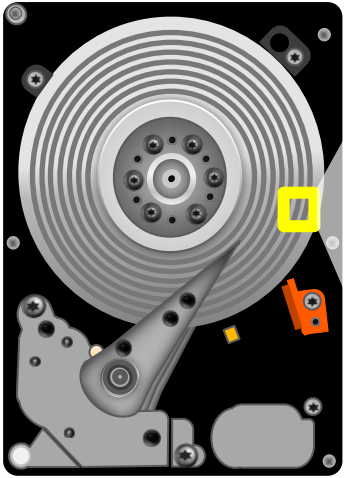# SMR:The Next Generation of Storage Technology
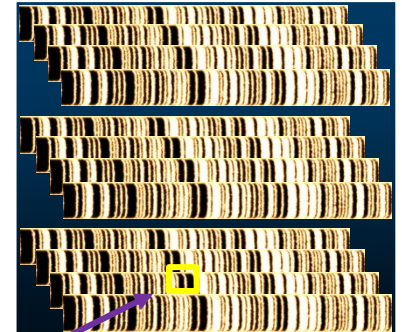
Jorge Campello

HGST, a Western Digital Company
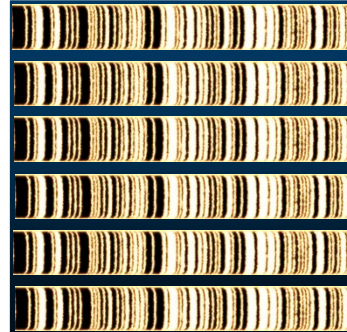
What is Shingled Magnetic Recording?
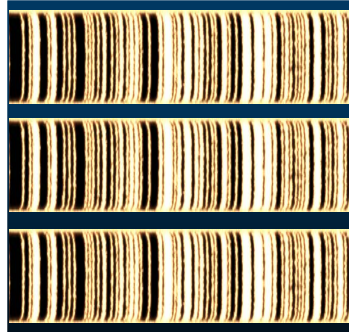
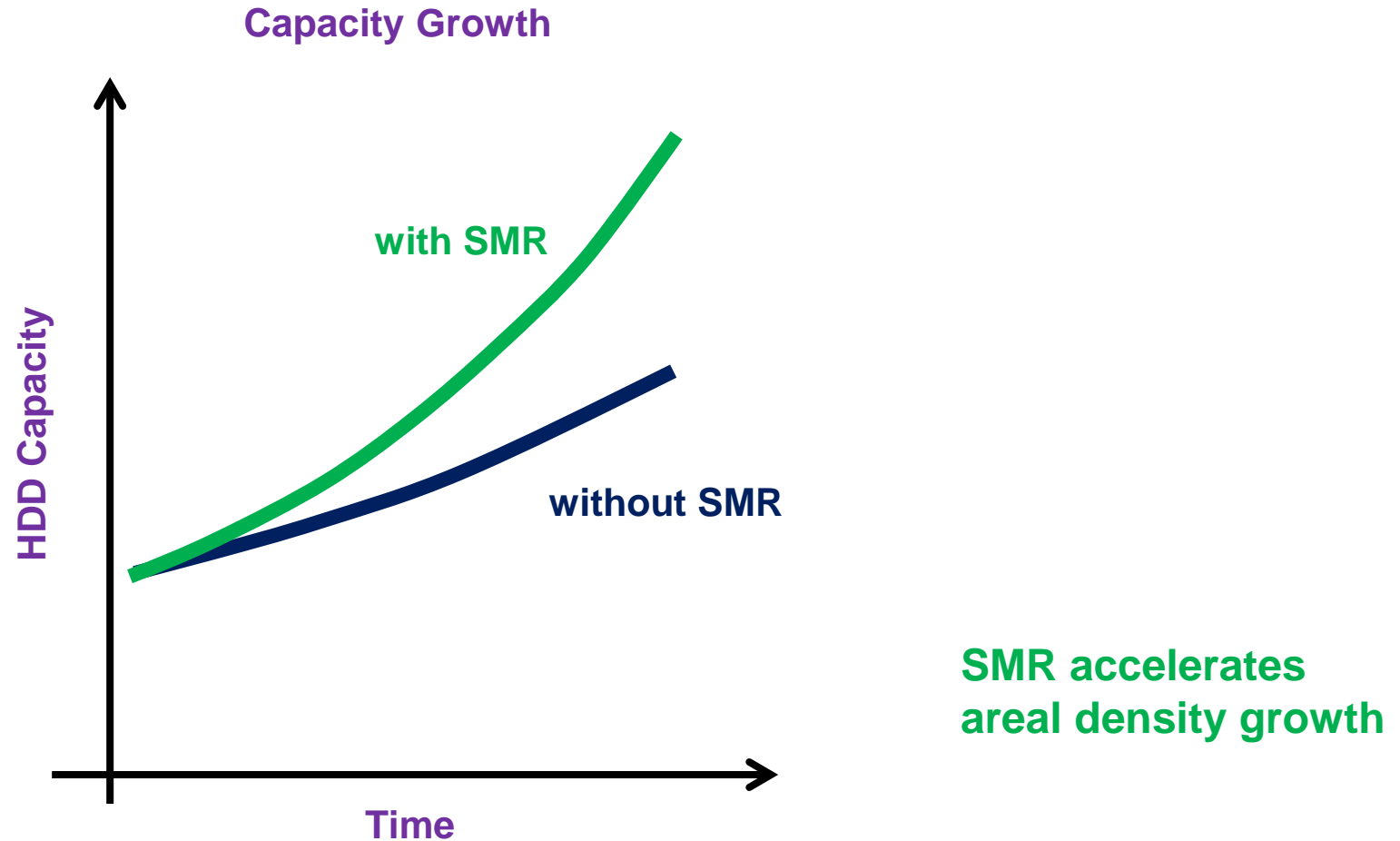Conventional PMR HDD

