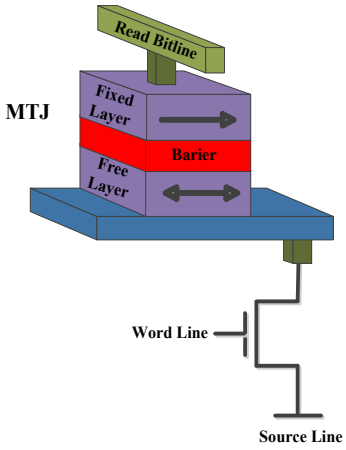


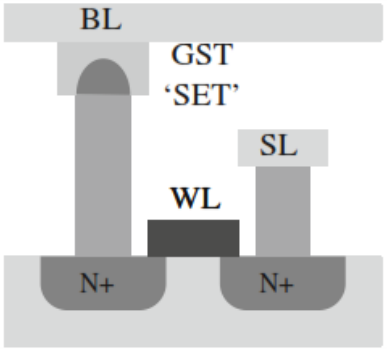
M-F. Chang, et al, IEEE ASPDAC, 2011.

Non-volatile Memory

- **STT-RAM**
  - Low leakage power
  - High endurance (10^{10}-10^{15})
  - Very fast in read
  - Write latency and energy!
  - Low scalability
- **PCM**
  - Low leakage power
  - Very high density
  - Scalable
  - Write latency and energy!
  - Low endurance (10^{6}-10^{7})

Solution?





Features	SRAM	eDRAM	STT-RAM	PCRAM
Density	Low	High	High	Very high
Speed	Very Fast	Fast	Fast for read; slow for write	Slow for read; very slow for write
Dynamic Power	Low	Medium	Low for read; very high for write	Medium for read; high for write
Leakage Power	High	Medium	Low	Low
Non-volatility	No	No	Yes	Yes

