Effective Storage Tiering for Databases

Murthy V Mamidi
Raghu Malige
Symantec Corporation

SNIA Legal Notice

- The material contained in this tutorial is copyrighted by the SNIA.
- Member companies and individuals may use this material in presentations and literature under the following conditions:
  - Any slide or slides used must be reproduced without modification
  - The SNIA must be acknowledged as source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA Education Committee.
- Neither the Author nor the Presenter is an attorney and nothing in this presentation is intended to be nor should be construed as legal advice or opinion. If you need legal advice or legal opinion please contact an attorney.
- The information presented herein represents the Author's personal opinion and current understanding of the issues involved. The Author, the Presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.
  NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

Agenda

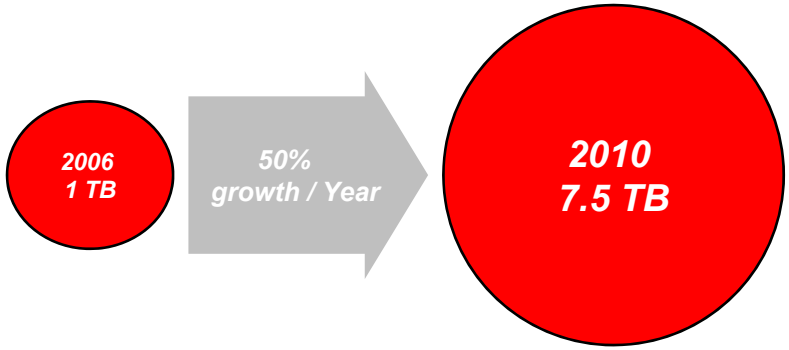
- The Problem
  - Inefficient utilization of storage
  - Increasing cost of storage due to data growth
  - Sub-optimal performance
- The Solution
  - Identification of hot and cold files
  - Dynamic, transparent, and automatic file relocation
- The Result
  - File relocation makes storage tiering possible for databases
  - Relieves DB administrators of extensive data management

Learning objectives

- Why multi-tier storage is valuable
- Why data placement and relocation is beneficial
- Identifying challenges to exploiting multiple storage tiers with structured (database) data
- The value of a novel solution that achieves storage tiering benefits for databases

Why Tier? Data Explosion …

Information is created, captured and replicated



- Causes
  - Regulatory compliance
    - Redundant & replicated content
  - New applications and services
    - Non-text data types
  - Online direct access is *sooooo* much better
  - *Because we can*
    - Plummeting raw storage cost

- Consequences
  - The data center is increasingly about data
  - Storage cost matters

Why Tier? (Contd.)

- Not all data is created equal
  - Importance
  - Frequency
  - Ready accessibility
  - Performance
  - Age
  - Dynamic
- Data's value to the business changes
  - Good or bad, potentially unlimited
  - Has some life cycle to It

Why Tier? (Contd.)

- Data's value changes over time
  - Age, activity, …
- Value drives access characteristics
  - Frequency
  - Ready accessibility
  - Performance
- Value drives availability needs
  - Highly available
  - Frequently backed up
  - …

Why Tier? (Contd.)

- Not all storage is created equal either
  - Cost – 4 or even 8X differential
  - Reliability
  - Performance
  - Storage features (e.g., thin provisioning)
- Storage cost constraints
  - Storage itself
  - Space, power, cooling
  - Administration

Exploit the Inequality!

- Files are natural objects to manage

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects
- Files have well understood attributes

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects
- Files have well understood attributes
- File-based placement makes storage content-aware

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects
- Files have well understood attributes
- File-based placement makes storage content-aware
- File-based placement avoids fragmentation

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects
- Files have well understood attributes
- File-based placement makes storage content-aware
- File-based placement avoids fragmentation
- Files enable control over business object placement

Exploit the Inequality!

- Files are natural objects to manage
- Files map to underlying business objects
- Files have well understood attributes
- File-based placement makes storage content-aware
- File-based placement avoids fragmentation
- Files enable control over business object placement

File-based scheme:
a high-value approach to tiering!

Why Care?

- Consider 100 terabytes of data
  - 20% active, critical
  - 80% relatively inactive, backup available

Why Care?

- Consider 100 terabytes of data
  - 20% active, critical
  - 80% relatively inactive, backup available

	Storage cost per terabyte	Cost of storage
Single-tier strategy	Tier 1: $7,500	Tier 1: $750,000

Why Care?

