

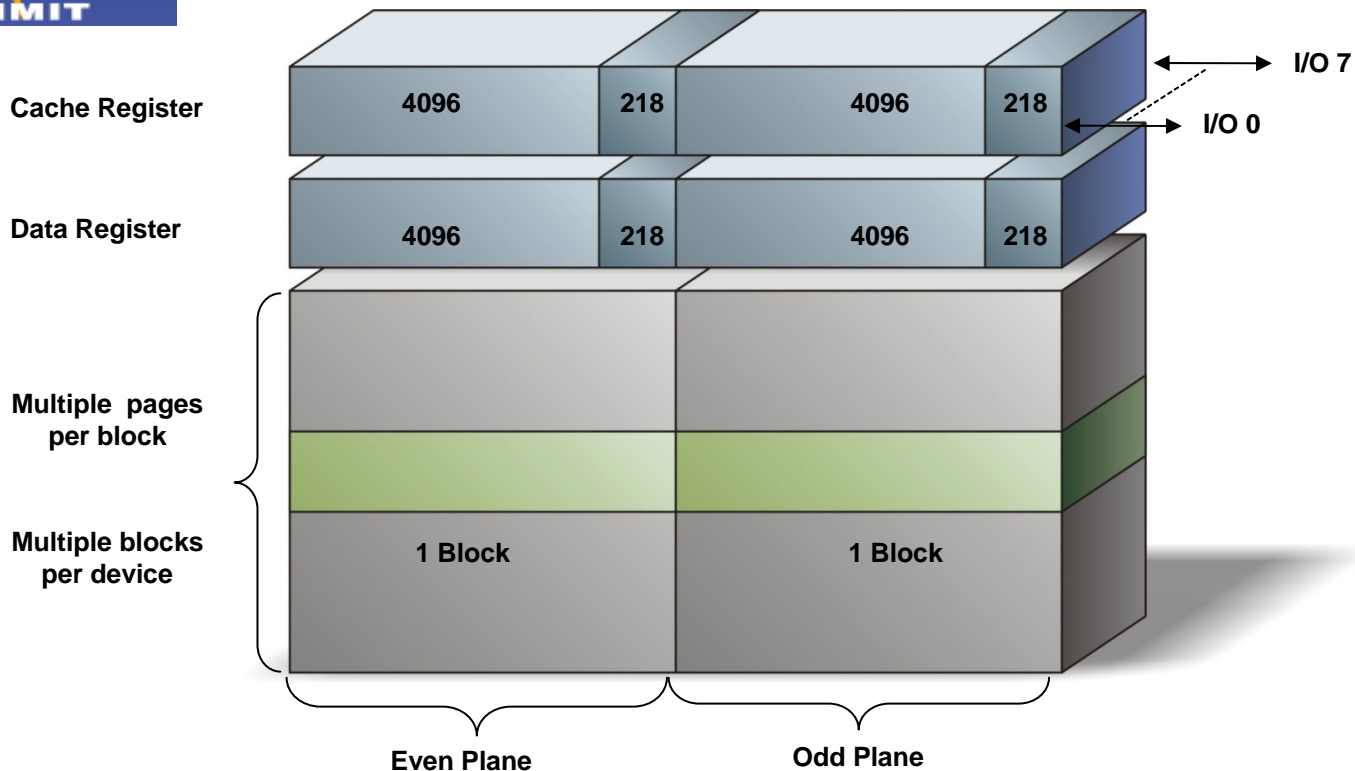


# Accommodating Solid State Storage in Your Favorite OS

Micron Technology

Justin Sykes – Director, SSD Marketing  
August 2009

# NAND Data Structure Primer



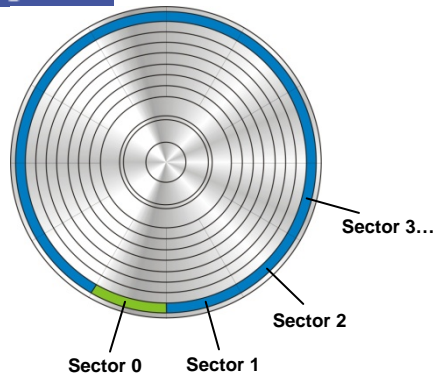
- Smallest Read/Write Unit = Page = 4KB – 8KB
- Smallest Erasable Unit = Block = 256KB – 1MB
- Read Time = 25uS – 50uS
- Write Time = 250uS – 900uS typ.
- Erase Time = 0.7mS – 3.5mS typ.