Data in Discrete Tracks

SMR HDD

Zone

Data in Zones of Overlapped Tracks

While Zones are independent, we can't change sectors independently within a Zone.

SMR Standards

- T10: ZBC

- T13: ZAC

Why SMR?

Capacity Growth

with SMR

without SMR

HDD Capacity

Time

SMR accelerates areal density growth

So How do We Deal with the Writes?

- Can't the drive manage it internally?
  - Drive Managed Model
- Can't we just present the host system with the constraint and let the host manage it?
  - Host Managed Model
- Isn't there something "in between"?
  - Host Aware Model

Disk Models

❑ Three models are defined

Model	Description	Impact on Host Software
Drive Managed	• Disk firmware handles random writes processing • Backward compatible (standard Device Type 0H) • No host changes necessary • <u>Performance can be unpredictable</u> in some workloads	**NONE**
Host Managed	• Host uses new commands and information to handle write operations • Not backward compatible (Device type 14h) • Predictable Performance	Host SW must write sequentially to the disk
Host Aware	• Disk firmware handles random writes processing • Backward compatible (standard Device Type 0H) • Host uses new commands and information to optimize write behavior "or" host can treat device as an autonomous device • Performance can be unpredictable if the host sends a "sub-optimal" request	**NONE ~ HIGH** Depends on the amount of optimization

SMR Drive Model

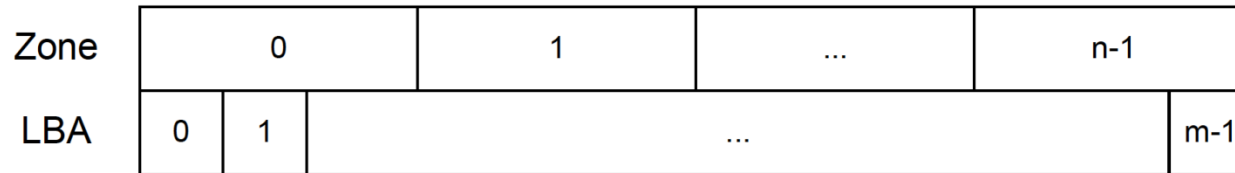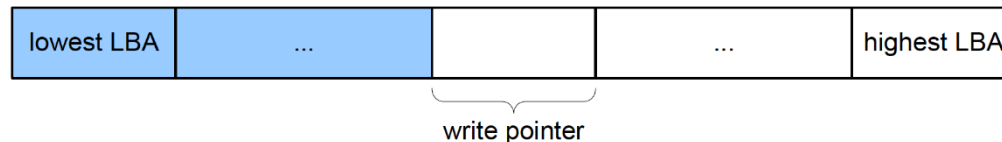- ❑ The drive is divided into zones

Figure 2 — Zones in a zoned block device

- ❑ Three types of zones
  - ❑ Conventional: Works like traditional PMR
  - ❑ Write Pointer Zones: They have a write pointer indicating the position for the next write. Two types
    - ❑ Sequential Write Required: Writes must be sequential in zone
    - ❑ Sequential Write Preferred: Writing at write pointer not required.