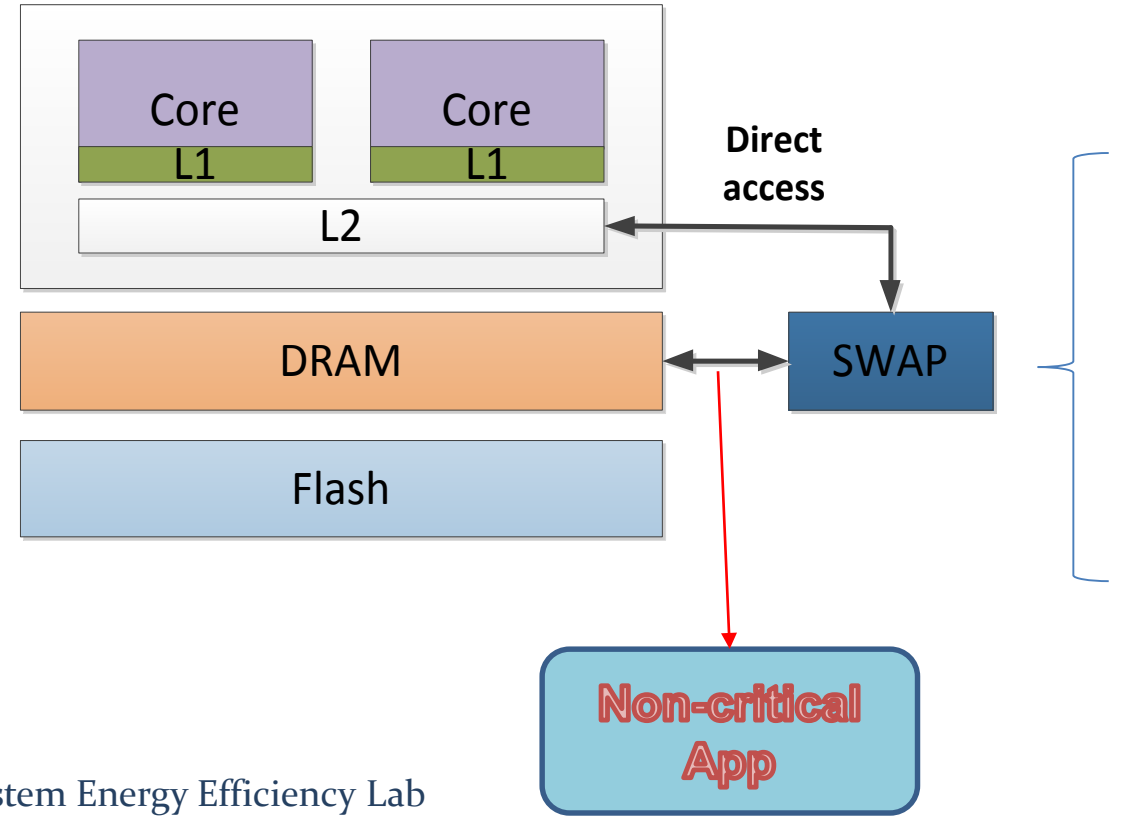
Challenges of NVM Based Swap Memory

- ***Software:***
  How to select apps which are good to be swapped?
  - Apps have different launching usage trends, resulting in distinct levels of criticality for user experience

- ***Hardware:***
  How to design NVMs for swap?
  - Apps have different access characteristics according to their status, e.g., foreground app and background service

CAUSE

- Critical Application Usage-Aware Memory System
