Recent Developments in the Linux I/O Stack

Martin K. Petersen
Oracle
Linux I/O Development Activity

![Linux I/O Development Activity Chart](chart.png)
Linux I/O Development Activity
Linux I/O Development Activity

blk-mq
Multiqueue Block Layer

- Legacy I/O submission path is single-threaded
- Major rework of the block I/O infrastructure to accommodate devices with multiple submission queues such as NVMe
- Lockless submission path and better scalability
- NVMe and SCSI are the two main users
Multiqueue Block Layer

- Legacy I/O path developed for spinning media
- I/O schedulers for fairness and coalescing
- High latency due to seek reduction
- blk-mq aims at low-latency devices
- However, some mq devices and workloads benefit from I/O scheduling
Another rewrite?
Multiqueue Block Layer Enhancements

- Preparation to remove legacy I/O path
- blk-mq now has I/O scheduling capability:
  - Kyber
  - Budget Fair Queueing
- Polling
- Opal/SED
Linux I/O Development Activity

- SCSI Core
- NVMe
- Block

Block Layer I/O Abstractions

- Not just reads, writes, and passthrough
- *Flush* operation for consistency
- *Discard* for deprovisioning block ranges
- *Write Zeroes* for clearing block ranges
- Persistent Reservations
- *Copy In* and *Copy Out* in pipeline
Block Layer I/O Abstractions

- Hinting
  - Data lifetime
  - Realtime and Background operations
- Streams & File IDs
  - Data Affinity
- Key-Value vs. General Purpose
Zoned Block Devices

- SMR drives, zones are append-only
- Challenging for existing applications and file systems
- dm-zoned
- Legacy I/O path only, MQ support in pipeline
- Key-Value vs. General Purpose
Linux I/O Development Activity

- Mark I driver
- Mark II driver

Year:
- 2013
- 2014
- 2015
- 2016
- 2017

Graph shows the increase in Linux I/O development activity from 2013 to 2017, with a notable rise in activity post-2015.
3rd iteration of the Linux NVMe driver
Mainly done to facilitate NVMe over Fabrics RDMA transport binding
Fibre Channel transport binding merged
TCP transport binding in the pipeline
NVM Express

- 1.2/1.3 features
- Power Management
- Device Quirks
- Persistent Reservations
- Fabrics NVMe target support
- Multipathing support in the pipeline
Linux I/O Development Activity

PMem
Persistent Memory

- Persistent Memory
- Block accesses vs. byte-addressable memory
- Device DAX vs. Filesystem DAX
- Combining fast flushes with benefits of file management
Linux I/O Development Activity

Block Storage Commits
Thank You!