Towards an Unwritten Contract of Intel Optane SSD

Kan Wu, Andrea Arpaci-Dusseau, and Remzi Arpaci-Dusseau
Outline

Background & Motivation

An Unwritten Contract of Intel Optane SSD

Implications from the Contract

Discussion
New Non-volatile Memory technologies provide unprecedented performance for persistent storage.
Background: Intel Optane SSD

The most cost-effective and widely available option
Motivation

Intel Optane SSD

How to use it effectively?
How to use a device effectively?

The **Written Contract**

- HDD: (Steven et al.)
  “Sequential accesses are the best, much better than non-sequential.”
- SSD: (Jun et al.)
  - Large Request Scale
  - Locality
  - Grouping by Death Time
  - ...

The **Unwritten Contract**

Intel Optane SSD
An Unwritten Contract of Intel Optane SSD
An Unwritten Contract of Intel Optane SSD

Immediate performance: (6)
- Access with Low Request Scale Rule
- Random Access is OK Rule
- Avoid Crowded Accesses Rule
- Control Overall Load Rule
- Avoid Tiny Accesses Rule
- Issue 4KB Aligned Requests Rule

Sustainable performance: (1)
- Forget Garbage Collection Rule
An Unwritten Contract of Intel Optane SSD

Rule 1: Access with Low Request Scale
Rule 4: Control Overall Load

Storage Hierarchy

Expected

DRAM Persistent Memory
Intel Optane SSD
Flash SSD

Real

DRAM Persistent Memory
Intel Optane SSD
Flash SSD
Rule 1: Access with Low Request Scale

Motivation:
- 3D XPoint Memory > NAND Flash (up to x1000 lower latency[2])
- Does Optane SSD always perform better than Flash SSD?

What is the rule?
- “To obtain low latency, Optane SSD users should issue small requests and maintain a small number of outstanding IOs”

Note: > stands for “is better than”
Rule 1: Access with Low Request Scale

Optane SSD vs. Samsung 970 Pro:

→ What we do:
  – Random read-only / write-only workloads
  – Each workload has two variables: Request Size and Queue Depth
Rule 1: Access with Low Request Scale

Optane SSD vs. Samsung 970 Pro:

What we observe:
- Similar Write Results (in paper)
- Optane SSD > / = / < Flash SSD

$$|T| = \frac{L_{\text{higher}} - L_{\text{lower}}}{L_{\text{lower}}}$$

$T > 0$ when Optane has smaller latency
$T < 0$ when Flash has smaller latency

Avg Latency of random workloads, Optane vs. Flash
Rule 1: Access with Low Request Scale

Uncover the internals of the Optane SSD

→ Internal parallelism dictates its behavior when serving workloads with high request scale

- Optane SSD: RAID-like organization of memory dies
- The interleaving degree (#channels)

RAID-like Architecture in Optane SSD
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

- What we do: (Feng et al. (HPCA 11), Timothy et al. (ASPLOS 04))
- Precondition: sequential writes => evenly distribute
- 4KB (chunk) read stream with stride $S$ ($S = \text{distance between consecutive chunks}$)

Different $S$ => Different throughput

Chunk Layout
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

→ What we do:
   - Precondition: sequential writes
   - 4KB (chunk) read stream with stride $S$ ($S = \text{distance between consecutive chunks}$)

S = 0 (chunk), QD = 4
Performance 😊

Chunk Layout
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

- What we do:
  - Precondition: sequential writes
  - 4KB (chunk) read stream with stride $S$ ($S = \text{distance between consecutive chunks}$)

\[ S = 1 \text{ (chunk)}, \quad QD = 4 \]

Performance 😞
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

- What we do:
  - Precondition: sequential writes
  - 4KB (chunk) read stream with stride S (S = distance between consecutive chunks)

S = 3 (chunk), QD = 4
Performance

Chunk Layout
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

What we observe:

Intuition:
Distance between the lowest dips in each line => the interleaving degree

Optane SSD

Flash SSD

HotStorage ’19
Rule 1: Access with Low Request Scale

Detecting Interleaving Degree of Optane SSD:

→ What we observe:

→ Internal parallelism: Optane SSD (7) << Flash SSD (128)
→ Explains Optane SSD’s worse behavior serving workloads with high request scale

![Graph showing throughput comparison between Optane SSD and Flash SSD with different QD values and stride values.](image)
Rule 4: Control Overall Load

Motivation:

→ Optane SSD facing mixed (read and write) workloads?