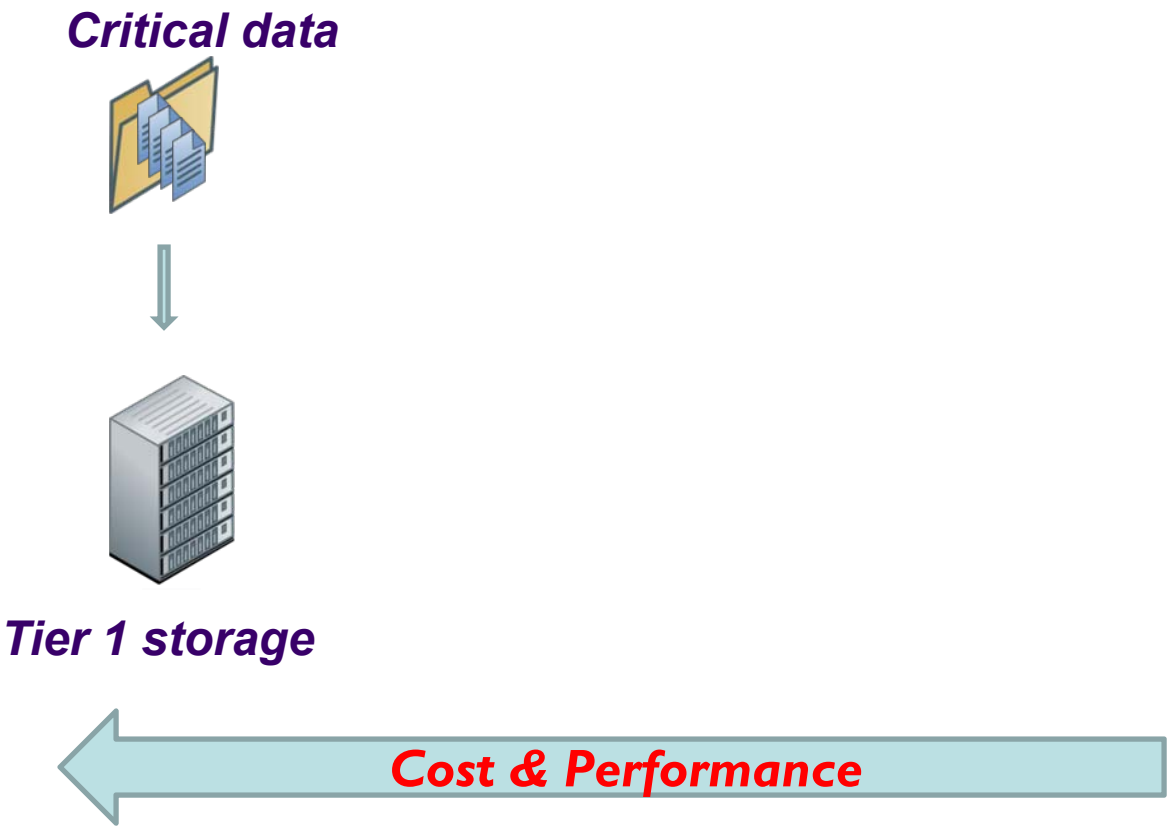
- Consider 100 terabytes of data
  - 20% active, critical
  - 80% relatively inactive, backup available

	Storage cost per terabyte	Cost of storage	Savings
Single-tier strategy	Tier 1: $7,500	Tier 1: $750,000	
Two-tier strategy	Tier 1: $7,500 Tier 2: $2,000	Tier 1: $150,000 Tier 2: $160,000 Total: $310,000	$440,000 (59%)*

