Transparent Smartphone Memory Expansion

Gyusun Lee
Yonsei University
Department of Computer Science
South Korea
gyusun.lee@csl.skku.edu

Jinyong Park*
Sungkyunkwan University
Department of Semiconductor and Display Engineering
South Korea
jinyong.park@csl.skku.edu

Jae W. Lee
Seoul National University
Department of Computer Science and Engineering
South Korea
jaewlee@snu.ac.kr

Jinkyu Jeong†
Yonsei University
Department of Computer Science
South Korea
jinkyu@yonsei.ac.kr

ABSTRACT

Traditionally, mobile systems avoid virtual memory swapping to flash memory due to wear-out and latency issues. Instead, mobile systems kill background applications to secure free memory but this leads to long-latency cold startup of applications, thereby resulting in degraded user experience. With the advance of mobile network technologies, network-based memory expansion can offer a solution, allowing users to leverage large back-end memory on edge servers. However, traditional memory expansion methods still suffer from high latency due to on-demand page faults and difficulty in identifying numerous cold pages. We propose a novel memory expansion scheme tailored for fast networks in mobile systems. Our scheme minimizes the on-demand network latency of remote I/O by bulk-prefetching pages based on access sequences during an application launches. Our system also effectively identifies and exports cold pages to the edge, freeing up local memory. Our evaluations show a 41% reduction in displayed time and 62% memory expansion with 42% faster launch times, outperforming both traditional swapping and app-killing mechanisms of conventional systems.

CCS CONCEPTS

• **Computer systems organization → Embedded software**; • **Software and its engineering → Virtual memory**; • **Information systems → Network attached storage**; • **Human-centered computing → Empirical studies in ubiquitous and mobile computing**.

KEYWORDS

Mobile systems, Edge computing, Secondary storage, Memory management

*Now at Samsung Electronics
†Jinkyu Jeong is the corresponding author.

SAC'25, March 31 –April 4, 2025, Sicily, Italy

ACM ISBN 979-8-4007-0629-5/25/03…$15.00
https://doi.org/10.1145/3672608.3707936

ACM Reference Format:
Gyusun Lee, Jinyong Park, Jae W. Lee, and Jinkyu Jeong. 2025. Transparent Smartphone Memory Expansion. In *Proceedings of ACM SAC Conference (SAC'25).* ACM, New York, NY, USA, Article 4, 10 pages. https://doi.org/10.1145/3672608.3707936

1 INTRODUCTION

Mobile smartphones have rapidly evolved into the most common personal computers globally [28, 30]. In smartphones, *application caching* is a crucial technique that enhances user experience under hardware limitations like screen and memory size. Although smartphones can run many applications, only one or a few applications can be active on the small screen. The memory not used by foreground applications is utilized for caching background apps. When users want to access a cached application, it loads almost instantly, as *switching* to a cached background app is simple and quick. This approach also preserves the application's state, eliminating the need for the time-consuming process of restarting the application and setting up its state.

Application caching becomes ineffective when mobile systems face memory shortages. When free memory is low, mobile operating systems like Android reclaim memory by terminating cached applications [3, 13]. Consequently, restarting an application can be time-consuming, requiring users to take extra steps to return to the application's previous state, which degrades the smartphone user experience. This issue is made more significant by the increasing memory demands of mobile applications [31]. Various methods have been proposed to enhance application caching, such as compressing memory [9, 16] or swapping pages to local or cloud storage [5, 17, 35]. While memory compression increases memory density, it still relies on the device's DRAM. Swapping techniques depend on traditional virtual memory swapping [4], leading to application responsiveness issues during page faults, resulting in increased latency when handling network faults.

This paper proposes memory expansion of mobile systems with edge computing. With advancements in mobile networks, modern smartphones are now just one hop away from high-performance servers. Technologies like 5G offer low latency and high bandwidth between mobile devices and edge servers. Our approach moves beyond traditional demand paging by proactively managing application memory across mobile systems and edge servers. Our scheme analyzes page access sequences and classifies them into

to remote storage. ACR-Swap$^+$[20] proposed remote swapping between networked devices, prioritizing remote pages based on the swap-in frequency of app and system processes. While these works reduce app switch latency through prefetching, some swap-ins still depend on on-demand I/O. Our sequence-based prefetching further reduces on-demand faults by retaining early pages locally, improving prefetching efficiency.

