SNIA DEVELOPER CONFERENCE
SDC 25
By Developers FOR Developers

Hyatt Regency Santa Clara, CA
September 15-17, 2025

Deterministic, Fast, Random Preconditioning Using Sprandom

Steven Sprouse, SanDisk

SANDISK™

www.sniadeveloper.org

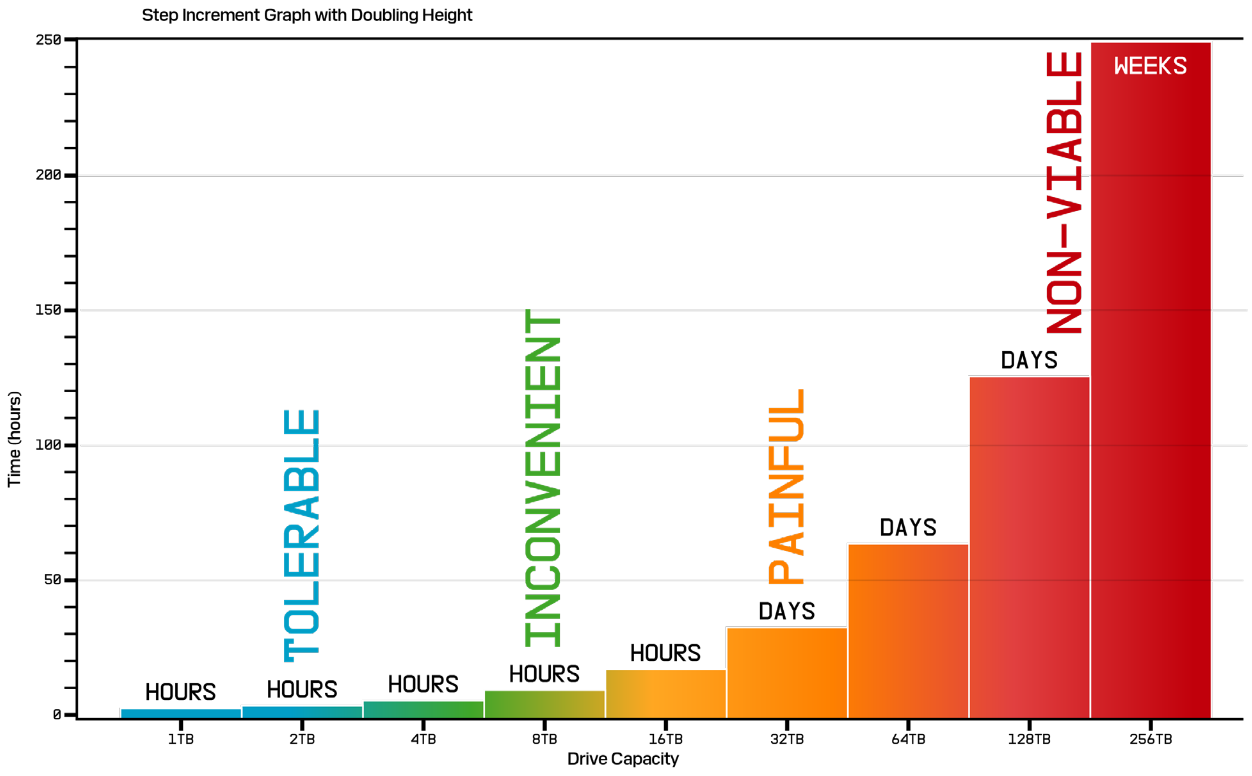
Time spent testing is more valuable than time spent preparing the test

Previous time constraints were convenient, but… :

- As SSD capacities increase beyond 16TB, the time to randomly precondition these drives has also increased from several hours to several days.

Brute force used to work, but… :

- Traditional methods involve at least one sequential write followed by multiple random writes to reach a steady state.
- A single sequential write on a 128TB drive can take 10 hours.