# NAND Data Structure Implications

- Issue #1: A writable unit is a page (4KB – 8KB), but ATA's LBA's address 512 bytes.
- Issue #2: A re-writable unit is an erase block, (256KB – 1MB), but any valid data in the erase block must be moved and the operation is slow.
- Planes can be accessed in parallel for higher sequential throughput but, dual plane operation doubles smallest write and erase units
- The page size tends to grow because: as the technology process shrinks geometry, write times tend to increase and we access more data in parallel to compensate
- So how do we make this look like a hard disk?
  - Flash translation layer in SSD firmware

# FTL Overview



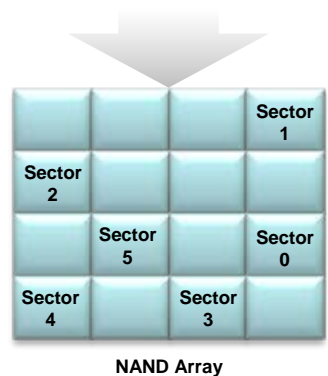
**OS Sector  
Number**

Operating systems address disk based storage by sectors



**FTL  
Mapping**

The Flash Translation Layer (FTL) “maps” the disk sectors that the Operating System is designed to address...



**Physical  
Location in  
NAND Flash**

...into physical location on the NAND array in the SSD



## FTL implications

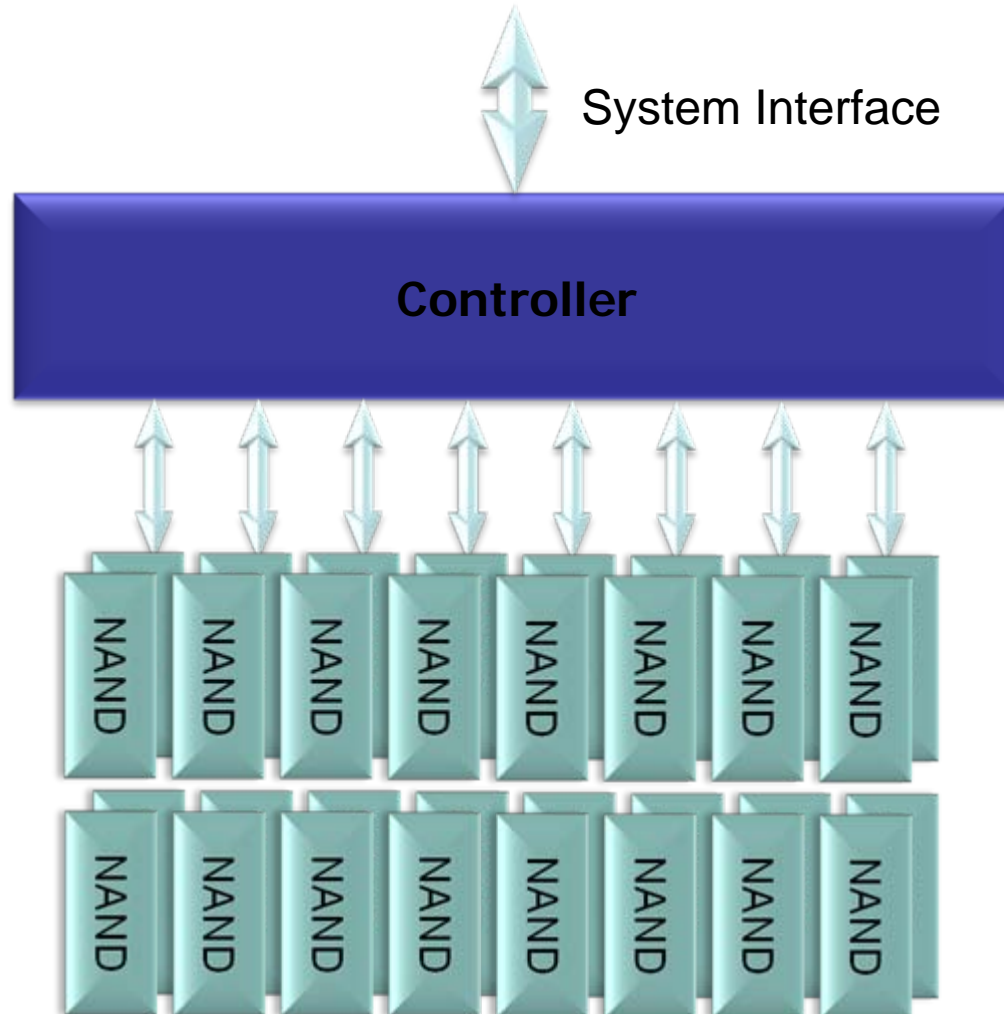
- To the OS, an SSD appears just like an HDD
- FTL's map logical LBA's to physical NAND addresses. The FTL retains this information in allocation tables.
- There are two mapping extremes for a simple FTL
  - Map contiguous LBA's to erase blocks
  - Map contiguous LBA's to pages



