ATA Trim/Delete Notification Support in Windows 7

Neal Christiansen
Principal Development Lead
Windows Core File Systems
Microsoft

What we'll cover

- Improve Windows performance on SSD (Solid State Devices) by increasing wear level effectiveness
  - Communicate deleted block information from the file system
  - Disable scheduled defrag operation

The ATA Command Set

ATA-7 Hard Drive Commands			
Read Commands	Write Commands	Power Management	Other
		CHECK POWER MODE	CONFIGURE STREAM
READ DMA	WRITE DMA	IDLE	DEVICE CONFIGURATION
READ DMA EXT	WRITE DMA EXT	IDLE IMMEDIATE	DOWNLOAD MICROCODE
	WRITE DMA FUA EXT	SLEEP	EXECUTE DEVICE DIAGNOSTIC
READ DMA QUEUED	WRITE DMA QUEUED	STANDBY	**FLUSH CACHE**
READ DMA QUEUED EXT	WRITE DMA QUEUED EXT	STANDBY IMMEDIATE	**FLUSH CACHE EXT**
	WRITE DMA QUEUED FUA EXT		IDENTIFY DEVICE
READ FPDMA QUEUED	**WRITE FPDMA QUEUED**		NOP
READ MULTIPLE	WRITE MULTIPLE		READ LOG EXT
READ MULTIPLE EXT	WRITE MULTIPLE EXT		READ NATIVE MAX ADDRESS
	WRITE MULTIPLE FUA EXT		READ NATIVE MAX ADDRESS EXT
READ SECTOR(S)	WRITE SECTOR(S)		SERVICE
READ SECTOR(S) EXT	WRITE SECTOR(S) EXT		SET FEATURES
READ STREAM DMA EXT	WRITE STREAM DMA EXT		SET MAX ADDRESS
READ STREAM EXT	WRITE STREAM EXT		SET MAX ADDRESS EXT
READ VERIFY SECTOR(S)			SET MULTIPLE MODE
READ VERIFY SECTOR(S) EXT			SMART
			WRITE LOG EXT

Most of the IO traffic from Windows : Queued IO and durability control

ATA Storage Interface

□ Notice there are lots of flavors of Read and Write

□ What was missing :

"I no longer need the data in this sector now"

□ "Trim" addresses this

Why Trim is Important?

- Trim is important to SSDs because:

  - They spend a lot of resources (processor time and internal storage) wear-leveling

  - Free blocks are used to make wear-leveling algorithms more effective
    - The more free blocks, the easier it is to wear level

Why Trim is Important? (cont'd)

- Some SSDs have tried to work around lack of Trim by trying to interpret and understand the file system format such that they can free blocks when a file is marked deleted.
    - Works for FAT since it's a published spec
    - Does not work well for NTFS
        - NTFS structures are much more complex than FAT
        - NTFS structures are not published and will change in the future

Trim Implementation in Windows

- Trim is only implemented by the NTFS File System
- How Trim is Implemented:
    - Application calls DeleteFile
    - File system metadata is updated and written to device
    - Metadata is flushed and checkpoint record written to log
    - Device is notified that blocks are no longer in use
    - Blocks are made marked as free in volume bitmap

Trim Semantics

- ❑ "A write will be issued before another read is issued for this range"... If a read comes down before the first write after a Trim, the data returned can be :
  - ❑ The last data written to the range
  - ❑ A fixed pattern (zeros or ones)
  - ❑ Can not be other data pages on the device
    - ❑ Trim should not expose security hole

Trim Semantics (cont'd)

- Trim differs from "Secure Delete" command
  - Secure Delete means it must be done
  - Trim is just an advisory… don't take the perf hit of flushing zeroes out

DRAT within Trim

- ❑ "DRAT" is Deterministic Read After Trim
  - ❑ Trim command can be done synchronously or in background
  - ❑ If DRAT is set, then results for read of trimmed area must not change after the first read after Trim.
  - ❑ If DRAT is not set, then results for read of trimmed area are non deterministic
    - ❑ Could be old data or all zeros or all ones

- ❑ DRAT is currently unused by Windows 7 because NTFS never lets applications read the trimmed data

Trim Methods

- Two ATA methods to send down a Trim
  - Both use Data Set Management (DSM) to set attributes
    - Trim in this case is the attribute that is set on the data

  - ATA DSM Command with Trim

  - FPDMA Send with DSM Trim

ATA DSM Trim

- Method #1 : ATA DSM Trim
  - One TRIM command potentially has multiple ranges to be trimmed
  - The list of ranges is sent in the data transfer phase of the request
  - Disrupts NCQ queuing

Identify for **ATA DSM with Trim**

DSM Command with Trim

LBA and length to be trimmed. Could be up to 64 ranges present per block. Up to 16K blocks per request.