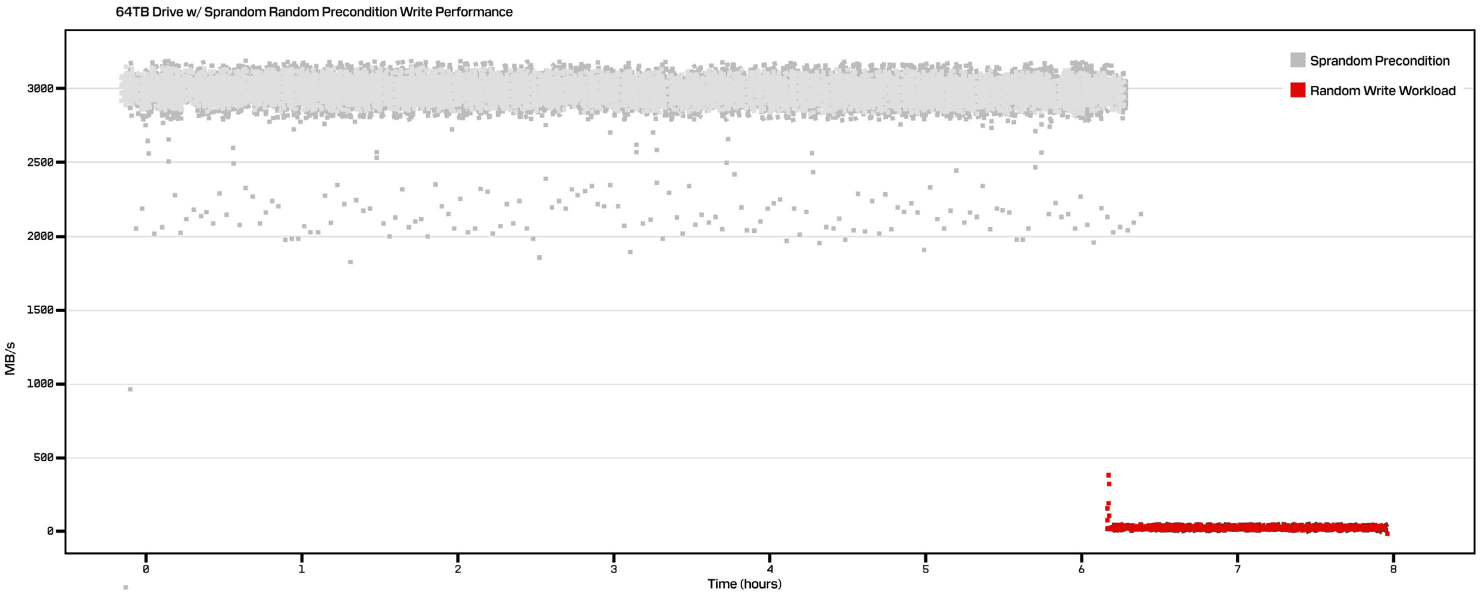
The Mechanics of Pre-Conditioning SSD's

Logically fill the drive:

- Every LBA shall be written once.
- Achieved by at least one logical drive write

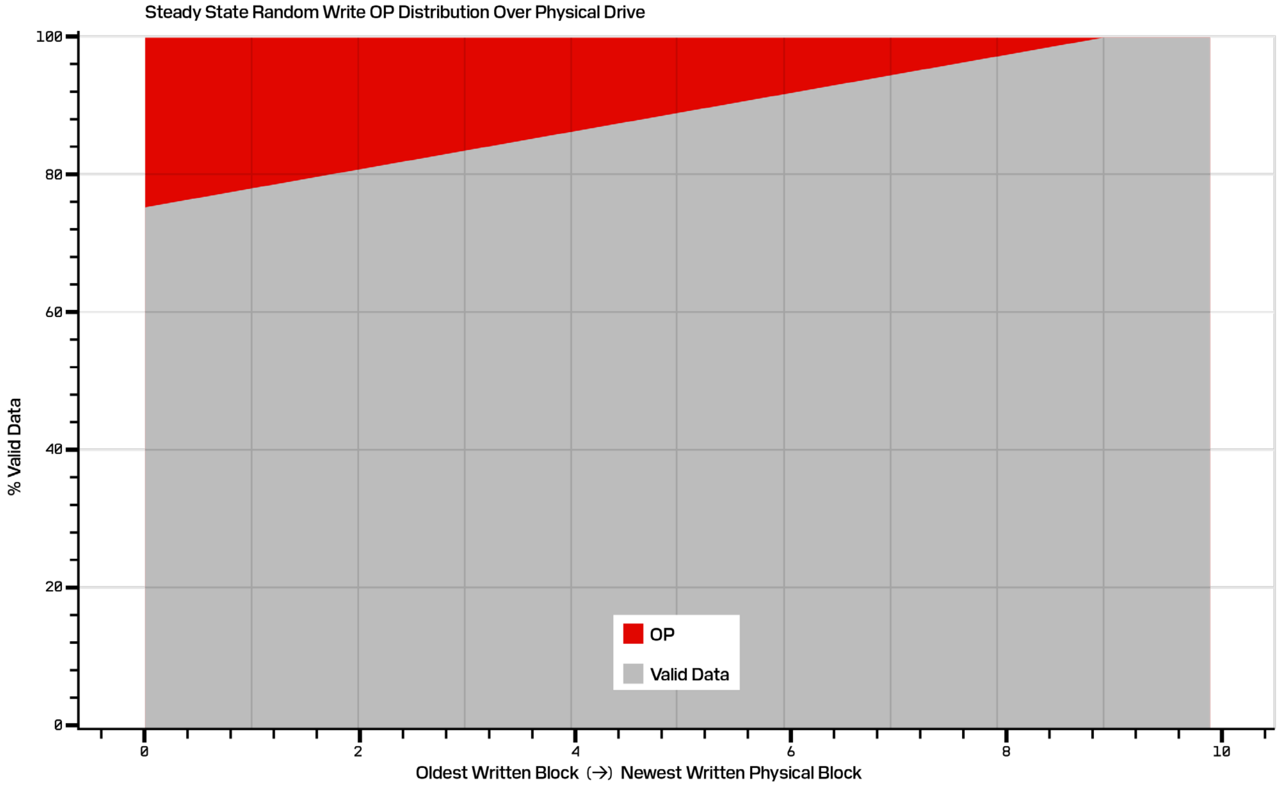
Reach a steady state throughput:

- Achieved through several random drive writes.
- Overprovisioning is distributed within the drive in a particular way.



What is the ideal "Pre-Conditioned" State?

- The figure to the right shows roughly how the OP is distributed across the physical media in an SSD when a random write workload reaches steady state.
- In this example, there is 15% OP in the drive.

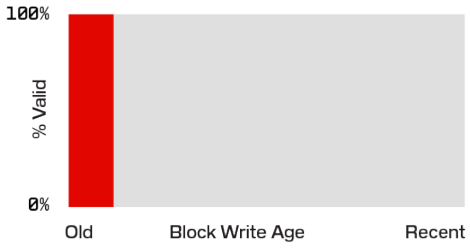


Sandisk Pseudo-Random Fast Preconditioning (SPRandom)

- **Goal: Combine the Fill and Steady State in a single pass.**
  - the Logical Drive fills were taking too long, and required too many passes the higher capacities became
  - Could we shape writes in a way that will still write every LBA at least once, while forcing the drive to go into garbage collection saving time on subsequent sequential or random writes?
- **Steady state throughput is measurable, so we can compare methodologies**
  - Test apples to apples after traditional and SPRandom pre-conditioning to ensure steady state operation
  - Report and measure overprovisioning use and compare between methodologies

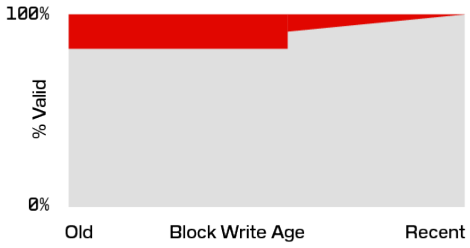
What happens during random preconditioning?

