
IBM
Tivoli Storage, IBM Software Group
Understanding Disk Storage
in Tivoli Storage Manager
Dave Cannon
Tivoli Storage Manager Architect
Oxford University TSM Symposium
September 2005
© 2005 IBM Corporation

Disclaimer

- Unless otherwise noted, functions and behavior described in this presentation apply to Tivoli Storage Manager 5.3, but may differ in past or future product levels
- Description of potential future product enhancements does not constitute a commitment to deliver the described enhancements or to do so in a particular timeframe

© 2005 IBM Corporation

Agenda

- Background
- Random- and sequential-access disk in TSM
- Special considerations for sequential-access disk
- Potential future enhancements for disk storage

© 2005 IBM Corporation

Disk vs. Tape

- Potential disk advantages
  - Faster access by avoiding delays for tape mounts and positioning
  - Reduced management cost (no tape handling)
  - Reliability (RAID, no tape robotics or media failures)
- Potential tape advantages
  - Removability/portability for off-site storage (disaster recovery)
  - High-speed data transfer for large objects
  - Cost effectiveness (especially for long-term, off-site archiving)
  - Reliability (less susceptible to hardware failure or file system corruption)
- Tiered approach with copies on offsite tape exploits strengths of disk and tape

© 2005 IBM Corporation

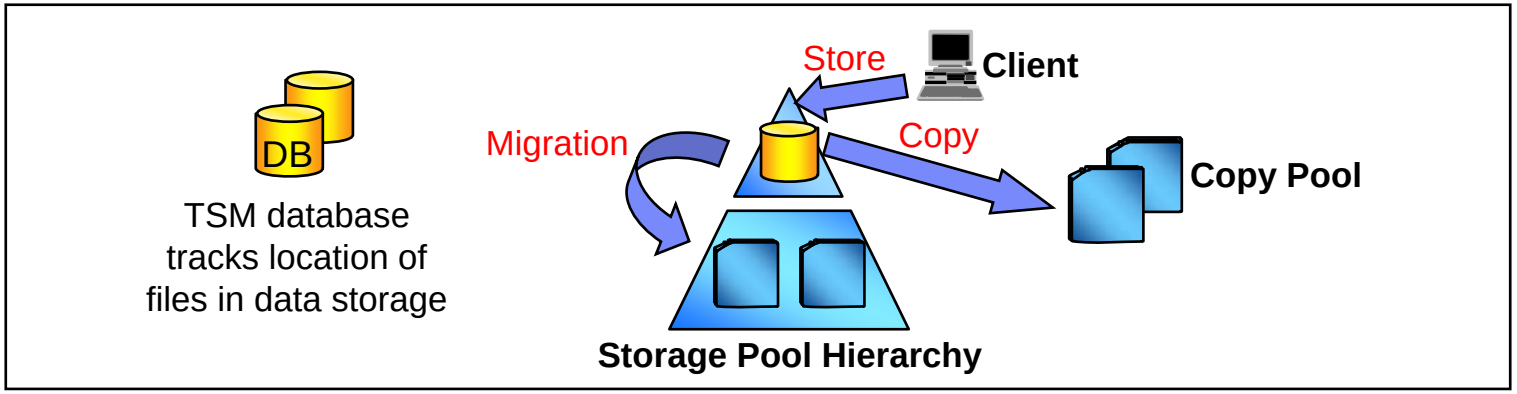
Industry Trend Toward Increasing Use of Disk

- Lower cost of disk storage (ATA, SATA)
- Promotion of disk-based appliances and solutions (EMC, Network Appliance)
- Virtual tape library (VTL) products comprised of preconfigured disk systems that emulate tape
- Disk-based technologies
  - Replication
  - Snapshots
  - Continuous data protection (CDP)

© 2005 IBM Corporation

TSM is Designed for Disk in a Storage Hierarchy

- Disk has been an integral part of the TSM data storage hierarchy since 1993
- Virtualization of disk volumes in a storage pool allows objects to be stored across multiple volumes and file systems
- Automatic, policy-based provisioning of disk storage pool space and allocation of that space during store operations
- Automatic, policy-based migration to tape or other media types in tiered hierarchy
- Incremental backup of objects from primary disk pool to tape copy pool for availability or offsite vaulting
- Objects automatically accessed in copy pool if not available in primary storage pool



