

# Optimizing Memory-mapped I/O for Fast Storage

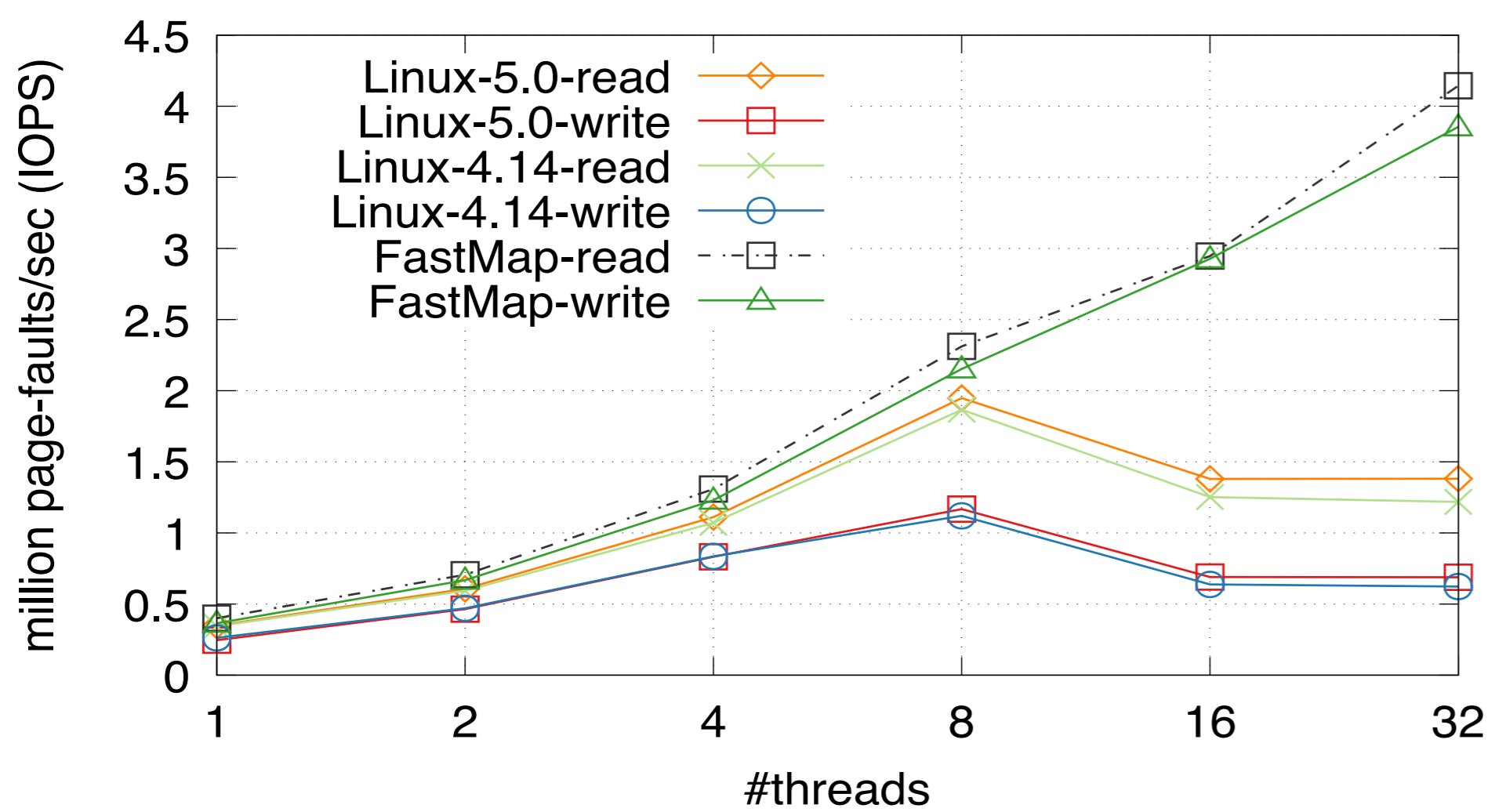
Anastasios Papagiannis, Giorgos Xanthakis, Giorgos Saloustros, Manolis Marazakis, and Angelos Bilas