- Preconditioning redistributes overprovisioning across the NAND media



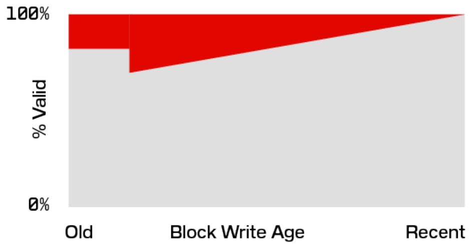
After 1 Sequential Drive Write

Fast
No GC



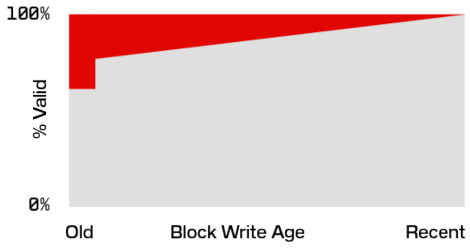
After .5 Random Drive Write

Slow
w/ GC



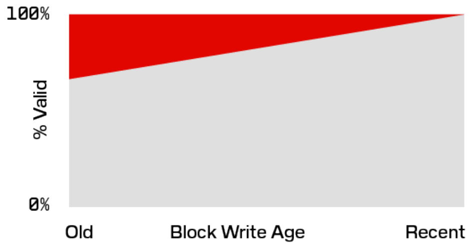
After .75 Random Drive Write

Slow
w/ GC



After 1.0 Random Drive Write

Slow
w/ GC

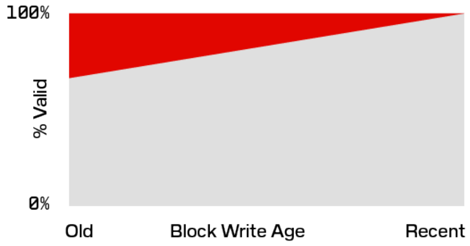


After 1.25 Random Drive Write

Slow
w/ GC

After 1 Sprandom Drive Write

Fast
No GC

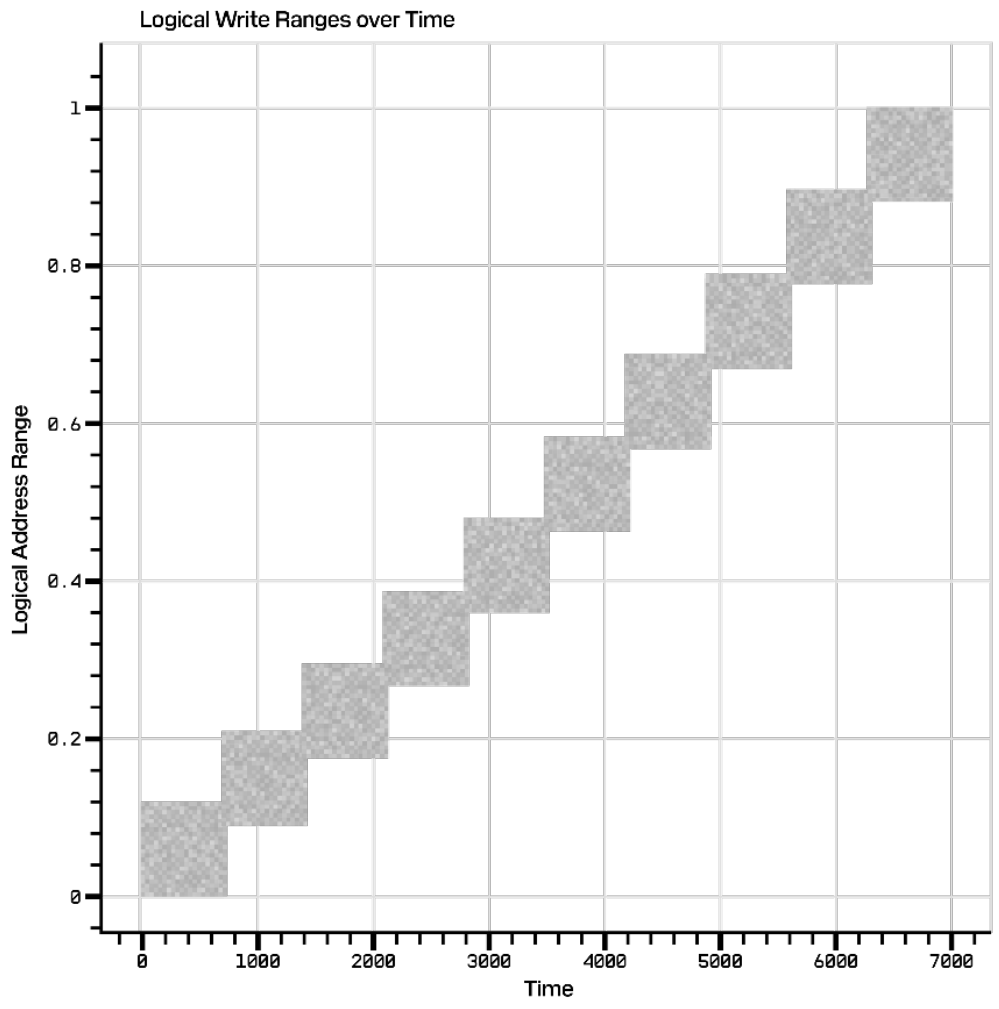


How Does Sprandomize Work?

- Sprandom starts with an empty, unmapped drive.
- Sprandomize needs to know the drive's:
  - Exported logical capacity
  - The amount of OP in the drive.
- Using the exported capacity and OP values it determines the physical capacity of the SSD.
- It segments the physical capacity into N regions, of