© 2005 IBM Corporation

Disk Usage Trend in TSM

Traditional Disk Usage

- LAN-based data transfer between client and disk storage
- Data initially stored on disk to allow concurrent client backups without tape delays
- Backup from disk to tape copy storage pool for availability and disaster recovery
- Most data migrated to tape within 24 hours

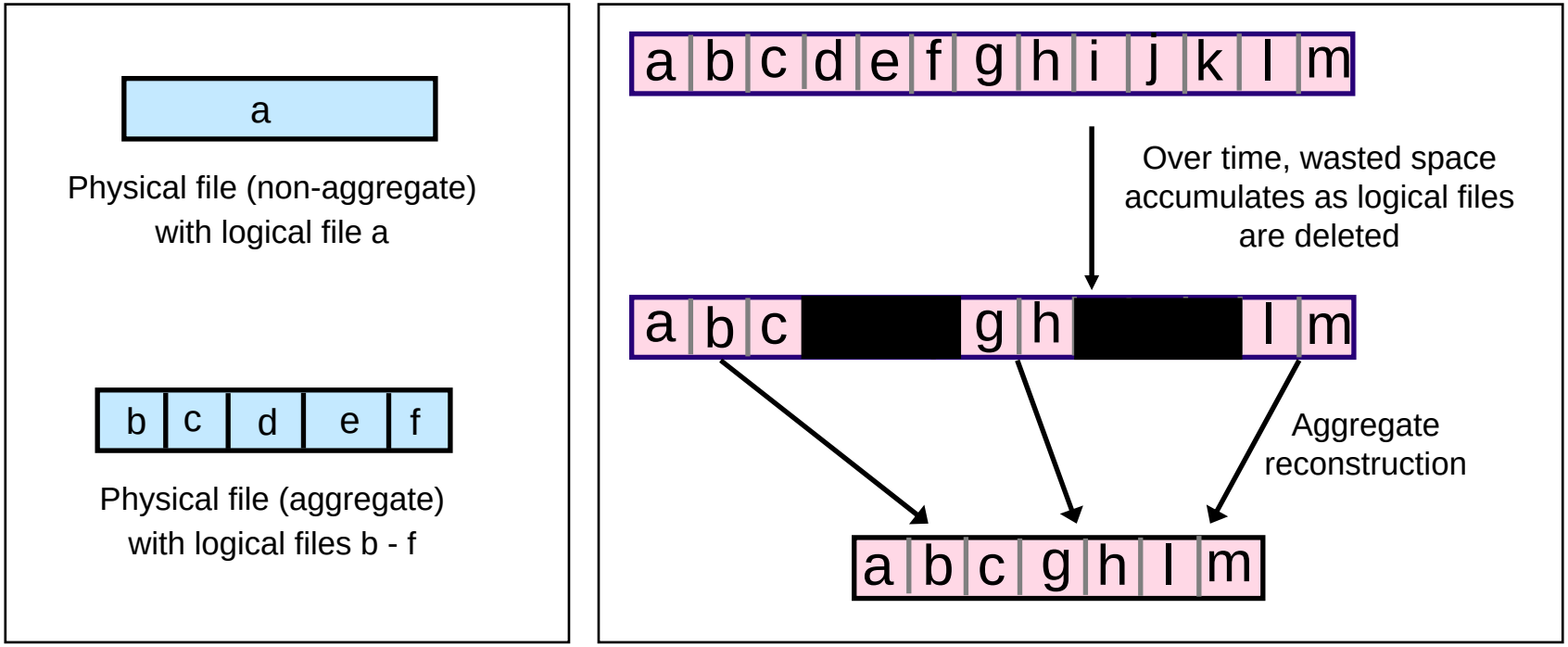
Emerging Disk Usage

- Increasing interest in LAN-free transfer between client and disk
- Data initially stored on disk to allow concurrent client backups without tape delays
- Backup from disk to tape copy storage pool (may be main reason for tape)
- Data may be stored on disk indefinitely

© 2005 IBM Corporation

A Detour on File Aggregation

- TSM server groups client objects into aggregates during backup or archive
- Information about individual client objects is maintained and used for certain operations (e.g., deletion, retrieval)
- For internal data transfer operations (migration, storage pool backup), entire aggregate is processed as a single entity for greatly improved performance