# FTL implications

- Mapping pages
  - Pros / cons
  
- Erase Blocks
  - Pros / cons
  
- Modern FTL's are hybrids that do some of both or map fractions of erase blocks

# SSD Performance Enablers



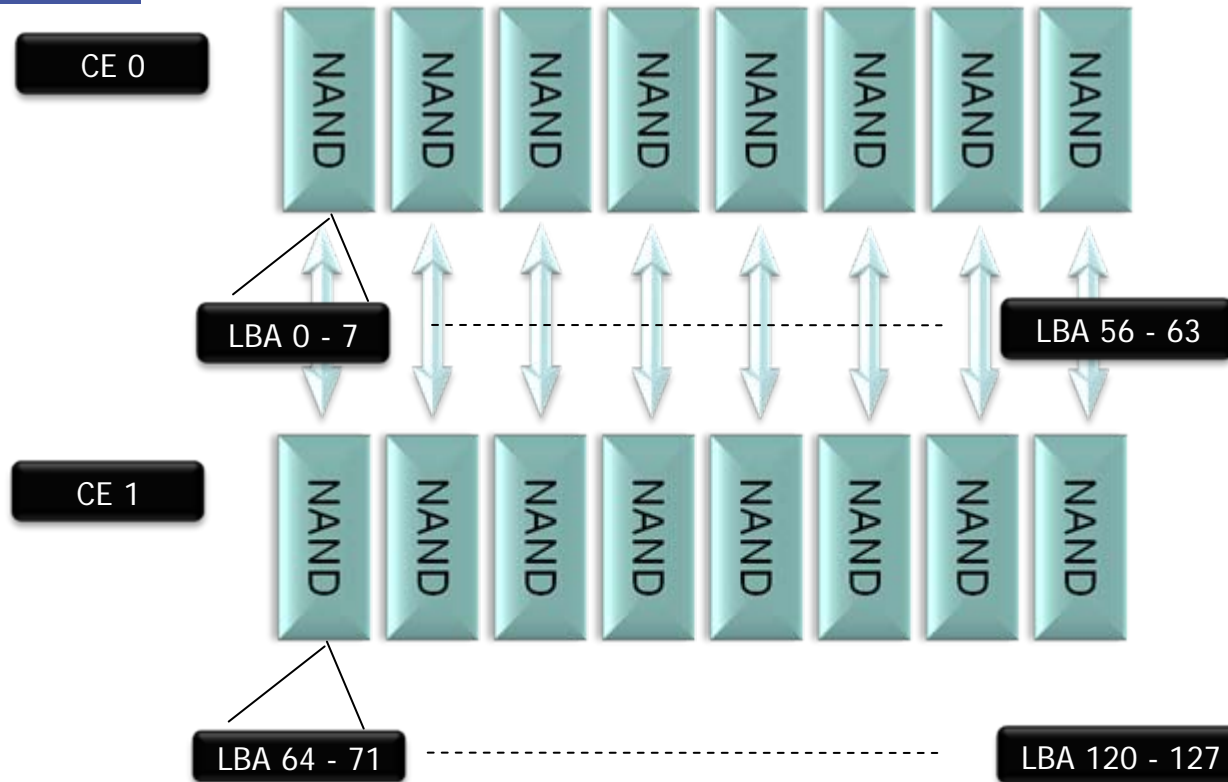
- NAND parallelism delivers high system performance