  `size=(physical_capacity/N)`

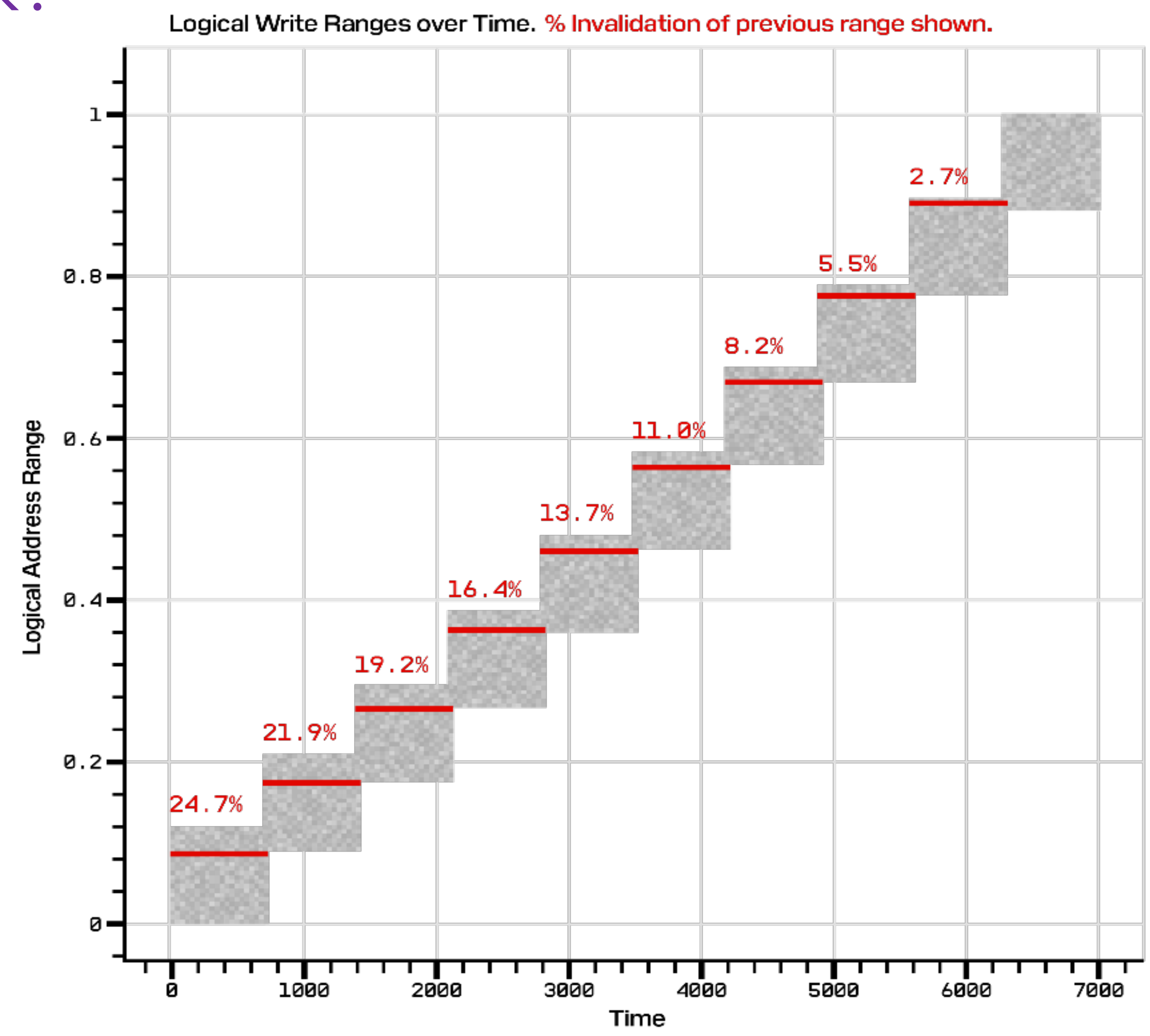
- Each of the N regions have an overlap with the previous region. The amount of overlap is determined by the amount of OP to be contained in the previous region.
- In each region, every logical address is written once and in a pseudo-random order. (Think of it as a shuffle.)



SANDISK™ SDC 25

How Does Sprandomize Work?

- This diagram shows more clearly the amount of overlap created between each region.
- For this example, the physical capacity is segmented into 10 regions, with 15% OP being distributed across the drive.
- Note that the early regions have more overlap than the last region.



SANDISK™

SDC 25

How To Calculate The OP Distribution.