© 2005 IBM Corporation

Agenda

- Background
- ➢ Random- and sequential-access disk in TSM
- Special considerations for sequential-access disk
- Potential future enhancements for disk storage

© 2005 IBM Corporation

Overview of Random- and Sequential-Access Disk

- TSM supports two methods for storing and accessing data on magnetic disk
  - Random-access storage pools (also known as DISK pools)
  - Sequential-access storage pools (also known as FILE pools)
- Random- and sequential-access disk pools differ in how TSM manages disk storage and the operations that are supported
- TSM development views sequential-access disk as strategic
  - Current functions on random-access disk supported for the foreseeable future
  - Future product enhancements involving disk storage may be offered only for sequential-access disk

© 2005 IBM Corporation

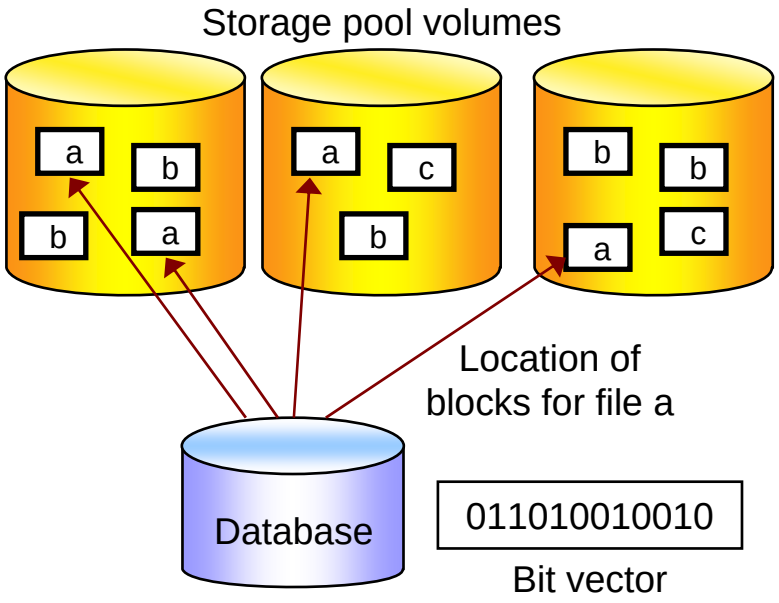
Basics of Random- and Sequential-Access Disk

	Random-Access Disk	**Sequential-Access Disk**
Storage pool definition	Predefined device class DISK	Device class with device type of FILE
Pools spanning multiple file systems	Supported	Supported
Storage pool volumes	Files or raw logical volumes	Files
Volume creation	▪Define Volume command ▪Space trigger	▪Define Volume command ▪Space trigger ▪Scratch volumes
TSM caching	Supported	Not supported
Use for copy storage pool	Not supported	Supported

© 2005 IBM Corporation

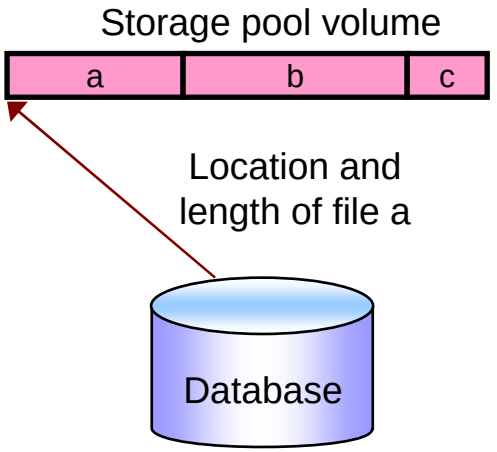
Space Allocation and Tracking

Random Access



- Space allocated in randomly located 4KB blocks
- TSM server tracks volumes and blocks on which each file is stored
- Bit vector in TSM database tracks allocated and free blocks for each volume
- Space allocation and tracking requires overhead
- May not scale well for extremely large files

Sequential Access +

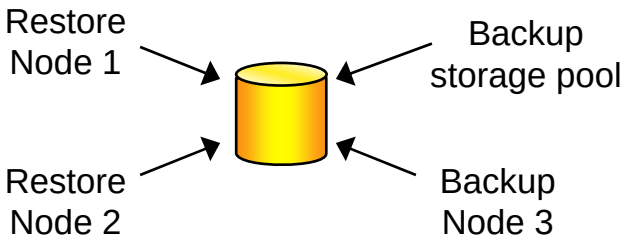


- Files written sequentially in FILE volume
- TSM database only tracks volume and offset at which each file is stored
- TSM has less overhead for space allocation and tracking