## Multiple levels of Parallelism

- Channel parallelism
  - Multiple channels
- Parallelism in the channel
- Parallelism in the NAND package
- In the die with planes
- With Parallelism comes complexity
  - ECC complication in controller
  - Complications in controller NAND sequencer
  - FTL tracking of device status

# SSD LBA to NAND Alignment



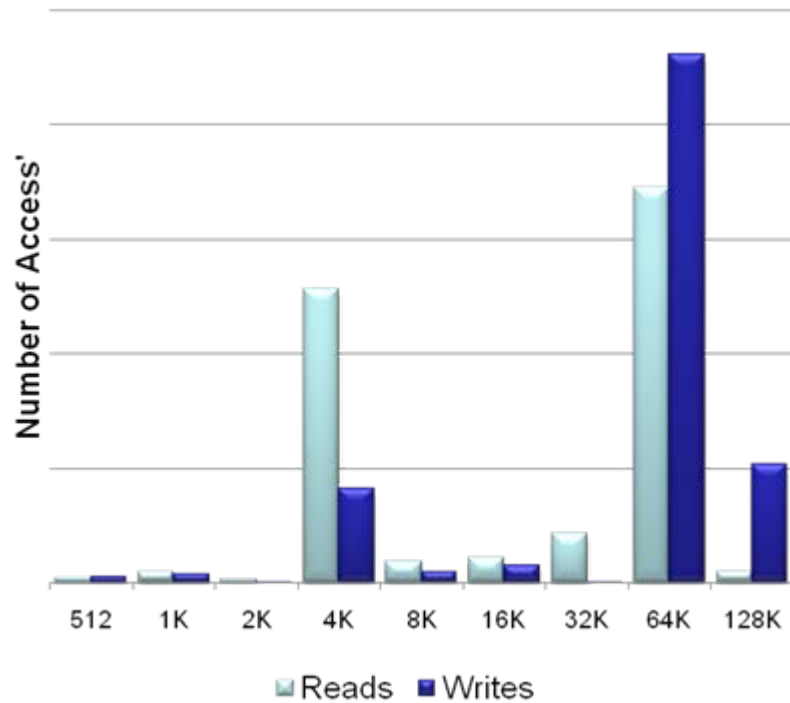
- Firmware attempts to keep LBA's evenly split among all NAND devices for optimum performance



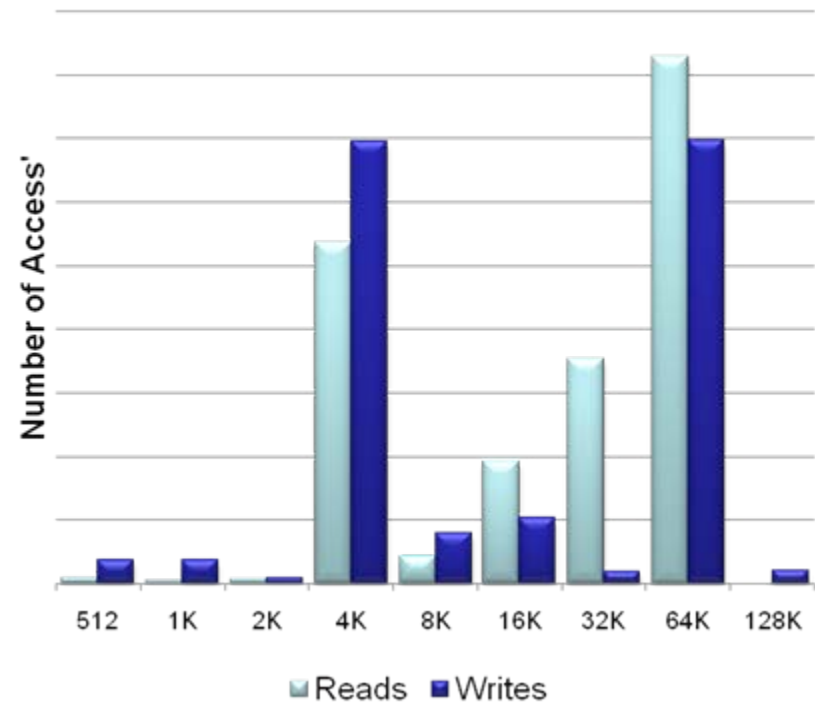
## What type of data workload are presented to the SSD?