Institute of Computer Science (ICS), Foundation of Research and Technology – Hellas (FORTH), Greece  
Computer Science Department, University of Crete, Greece



## MOTIVATION

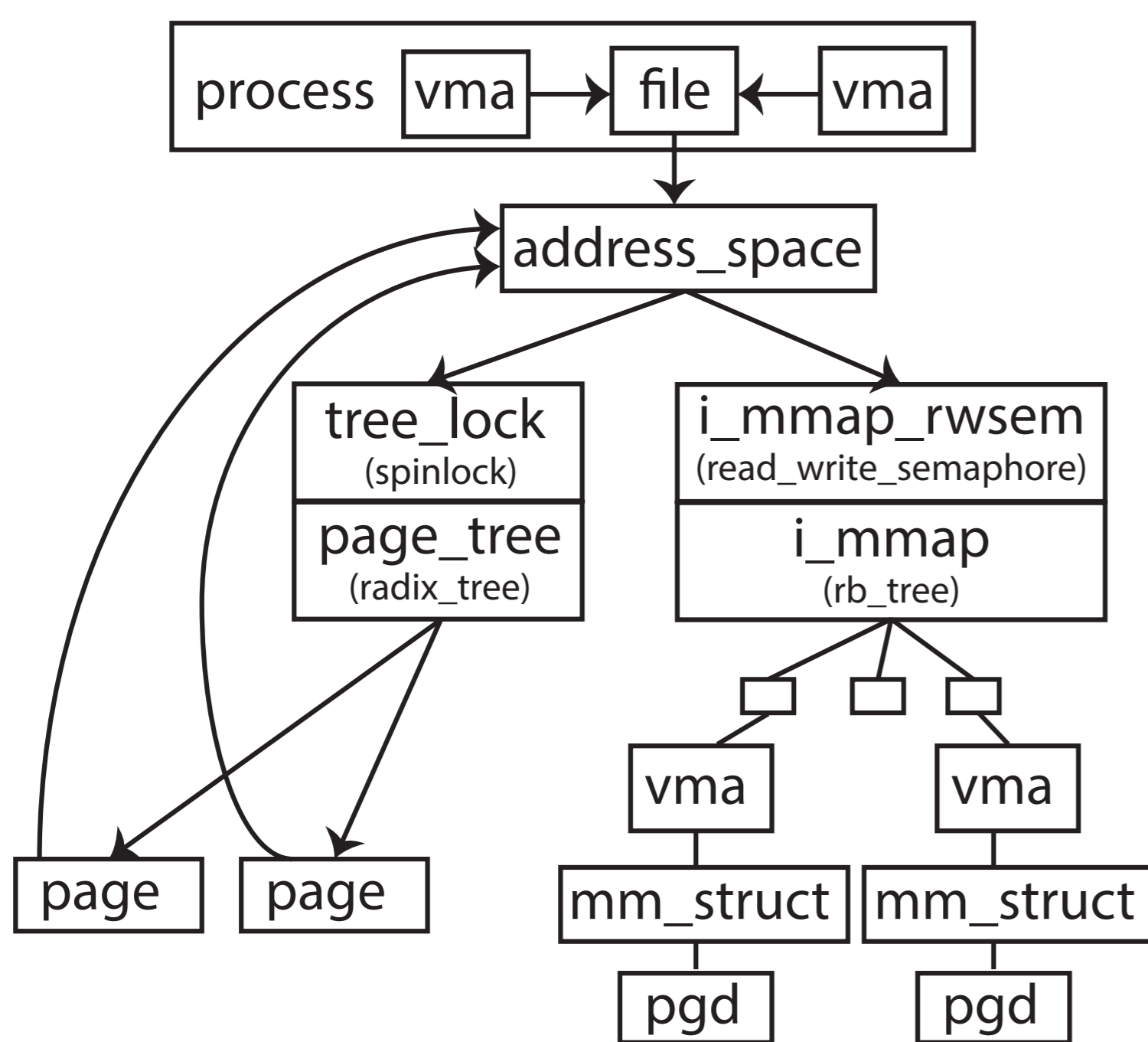
- Use Memory-mapped I/O instead of Read/Write API for Fast Storage Devices
- Removes the need of system-calls
- Removes the need of data copying between user and kernel
- Extend the virtual address space beyond the physical memory size over a fast storage device
- Lots of random memory accesses
- But:



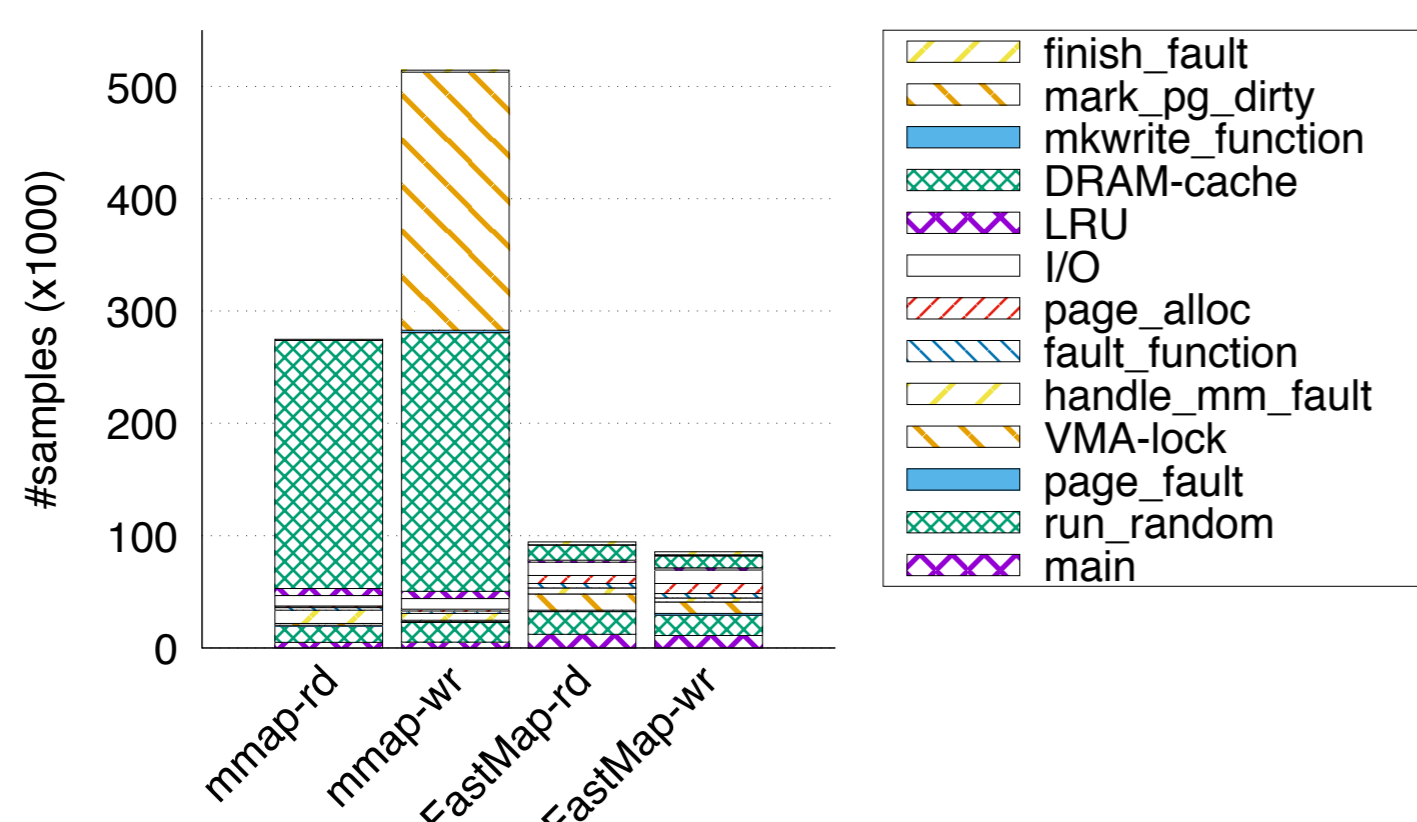
- Which also results in limited concurrency to the devices