** Symantec's DST Analyzer can help estimate potential savings*

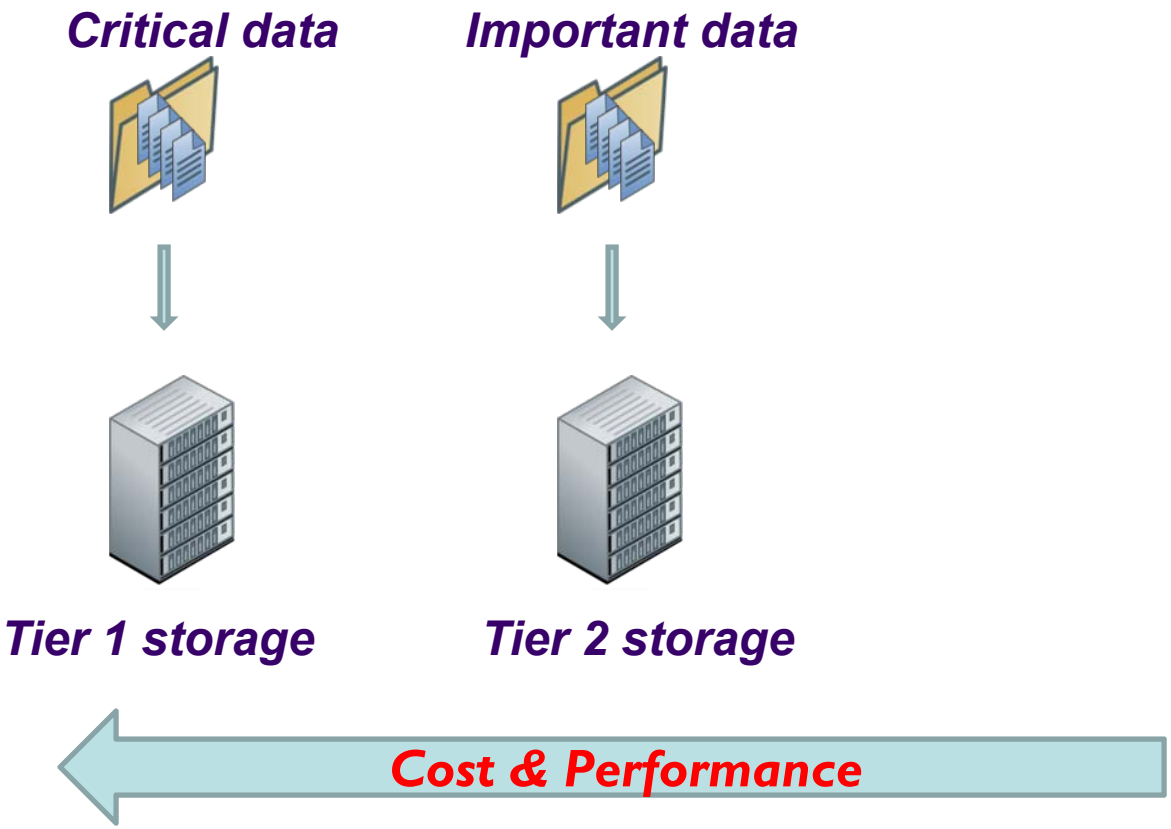
The secret to success with storage tiering