6

SMR Drive Model

Zone Type	Disk Model	Characteristics
Conventional zone	Host-aware and Host-managed	Unconstrained writes
Sequential write preferred	Host-aware	Unconstrained writes possible
Sequential write required	Host-managed	Write operations MUST BE sequential

Disk LBA range

Conventional zone

Sequential write zones

Zone empty

Partially written zone

Zone full

Write pointer
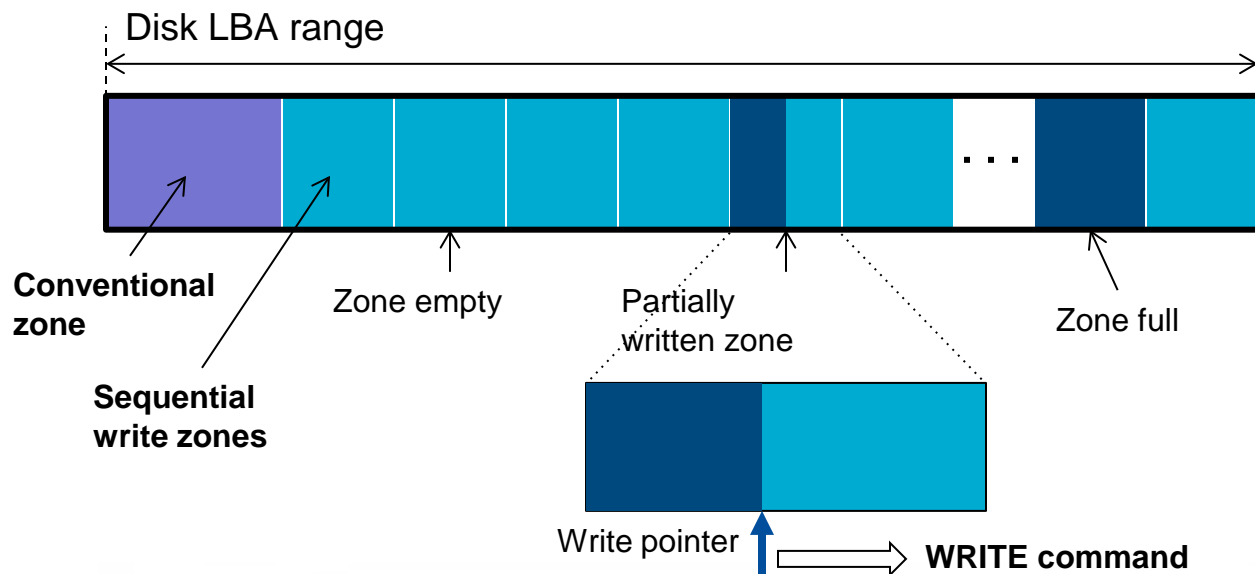
WRITE command

ZBC/ZAC Command Set

- Two main commands
  - **REPORT ZONES**: discover disk zone configuration and zone write pointer position
  - **RESET WRITE POINTER**: reset a zone write pointer position to the beginning of a zone
    - » Destructive operation: data previously written in the zone becomes inaccessible
- Disk write performance in sequential write zones can be optimized using additional commands
  - **OPEN ZONE**: keep a zone FW resources locked until the zone is closed
  - **CLOSE ZONE**: release a zone FW resources
  - **FINISH ZONE**: fill a zone