© 2005 IBM Corporation

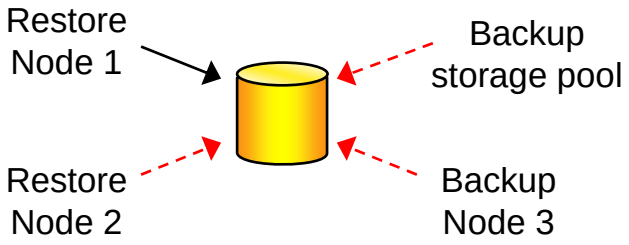
Concurrent Volume Access

Random Access +



- Multiple TSM sessions or processes can concurrently use the same disk volume
- However, individual I/O operations for each volume are serialized

Sequential Access



- Disk volume is locked by a single process or session using that volume
- Other operations cannot access the volume until the lock is released, usually when the locking operation has completed all work on the volume

To avoid volume contention, sequential-access volume sizes should be much smaller than random-access volume sizes

© 2005 IBM Corporation

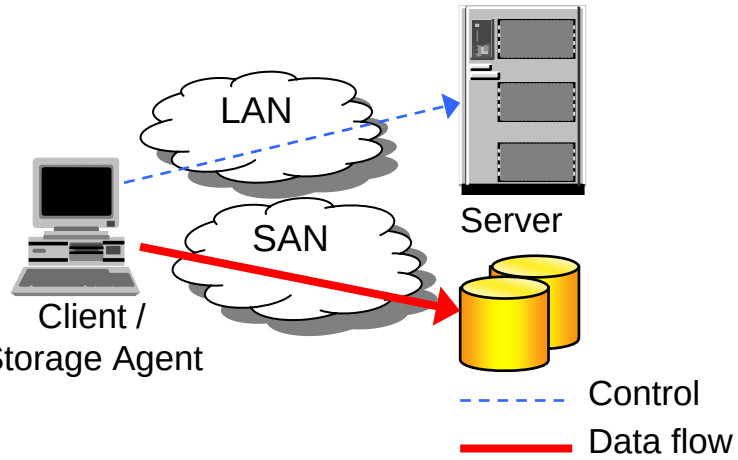
LAN-free Backup/Restore

Random Access

Not supported

Sequential Access +

- Supported using either of the following to control shared access to sequential disk volumes
  - SANergy
  - Storage pool volumes on SAN FS (supported in TSM 5.3.1)
- Reduces CPU cycles on TSM server and moves network traffic from LAN to SAN

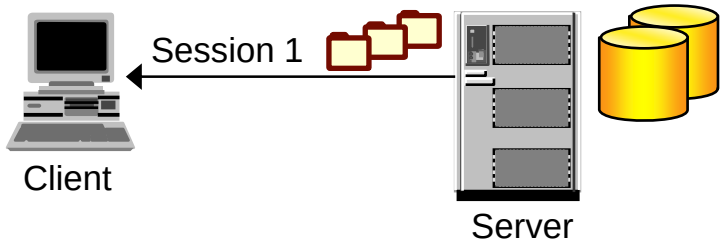


Alternative approach for LAN-free would be a virtual tape library (VTL) appliance