  - Fast app launch: Better user experience!
  - Better process service time: More memory space for foreground apps



Optimized NVM for swap

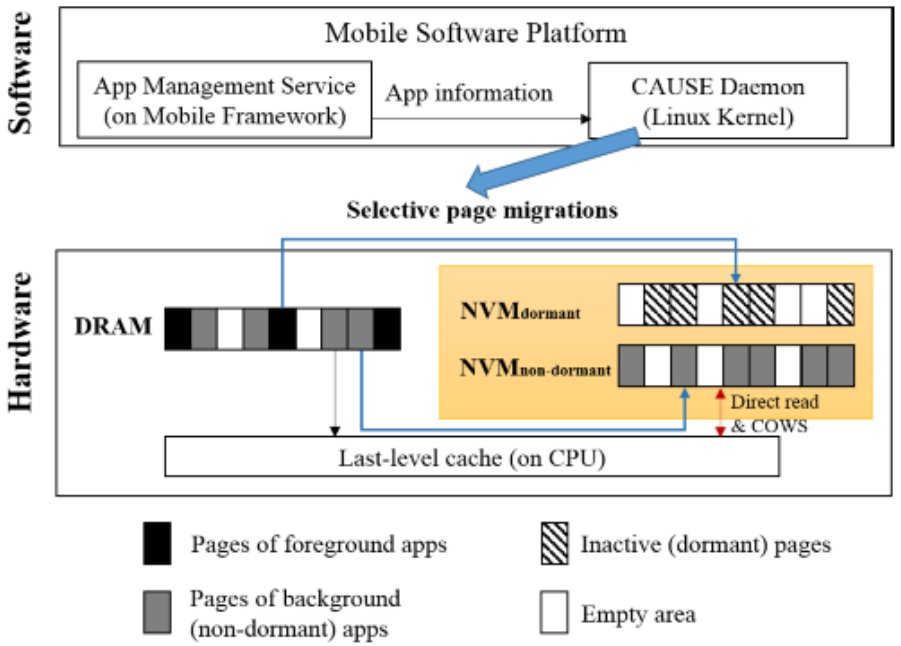
Low leakage power

Fast as DRAM

Memory Systems with NVM

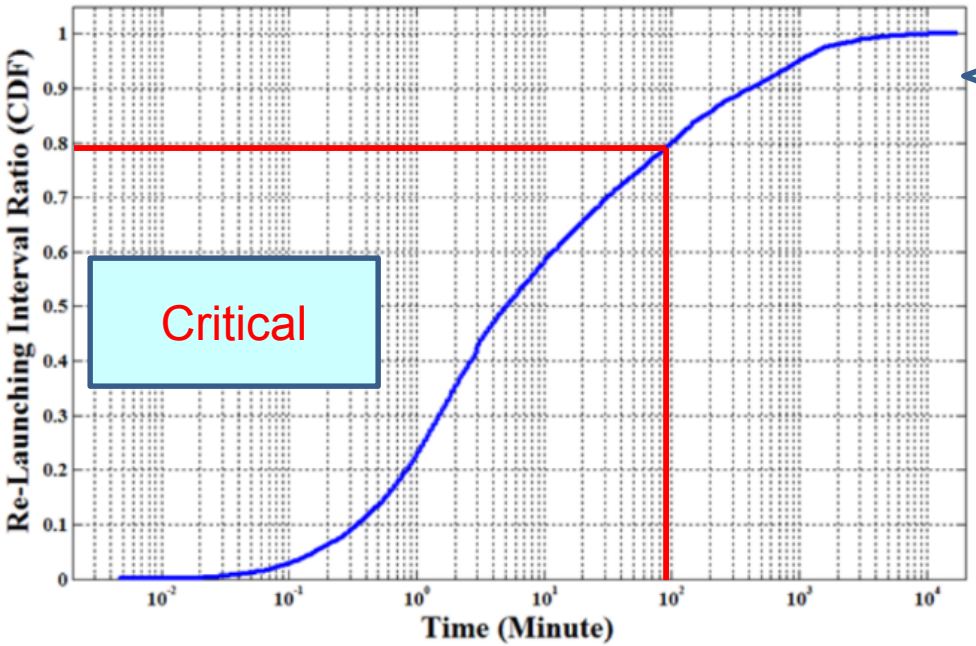
- **App management service:**
  - Recognizes applications launched by users
  - Tracks the applications recently launched in foreground
  - Sends the application information that is likely to be used in near future