- Server, client different workloads
- Different access patterns within the workloads
- Introduction to sample workloads.....

PCMark in Windows Vista

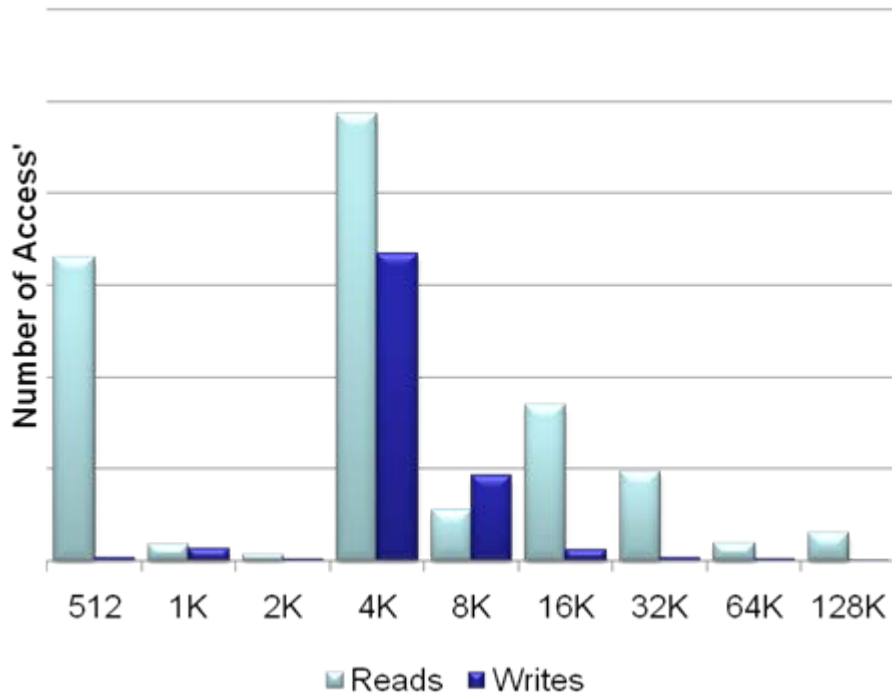


Mobile Mark in Windows XP

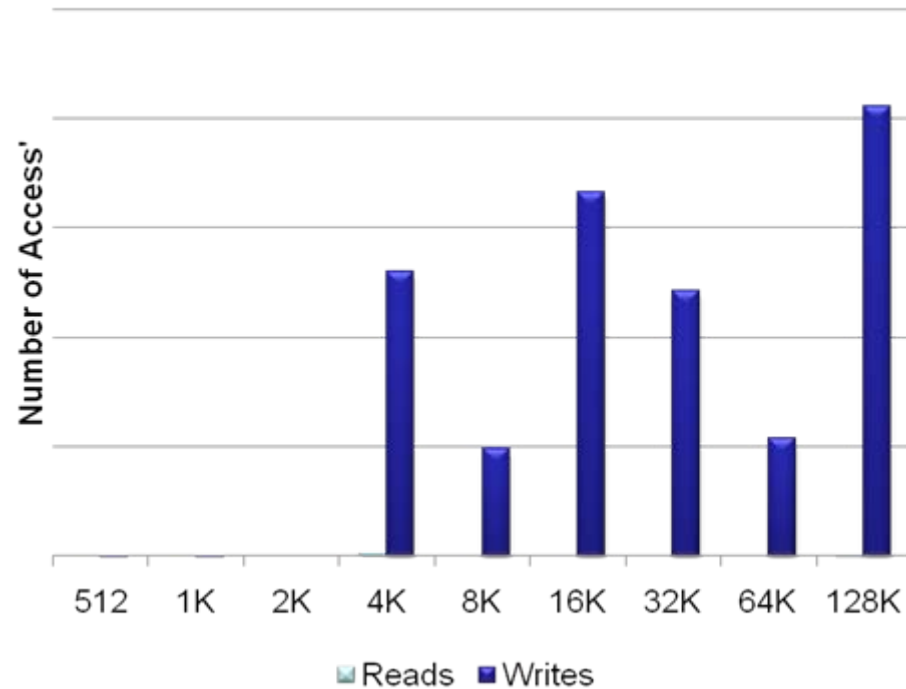


- Data workloads seen by the SSD in two different productivity benchmarks
- XP data is misaligned