SMR Implementation Models

	Host		
	No SMR Management	Varying amount of SMR Management	Full SMR Management

Host

IOs

SMR Drive

Unconstrained

Constrained

Drive Managed   **Host Aware**   **Host Managed**

9

SDC 15
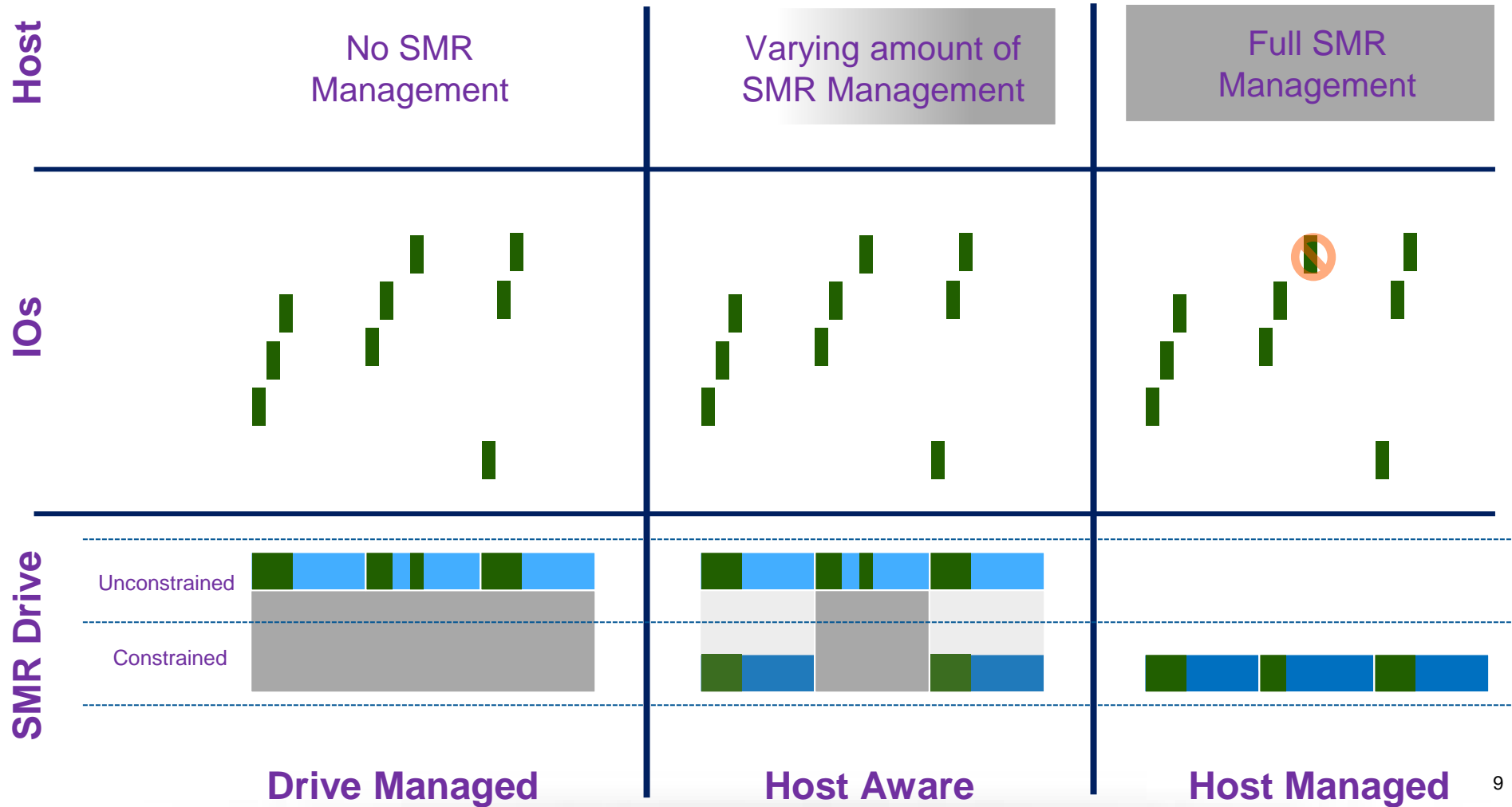
Drive Managed Model

- Sequential Read
  - Similar to PMR

- Random Read
  - Similar to PMR
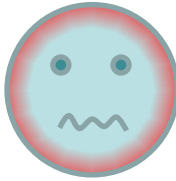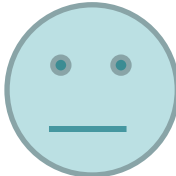
- Sequential Write
  - Similar to PMR

- Random Write
  - YMMV

  ?