© 2005 IBM Corporation

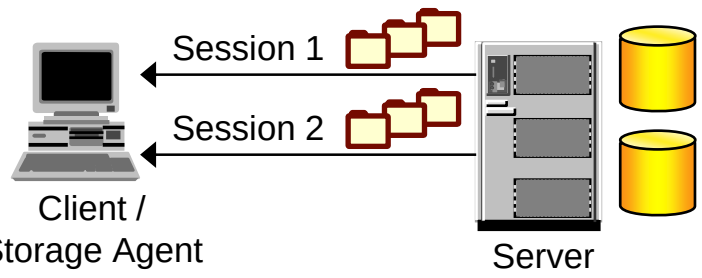
Multi-Session Restore

Random Access



Multi-session restore allows only one session for all random-access disk volumes

Sequential Access +



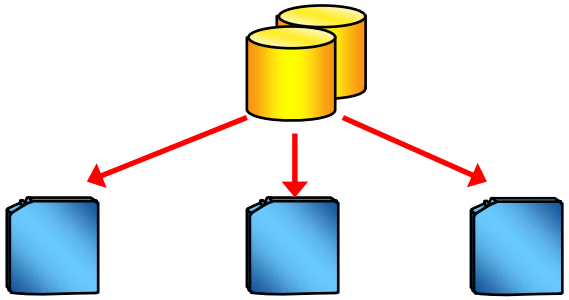
Multi-session restore allows one session per sequential-access volume

Multi-session restore is performed only for no-query restore (NQR) operations

© 2005 IBM Corporation

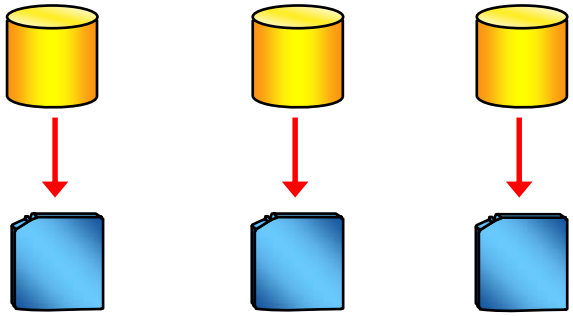
Migration

Random Access



- High/low migration thresholds based on percentage occupancy of the pool
- If node is grouped and target pool is collocated by group, parallel migration processes each work on a different group
- Otherwise, parallel migration processes each work on a different node
- Optimized for transfer by node and file space, making it an ideal intermediate buffer for transfer from non-collocated tape to collocated tape (e.g., restore from copy pool to collocated tape pool)

Sequential Access

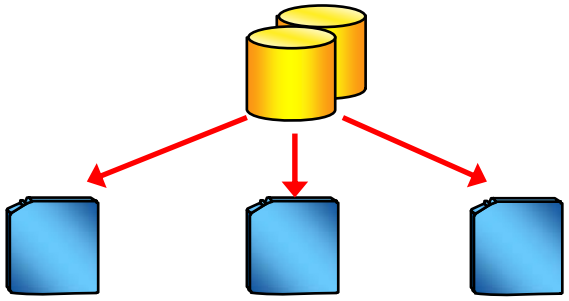


- High/low migration thresholds based on percentage of volumes containing data
- Parallel migration processes each work on a different source volume, possibly dividing work more evenly among processes
- Collocated sequential disk can be used as a buffer for transfer from non-collocated tape to collocated tape

© 2005 IBM Corporation

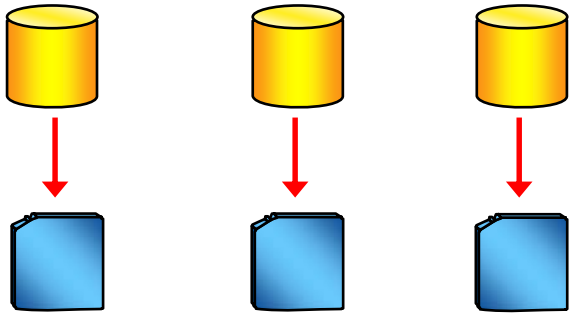
Storage Pool Backup

Random Access

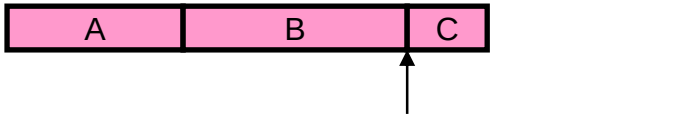


- If node is grouped and target pool is collocated by group, parallel backup processes each work on a different group
- Otherwise, parallel backup processes each work on a different node
- Each physical file (aggregate or non-aggregated file) must be checked during every storage pool backup

Sequential Access +



- Parallel backup processes each work on a different source volume
- Optimization: For each primary pool volume and copy pool, database stores offset of volume that has already been backed up (no need to recheck during each backup)
- Optimization can be especially important for long-term storage of data on disk