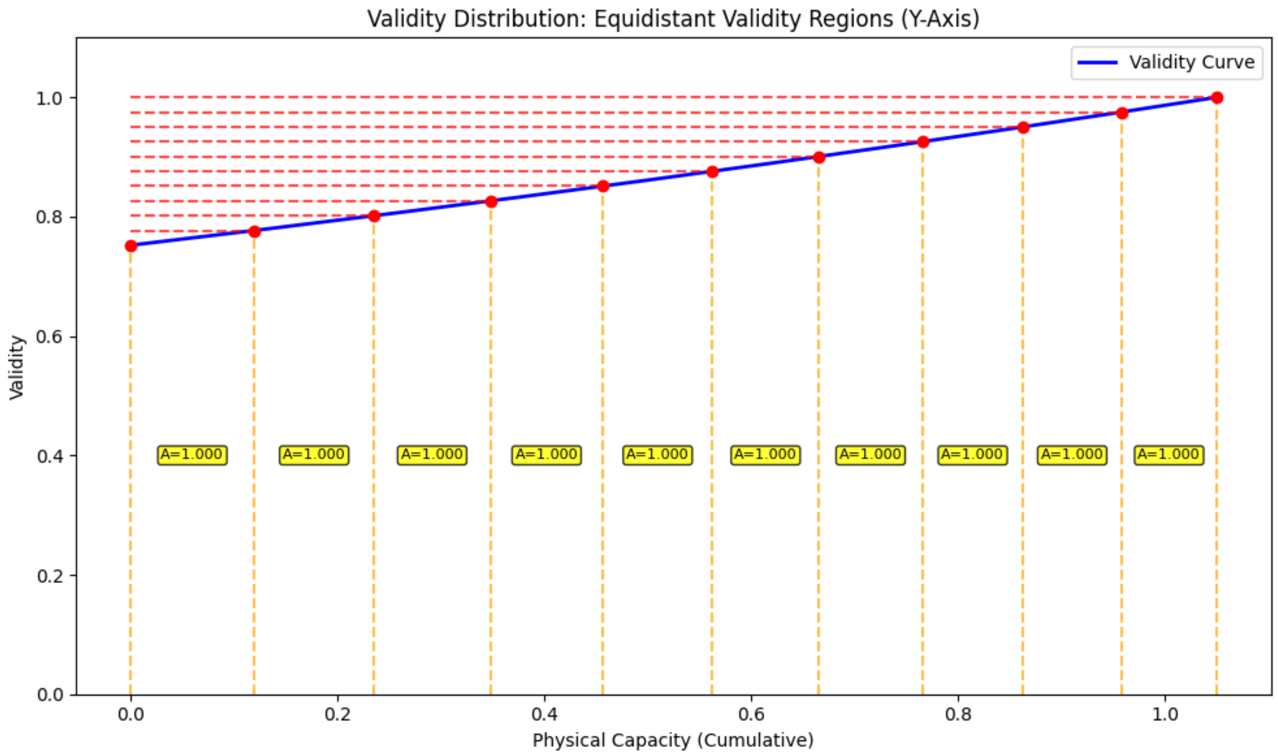
- Based on a 2012 paper : P. Desnoyers, **"Analytic Models of SSD Write Performance,"** ACM Transactions on Storage, vol. 8, no. 2, pp. 1–18, Jun. 2012, doi: 10.1145/2133360.2133364.
- We use equation (5) from this paper, $i \times f_i = k$, which describes for a given validity range, the area under the curve is a constant.
- f_i are the number of blocks with i valid pages.
- $i \times f_i = k$ states for a number of blocks with the same average validity, the product of the two numbers is a constant.
- For the left side we know the the average validity of the block selected for greedy reclaim is $gc = 1 - \frac{1}{WA}$
- Let the k = 1.

  Then $f_{gc} = \frac{1}{1 - \frac{1}{WA}}$
- Then for every other region with validity j:

$$f_j = \frac{1}{j}$$

- Equidistant ranges are selected on the Y axis.
- This gives equal area regions with varying widths on the x-axis. **Note: The area of each region is the same.**

$$i \cdot f_i = k \qquad (5)$$



Write Amp and Overprovisioning (OP)

- The Desnoyer's and another paper by Kurkowski et al.* derive an equation that describes the relationship between WA and OP. They differ slightly in how they define OP.
- At right is the equation (12) from Kurkowski's paper.
- W is the Lambert W function.
- A good approximation for this equation is:

$$WA = \frac{.5}{OP} + .7$$

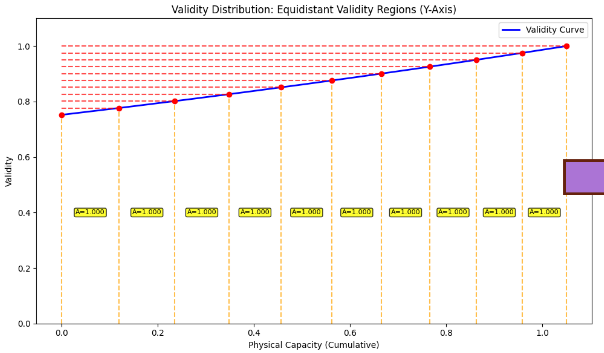
$$WA = \frac{-1-\rho}{-1-\rho-W(-1-\rho)e^{-1-\rho}}$$

$$\text{OP} = \rho = \frac{physical\_cap - logical\_cap}{logical\_cap}$$

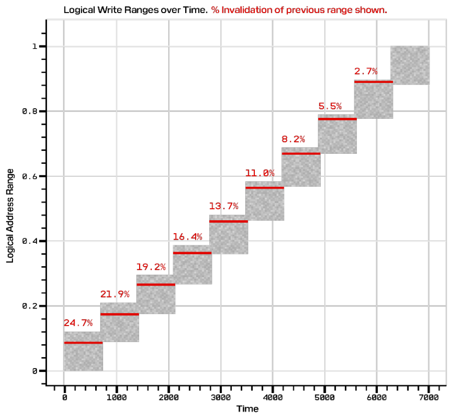
* Xiang, Luojie, and Brian M. Kurkoski. 2012. "An Improved Analytic Expression for Write Amplification in NAND Flash." In International Conference on Computing, Networking and Communications, 497–501. IEEE. https://doi.org/10.1109/ICCNC.2012.6167472.