### Linux Installation



### Linux File Write



- Data workloads seen by the SSD under a couple different activities

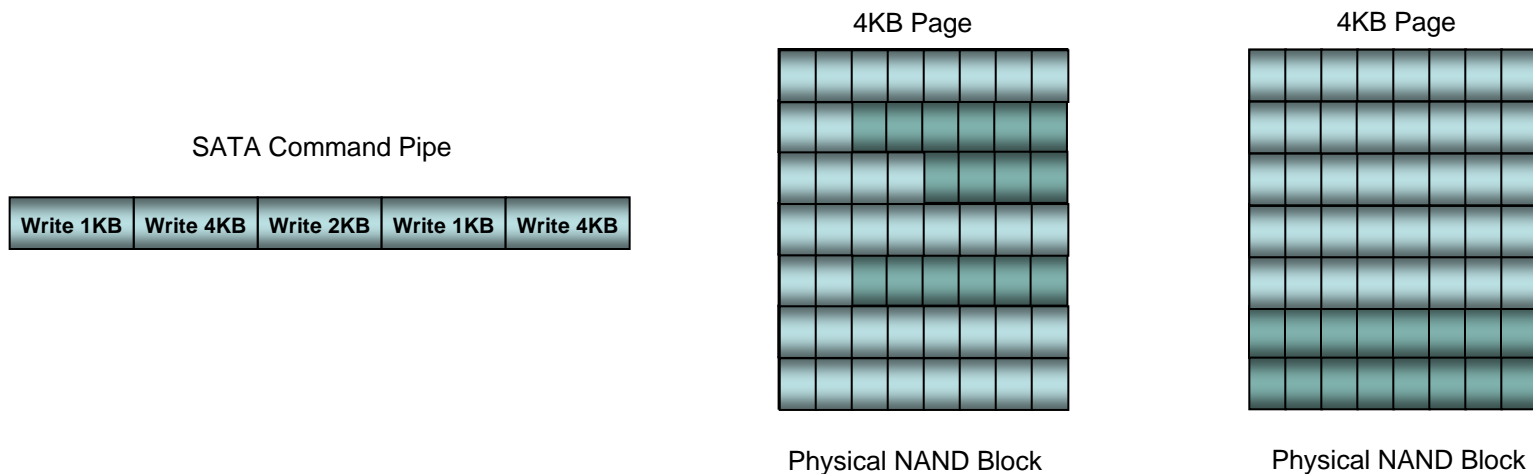
# Potential for improvement

- Potential for improvement
  - Optimize sector size
  - Trim
  - Defragmentation
  - Tagging Hot Data



# Optimize sector size

- Ideal minimum Transfer size = NAND page size
  - ▶ 2X NAND page size for dual plane operation
- Today NAND page size = 4KB
- Future NAND page size = 8KB