Drive Managed Model: Random Write

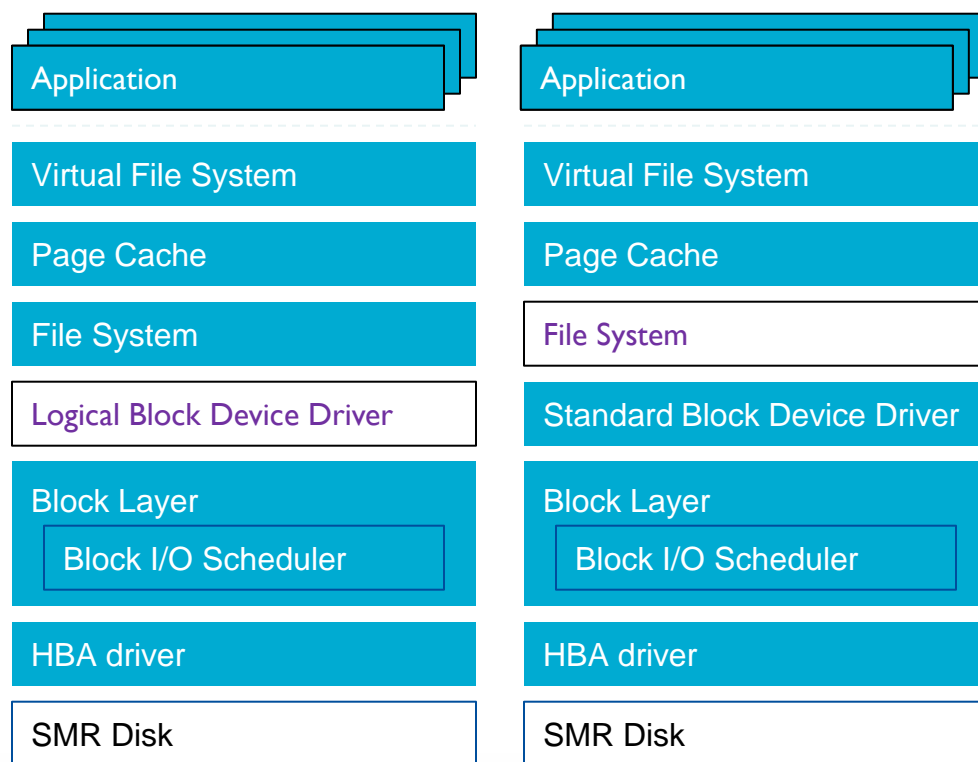|	**Small Block**	**Large Block**	**Huge Block**
High Duty Cycle	Performance dominated by seek time. Caching writes on media and moving later has good performance.	Seek time no longer dominates. Writing twice has overhead. High duty cycle fills up cache quickly and doesn't allow time for recovery.	Behaves close to sequential writes.
Low Duty Cycle	Performance dominated by seek time. Caching writes on media and moving later has good performance.	Seek time no longer dominates. Writing twice has overhead. Low duty cycle allows drive to hide overhead.	Behaves close to sequential writes.

Host Managed Model

❏ New device type.

  ❏ Not backwards compatible

  ❏ Doesn't work with existing software

❏ How do we make use of Host Managed SMR?

  ❏ HW Solutions: HBA, appliance, etc, virtualizes

  ❏ SW Solutions:

    ❏ Kernel level support

    ❏ Application level support

Kernel Level Support

☐ Basically, two approaches here

- Device mapper logically standard storage device
  » STL: Similar to flash FTL
  » Allows reusing existing software components (e.g. file systems)
- SMR compliant file system
- Combination of both

☐ Currently

- Support for XFS announced
- Other FS branches and news FS projects started.
- Device mapper implementations exist.
- Zone caching schemes being investigated.

Application	Application
Virtual File System	Virtual File System
Page Cache	Page Cache
File System	File System
Logical Block Device Driver	Standard Block Device Driver
Block Layer Block I/O Scheduler	Block Layer Block I/O Scheduler
HBA driver	HBA driver
SMR Disk	SMR Disk

