Mercury: host-side flash caching for the datacenter

NetApp™

Steve Byan, James Lentini, Luis Pabón, Christopher Small, Mark W. Storer

Shared-pool datacenter + Integrated flash memory

- No persistent state
- VMs not tied to servers, can be dynamically added, moved
- Servers can be added, upgraded, repurposed on-the-fly
- Unified central storage management
- Shared pool scales more easily; resources can be reassigned on-the-fly
- Data can be moved to most appropriate media

- 10s-100s GB of flash being integrated into servers
- New price/perf tier between disk and DRAM
- Flash is 10-100x faster than disk, ¼ price of DRAM
- High IO-per-second (IOPS) storage close to CPU

…but using integrated flash for primary storage breaks the shared-pool datacenter model

- *Binds software services to specific servers*
- *Puts flash primary storage out of reach of storage management tools*

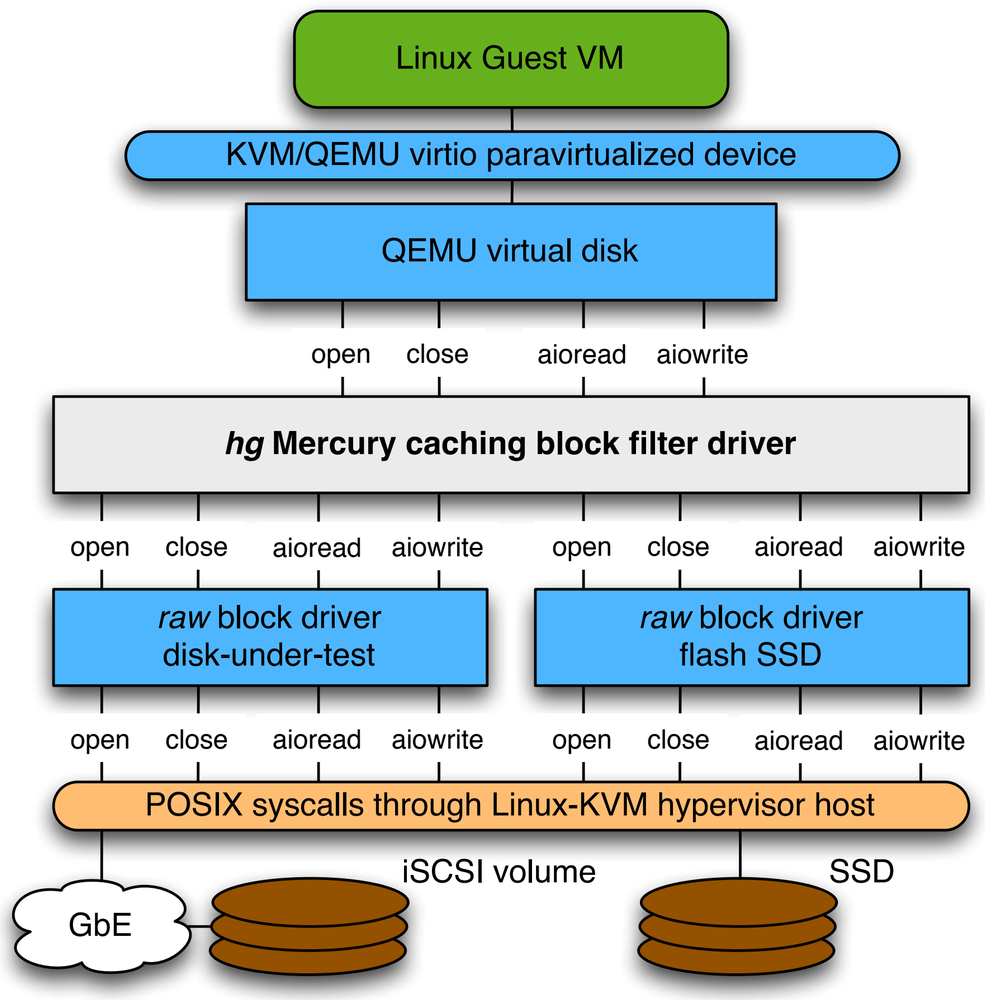
Mercury flash cache implementation

Mercury portable flash cache

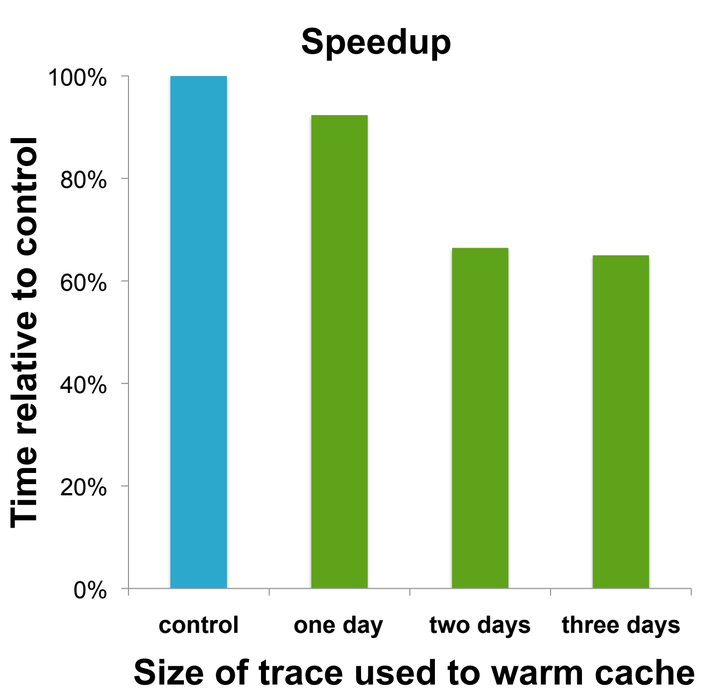
- Uses integrated flash or other local storage as a cache for centrally-managed shared storage