Sprandom – Putting It All Together

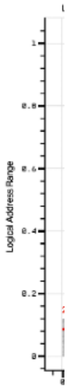
Invalidity Distribution From OP and WA



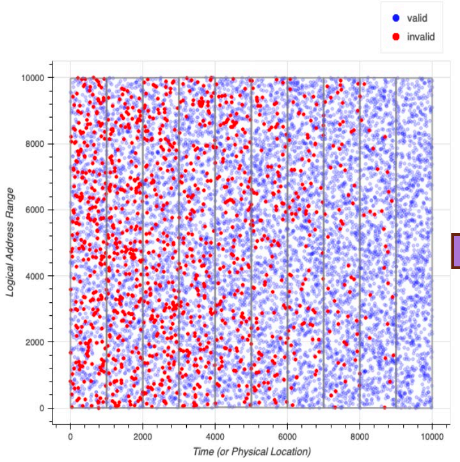
Localized Writes with Tapered Invalidation



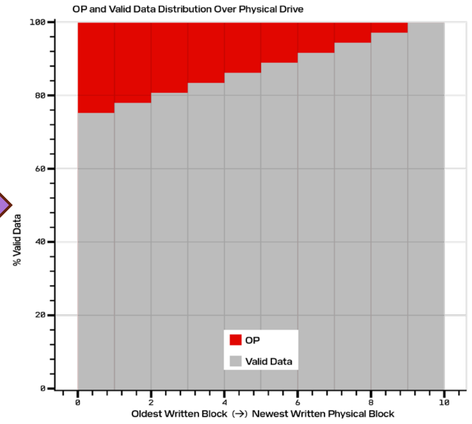
Logical Address Shuffle



De-localized Writes with Tapered Invalidation



Physical OP Distribution After One Physical Drive Write

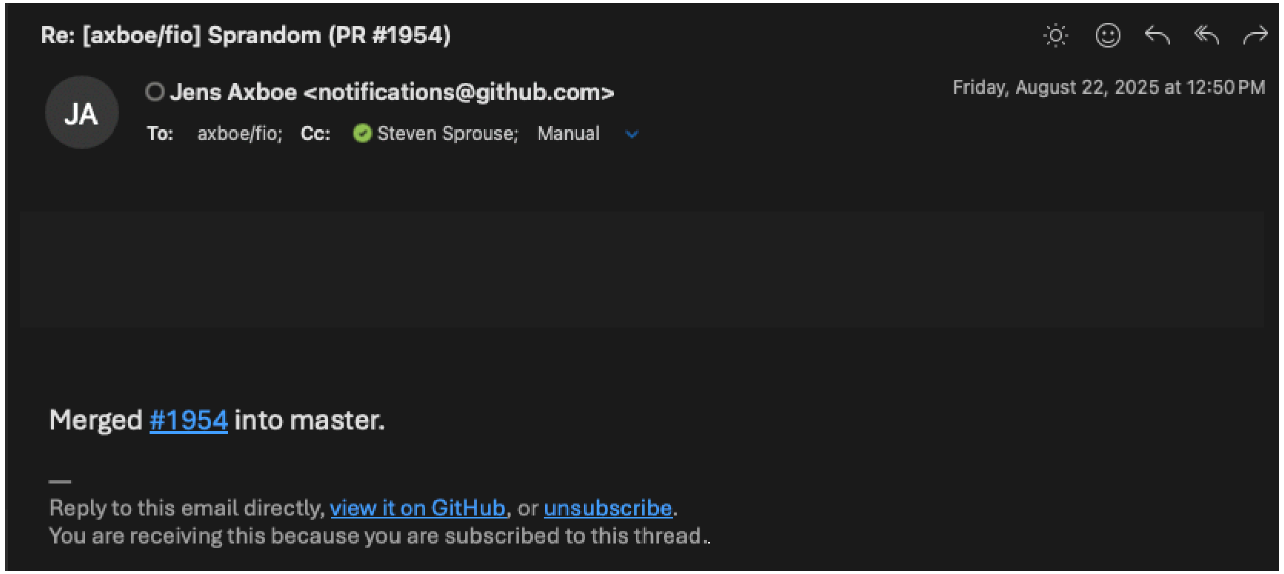


 |

SANDISK™ SDC25

Integration Into FIO Open Source.