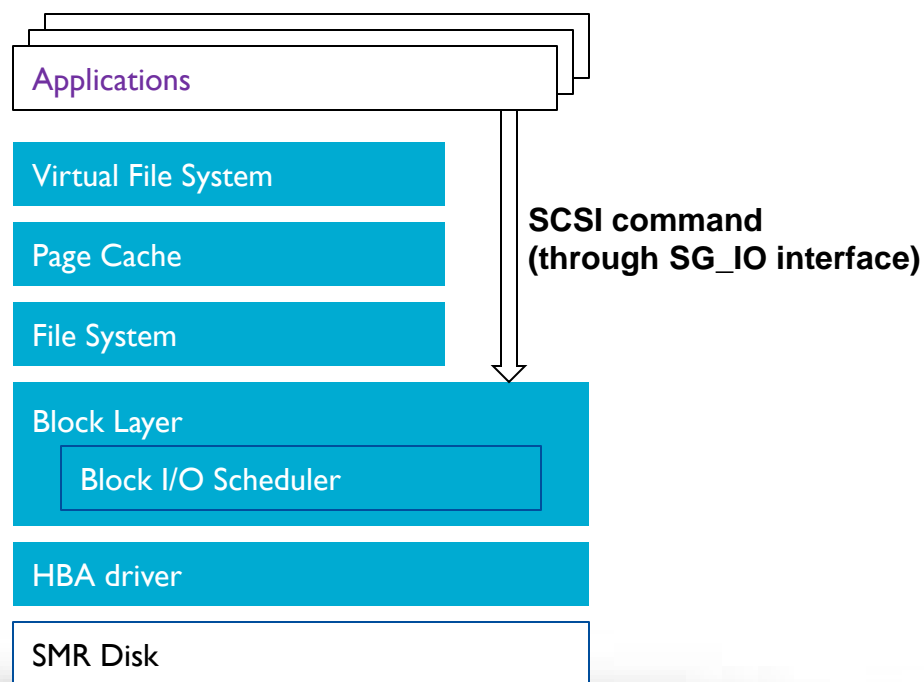http://xfs.org/index.php/File:Xfs-smr-structure-0.2.pdf

Direct Application Management

□ Direct disk accesses from applications through SG_IO

 - Application level code issues ZBC commands and write-pointer aligned write operations through SG_IO interface

 - Kernel level support is minimized

  » Device type recognition on HBA bus scan for device SG node creation

□ Fits many use cases were kernel file systems are not strongly necessary

 - Distributed object stores

✓ **Simple for single application disk use**
✓ **Can be implemented NOW !**

✗ **Functionality loss (caching, …)**

Applications

SCSI command (through SG_IO interface)

Virtual File System

Page Cache

File System

Block Layer
Block I/O Scheduler

HBA driver

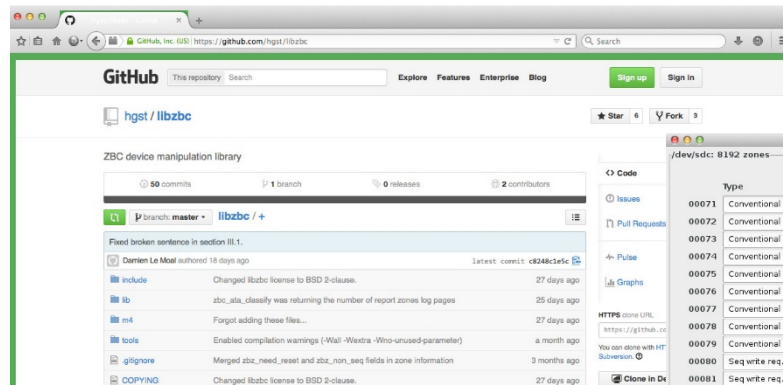| SMR Disk |
| --- |

Libzbc Open Source Project

- User space library providing applications with direct access to ZBC/ZAC drives.

- Enabling application level innovations!

User level

Application   Application

libzbc

Native mode   Emulation mode

Standard system calls (read/write)

Kernel

Kernel VFS   Kernel VFS

Block Layer   Block Layer

SCSI command (SG_IO)