© 2005 IBM Corporation

Space Recovery

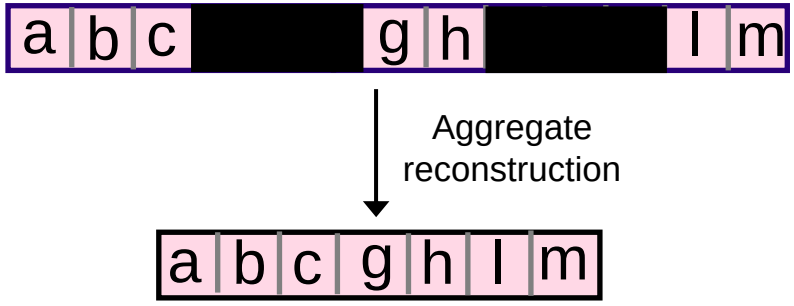
Random Access

- When physical file is moved to another pool (if caching not enabled)
- Space occupied by cached data is recovered as needed
- When physical file is deleted (for aggregates, all files in aggregate must be deleted)
- No reconstruction of empty space within aggregates, a disadvantage if aggregated files are stored for long periods of time

Empty space accumulates until entire aggregate is deleted

Sequential Access +

- Space is not immediately recovered after movement or deletion
- Reclamation recovers empty space created by deletion or movement of files



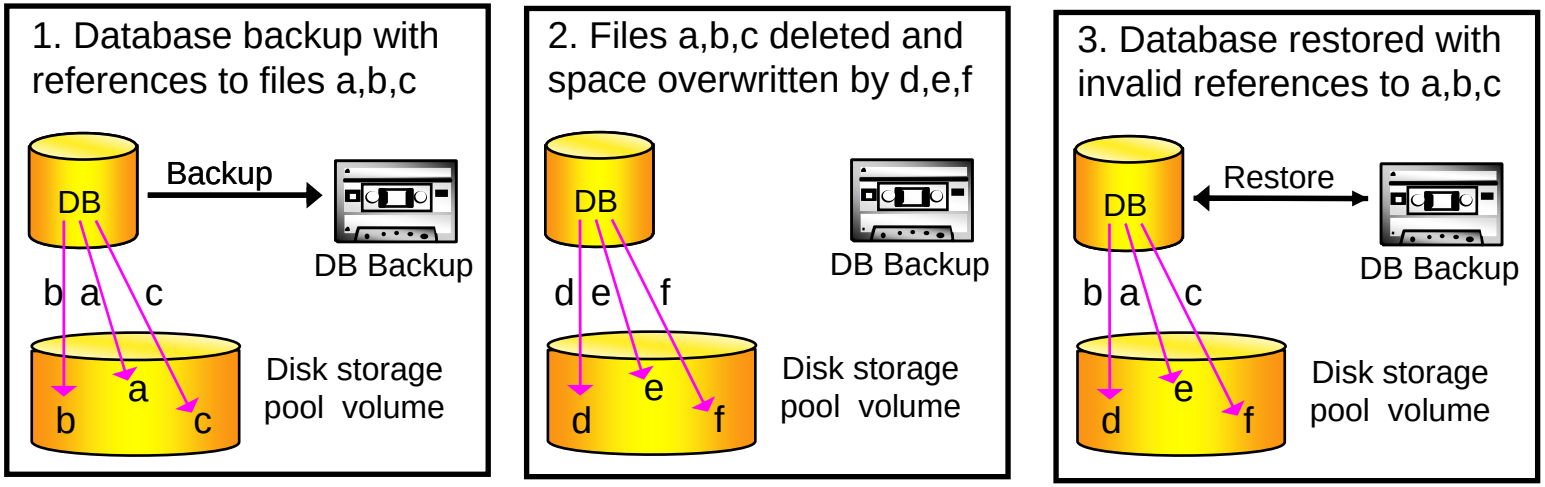
© 2005 IBM Corporation

Fragmentation

	Random-Access Disk	**Sequential-Access Disk +**
Aggregate fragmentation caused by expiration of files	Empty space accumulates in aggregates until all logical files in aggregate are deleted. May result in wasted space for long-term storage on disk.	Empty space is recovered by aggregate reconstruction during reclamation.
Fragmentation of space within TSM volumes caused by deletion of physical files	Volume fragmentation can also occur due to allocation of multiple extents if client size estimate is too low. Fragmentation can degrade performance, but is relieved by migration if no TSM caching.	Deletion of physical files results in empty space within volumes, but this is recovered during reclamation.
File system fragmentation leading to fragmentation of files that constitute TSM volumes	Fragmentation is usually minimal because volumes are predefined or created by space trigger.	Use of scratch volumes causes fragmentation because volumes are extended as needed. Fragmentation can be avoided either by predefining volumes or using space trigger.