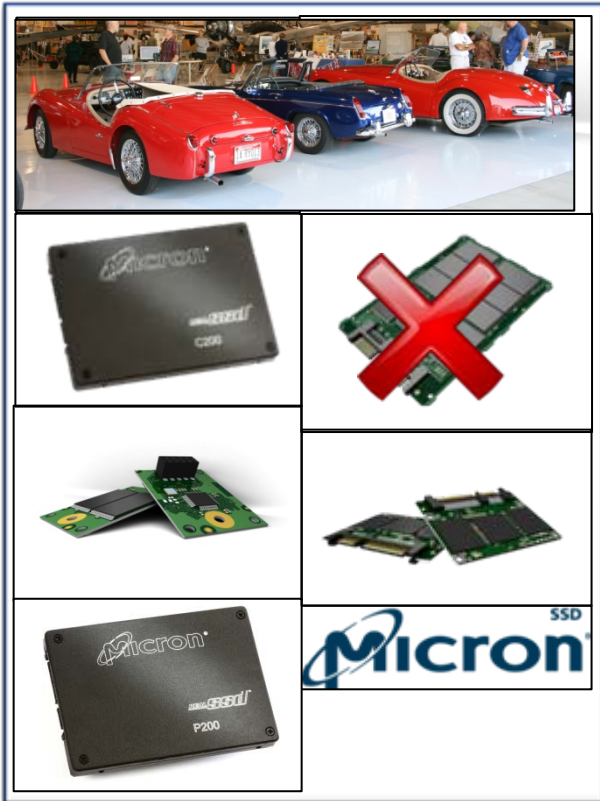


## What is Trim

- Proposed in T13 as an addition to the ATA command set
- Trim is a newly defined command that provides a mechanism for the operating system to provide information to the SSD about LBA's that are no longer in use
- Used properly has the potential to improve SSD performance in client platforms

