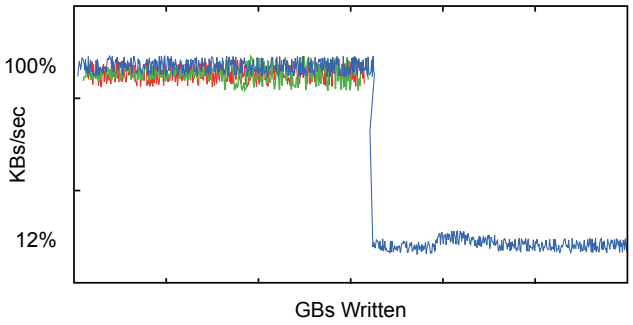


SSD Performance Tips: Avoid The Write Cliff

An Inexpensive and Highly Effective Method to Keep SSD Performance at 100% Through Content Locality Caching

VeloBit
Faster Storage Made Easy

Share this ebook

Start

INTRODUCTION

Solid state disk (SSD) based on flash memory has emerged as a popular storage medium. Because SSD uses semi-conductor chips, it provides great advantages over mechanical disk in random read speed, power consumption, size, and shock resistance. However, SSD has limitations in write performance and durability because of the physical properties of flash. These limitations are exacerbated when the SSD hits the so called *"write cliff"*, the point at which SSD write-performance slows down significantly. The SSD hardware vendors attempt to delay the SSD write cliff through *overprovisioning,* a costly solution. This paper discusses a simple, inexpensive, and non-disruptive solution that greatly minimizes the SSD write cliff effect.

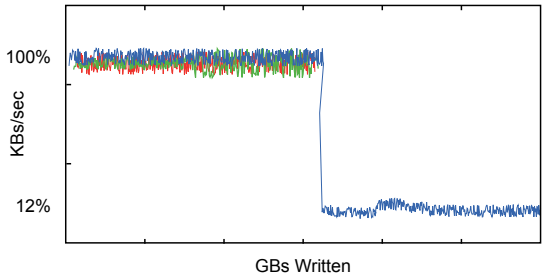
CONTENTS

WHAT IS THE SSD WRITE CLIFF? WHY DOES IT OCCUR?

SSD *write cliff* is the effect where SSD write performance drops off after all the free SSD (Flash) cells have been initially written to and the device cannot provide enough free blocks to keep up with write requests. The performance drop can be dramatic, often in the range of 6-10x reduction in write speed. The less Flash capacity a system has the sooner this effect occurs and the more significant the drop off.

Let's explore the design of flash-based SSD so that we better understand what's causing these limitations. A typical NAND-gate array flash memory chip consists of a number of blocks, each of which contains a number of pages (e.g. a block with 64/128 pages of 4KB/8KB each). In NAND flash, blocks are the smallest erasable units whereas pages are the smallest programmable (i.e., writable) units. When a write operation is performed, the SSD needs to first find a free page for the write. If there is no free page available, an erase operation is necessary to make free pages. A read operation usually takes a few or tens of microseconds, whereas a write takes tens to hundreds of microseconds and an erase operation takes 1.5 to 3 milliseconds.

After an SSD is filled with data, unused (*"stale"*) pages must be erased before new data is written. Since a block is the smallest erasable unit in an SSD, in order to erase the stale pages the SSD controller must erase all pages on the block (stale or active). Therefore, before a block is erased, all pages with active data must be re-written into another block with free cells. So, writing a single page to SSD can result in multiple corresponding *re-write* operations. More-over, the re-writes can trigger additional erase operations. This *"write amplification"* is a major cause of SSD wear, and is the primary cause of the SSD write performance cliff.



SSD Write Cliff

Garbage collection is a background process that occurs when there are spare SSD controller cycles. Garbage collection makes use of "quiet time" for SSDs to reclaim previously used SSD blocks and free pages for new *writes.* During garbage collection, the SSD controller pro-actively scans the contents of the SSD blocks to identify stale pages, re-writes any active pages found in the same block, and then erases the block. Unfortunately, in the 7x24x365 enterprise data center there is rarely a "quiet time" for an SSD to perform its housekeeping duties while not having to process critical data requests at the same time. When garbage collection cannot keep pace with new writes, the SSD hits the *write cliff.*

OVERPROVISIONING: THE SSD HARDWARE VENDORS' METHOD TO POSTPONE THE WRITE CLIFF

SSD hardware vendors delay the *write cliff effect* through *overprovisioning.* The basic principle of overprovisioning is to put more capacity (NAND) on the SSD than the drive actually reports (some high end SSD drives overprovision by 25% or more). Even though there may be 400GB of physical capacity on the SSD, the SSD tells the operating system that it has only 320GB. The SSD controller uses the hidden capacity to scatter and stage data during housekeeping functions like garbage collection. Because there is extra (overprovisioned) capacity, it is less likely that the SSD will run out of free pages for new writes and the likelihood of the write cliff is reduced.

Overprovisioning enables SSD vendors to delay the *write cliff effect* but this benefit comes with added expense for unused capacity. And, overprovisioning does not guarantee that the SSD will not experience a write cliff when it is used in a 24/7 production environment.