- **Page characteristics:**
  - *Dormant:*
    - Foreground applications
    - Indeed not recently used
  - *Non-dormant:*
    - Likely to be accessed soon or periodically
    - Background applications and widget



Software: Application Launching Usage Trend

- Collected logs for two weeks from 10 users
- **Re-launching interval:** the time interval between two application launches for a certain application.
- 80% of applications were reused within 100 minutes!



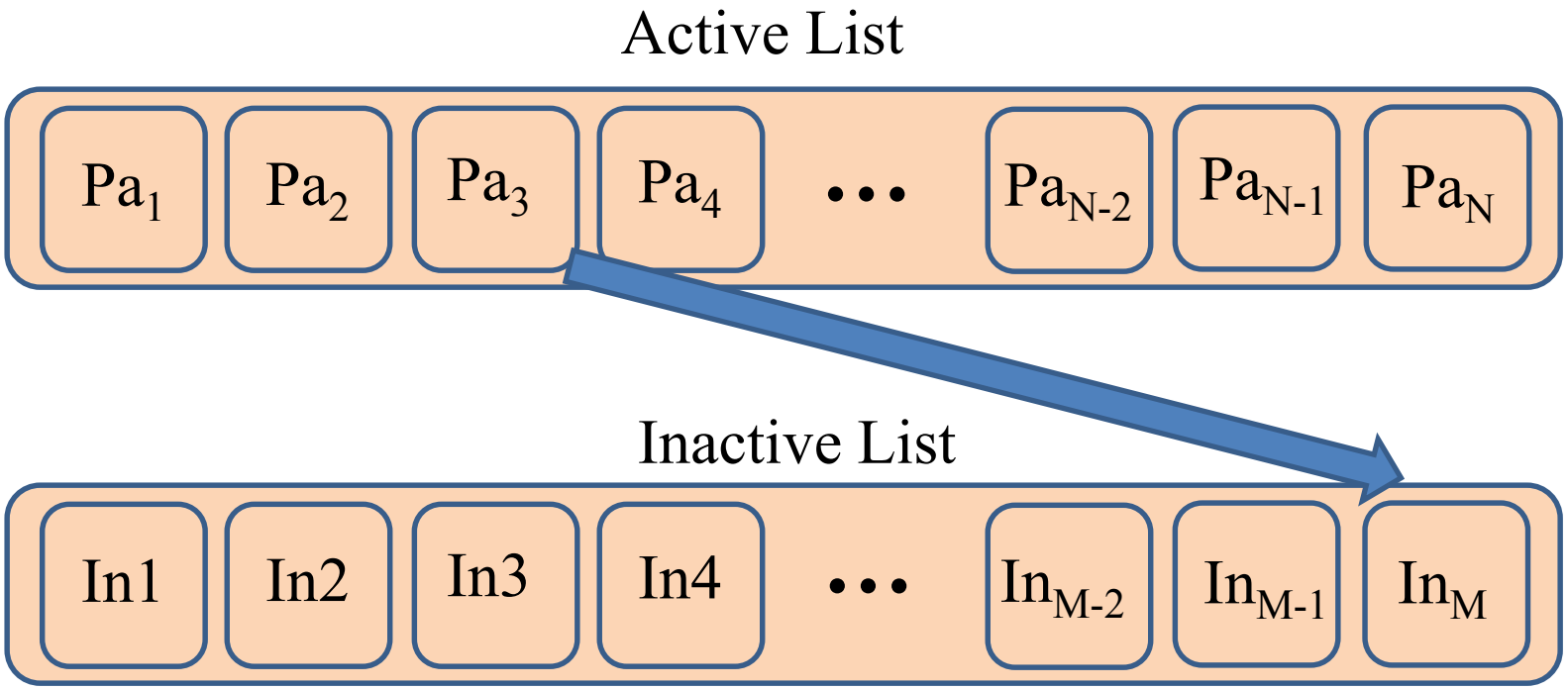
80% of app reused within 100 minutes!

