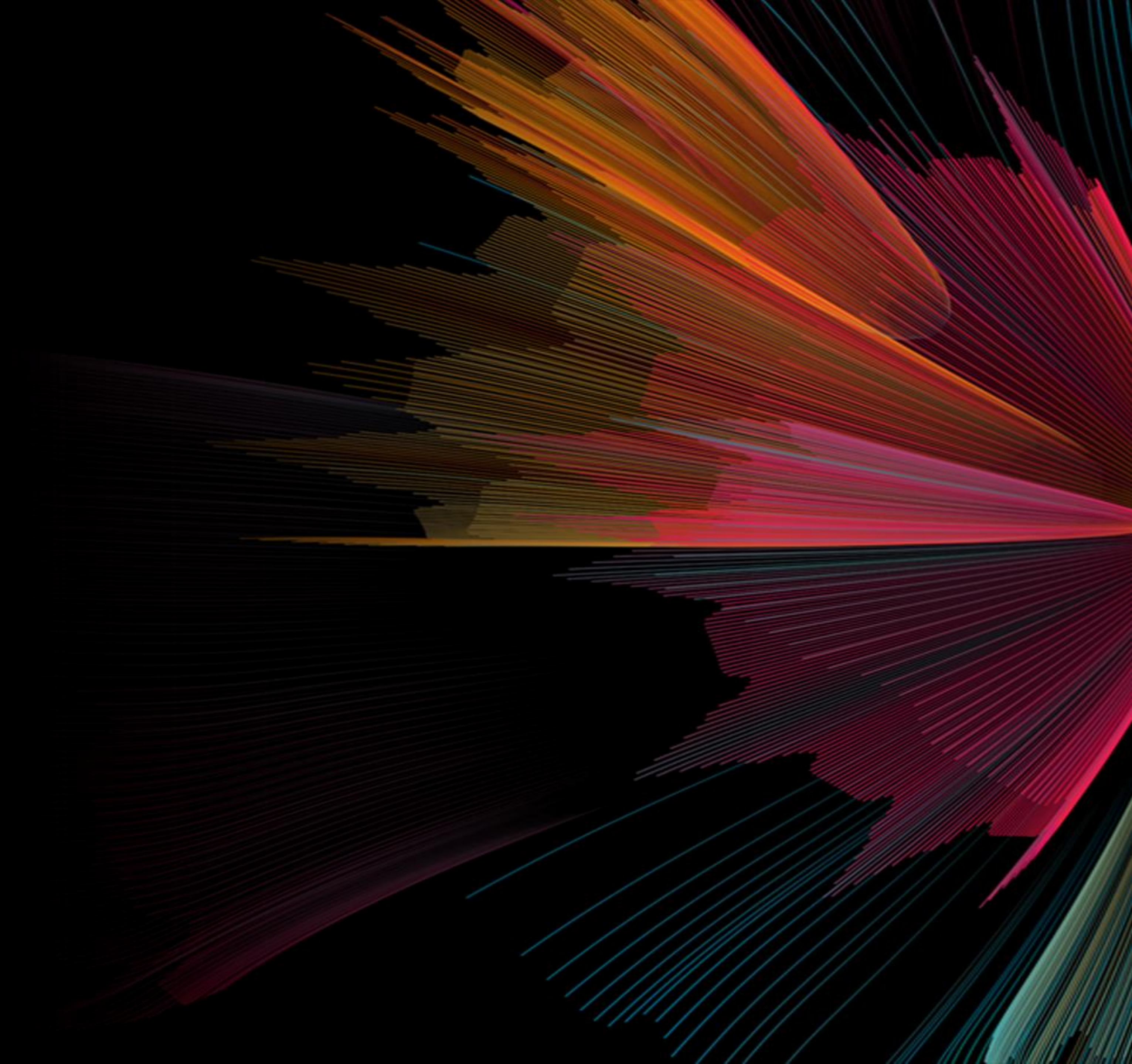
Western Digital

Realms API



December 12, 2017
By
Bill Boyle and Curtis E. Stevens

Agenda

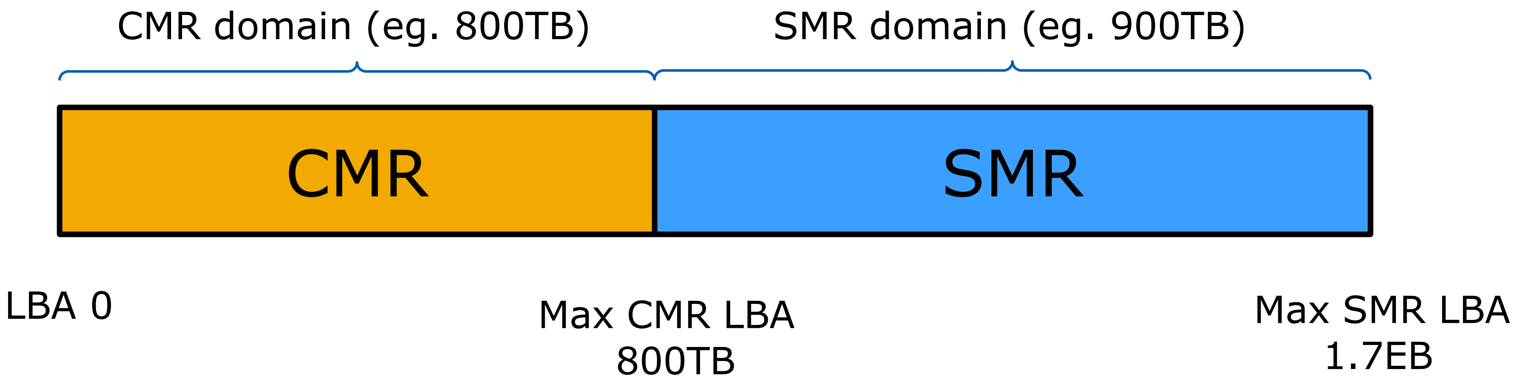
- Realms API Goals
- LBA Address System
- Realms
- SMR and CMR Zones
- Conversion of Realms
- CMR Sequential Write Pointers
- Standards Impacts
- Backward Compatibility

Realms API Goals

- The Realms API is an extension of Host Managed SMR
- Allows host systems to gradually convert from using CMR type recording to SMR recording
- Host system can segregate frequently written from infrequently written data