- Match data to storage



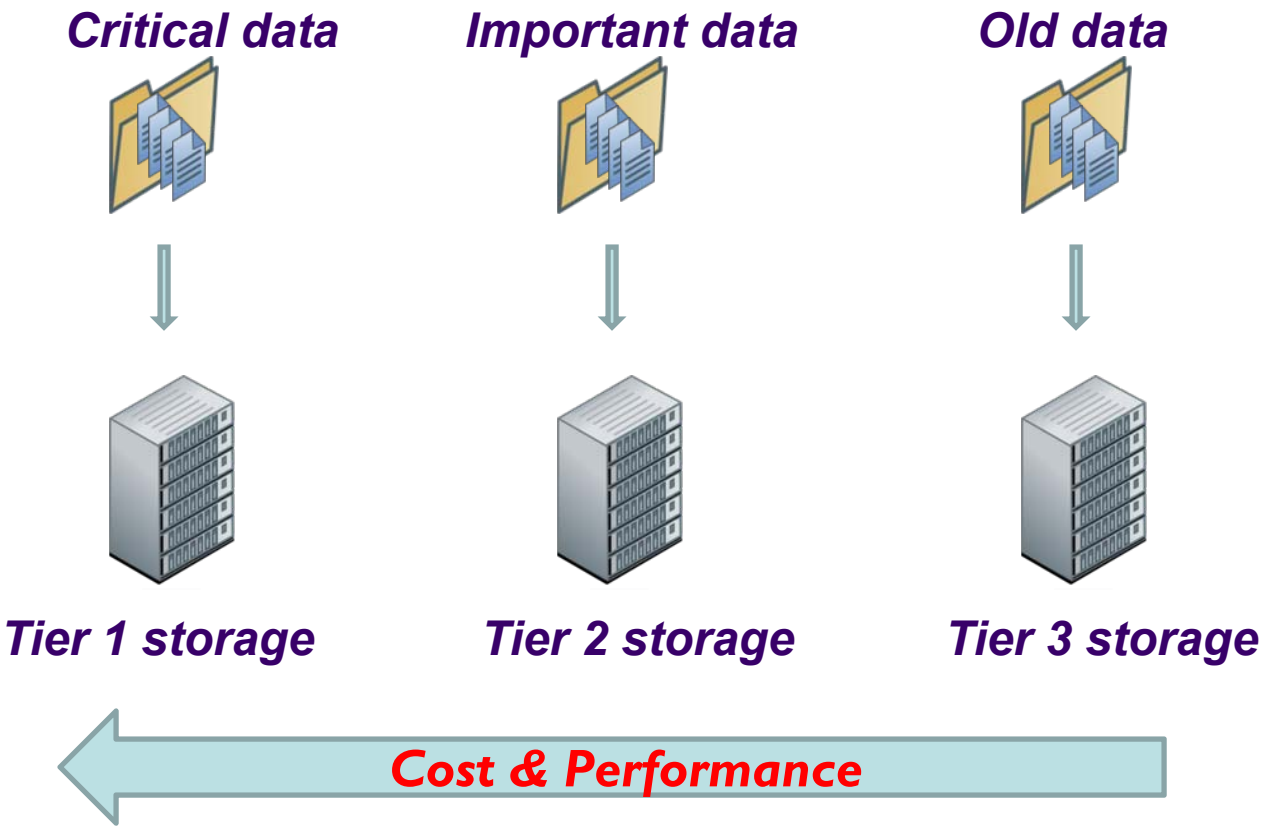
The secret to success with storage tiering