Mobile systems with Edge or Cloud. The issue of limited resources in mobile systems is often addressed through offloading data and computation to edge or cloud servers [15, 23]. However, we focus on overcoming memory shortages in mobile devices by designing a sophisticated system to expand their memory.

8 CONCLUSION

With the emergence of fast networks, memory expansion via edge servers in mobile systems is now feasible. However, traditional methods like swapping face limitations due to network overhead from on-demand paging. To address this, we propose a new memory management scheme that expands the memory of mobile systems by selectively exporting or prefetching pages using a sequence-based prefetch method, reducing network faults. Additionally, we identify and offload cold pages to edge storage, further expanding memory. This approach improves application performance over vanilla kernel or traditional flash swap methods, enabling seamless use of edge storage as local memory.

9 ACKNOWLEDGEMENTS

We would like to thank the anonymous reviewers for their valuable comments. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (RS-2024-00395134, DPU-Centric Datacenter Architecture for Next-Generation AI Devices) and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2023-00321688).

REFERENCES

[1] Android. 2020. Define Custom Permissions. https://developer.android.com/guide/topics/permissions/defining.
[2] Android. 2022. Android App startup time. https://developer.android.com/topic/performance/vitals/launch-time.
[3] Android open source project. 2018. low memory killer. https://source.android.com/devices/tech/perf/lmkd.
[4] Remzi H Arpaci-Dusseau and Andrea C Arpaci-Dusseau. 2018. *Operating systems: Three easy pieces.* Arpaci-Dusseau Books LLC Boston.
[5] Dongju Chae, Joonsung Kim, Youngsok Kim, Jangwoo Kim, Kyung-Ah Chang, Sang-Bum Suh, and Hyogun Lee. 2016. CloudSwap: A cloud-assisted swap mechanism for mobile devices. In *2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid).* IEEE, 462–472.
[6] Android Developers. 2022. Memory allocation among processes. https://developer.android.com/topic/performance/memory-management.
[7] George Dramitinos and Evangelos P Markatos. 1999. Adaptive and reliable paging to remote main memory. *J. Parallel and Distrib. Comput.* 58, 3 (1999), 357–388.
[8] Juncheng Gu, Youngmoon Lee, Yiwen Zhang, Mosharaf Chowdhury, and Kang G Shin. 2017. Efficient memory disaggregation with infiniswap. In *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17).* 649–667.
[9] Nitin Gupta. 2014. zram: Compressed RAM based block devices. In *Proc. Linux Kernel User's Administrator's Guide.*
[10] Sangwook Shane Hahn, Sungjin Lee, Inhyuk Yee, Donguk Ryu, and Jihong Kim. 2018. {FastTrack}: Foreground {App-Aware}{I/O} Management for Improving User Experience of Android Smartphones. In *2018 USENIX Annual Technical Conference (USENIX ATC 18).* 15–28.
[11] Najmul Hassan, Kok-Lim Alvin Yau, and Celimuge Wu. 2019. Edge Computing in 5G: A review. *IEEE Access* 7 (2019), 127276–127289.
[12] Jongseok Kim, Cheolgi Kim, and Euiseong Seo. 2019. *ezswap*: Enhanced compressed swap scheme for mobile devices. *IEEE Access* 7 (2019), 139678–139691.
[13] Sang-Hoon Kim, Jinkyu Jeong, Jin-Soo Kim, and Seungryoul Maeng. 2016. SmartLMK: A memory reclamation scheme for improving user-perceived app launch time. *ACM Transactions on Embedded Computing Systems (TECS)* 15, 3 (2016), 1–25.
[14] Sung-hun Kim, Jinkyu Jeong, and Joonwon Lee. 2014. Efficient memory deduplication for mobile smart devices. In *2014 IEEE International Conference on Consumer Electronics (ICCE).* IEEE, 25–26.
[15] Yeongjin Kim, Hyang-Won Lee, and Song Chong. 2018. Mobile computation offloading for application throughput fairness and energy efficiency. *IEEE Transactions on Wireless Communications* 18, 1 (2018), 3–19.
