A unified object oriented storage architecture

Andy Hospodor, Ethan Miller, Rekha Pitchumani,Yangwook Kang, Darrell Long
University of California, Santa Cruz
Ahmed Amer, Santa Clara University
Yulai Xie, Huazhong University of Science and Technology
hospodor@soe.ucsc.edu, elm@soe.ucsc.edu, rekhap@soe.ucsc.edu,

Definition of Virtualization

Virtualization is the art of making modern storage devices (Flash, Phase Change Memory, Solid State Disk, Shingled Disk) look like a 1986 SCSI Disk Drives to preserve the integrity of device drivers and file systems.

Basic Concept

We introduce a unified storage architecture based upon object oriented storage devices (OOSD) that obfuscates the physical implementation issues of the underlying hardware. Metadata is decoupled from the data to permit the management of objects rather than blocks. Storage objects are managed through associated methods over a wide range of storage devices, including Magnetic Disk, Optical Disk, Storage Class Memories, Tape, Flash, and anticipates new storage devices such as Shingled Disk.

Previous Work

Previous work focussed on Disk Drives, the largest segment of the storage market, rather than the storage architecture. Specific OO commands for disk drives, rather than methods for storage architectures, were created as extensions of the 1986 SCSI command set.

References

[1] "The Object-Based Storage Device Commands - 2 (OSD-2) Rev 05a," ANSI INCITS 458-2011, 16 January 2009.

[2] "Inteĺ High Performance Solid State Drive - Advantages of TRIM". Intel Corporation, Santa Clara, CA,14 Sept 2010.

[3] "Q200 Series Intelligent Disk Drives", Quantum Corporation, Milpitas, CA, Nov 1986.

[4] Riedel, E., Faloutsos, C., Gibson, G.A. and Nagle, D.F. "Active Disks for Large-Scale Data Processing". IEEE Computer, June 2001.

[5] "Small Computer System Interface (SCSI)", ANSI X3.131-1986.

Acknowledgements

The authors received funding for this research from Center for Research in Intelligent Storage (CRIS).

Methods of an Object Oriented Storage Architecture

Method	***Description***
new storage = StorageObject(size)	Instantiate new Storage Objects
new moreStorage = StorageObject(size)	
metadata = storage.finger	Obtain metadata from a Storage Object
buffer = storage.read	Retreive data from Storage Object into memory
storage.write = buffer	Store data from memory onto Storage Object
storage.append = buffer	Extend Storage Object with additional data
storage.find(regExp)	Search a Storage Object
sortedStorage = storage.sort(keyValue)	Sort a Storage Object by key value pair
moreStorage = storage.replicate	Replicate a Storage Object

Motivation

The authors believe that an OO Storage Architecture is best implemented by scrapping the existing infrastructure of file systems and devices drivers. While this may be considered blasphemy within the data storage industry, not doing it has the effect of locking us into storage architectures of the 1980s.

Modern storage devices have microprocessor based control systems that decode commands, position read/write elements, translate logical to physical addresses, transfer data and report status. However, today's disk drives are block level devices. Flash Translation Layers make flash memory look like a SCSI disk, while Hardware Abstraction Layers make USB memory sticks, look like SCSI disks and RAID subsystems make groups of SATA disks also appear as SCSI disks .

Modern microprocessors are capable of managing a storage address space as a collection of objects, rather than blocks. Such an OO storage device would map its objects and access them using specified methods, as shown in Table 1. This approach offers design flexibility and permits efficient allocation and placement policies based up the device's intimate knowledge of its own physical geometry. Legacy support could be as simple as using a Logical Block Address(LBAs) in place of an object name, so, storage.read(12345) would read LBA 12345 in the same manner a SCSI or ATA storage device would read it today.

Conclusion

Object Oriented Storage Architectures have the promise of integrating new storage technologies and new features without the continuos modification of operating systems, file systems and device drivers. As the industry embraces Flash, Phase Change and other other non-volative memories, the ability to integrate and bring products to market quickly offers a distinct competitive advantage - one that Active Disk never had.