- Match data to storage



The secret to success with storage tiering

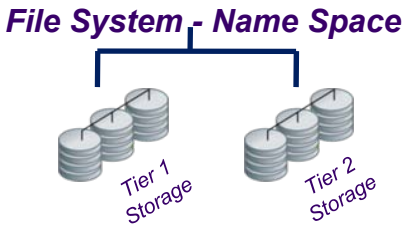
- Match data to storage



Requirements of a solution

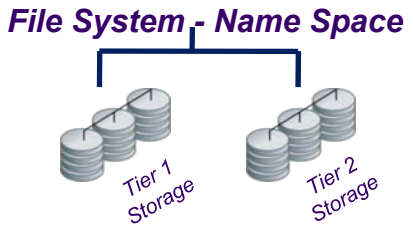
Single Name Space

Realized by

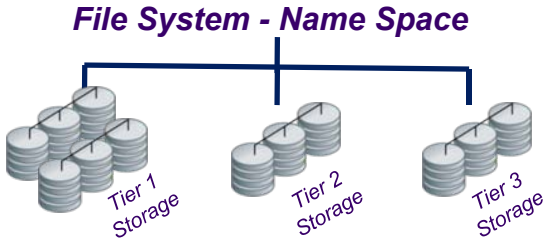


Requirements of a solution

Single Name Space → **Realized by**

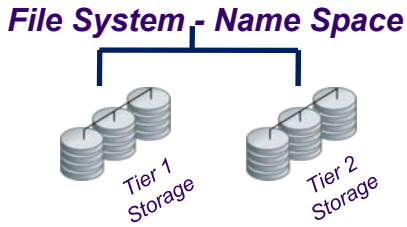


Expandability → **Manage by**

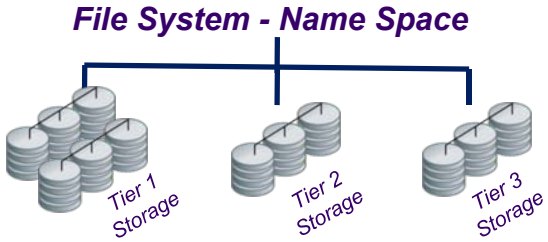


Requirements of a solution

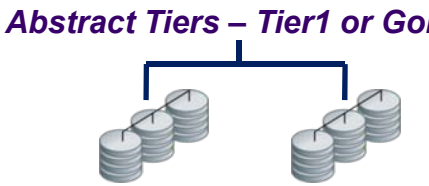
Single Name Space → **Realized by**



Expandability → **Manage by**

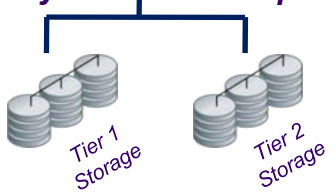


Flexibility → **Just tags**

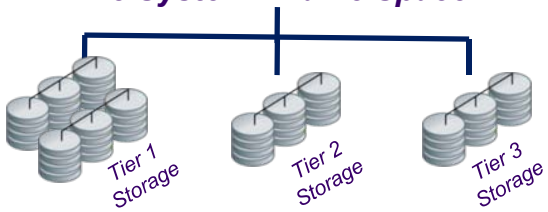


Requirements of a solution

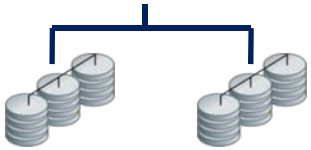
Single Name Space → **Realized by** → *File System - Name Space*



Expandability → **Manage by** → *File System - Name Space*

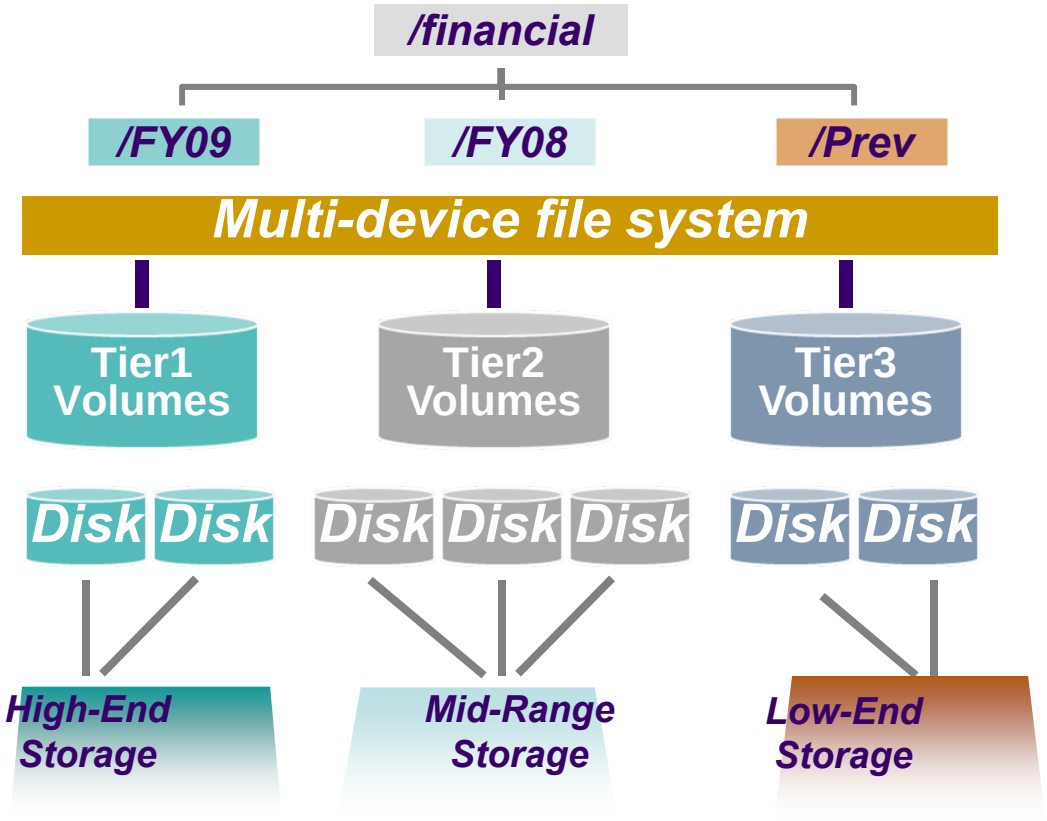


Flexibility → **Just tags** → *Abstract Tiers – Tier1 or Gold*



Automation → **Translates to** → *Right data on right storage at right time*

The Symantec multi-device file system



XML-based placement policy framework

- SELECT part of name space
  - Directory
  - Name patterns
  - Ownership
- CREATE on specific tiers
- RELOCATE or DELETE based on conditions
  - Size
  - Age
  - IO temperature

Implementing Dynamic Storage Tiering (DST)