Apps after 100min consider as inactive

CAUSE Management Policy

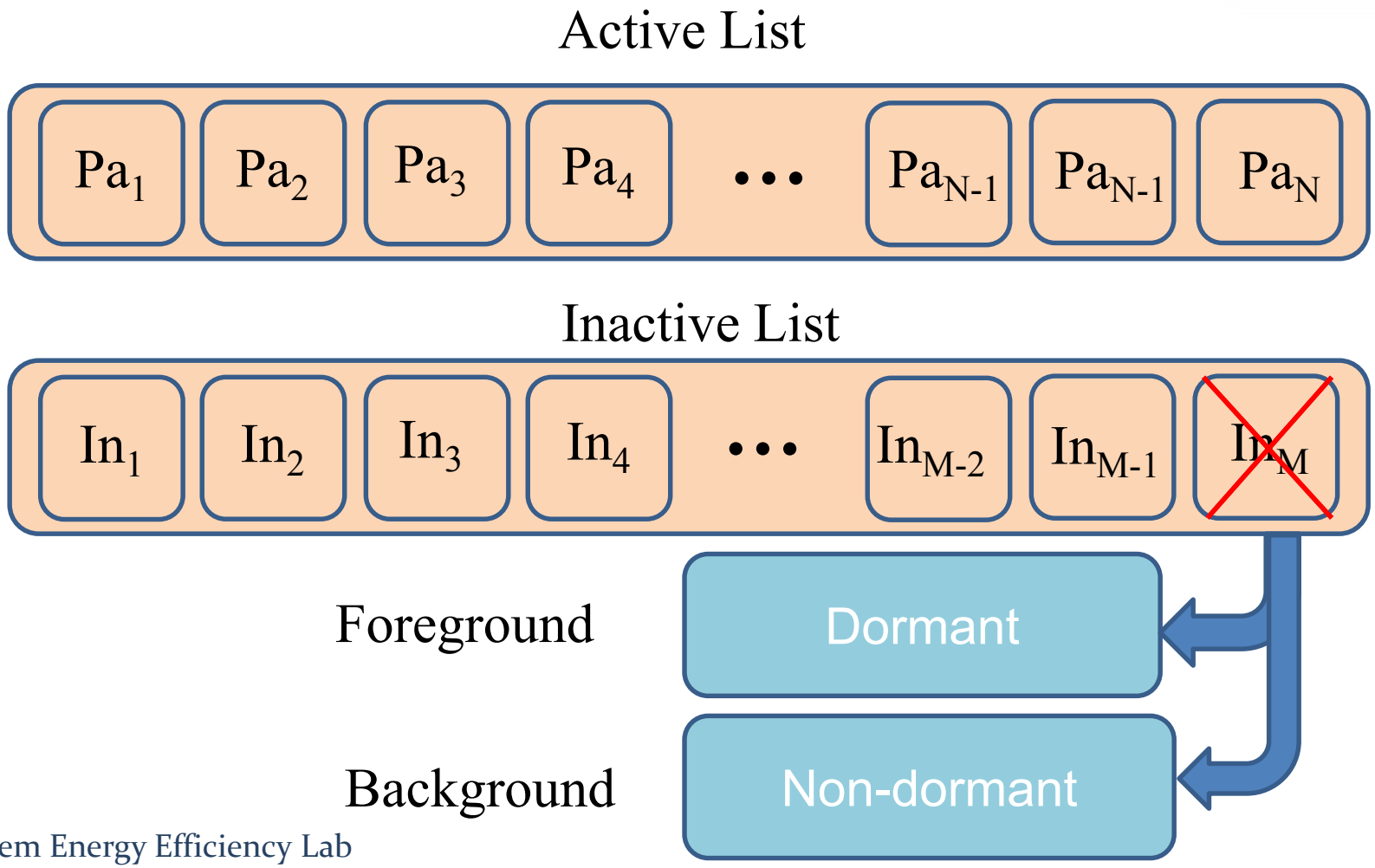
Active List Management

- Linux policy: balancing the number of active and inactive pages

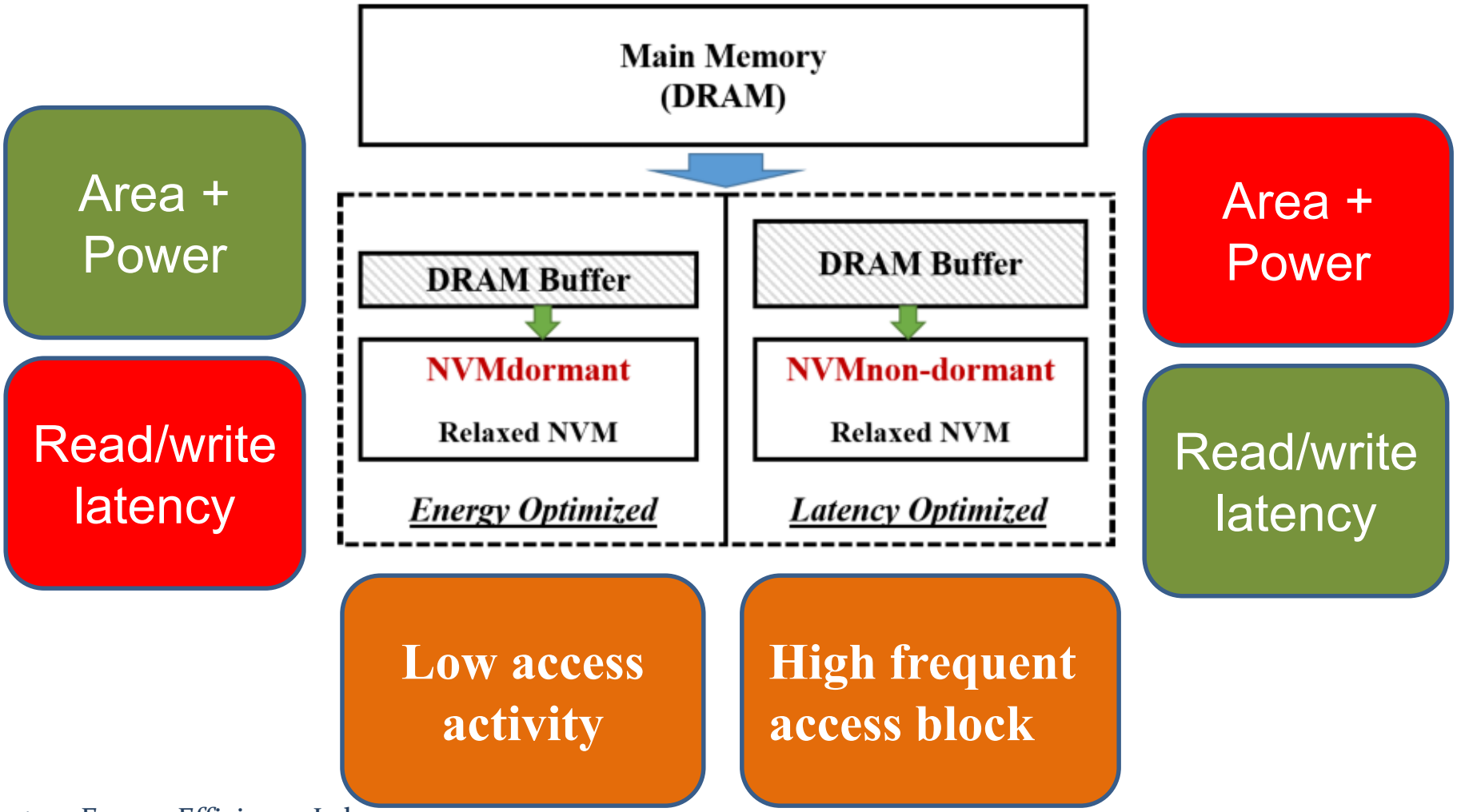


CAUSE Management Policy

Freeing memory pages