- Implements a block-oriented cache; write-through to maintain coherence with backing store
- Deployable as
  - Hypervisor filter driver, transparent to guest OS
  - OS filter driver, transparent to applications
  - Application cache
  - Proxy cache for a network storage protocol

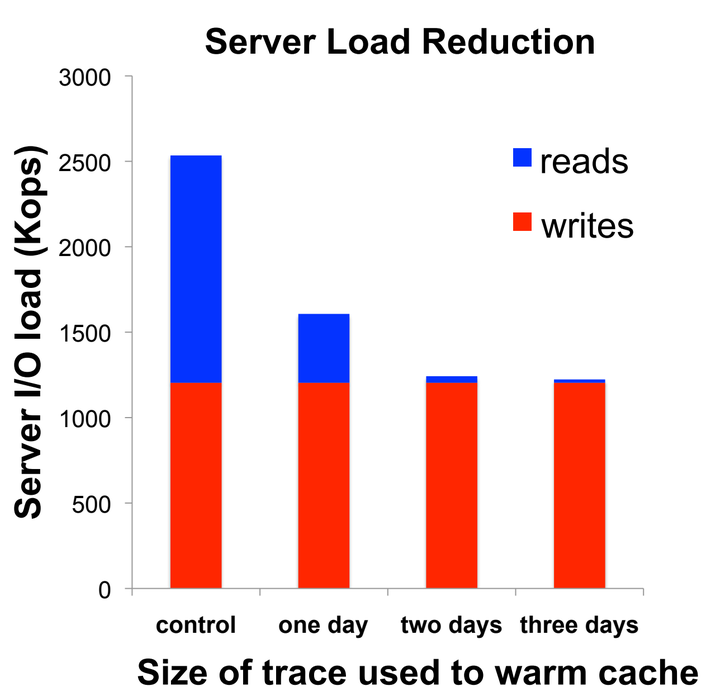
Prototype deployment

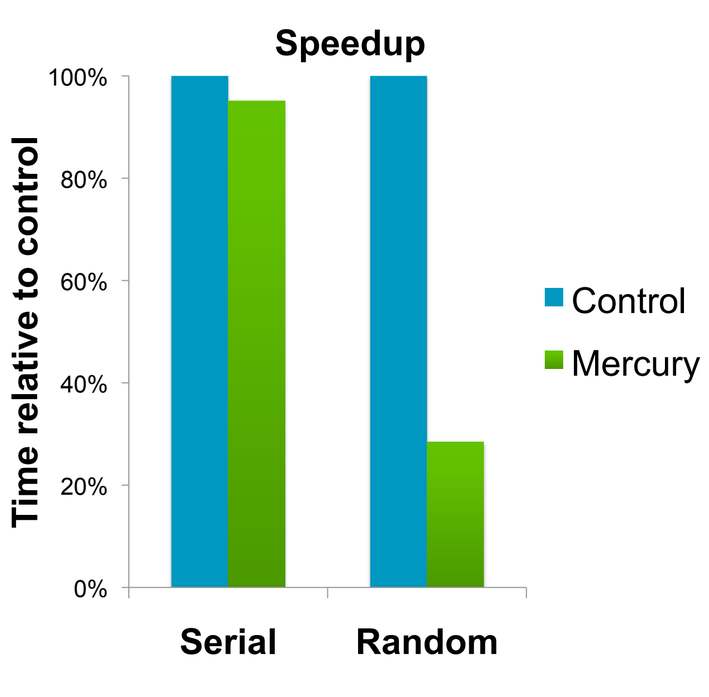
- KVM/QEMU block driver, loads into stock QEMU
- Provides new disk format, *hg*
- Requests sent to *hg* device handed to SSD cache or passed to *raw* backing device



Performance results







Desktop traces

- Replayed disk-level traces from Windows XP desktop
- Cache warmed with one to three days of traces
- All tests run on same trace
- Nearly 40% reduction in mean I/O service time
- Near 50% reduction of requests sent to server (almost all reads handled by Mercury)

iozone

- iozone run directly against server iSCSI volume and via Mercury cache
- Serial I/O showed small improvement
- Random I/O had substantial speedup (almost all reads handled by Mercury)