OPTIMIZE DATA FOR SSD AND AVOID THE WRITE CLIFF WITHOUT OVERSPENDING ON UNUSED SSD

If the system were able to separate static from dynamic data (i.e. separate data that doesn't change from data that changes often), write amplification would be reduced and garbage collection would be simplified. Furthermore, if data were optimized to maximize *reads* and minimize random *writes* to SSD, both performance and reliability would be improved. This is where content locality caching helps. Content locality caching optimizes the performance of solid state disk (SSD) and overcomes two of the main limitations of SSD – write speed and durability.

The content locality caching algorithm stages data blocks based on the popularity of their contents. There are two types of data blocks stored in cache. Popular blocks are designated as *"reference blocks"*. Blocks that are sufficiently similar to reference blocks are defined as *"associated blocks"*, and are stored in a compressed form using a pointer to a reference block and a *"delta"* that describes the differences between the two blocks. When the application requests data and a requested data block is in cache, either a *reference block* is delivered directly or an *associated block* is re-constructed at line speed. Reconstructing an *associated block* from its compressed form consumes CPU cycles. However, since CPU speed is much faster than storage disk I/O speed, *associated blocks* are reconstructed and delivered much faster than if they were fetched from disk.

Content locality caching optimizes data for SSD in three major ways that reduce the likelihood of a write cliff:

- **Separates data into read-only vs. actively changing data blocks**

  The content locality caching algorithm effectively separates data into read-only reference blocks and actively changing associated blocks. Since read-only data is stored in separate blocks from actively changing data, fewer active pages have to be re-written during garbage collection. This means that the garbage collection process can complete all housekeeping tasks faster and keep up with new incoming writes.

- **Converts SSD into (primarily) read cache**

  Reference blocks are written once but read many times (as they are used to describe *associate blocks*). Associated blocks have a smaller footprint since they are stored in a compressed form. Because a large portion of the SSD is used for read-only operations, an SSD used with content locality caching is faster than an SSD used for random reads and writes. Content locality caching also reduces the likelihood that the SSD will hit the *write cliff.*

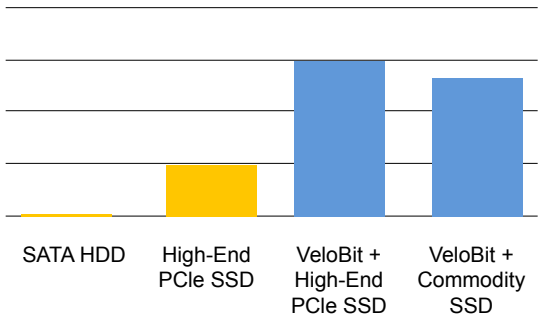
- **Simplifies data staging and maintenance operations on SSD**

  Compressed associated blocks are easier to scatter and write in existing open page slots. As a result, the SSD controller needs to perform fewer data re-organization tasks and fewer erase operations. Therefore, the SSD is less likely to experience the *write cliff.*

  As a result, both SSD performance and the predictability of SSD performance are improved. Content locality caching can be used as cost-effective complement or alternative to overprovisioning for eliminating the write cliff and improving SSD performance and reliability.

VELOBIT HELPS AVOID THE WRITE CLIFF WHILE IMPROVING PERFORMANCE BY 10X

VeloBit is the first solution that leverages content locality caching to improve performance. VeloBit is plug & play caching software that uses SSD to create a transparent acceleration layer that eliminates I/O bottlenecks and improves performance by 10x. VeloBit employs a number of other proprietary features, such as flash drive optimization, horizontal architecture, and automatic configuration, to enhance performance while reducing cost and minimizing disruption and complexity. The software installs seamlessly in <10 minutes and automatically tunes for fastest application speed. VeloBit deploys and operates without disruption to applications, storage, or data.



VeloBit Increases Performance by 10x

Applications that can benefit from VeloBit include:

- Financial analysis, research, simulation, or modeling applications
- Databases, such as *SQL Server, MySQL, Oracle, PosgreSQL, Cassandra,* etc.
- Enterprise applications such as *enterprise search, Apache Solr (Lucene),* etc.
- E-commerce and social networking applications

VeloBit can also be a benefit when reducing storage latency or increasing storage IOPS will **reduce the amount of hardware required to support a given workload.**

Specific examples include:

- Distributed applications such as *Hadoop*

SUMMARY

SSD write cliff is the effect where SSD write performance drops by 6-10x after the SSD is full and the device cannot provide enough free blocks to keep up with the write requests. SSD hardware vendors delay the write cliff effect through overprovisioning — reserving up to 25% or more of overall drive capacity for internal housekeeping operations. Overprovisioning delays the write cliff effect but this benefit comes with added expense for unused capacity. Plus, overprovisioning does not guarantee that the SSD will not experience a write cliff when it is used in a 24/7 production environment.

Content locality caching offers a simple, inexpensive means to reduce the likelihood of the SSD write cliff. The content locality caching algorithm structures data into separate read-only blocks and actively-changing blocks, converts most of the SSD into read-only cache, and simplifies data staging and maintenance operations. As a result the SSD runs faster, does not hit the write cliff, and the user experiences 10x acceleration in storage IOPS and application speed. VeloBit is the industry's first solution that leverages content locality caching to improve performance while reducing cost and minimizing disruption. VeloBit is a software only solution that seamlessly installs, automatically configures, and is transparent to applications and primary storage.