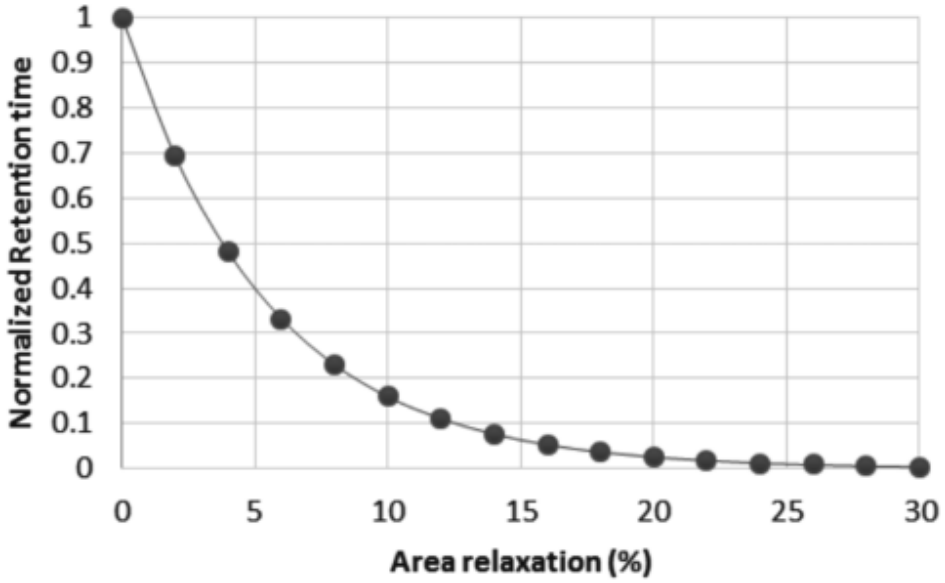


Hardware:
Buffer Optimization



Retention Relaxation

- **STT-RAM retention** [Smullen et al, HPCA 2011]
  - 20% MTJ area relaxation
  - 83% write latency improvement
  - Retention time from 20 years to 1 month
  - Possibility of refresh