What is the rule?

→ Distinctive from Flash SSD!
→ “To achieve optimal latency from Optane SSD, the client must control the overall load of both reads and writes.”
Rule 4: Control Overall Load

Experiments: Optane SSD serving mixed workloads

- What we do?
  - Random 4KB requests (reads + writes, QD=64), varying write%
Rule 4: Control Overall Load

Experiments: Optane SSD serving mixed workloads

→ What we observe?

→ Optane SSD (throughput yield similar results)
  
  Reads = Writes;

Latency is related to the overall load, not to write%
Rule 4: Control Overall Load

Experiments: Optane SSD serving mixed workloads

What we observe?

- Optane SSD vs. Flash SSD: distinctive behavior

![Graph showing latency comparison between Optane Read Latency, Optane Write Latency, Flash Read Latency, and Flash Write Latency. The graph illustrates a 5x difference in latency between read and write operations for Optane SSD compared to Flash SSD.]
An Unwritten Contract of Intel Optane SSD

Rule 1: Access with Low Request Scale
Rule 4: Control Overall Load

Expected

Storage Hierarchy

Real

**DRAM Persistent Memory**

**Intel Optane SSD**

**Flash SSD**

1. Low request scale
2. Write-dominated

1. High request scale
2. Read-dominated
Other Rules…
Rule 2: Random Access is OK

Motivation:
- Optane SSD: Random vs. Sequential?

What is the rule?
- “Optane SSD is a random access block device, where clients can observe the same performance for random and sequential workloads”
Rule 3, Rule 5, Rule 6

Motivation:
- Byte-addressability of 3D XPoint Memory
  => Efficient tiny accesses to Optane SSD?

What is the rule?
- Rule 3: Avoid Crowded Accesses (4.6x)
  - Clients of Optane SSD should never issue parallel accesses to a single chunk (4KB)
- Rule 5: Avoid Tiny Accesses (5x)
  - To exploit bandwidth of the SSD, the client must not issue requests less than 4KB.
- Rule 6: Issue 4KB Aligned Requests (1.2x)
  - To achieve the best latency, requests issued to Optane SSD should always align to eight sectors.
Rule 7: Forget Garbage Collection

Motivation:
- Optane SSD maintains MAX throughput for sustained writes
- Insights of this?
  Optane: LBA-based mapping vs. Flash: written-order based

What is the rule?
- There is no need to worry about garbage collection in Optane SSD.
An Unwritten Contract of Intel Optane SSD

Immediate performance: (6)

- Access with Low Request Scale Rule
- Random Access is OK Rule
- Avoid Crowded Accesses Rule
- Control Overall Load Rule
- Avoid Tiny Accesses Rule
- Issue 4KB Aligned Requests Rule

(Feedback)

Sustainable performance: (1)

- Forget Garbage Collection Rule

More interesting questions to answer?
Implications from the Contract

Users design systems for Optane SSD

→ Random Access is Okay.
  ● Restructuring of external data structures
    Much effort: random -> sequential accesses ; Less necessary
    E.g. Single Machine Graph Processing Systems (Nima Elyasi et al. FAST’19)
  ● Applications which behave poorly on Flash thus become potential consumers

→ No Crowded Accesses, No Tiny Access, and Alignment rule
  ● Pitfalls that fine-grained external data structure must be aware
Implications from the Contract

Users who combine Flash and Optane in a hybrid setting

- Access with Low Request Scale Rule
- Control Overall Load Rule
- Forget Garbage Collection Rule

Classic concept of hierarchy need to be reconsidered

- How to split accesses?

1. Low request scale
2. Write-dominated
3. Full device

1. High request scale
2. Read-dominated

DRAM Persistent Memory

Intel Optane SSD

Flash SSD

Real

HotStorage ’19
Conclusion

We analyze a NVM-based block device: the Intel Optane SSD
We formalize the rules that Optane SSD users should follow
Implications from this Contract

Interesting thing we can do with the contract?
Acknowledgement
Thanks!

Questions?