SCSI Layer   SCSI Layer

HBA driver   HBA driver

Hardware

ZBC/ZAC disk   PMR disk
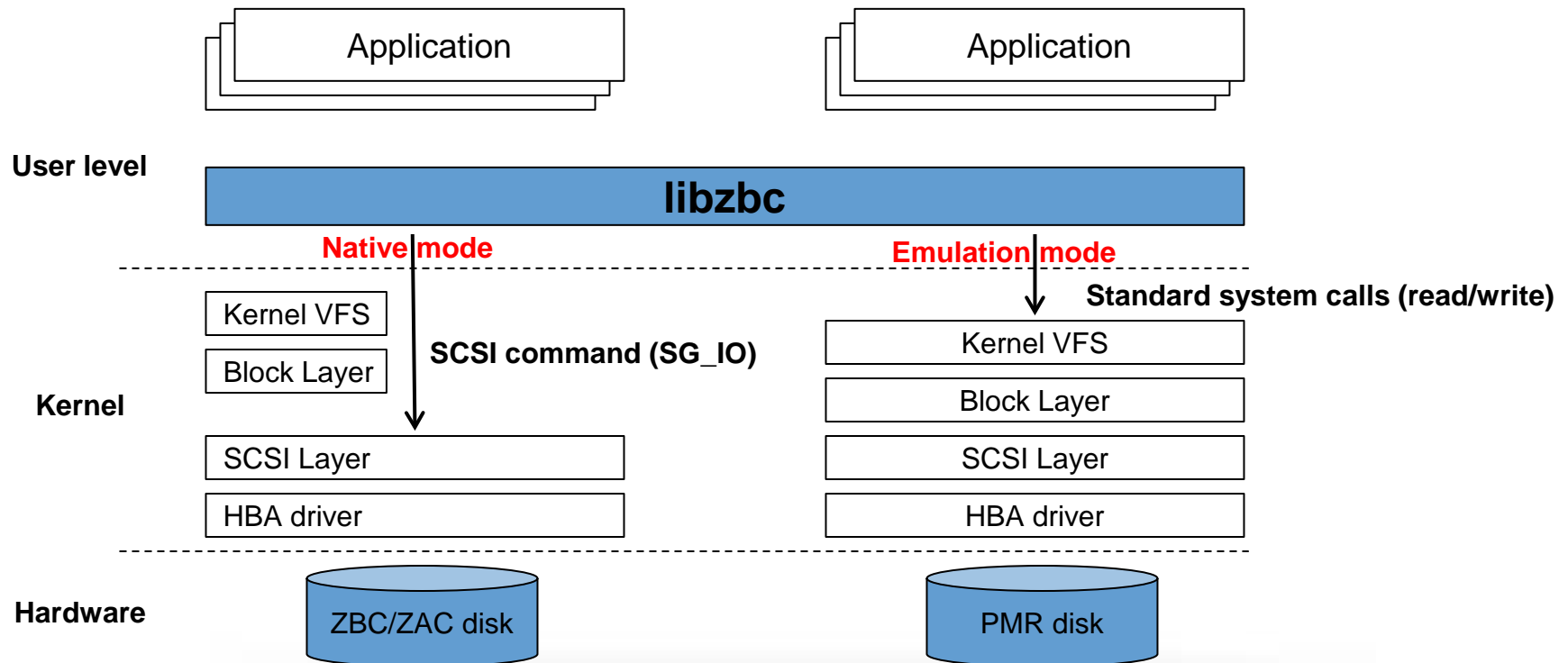
Libzbc Project

Download NOW: http://github.com/hgst/libzbc

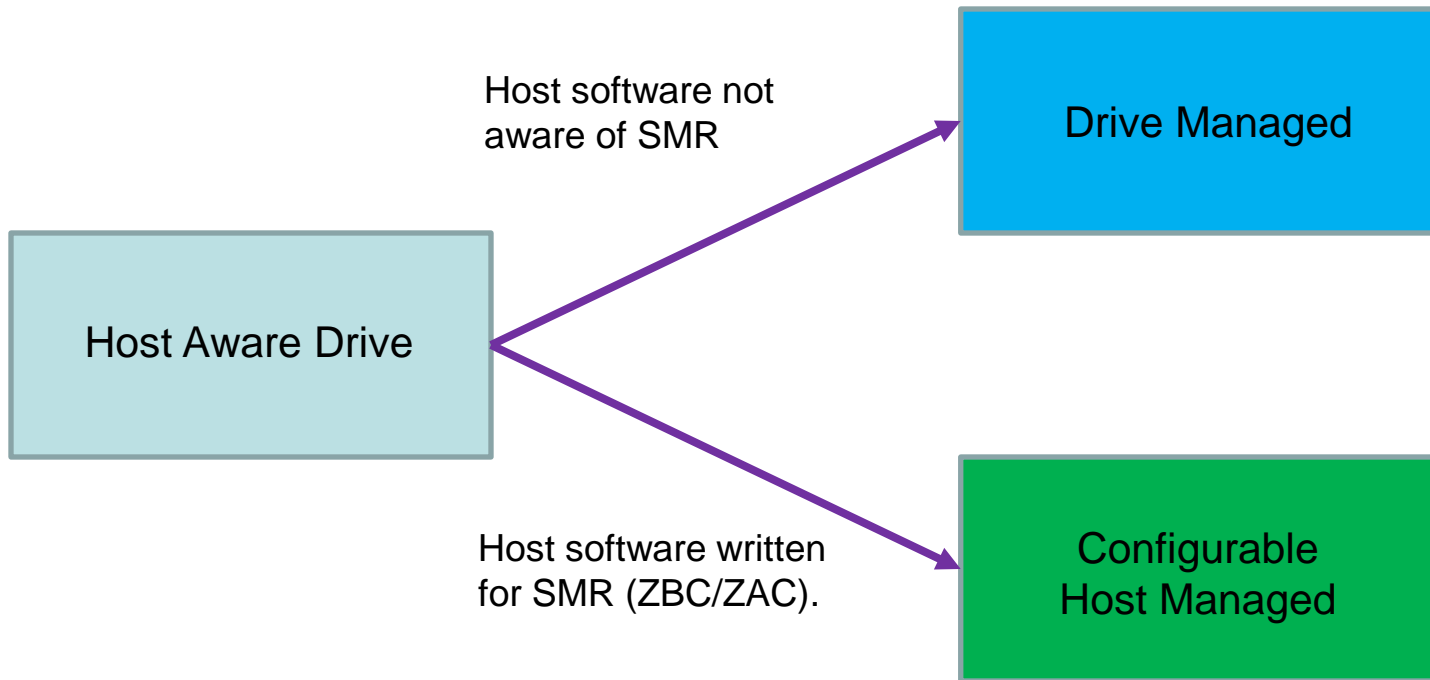| Command	Description
zbc_report_zones	Display a disk zone information
zbc_reset_write_pointer	Reset a zone or all zones write pointers
zbc_read_zone	Read data from a zone
zbc_write_zone	Write data or a file to a zone
zbc_set_zones	For emulation mode: configure the disk zones
zbc_set_write_pointer	For emulation mode: manually change the value of a disk write pointer