- The code was developed by SanDisk and integrated into FIO.
- Merged on Aug 25, 2025
- Simple interface.
- Sprandom is a new job type enabled with `sprandom=1`
- User sets the OP and # sub-regions to partition the drive.


Re: [axboe/fio] Sprandom (PR #1954)

JA Jens Axboe <notifications@github.com>

Friday, August 22, 2025 at 12:50 PM

To: axboe/fio; Cc: Steven Sprouse; Manual

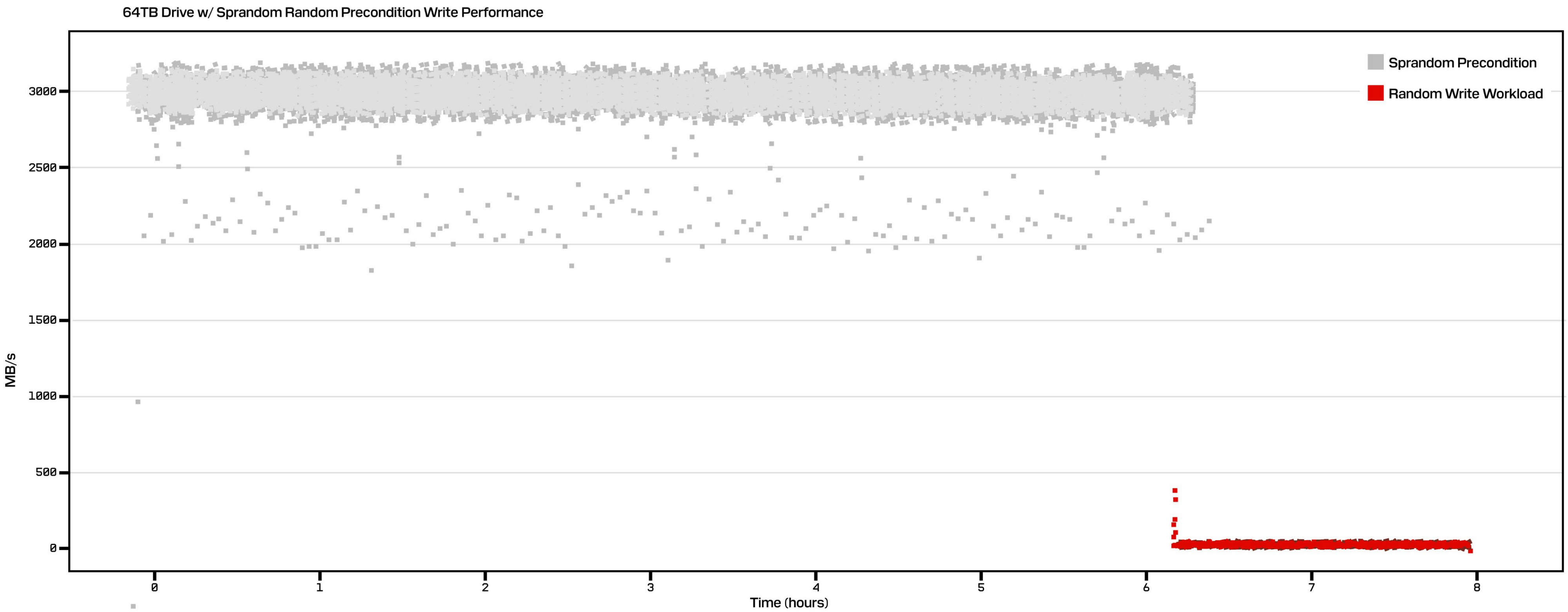
Merged #1954 into master.

—
Reply to this email directly, view it on GitHub, or unsubscribe.
You are receiving this because you are subscribed to this thread.

FIO Jobfile

```
[preconditioning]
sprandom=1
spr_op=0.15
spr_num_regions=400
```


Example Preconditioning of a 64TB Drive*



Conventional Precon = 6 hours (seq precon) + 138 hours (random precon) = 144 hours
Sprandom Precon = < 7 hours

*We gratefully acknowledge Western Digital Platforms Business Unit and the use of their OCCL Lab (www.opencomposable.com) for their valuable collaboration, use of facilities, and support in product testing and documentation.

Next Steps

- Sprandom is proposed as a method to address the prohibitively long times it takes to randomly precondition high-capacity SSDs.
- Looking to foster adoption of Sprandom as an industry wide test methodology.
- How to get involved:
  - Try out the feature.
  - Look for applications: Development, Qual, Production, and Deployment.
  - Propose improvements
- fio with Sprandom integration:
  - https://github.com/axboe/fio (code merged on Aug 22, 2025, release version is TBD.)

Thank you for attending!

Please remember to rate this session. You get access the presentations at
http://sniadeveloper.org/conference