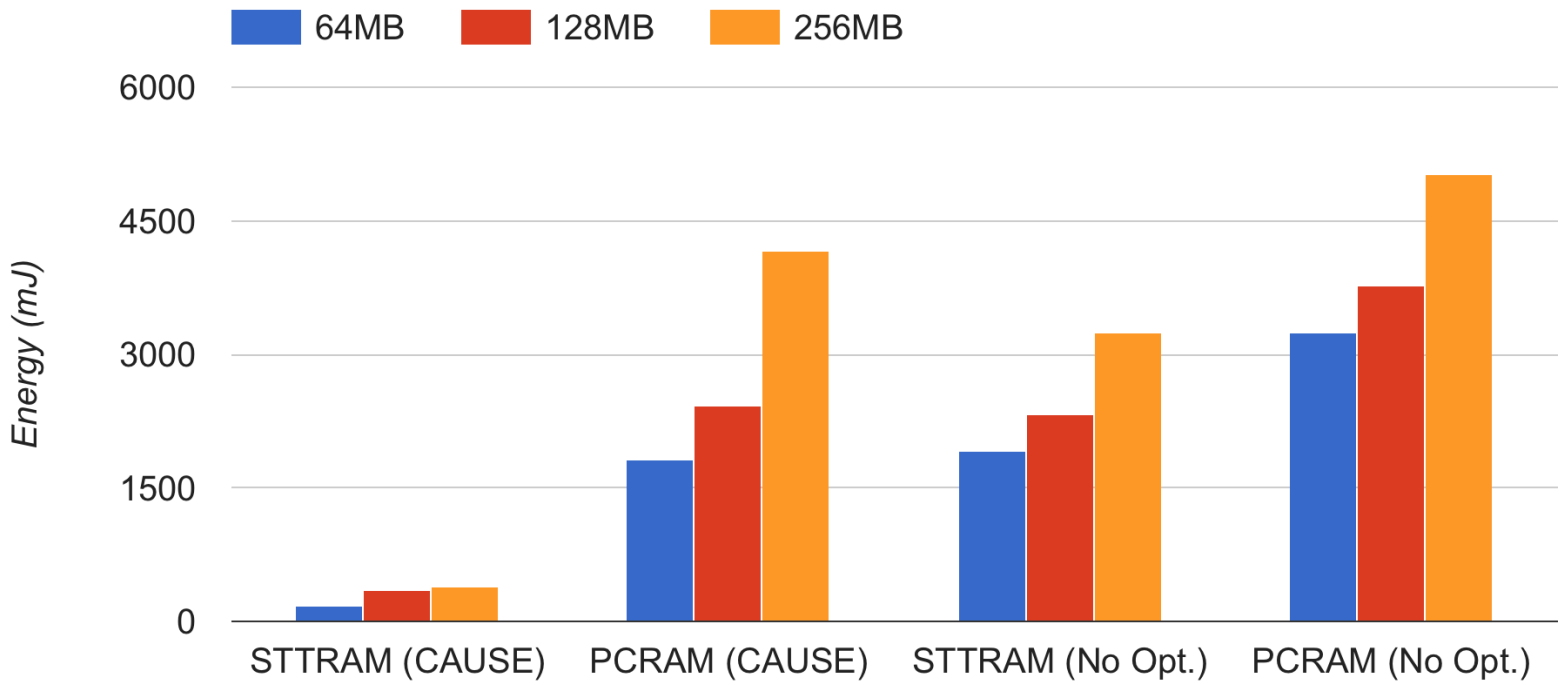


Experimental Setup

- Qualcomm MSM8660 smartphone
  - Running Android 4.1 with Linux kernel 3.0.6
  - 1GB Main memory; 768MB DRAM; 64KB-256KB NVM
- HSPICE for circuit level simulation
  - Retention time relaxation
  - Circuit level optimization
- NVsim & NVmain simulators for energy estimation
  - DRAM buffer
  - Buffer design
- 10 users launch 20 apps for 2 weeks
  - Scale down the executed time to 20 mins