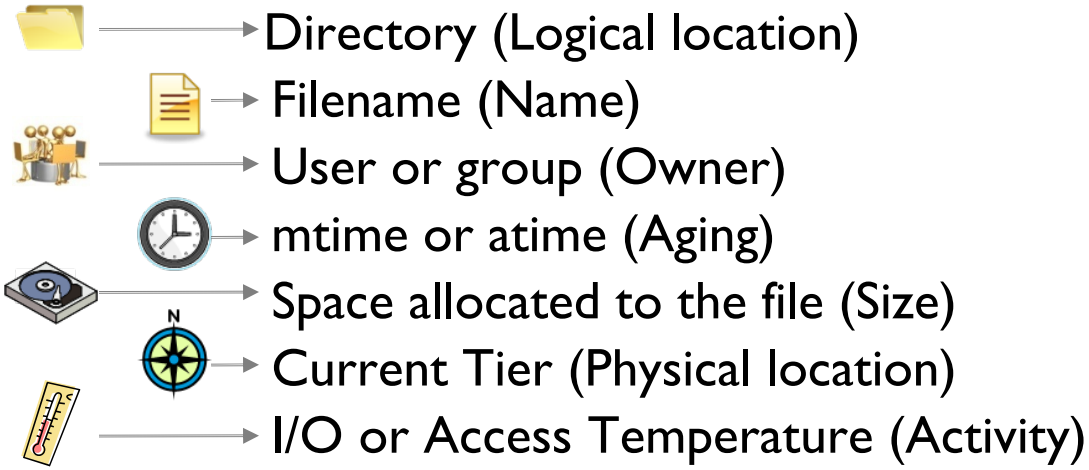
- Create storage tiers
  - Tier: a volume or group of volumes

Implementing Dynamic Storage Tiering (DST)

- Create storage tiers
  - Tier: a volume or group of volumes
- Add a "policy" to relocate file data based on changes to file attributes or metadata

Implementing Dynamic Storage Tiering (DST)

- Create storage tiers
  - Tier: a volume or group of volumes
- Add a "policy" to relocate file data based on changes to file attributes or metadata
- Specify various placement criteria

