

A User-level Toolkit for Storage I/O Isolation on Multitenant Hosts

Giorgos Kappes, Stergios V. Anastasiadis – University of Ioannina, Ioannina 45110, Greece

Multitenancy with Containers

Software Containers

- Run in multitenant hosts
- Host data-intensive applications
- Achieve bare-metal performance & flexibility

Host OS Kernel

- Serves the containers of different tenants
- Mounts container root & application filesystems
- Handles I/O to local & network storage devices

Limitations of sharing the host kernel

- **Unfair use of resources**
- **Global configuration rather than per tenant**
- **Software overheads**
- **Slow software development**
- **Large attack surface**

The Polytropon Toolkit

Idea

- **Per tenant** user-level filesystems
- **Container Pool**: Tenant containers per machine

Goals

- **Compatibility**: POSIX-like interface
- **Isolation**: Per-tenant I/O paths
- **Flexibility**: Per-tenant configuration & features
- **Efficiency**: Lightweight on resources

Filesystem Library

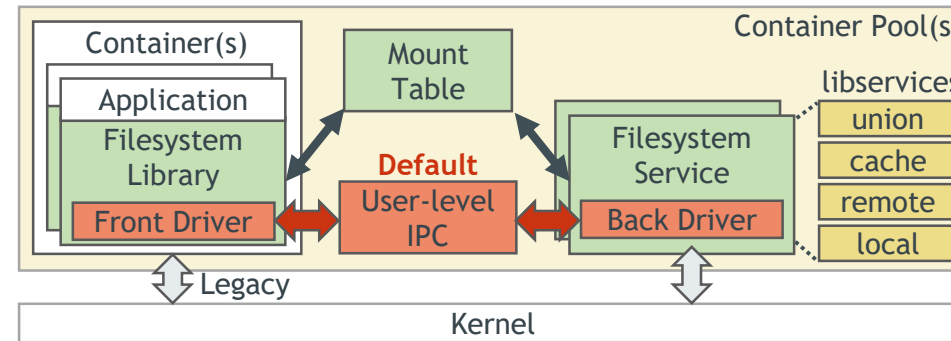
Filesystem access to processes

- Preloaded
- POSIX-like interface for process management, memory mappings, library functions, asynchronous I/O

Filesystem Service

Handles container I/O in a pool

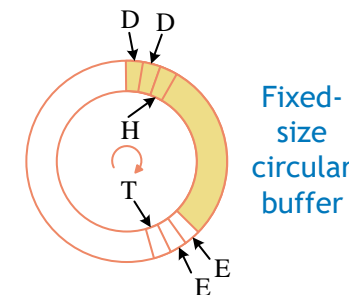
- Collections of libservices
- **libservice**: standalone user-level storage functionality implemented as library



Interprocess Communication

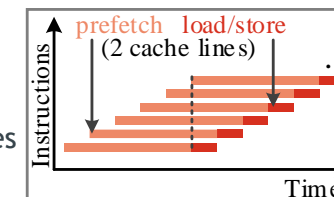
Request Concurrent Queue Blocking (RCQB)

- **Optimized queue for request communication**
- 1st Stage: Distributes operations sequentially
- 2nd Stage: Let them complete in parallel potentially out of FIFO order



Shared Memory Optimized (SMO) Copy

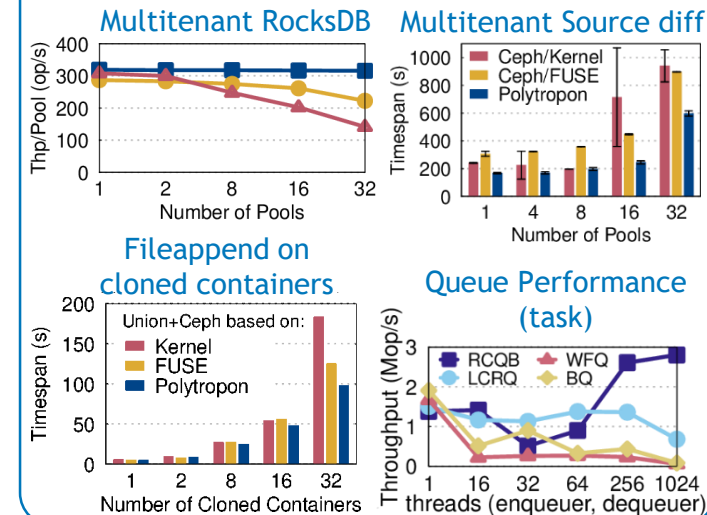
- **Optimized copy algorithm for data transfer**
- Copy: Source -> Shared Memory -> Destination
- 1st Stage: Non-temporal prefetch of 2 cache lines
- 2nd Stage: Non-temporal store of 2 cache lines



Results

Summary

- Polytropon achieves faster I/O response & more stable performance
- RCQB benefits from parallel completion of enqueue and dequeue operations



Conclusions

Problem

- **I/O contention on shared host kernel** limits performance of containerized applications

Solution: The Polytropon toolkit

- User-level components for filesystem provisioning on multitenant hosts
- **Tenant I/O isolation**
- **Scalable and stable performance**