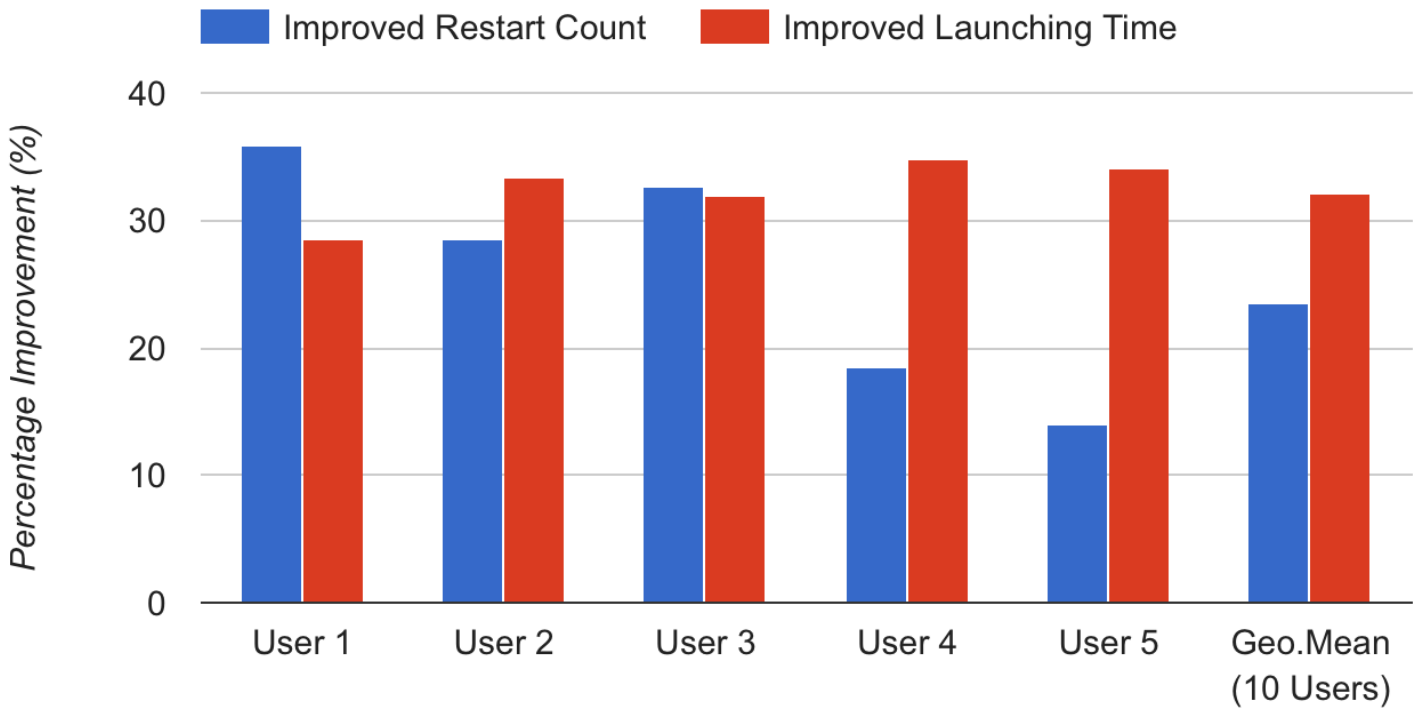
CAUSE Energy Consumption

- Comparison of energy consumption with different memory technologies
  - 90% and 44% energy savings for STT-RAM and PCM



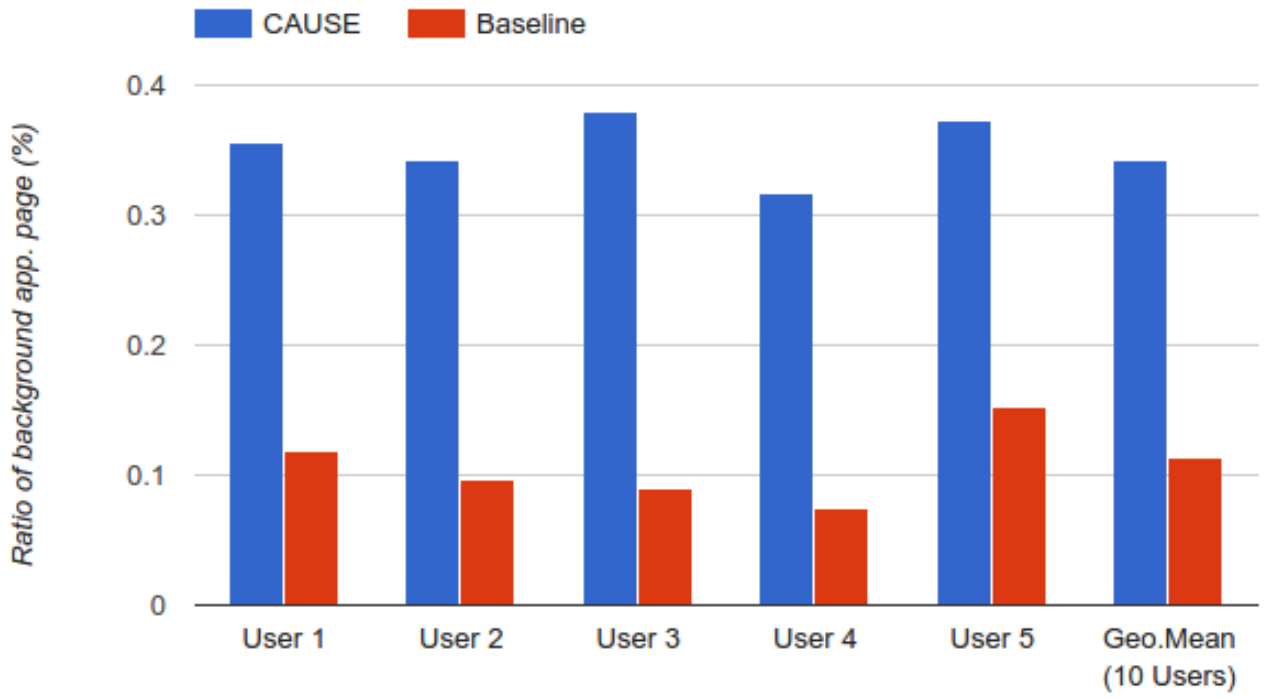
Launch Experience

- 32% launching time speed up
  - Better user experience ☺



Background Page Balancing

- 23% more background page migration
  - Provides space for foreground applications
  - Better process service time ☺



Summary

- Addressed limited main memory capacity of mobile devices
- Proposed new swap architecture to save the inactive pages based on applications and users
- Proposed & optimized dormant and non-dormant memory components for background and foreground applications
- 23% more background migrations + better process service time
- 32% launching time speed up + better user experience