© 2005 IBM Corporation

Database Regression



Random Access

- After database regression, all volumes must be audited
- This may be time-consuming for large DISK pools (for example, pools used for long-term data storage)

Sequential Access +

- After database regression, audit only volumes that were reused or deleted after database backup OR
- With REUSEDELAY set, volume audit can be avoided completely
- Time delays for volume audits during critical recovery operations can be minimized or eliminated

© 2005 IBM Corporation

Random vs. Sequential Disk: Which is Best?

Disk Storage Usage	Recommendation
Traditional disk usage ▪LAN-based storage to disk ▪Daily migration from disk to tape	Either random or sequential, depending on requirements
LAN-free data transfer between client and disk storage	Sequential (or possibly VTL)
Long-term storage of data on disk	Sequential offers significant advantages ▪Reconstruction recovers space in aggregates ▪Optimized storage pool backup ▪Reduced volume fragmentation ▪Multi-session restore ▪Avoidance of volume audit
Exploitation of new disk storage features	Sequential-access disk may be required

© 2005 IBM Corporation

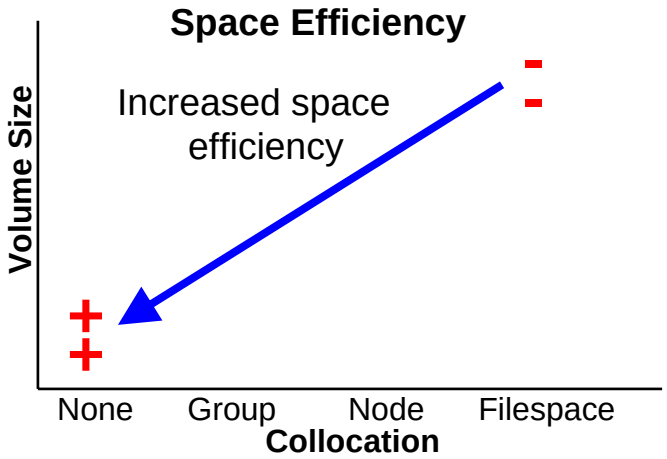
Agenda

- Background
- Random- and sequential-access disk in TSM
- ➢ Special considerations for sequential-access disk
- Potential future enhancements for disk storage

© 2005 IBM Corporation

Optimizing Space Efficiency

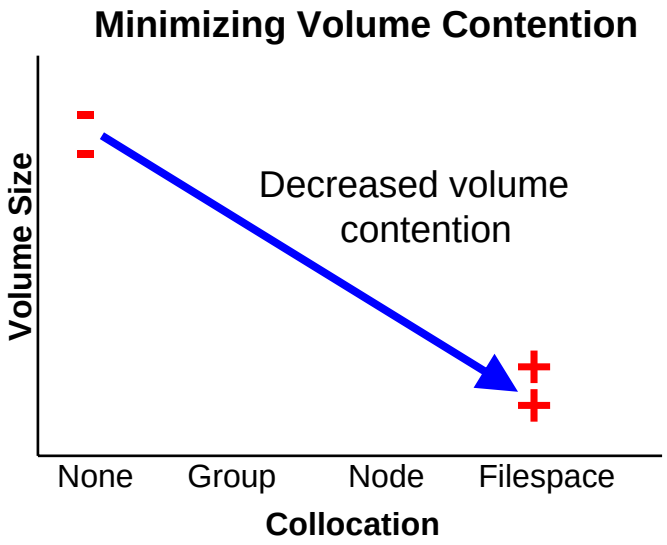
- Scratch volumes
  - Volumes are created and extended only as needed
  - Space is conserved at the expense of file-system fragmentation
- Non-scratch volumes (created by Define Volume command or space trigger)
  - No collocation is more space-efficient
  - Smaller volumes are more space-efficient
- Reclamation should be performed regularly to recover space
  - Efficient because no mount/dismount
  - Many volumes can be reclaimed concurrently