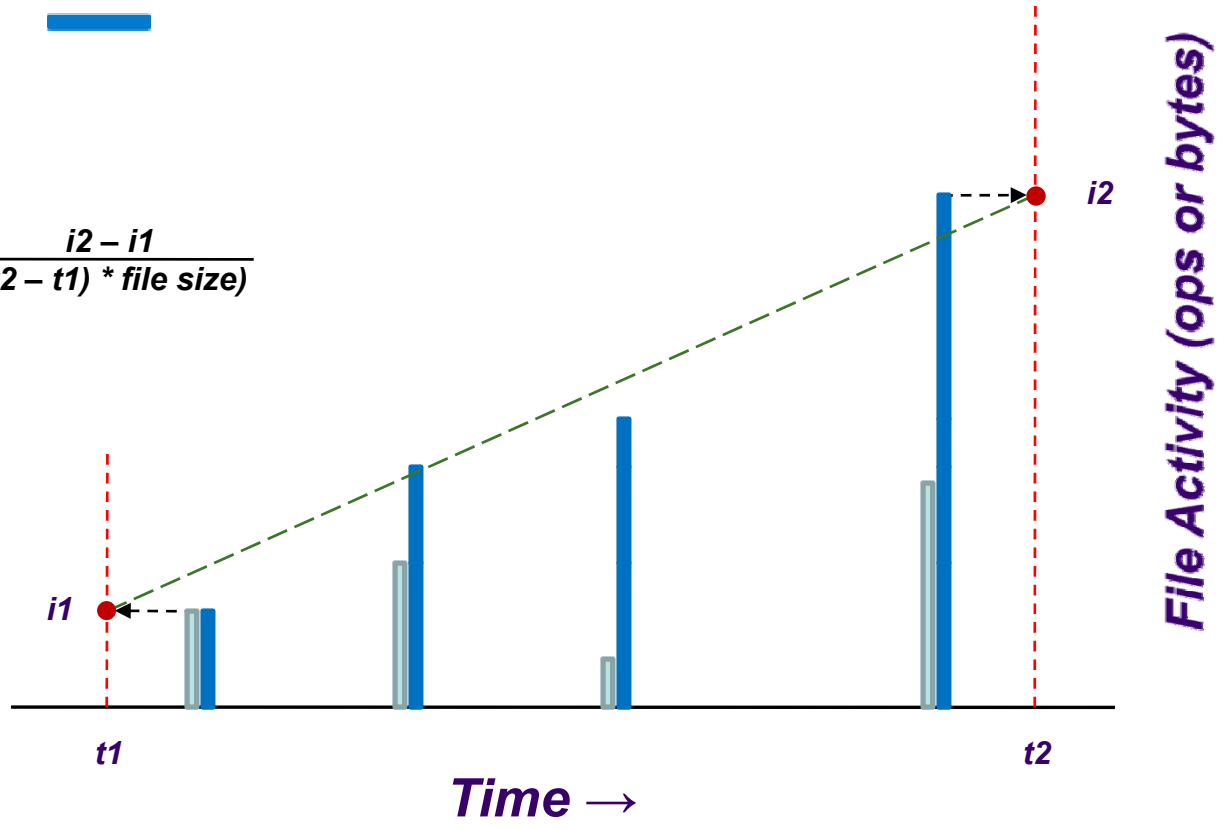


IO Temperature

Amount of IO

Cumulative IO

$$\text{Temperature} = \frac{i2 - i1}{((t2 - t1) * \text{file size})}$$

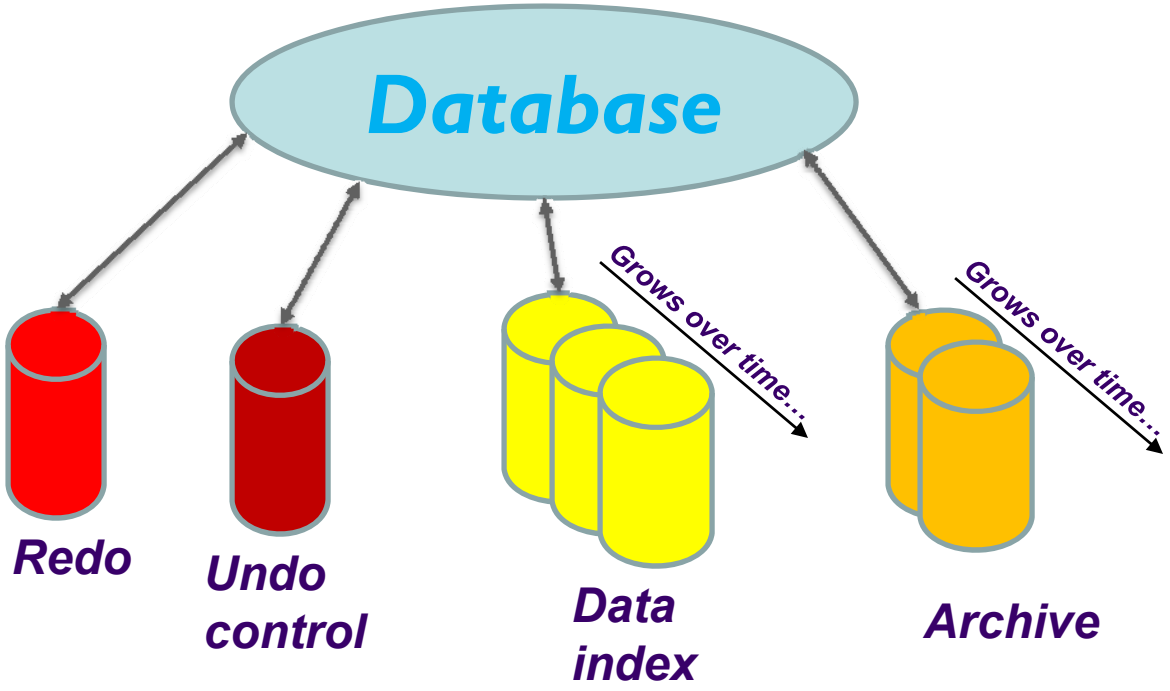


DST "set and forget" data management

- Set
  - Construct and assign a placement policy
- Forget
  - Efficient sweeping via *File Change Log* and inode list walk
- Extra benefits
  - "What if" analysis
  - Query for file location
  - Per-tier utilization reports