FPDMA Send with DSM Trim

- Method #2 : FPDMA Send with DSM Trim
  - Does not disrupt NCQ queuing
  - Not currently used by Windows 7 inbox ATA driver
  - May be supported in future version of Windows inbox ATA driver

IDENTIFY

Read Log Pages Supported

Read Log Page DSM

FPDMA Send with DSM Trim

Disabling Trim

- Trim is always sent by NTFS
  - Registry key exists to disable NTFS from sending Trims:
    - \HKLM\System\CurrentControlSet\Control\ FileSystem\DisableDeleteNotification set to 1
    - Useful in situations where data recovery is more important than wear leveling:
      - Undelete that do not use a filter architecture
      - Unformat tools

Future Work in Trim

- Trim is ATA version coming out of T13 committee
  - ACS-2 (document ID d2015r1a)
  - FPDMA Send with DSM Trim may be supported

- Unmap is SCSI version coming out of T10 committee
  - Proposal still being negotiated

Future Work in Trim

- Difference in semantics between Unmap & Trim
  - Unmap: Underlying storage may be repurposed
  - Trim: Underlying storage is still present after trim. Future write won't fail due to space allocation
- Windows inbox storage drivers may support Unmap in future versions
- Other file systems such as FAT32 and ExFAT may support Trim in future versions of Windows

Defrag on SSD

- Flash devices have limited number of erase cycles
- Rewriting blocks to get them into adjacent LBA ranges can lead to greater FLASH burnout rate
- Two things required:
  - Detect underlying storage is FLASH device
  - Disable defrag program from "optimizing" the device

Defrag on SSD

- Storage device is not scheduled for defrag by default under the following circumstances:
    - SSD reports rotation speed of 1 or
    - The WinSat performance shows that the device performs over 8 MB / sec in the random read disk test
- User can still manually configure defrag for device

Identify for No Spindle Speed

State Listing ☐ Apply Filter | Filtering... | Columns...

Store #	Timestamp	L1 I->T Data	L1 I->T Count	L1 I->T Description
124,387	16 ns	2780EC00	0	FIS 27 - Reg Host->Device
				Features = 0x00=Reserved
				Command = 0xEC=IDENTIFY DRIVE
				C = 1 - Command Register Updated
				PM Port = 0x0 - Default Port
124,388	12 ns	000000A0	1	Dev/Head = 0xA0
				Cyl High = 0x00
				Cyl Low = 0x00
				Sec Num = 0x00
124,389	12 ns	00000000	2	Features(exp) = 0x00
				Cyl High(exp) = 0x00
				Cyl Low(exp) = 0x00
				Sec Num(exp) = 0x00
124,390	16 ns	01000000	3	Control = 0x00
				Sec Cnt(exp) = 0x00
				Sec Cnt = 0x01
124,391	12 ns	00000000	4	Reserved
124,392	12 ns	2FC1492A	5	
124,393	16 ns			CRC - Good
		7CB5D5D5		SATA_EOF