© 2005 IBM Corporation

Reducing Volume Contention

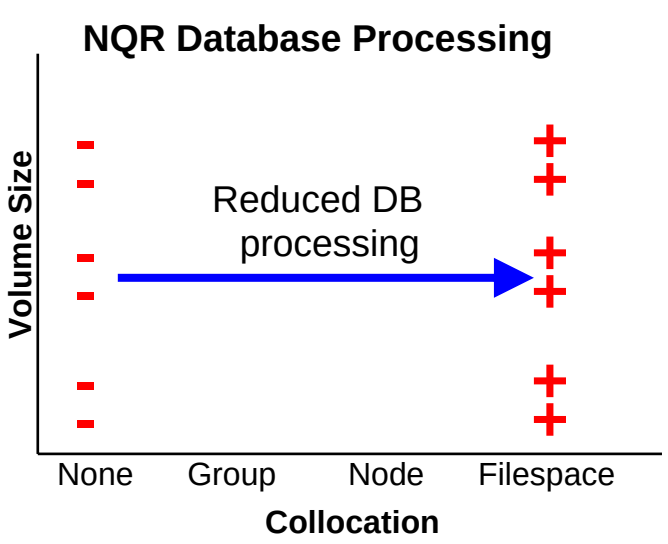
- High-granularity collocation (by node or filespace) reduces contention
- Smaller volume sizes reduce contention

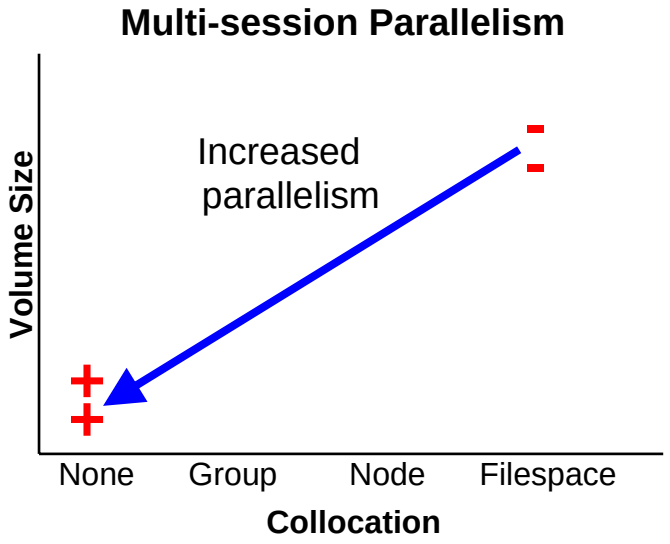


© 2005 IBM Corporation

Improving Client Restore Performance

- For no-query restore operations (used for most large restores), database scanning is greatly reduced if data is well collocated
- Multi-session restore operations achieve greater parallelism if data is spread over multiple sequential-access volumes, indicating that parallelism may be increased by
  - Lower-granularity collocation
  - Smaller volumes





© 2005 IBM Corporation

Avoiding Fragmentation

- Perform reclamation regularly
- Avoid use of scratch volumes
- Predefine volumes using Define Volume command
- Use space trigger to provision additional volumes as needed

© 2005 IBM Corporation

Striking a Balance

- Configuration of sequential-access pools involves tradeoffs, but the following may be a reasonable starting point for most environments
- Define volumes and use space triggers for additional volume provisioning
- Collocate by node
- Use volume size scaled to the size of stored objects
  - For file systems, volume size of 500 MB to 10 GB
  - For databases and other large objects, volume size of 100 GB
- Set reclamation threshold at 20-60% and allow multiple reclamation processes

© 2005 IBM Corporation

Agenda

- Background
- Random- and sequential-access disk in TSM
- Special considerations for sequential-access disk
- Potential future enhancements for disk storage

© 2005 IBM Corporation

Potential Future Enhancements for Disk Storage

- Enhancements specifically for sequential-access disk pools
  - Migration thresholds based on percentage occupancy rather than volumes with data
  - Support for raw logical volumes
  - Concurrent read access for volumes
  - Performance improvements on z/OS server
- Collocation of active data (probably sequential-access disk only)
- Management of redundant files
- Data shredding (overwrite) as data is deleted from disk
- Additional snapshot support
- Additional exploitation of continuous data protection (CDP) technology

© 2005 IBM Corporation