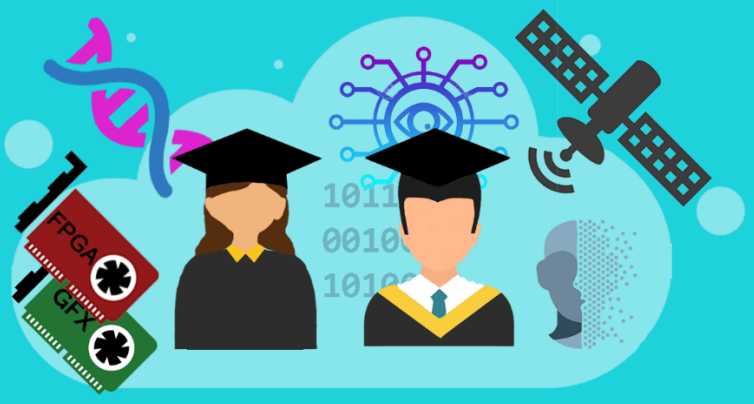
Diploma Thesis

Microproccessors and Digital Systems Laboratory



Memory tiering for Cloud-native services

Modern cloud platforms are starting to use **heterogeneous, multi-tier memory** instead of only DRAM. In practice, this means combining fast but expensive DRAM with slower, cheaper options such as NVM/PMem or CXL-attached far memory. By pooling and organizing these different types of memory, operators can reduce **fragmentation and memory stranding** (wasted memory that cannot easily be reused) while still meeting application latency requirements, as shown in Pond's study of CXL-based memory pools [1]. However, using remote or disaggregated memory comes with clear **trade-offs**: local DRAM has very low access latency (tens of nanoseconds), whereas remote or pooled memory can be much slower, even if it is cheaper and easier to share across applications [2].

To handle these trade-offs, several systems use **dynamic memory management**, where the placement of data in fast or slow memory changes at runtime based on how it is used. Memtis [3] tracks how frequently pages are accessed ("hotness") and moves them between fast and slow tiers, but it mainly tries to keep the fast tier as full as possible instead of exploring different splits between local and remote memory. In the serverless setting, FaaSMem [4] introduces a shared memory pool and moves cold pages from idle functions to a remote pool, reducing local memory usage while keeping performance acceptable. Overall, these systems show that dynamic offloading works in practice and can even limit the use of fast memory, but they do not give cloud operators an easy way to choose among multiple local/remote configurations (for example different latency vs. memory-footprint trade-offs), nor do they provide a clear mechanism to say "this service may use at most X% of its memory in local DRAM" at the container or function level.

In this thesis, the student will work with **DAMON** [5] and its **DAMOS** policies as the core tools for access-aware multi-tier memory management. The thesis will first introduce how DAMON monitors memory access patterns and how DAMOS can apply actions (promotion, demotion, reclamation), focusing especially on the newer **per-cgroup policies**, which allow control at the container level. Building on this, or on a simpler and faster custom alternative if needed, the main goal is to implement **serverless-like provisioning** of local and remote memory, where each function gets a target split between fast local DRAM and slower remote memory. The student will experimentally study how different applications react to various local/remote ratios, always keeping the hottest pages in local memory, and will use the available **Intel PMem** to emulate remote memory. The choice of local/remote ratio for each workload will take into account (i) the application's latency requirements, (ii) how much local and remote memory is free, and (iii) the current memory bandwidth, with the overall aim of balancing performance (QoS), cost, and reduced memory stranding in a cloud-native environment.

References:

[1] Li, Huaicheng, et al. "Pond: Cxl-based memory pooling systems for cloud platforms." Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2. 2023.

[2] Masouros, Dimosthenis, et al. "Adrias: Interference-aware memory orchestration for disaggregated cloud infrastructures." 2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA). IEEE, 2023.

[3] Lee, Taehyung, et al. "Memtis: Efficient memory tiering with dynamic page classification and page size determination." Proceedings of the 29th Symposium on Operating Systems Principles. 2023.

[4] Xu, Chuhao, et al. "Faasmem: Improving memory efficiency of serverless computing with memory pool architecture." Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3. 2024.

[5] DAMON: https://docs.kernel.org/mm/damon/index.html

Contact:

Achilleas Tzenetopoulos, Ph.D. candidate NTUA: (atzenetopoulos@microlab.ntua.gr)
Dimosthenis Masouros, Post Doc NTUA
Sotirios Xydis, Ass. Prof., NTUA
Dimitrios Soudris, Professor NTUA