L1 I->T | L2 I->T | L3 I->T | L4 I->T
L1 T->I | L2 T->I | L3 T->I | L4 T->I

Reverse Byte Order

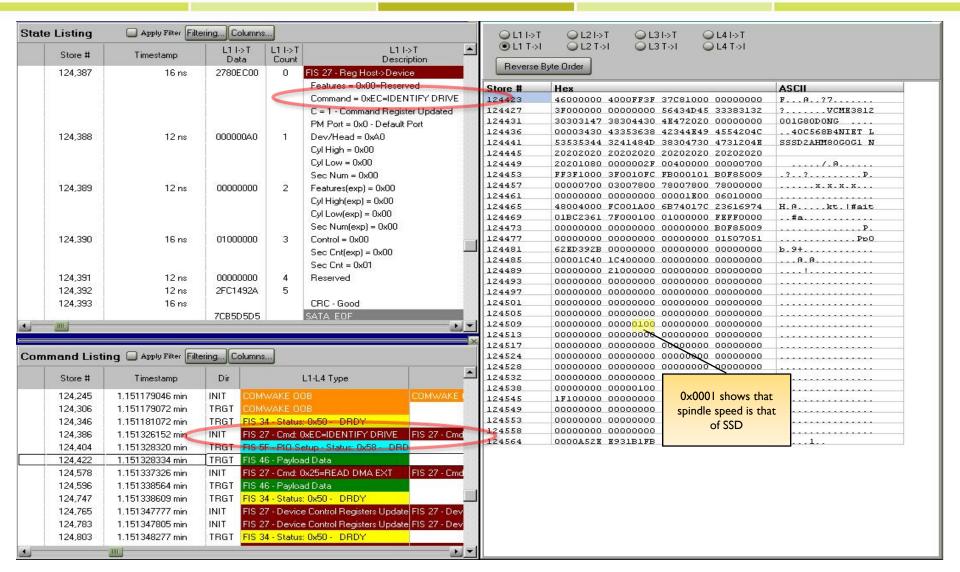| Store #	Hex	ASCII
124423	46000000 4000FF3F 37C81000 00000000	F...@..??.......
124427	3F000000 00000000 56434D45 33383132	?.......VCME3812
124431	30303147 38304430 4E472020 00000000	001G80D0NG ....
124436	00003430 43353638 42344E49 4554204C	..40C568B4NIET L
124441	53535344 3241484D 38304730 4731204E	SSSD2AHM80G0G1 N
124445	20202020 20202020 20202020 20202020	
124449	20201080 0000002F 00400000 00000700	......./.@......
124453	FF3F1000 3F0010FC FB000101 B0F85009	.?..?.........P.
124457	00000700 03007800 78007800 78000000	......x.x.x.x...
124461	00000000 00000000 00001E00 06010000	....
124465	48004000 FC001A00 6B74017C 23616974	H.@.....kt.|#ait
124469	01BC2361 7F000100 01000000 FEFF0000	..#a............
124473	00000000 00000000 00000000 B0F85009	..............P.
124477	00000000 00000000 00000000 01507051	.............PpQ
124481	62ED392B 00000000 00000000 00000000	b.9+............
124485	00001C40 1C400000 00000000 00000000	...@.@.........
124489	00000000 21000000 00000000 00000000	....!...........
124493	00000000 00000000 00000000 00000000	....
124497	00000000 00000000 00000000 00000000	....
124501	00000000 00000000 00000000 00000000	....
124505	00000000 00000000 00000000 00000000	....
124509	00000000 00000100 00000000 00000000	....
124513	00000000 00000000 00000000 00000000	....
124517	00000000 00000000 00000000 00000000	....
124524	00000000 00000000 00000000 00000000	....
124528	00000000 00000000 00000000 00000000	....
124532	00000000 00000000 00000000 00000000	....
124538	00000000 00000100		....
124545	1F100000 00000000		....
124549	00000000 00000000		....
124553	00000000 00000000		....
124558	00000000 00000000		....
124564	0000A52E E931B1FB		....1..

0x0001 shows that spindle speed is that of SSD

Command Listing ☐ Apply Filter | Filtering... | Columns...

Store #	Timestamp	Dir	L1-L4 Type	
124,245	1.151179046 min	INIT	COMWAKE OOB	COMWAKE
124,306	1.151179072 min	TRGT	COMWAKE OOB	
124,346	1.151181072 min	TRGT	FIS 34 - Status: 0x50 - DRDY	
124,386	1.151326152 min	INIT	FIS 27 - Cmd: 0xEC=IDENTIFY DRIVE	FIS 27 - Cmd
124,404	1.151328320 min	TRGT	FIS 5F - PIO Setup - Status: 0x58 - DRD	
124,422	1.151328334 min	TRGT	FIS 46 - Payload Data	
124,578	1.151337326 min	INIT	FIS 27 - Cmd: 0x25=READ DMA EXT	FIS 27 - Cmd
124,596	1.151338564 min	TRGT	FIS 46 - Payload Data	
124,747	1.151338609 min	TRGT	FIS 34 - Status: 0x50 - DRDY	
124,765	1.151347777 min	INIT	FIS 27 - Device Control Registers Update	FIS 27 - Dev
124,783	1.151347805 min	INIT	FIS 27 - Device Control Registers Update	FIS 27 - Dev
124,803	1.151348277 min	TRGT	FIS 34 - Status: 0x50 - DRDY	

Summary

- We've discussed the following:
    - Trim semantics and how a device can use it
    - How a device can report back that it wants to receive Trim information
    - How a device can report back that it should not be defragmented by default
- Windows 7 works well with SSDs
    - More work to be done in future versions of Windows

Questions?

Separator Slide