The CASE of FEMU: Cheap, Accurate, Scalable and Extensible Flash Emulator

Huaicheng Li, Mingzhe Hao, Michael Hao Tong,
Swaminathan Sundararaman, Matias Bjørling, Haryadi S. Gunawi

THE UNIVERSITY OF CHICAGO

ucare.cs.uchicago.edu

CERES Center for Unstoppable Computing

ceres.cs.uchicago.edu

Why a new SSD Emulator?

Platform	Pros	Cons
Simulator	Cheap; Easy; Time-saving	Trace-driven; Internal research only
Emulator	Cheap; Full-stack research support	Poor scalability; Poor accuracy
Hardware	Full-stack research support; Accurate	Expensive; Complex; Wear-out

Why Emulator?

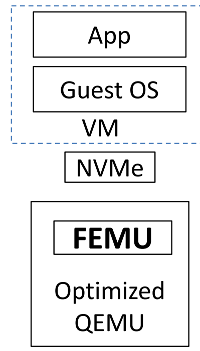
- Get the benefits of both simulators and hardware platforms
- Enable wide range of SSD research, including SDF/Split-level Architecture and host-SSD co-designs, etc.

Why FEMU?

Bleak Status of Existing SSD Emulators

- FlashEmu: no longer maintained
- VSSIM: non-scalable; inaccurate$_{\text{app-level}}$
- LightNVM's QEMU: only single-channel support; non-scalable

FEMU Use Case

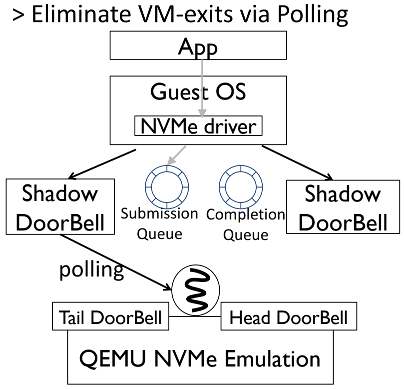


Supported Research:

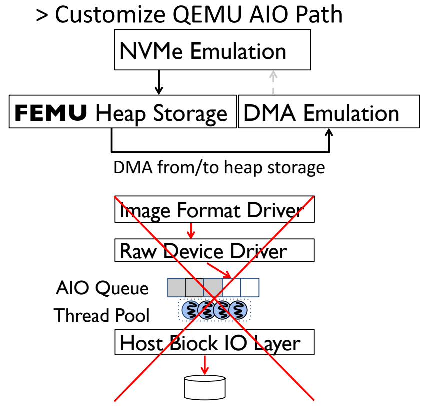
- Kernel changes ✓
- Interface changes ✓
- SSD FTL changes ✓

FEMU Scalability: QEMU Virtual IO Optimization

> Eliminate VM-exits via Polling



> Customize QEMU AIO Path



Why bother optimizing QEMU?

A high performance base environment is needed to:

- Emulate NAND operations at ~100us level
- Emulate tens of parallel NAND flash chips

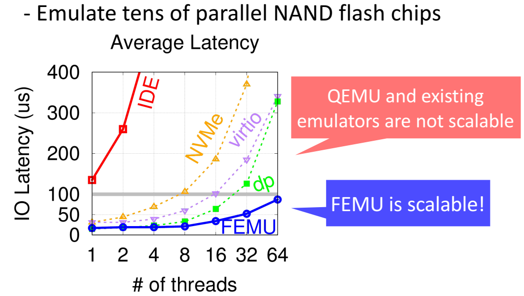


Figure 1: FEMU Scalability

FEMU Accuracy

Delay Emulation:

- Endio queue: requests sorted according to completion time
- Periodic polling for request completion time expiration

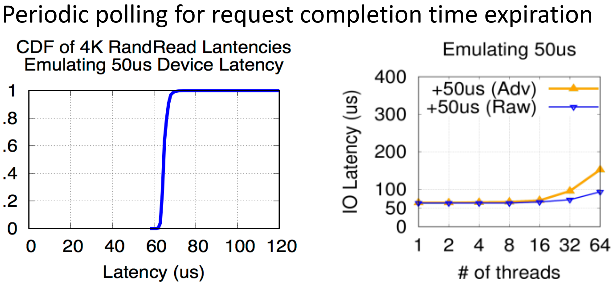


Figure 2: FEMU Accuracy on emulating 50us Device Latency

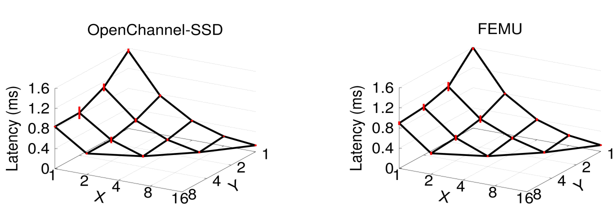
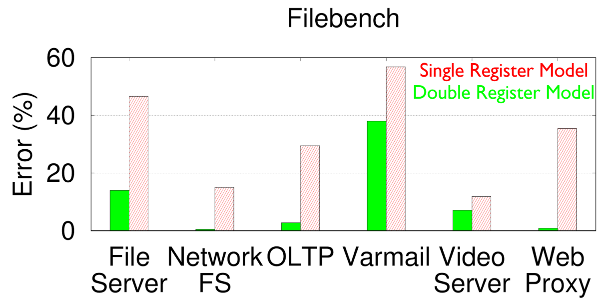


Figure 3: FEMU Accuracy on emulating OpenChannel-SSD



Latency Error: 11-47% ⇒ 0.5-38%

Figure 4: FEMU v.s. OpenChannel-SSD on Filebench

Future Work

- ❑ Further QEMU optimizations to support more scalability
- ❑ Improve accuracy by integrating more detailed SSD information
- ❑ Integrate well-implemented FTLs in popular SSD Simulators
- ❑ Multi-core support

Appears in the 16th USENIX Conference on File and Storage Technologies (FAST), 2018