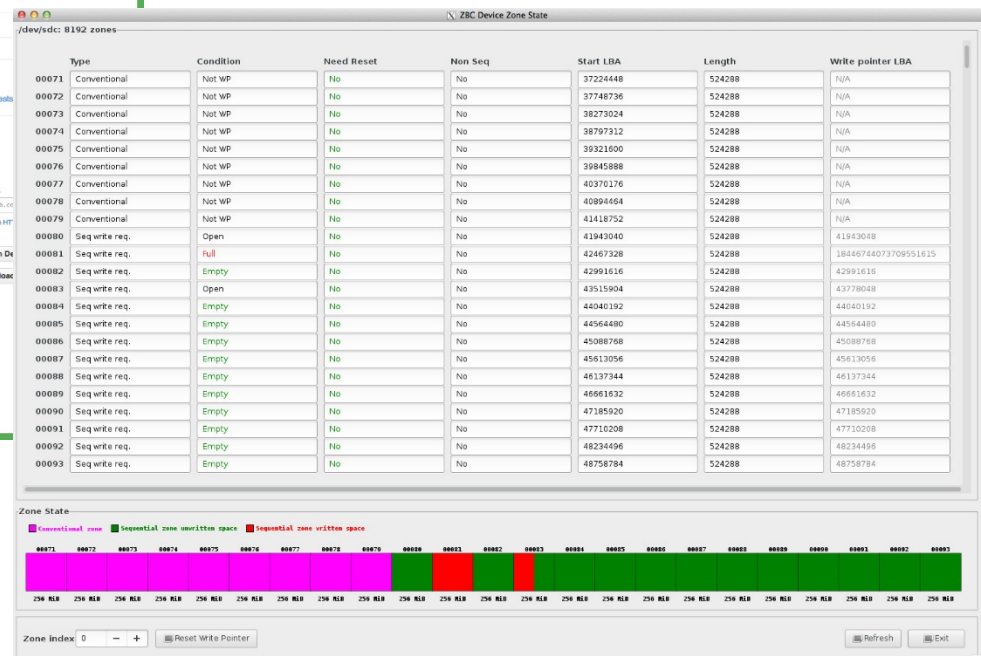
Host Aware Model

- Backwards Compatible
- Implements ZBC/ZAC
- In essence, a hybrid Model.

Host Aware Model

Host software not aware of SMR

Drive Managed

Host Aware Drive

Host software written for SMR (ZBC/ZAC).

Configurable Host Managed

The SMR Opportunity

❒ Those who learn how to manage the SMR constraint first will be the new leaders in mass storage solutions.

❒ New storage solutions designed with SMR in mind will have a technological advantage.

THANK YOU

HELPING THE WORLD HARNESS THE POWER OF DATA WITH SMARTER STORAGE SOLUTIONS. WWW.HGST.COM