- Allows system to dynamically adjust the amount of CMR vs. SMR capacity in the device
- Enables backward compatibly to existing BIOS systems.

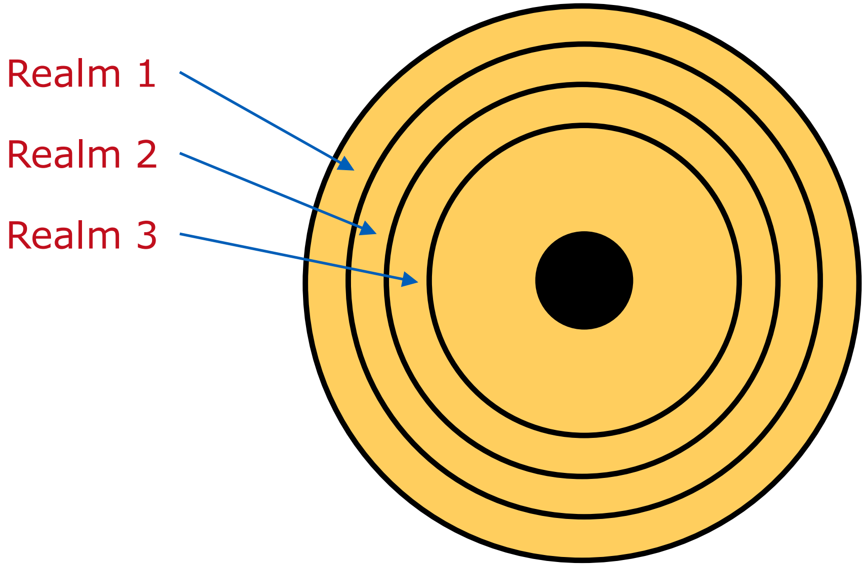
LBA Address System

- LBA space is partitioned into two domains, CMR and SMR
  - Each partition is the maximum possible size for the media type
- CMR addressing will start at LBA 0 and extend to the CMR capacity capabilities of the drive
- SMR addressing will start immediately after the CMR domain and extend to the SMR capacity capabilities of the drive.