[16] Sejun Kwon, Sang-Hoon Kim, Jin-Soo Kim, and Jinkyu Jeong. 2015. Managing GPU buffers for caching more apps in mobile systems. In *2015 International Conference on Embedded Software (EMSOFT).* IEEE, 207–216.
[17] Niel Lebeck, Arvind Krishnamurthy, Henry M Levy, and Irene Zhang. 2020. End the senseless killing: Improving memory management for mobile operating systems. In *2020 USENIX Annual Technical Conference (USENIX ATC 20).* 873–887.
[18] Changlong Li, Yu Liang, Rachata Ausavarungnirun, Zongwei Zhu, Liang Shi, and Chuan Jason Xue. 2023. Ice: Collaborating memory and process management for user experience on resource-limited mobile devices. In *Proceedings of the Eighteenth European Conference on Computer Systems.* 79–93.
[19] Changlong Li, Liang Shi, and Chun Jason Xue. 2021. MobileSwap: Cross-Device Memory Swapping for Mobile Devices. In *2021 58th ACM/IEEE Design Automation Conference (DAC).* IEEE, 115–120.
[20] Wentong Li, Yina Lv, Longfei Luo, Yunpeng Song, and Liang Shi. 2024. Access Characteristic-Guided Remote Swapping Across Mobile Devices. *ACM Transactions on Architecture and Code Optimization* (2024).
[21] Shuang Liang, Ranjit Noronha, and Dhabaleswar K Panda. 2005. Swapping to remote memory over infiniband: An approach using a high performance network block device. In *2005 IEEE International Conference on Cluster Computing.* IEEE.
[22] Yu Liang, Jinheng Li, Rachata Ausavarungnirun, Riwei Pan, Liang Shi, Tei-Wei Kuo, and Chun Jason Xue. 2020. Acclaim: Adaptive memory reclaim to improve user experience in android systems. In *2020 USENIX Annual Technical Conference (USENIX ATC 20).* 897–910.
[23] Juan Liu, Yuyi Mao, Jun Zhang, and Khaled B Letaief. 2016. Delay-optimal computation task scheduling for mobile-edge computing systems. In *2016 IEEE international symposium on information theory (ISIT).* IEEE, 1451–1455.
[24] Linux man page. 2001. Linux man page, "tc". https://man7.org/linux/man-pages/man8/tc.8.html.
[25] NIMBUSDATA. 2022. World's Highest Capacity and Most Efficient SSDs. https://nimbusdata.com/products/exadrive/.
[26] OPENSIGNAL. 2022. Benchmarking the Global 5G Experience. https://www.opensignal.com/2022/06/22/benchmarking-the-global-5g-experience-june-2022.
[27] Zhouyue Pi, Junil Choi, and Robert Heath. 2016. Millimeter-wave gigabit broadband evolution toward 5G: Fixed access and backhaul. *IEEE Communications Magazine* 54, 4 (2016), 138–144.
[28] Vijay Janapa Reddi, Hongil Yoon, and Allan Knies. 2018. Two Billion Devices and Counting. *IEEE Micro* 38, 1 (2018), 6–21. https://doi.org/10.1109/MM.2018.011441560
[29] Sam Son, Seung Yul Lee, Yunho Jin, Jonghyun Bae, Jinkyu Jeong, Tae Jun Ham, Jae W Lee, and Hongil Yoon. 2021. {ASAP}: Fast Mobile Application Switch via Adaptive Prepaging. In *2021 USENIX Annual Technical Conference (USENIX ATC 21).* 365–380.
[30] Statista. 2022. Number of smartphone subscriptions worldwide from 2016 to 2021. https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/.
[31] Stephanie Chan. 2021. The iPhone's Top Apps Are Nearly 4x Larger Than Five Years Ago. https://sensortower.com/blog/ios-app-size-growth-2021.
[32] Wouter Verhelst. 2022. Network Block Device. https://github.com/NetworkBlockDevice/nbd.
[33] Mengwei Xu, Zhe Fu, Xiao Ma, Li Zhang, Yanan Li, Feng Qian, Shangguang Wang, Ke Li, Jingyu Yang, and Xuanzhe Liu. 2021. From cloud to edge: a first look at public edge platforms. In *Proceedings of the 21st ACM Internet Measurement Conference.* 37–53.
[34] Roger Ye. 2017. *Android System Programming.* Packt Publishing Ltd.
[35] Xiao Zhu, Duo Liu, Kan Zhong, Jinting Ren, and Tao Li. 2017. SmartSwap: High-performance and user experience friendly swapping in mobile systems. In *Proceedings of the 54th Annual Design Automation Conference 2017.* 1–6.