## LINUX MEMORY-MAPPED I/O STACK

- Contention in `tree_lock`
- `i_mmap_rwsem` limits concurrency even as read-lock

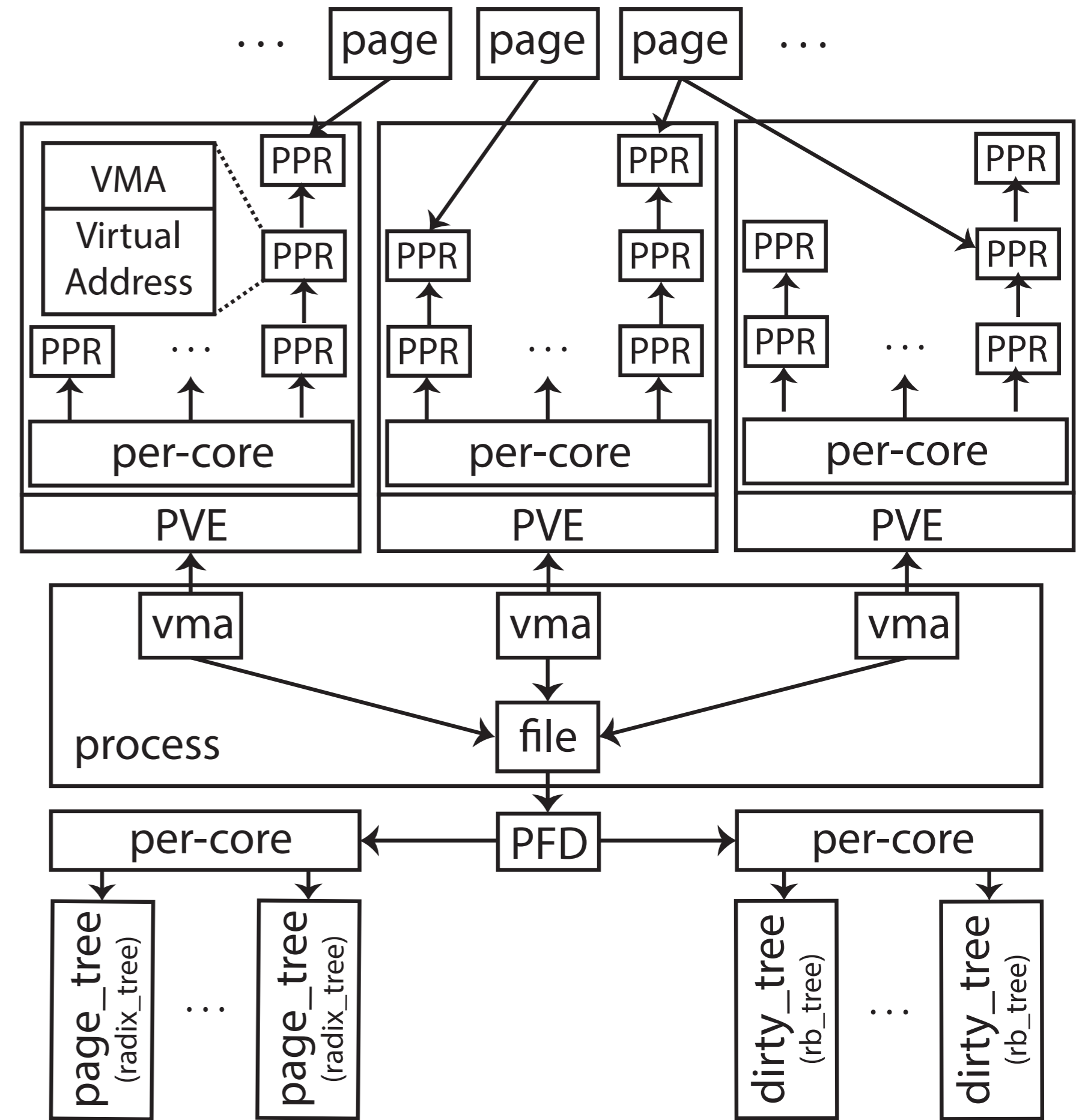


## EXECUTION TIME BREAKDOWN - MICROBENCHMARK



## FASTMAP DESIGN

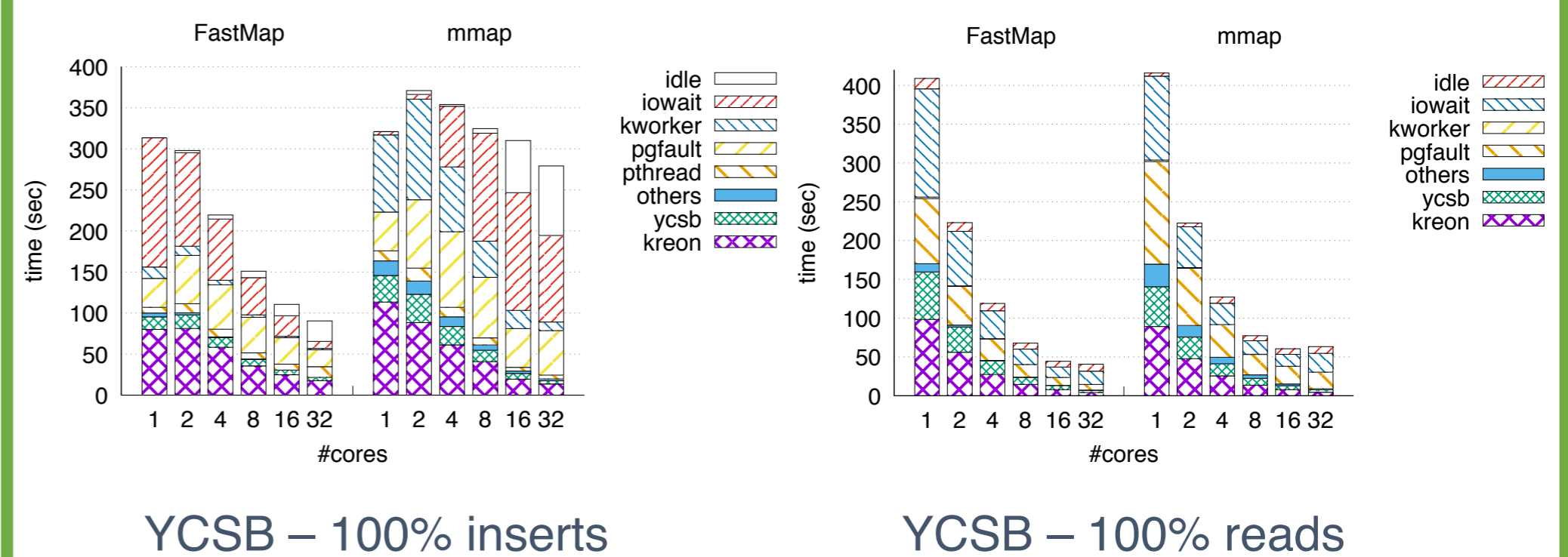
- FastMap provides a scalable design for Memory-Mapped I/O
- ... which results also in higher concurrency to devices
- Scalable (per-core and/or sharded) data structures
- Full reverse mappings



- PFD: Per-File-Data
- PVE: Per-Vma-Entry
- PPR: Per-Pve-Rmap

## EXECUTION TIME BREAKDOWN - KREON

- Kreon is a persistent key-value store designed over mmap()



## REFERENCES

1. Anastasios Papagiannis, Giorgos Saloustros, Pilar González-Férez, and Angelos Bilas. 2018. An Efficient Memory-Mapped Key-Value Store for Flash Storage. In Proceedings of the ACM Symposium on Cloud Computing (SoCC '18)

## ACKNOWLEDGEMENTS

We thankfully acknowledge the support of the European Commission under the Horizon 2020 Framework Programme for Research and Innovation through the ExaNeSt (GA 671553) and EVOLVE (GA 825061) projects and the General Secretariat of Research and Technology in Greece through project Sentitour at Scale (T1EDK-02857).