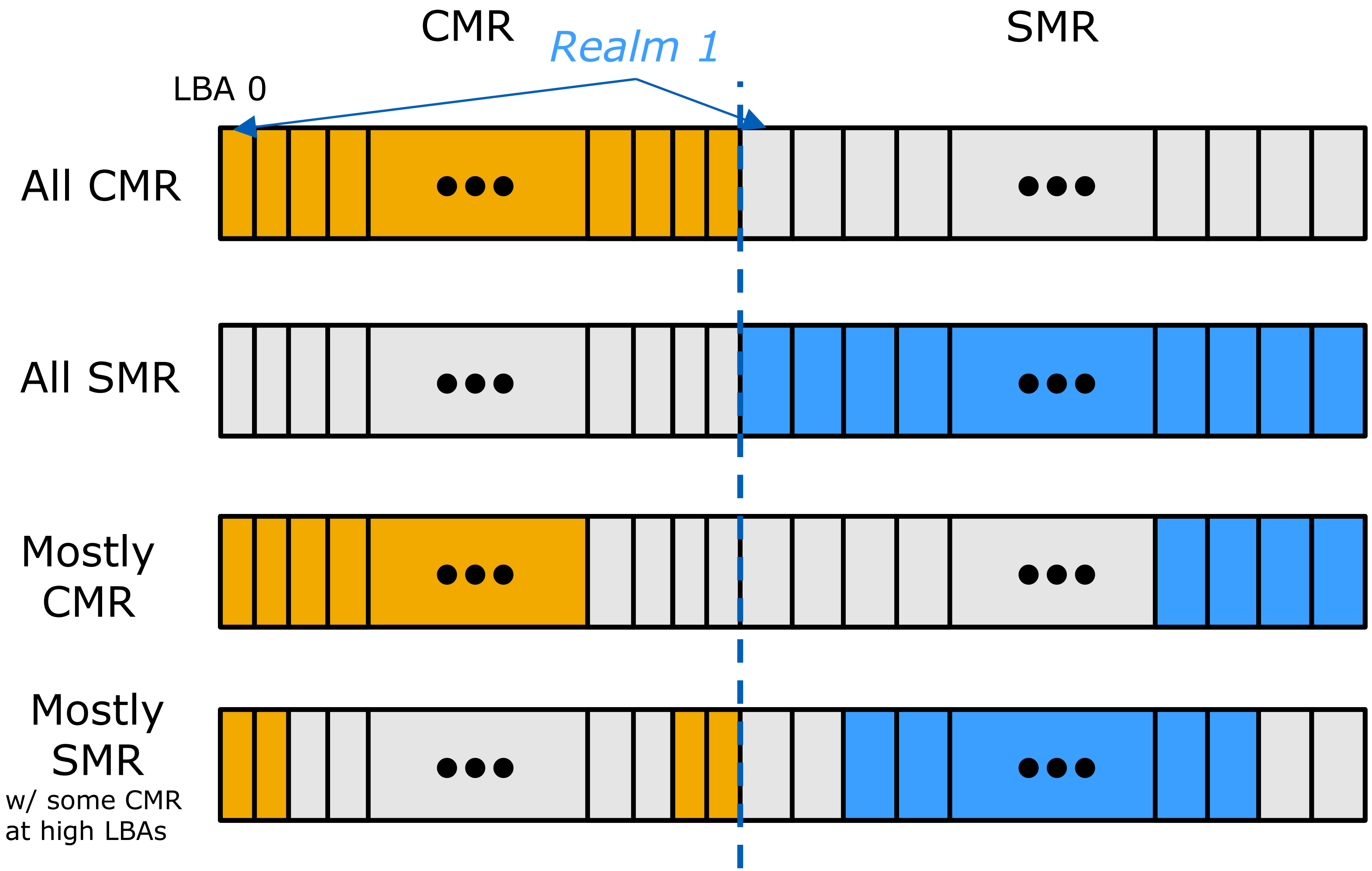
Realms

- A Realm is a physical portion of the device that stores user data
- A Realm is intended to be an allocation unit for the application client
- Each Realm is either CMR or SMR recording technology
  - CMR recording will provide less capacity than SMR recording
- A Realm is the conversion unit size between CMR and SMR modes
- All Realms in SMR mode will have the same capacity
- CMR Realms will likely vary in size but will always be less than the SMR size
- Realms can be converted between CMR <-> SMR by the host dynamically



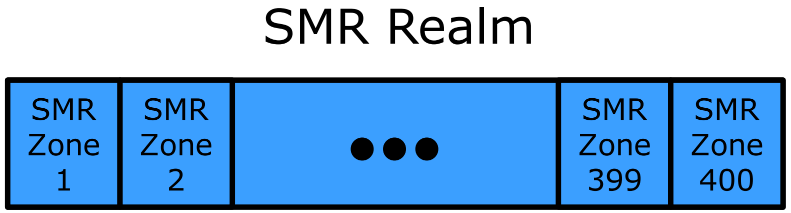
Realm Configuration Examples

- A Realm can either be in CMR mode or SMR mode.

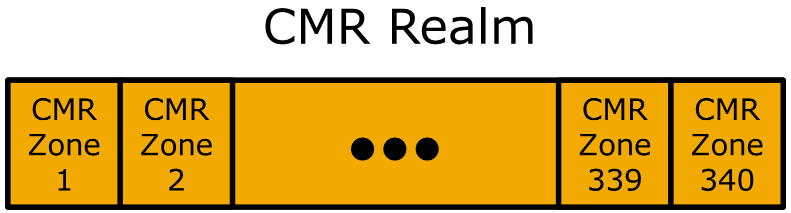


SMR and CMR Zones

- Realms are further subdivided into zones
- SMR Realms will have SMR Zones
  - SMR zones will be 256MiB each
    - A 100GiB SMR Realm has 400 SMR Zones
- CMR Realms will have CMR Zones
  - CMR Zones will also be 256MiB each
  - A CMR Realm will have a variable amount of CMR Zones depending on the CMR Realm size.
    - Example: an 85GiB CMR Realm will have 340 Zones
- Each SMR Zone has have a sequential write pointer (SWP)
- Each CMR Zone may have a sequential write pointer based on the conversion type and the usage of the zone
- SMR Zones behave as in normal Host Managed SMR drives