# Without Trim

## Operating System Content

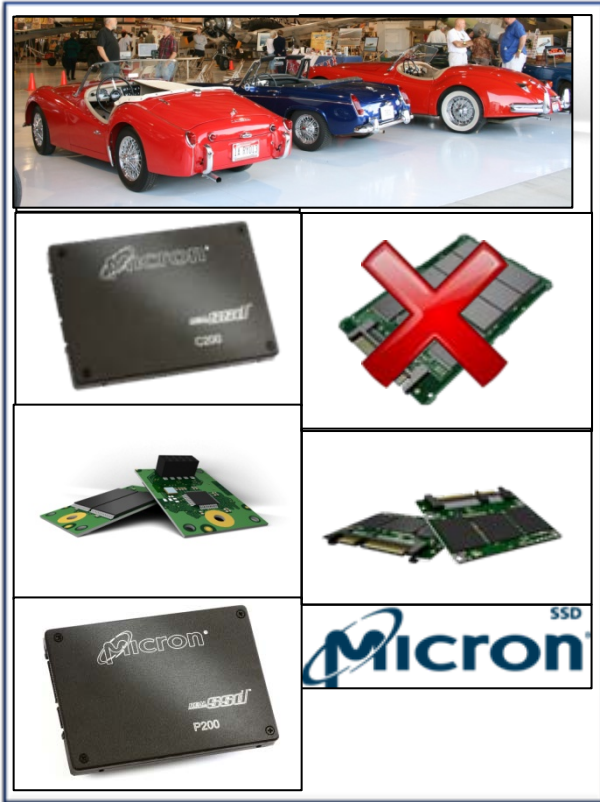


## SSD Content

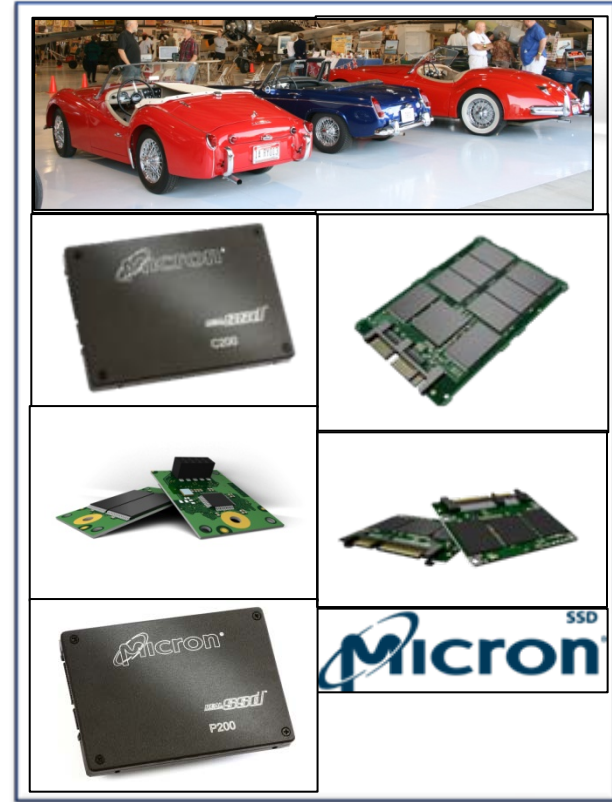


# Using Trim

## Operating System Content



## SSD Content

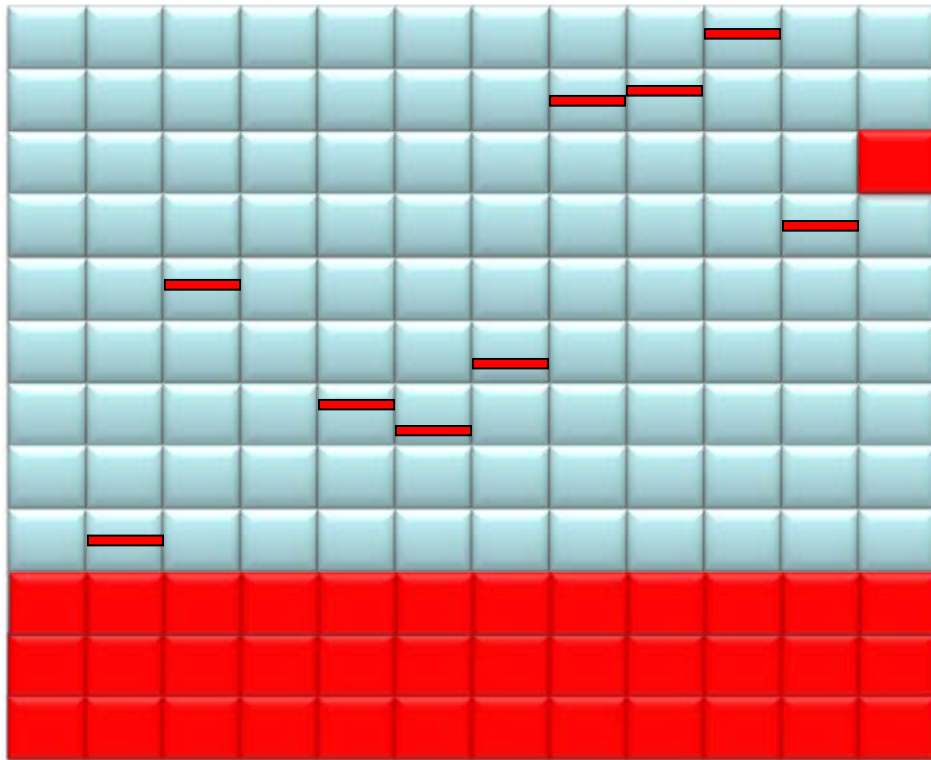


# Defragmentation

- Like an HDD, SSD's can also benefit from files being defragmented and sequential in nature
  - The difference is the frequency of performing the defragmentation and the performance impact
- Must be coupled with the use of Trim

# Defragmentation Example

Array of NAND Devices



Physical NAND Block





## Tagging Hot Data

- If the data is known to the drive to be active and changing often, it can be managed differently to improve drive performance and life
- Can be performed by the SSD firmware
  - Adds processing cycles to SSD controller and FTL complexity
- OS could notify the SSD through a special command or metadata
  - Reduces SSD controller processing load and FTL complexity



## Summary

- SSD's are delivering on the promise of higher system performance
- SSD architectures and the handling of data are different than HDD's that the current OS storage infrastructures are designed around
- Future changes to OS storage stacks can further improve SSD system performance

