



Longhorn Large Sector Size Support

Anuraag Tiwari
Program Manager
Core File System

anuraagt@microsoft.com

Agenda

- Historical OS Support for Large Sector Size Drives
- A Brief Overview of the OS Disk I/O Components
- Longhorn Support for Large Sector Size Drives
- Legacy OSes vs. Longhorn Comparison
- Dates and Schedule for Upcoming Work
- Q&A

Historical OS Support for Large Sector Size Drives

- **Windows 2000, XP and 2003 users could use data disks with 512-byte multiple sectors**
 - Several customer scenarios for disks with such characteristics, including SCSI and P.D. optical media
 - There was no OS software RAID support
- **Users could not boot from large sector size drives**
 - Components involved in boot (BIOS, boot code, file system boot code, loader) did not support such devices

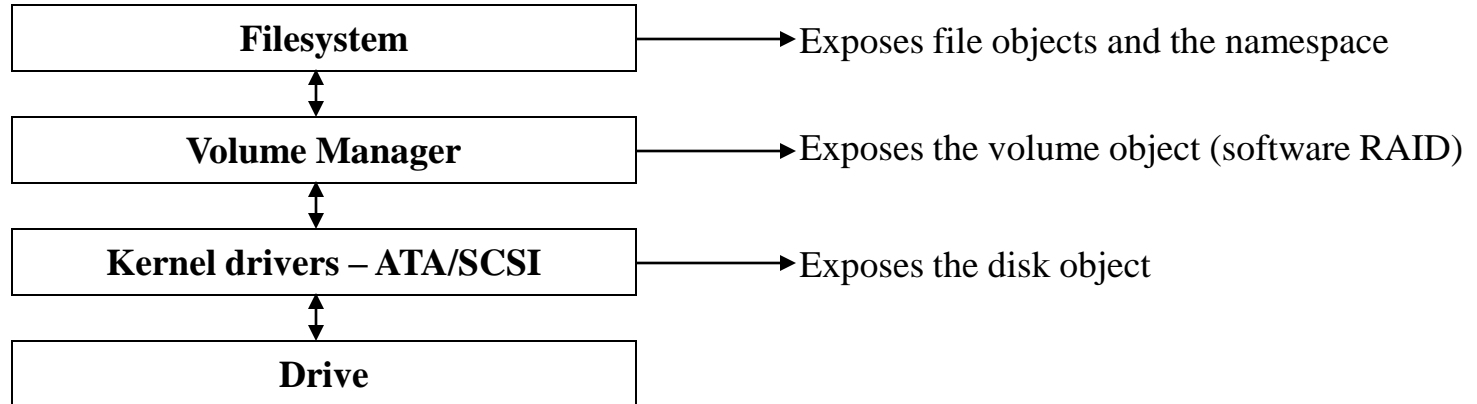
A Brief Overview of the OS Disk I/O Components

■ Boot Sequence

- ❑ BIOS initializes itself
 - Locates disk at 0x80
 - Finds first sector of this disk, and starts executing
- ❑ MBR Boot Code
 - Boot code exists to read the partition table, find the first sector of the active partition, and start executing it
- ❑ File System Boot Code
 - Only job of the file system boot code is to find the NTLoader/Boot Manager
- ❑ Windows Boot Manager
 - The Windows Boot Manager will start loading the operating system

A Brief Overview of the OS Disk I/O Components

■ I/O Path Post-Boot



Longhorn Support for Large Sector Size Drives

■ Two key scenarios

- ❑ Large Sector Size Drive is used as a Data Disk with Windows Longhorn
- ❑ Large Sector Size Drive is used as a Boot Disk with Windows Longhorn

Longhorn Support for Large Sector Size Drives

■ Large Sector Size Drive is used as a Data Disk with Windows Longhorn

□ Disk driver

- The driver stack issues I/O in sector size multiples, and at sector size boundaries
- We are adding sensitivity to an alignment offset

□ Volume manager

- The volume manager issues I/O in sector size multiples, and at sector size boundaries
- Modify partition alignment scheme in the volume to allow for default creation of partitions at 64/128/256/1024K byte intervals

□ File system

- The file system issues I/O in sector size multiples, and at sector size boundaries
- As currently designed, NTFS will align files at the cluster size for the partition (which is 4K bytes by default)

□ Note

- Applications that issue unbuffered I/O against disks, volumes or files need to ensure that the I/O is aligned on the queried sector size
- Applications which use log files may have to be fixed to understand larger sector size if they are logging based on a smaller sector size

Longhorn Support for Large Sector Size Drives

■ Large Sector Size Drive is used as a Boot Disk with Windows Longhorn

□ Extended Int 13 BIOS

- Uses LBA addressing and requires reporting Large Sector Size size for SCSI controllers
- BIOS work needs to be done for ATA disks to read the ATA command for sector size
- BIOS drivers for option ROMs will need to be updated

□ Extended Firmware Interface

- EFI specifies the use of LBA addressing and reports sector size through the Block I/O Interface for Block Device → Block Size

□ Longhorn File System Boot Code/Boot Manager correctly issue I/O calls to disk based on sector size reported by Extended Int 13/EFI BIOS

Legacy OSes vs. Longhorn Comparison

	Win2k/XP/2003		Longhorn	
512B Logical	Data	✓	Data	✓
512B Physical	Boot	✓	Boot	✓
512 B Logical	Data	✓	Data	✓
Large Sector Phys.	Boot	✓	Boot	✓
Large Sector Log.	Data	✓	Data	✓
Large Sector Phys.	Boot	✗	Boot	✓
	SCSI only – ATA not supported Software RAID not supported		SCSI/ATA supported Software RAID supported	

Dates and Schedule for Upcoming Work

■ Volume Manager changes

- Completed as of Longhorn Beta 1 (summer 2005)

■ File system changes

- In testing, targeted for Longhorn Beta 2 (Q4 2005)

■ Windows OS Boot Manager and Setup

- Completed as of Longhorn Beta 1 (summer 2005)

■ ATA Driver updates for sector size

- In design phase, targeted for Longhorn Beta 2 (Q4 2005)

Q&A