SMR Realm

SMR Zone 1	SMR Zone 2	●●●	SMR Zone 399	SMR Zone 400


CMR Realm

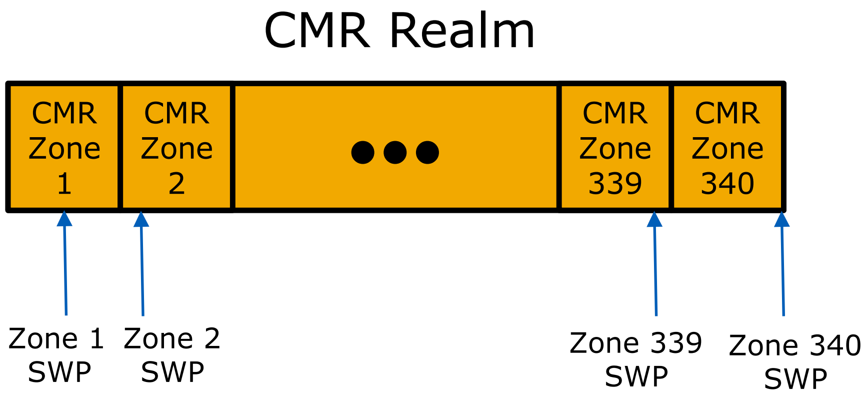
CMR Zone 1	CMR Zone 2	●●●	CMR Zone 339	CMR Zone 340

Conversion of Realms

- Conversion Prerequisites
  - Realm must be configured as convertible by manufacturer
  - Converting to CMR mode requires all SMR Zones to have their SWP reset first
    - This should be configurable for checking. This allows the device to inform the host that data may not have been moved out of the zone or zones
- Converting Realms causes all the previous data to be unreadable.
- SMR Realms have an inherent mechanism to enable initialization of the blocks through the enforcement of the Sequential Write Pointer for write commands.
- CMR Realms are intended to
  - Allow random writes without background data management activities to reorder the data
  - Allow random reads from any block in the Realm.
    - This may cause an initialization issue when a Realm is converted to CMR mode.
      - **This is why a CMR realm has a write pointer**

CMR Sequential Write Pointer

- To solve CMR Realm initialization issue, a CMR SWP is introduced
- Each CMR Zone will have a SWP to keep track of the progress of block initializations (written).
- Incoming command checking (optionally turned on by host)
  - Write commands are allowed to start <= SWP (aborted otherwise)
  - Read commands must address only LBAs that have been written
    - If checking is turned off, read commands succeed and return substitute data pattern for unwritten data
- If command checking is disabled then drive will allow full random writes and reads with potential for longer latency first write commands of blocks.



Standards Impacts

- ZAC and ZBC Interface standard changes
  - Modifications of “Report Zones Log Page” command
    - Use SWP for CMR zones also
    - Need new zone condition, UNASSIGNED. LBA’s in these zones are “deallocated”.
      - This is for zones that are in a realm that was converted from or may be converted to.
    - CMR Zones that need initialization shall be reported with “INIT_REQUIRED” Zone Condition.
      - The zone shall transition to normal “NOT WRITE POINTER” Zone Condition once all the blocks have been written.
    - Add the above to the REPORTING OPTIONS filter.
  - Addition of “Report Realms Log Page” command
    - Size of “Keep Out” for CMR Realms
    - Current Realm states
    - SMR Start LBAs
    - SMR End LBAs
    - CMR Start LBAs
    - CMR End LBAs
    - Convertible flags
  - Addition of “Convert Realm” command
    - Specifies list of Realms to convert
    - Command is blocking and ending status is not returned until conversions are complete
    - For SATA – this is a non-NCQ command

Backwards Compatibility

- Configuration of drive to support backward compatibility to traditional CMR drives.
  - Factory shipped with all Realms configured in CMR mode with all blocks initialized
  - First Realm would be in CMR mode and set to “Non-Convertible” in the factory.
  - Special case last Realm is in CMR mode and is set to “Non-Convertible” in the factory.
    - This Realm could be a much smaller size (maybe 1MB)
    - Used to support the GUID Partition Table (GPT)
  - Device signature set to traditional CMR drive.
- Higher level host device drivers would provide the support for Realms API functionality after drive/BIOS boots.

Western Digital®