Storage Tiering for Databases

Database storage components



Database storage requirements

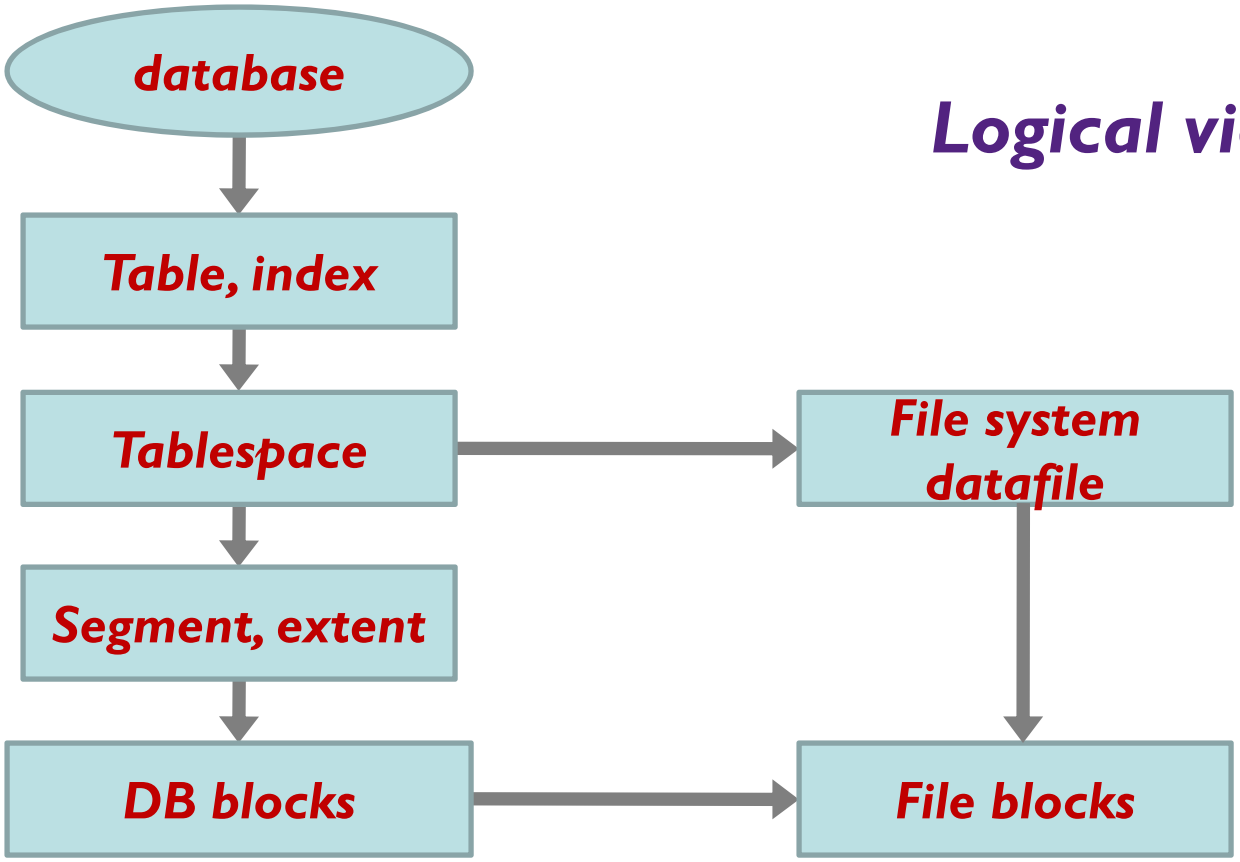
- Database sizes increasing at fast rate
  - Multi-terabyte databases are common
- Data, index and archive logs increase over time
- TELCO databases normally have high transaction rate
  - e.g., archive logs can grow at several GB/hour
- Governance rules/regulations, audit requirements mandate storing archive logs for 5-7 years

Storage cost of archive logs

- Typical size of each archive log : 300-500MB
- Recent archive logs needed for database recovery
  - Age based DST policy works well
- Example : Cost per month

	Storage cost/GB	DST policy	Cost of storage	Savings
Single-tier Strategy	Tier1 → $7.50	None	3TB = $22,500	
Three-tier strategy	Tier1 →$7.50 Tier2 →$2.00 Tier3 →$0.50	< 3 days 4 – 14 days > 14 days	300GB → $2,250 1000GB → $2,000 2700GB → $1,350 ========== Total → $5,600	$16,900 (~80%)

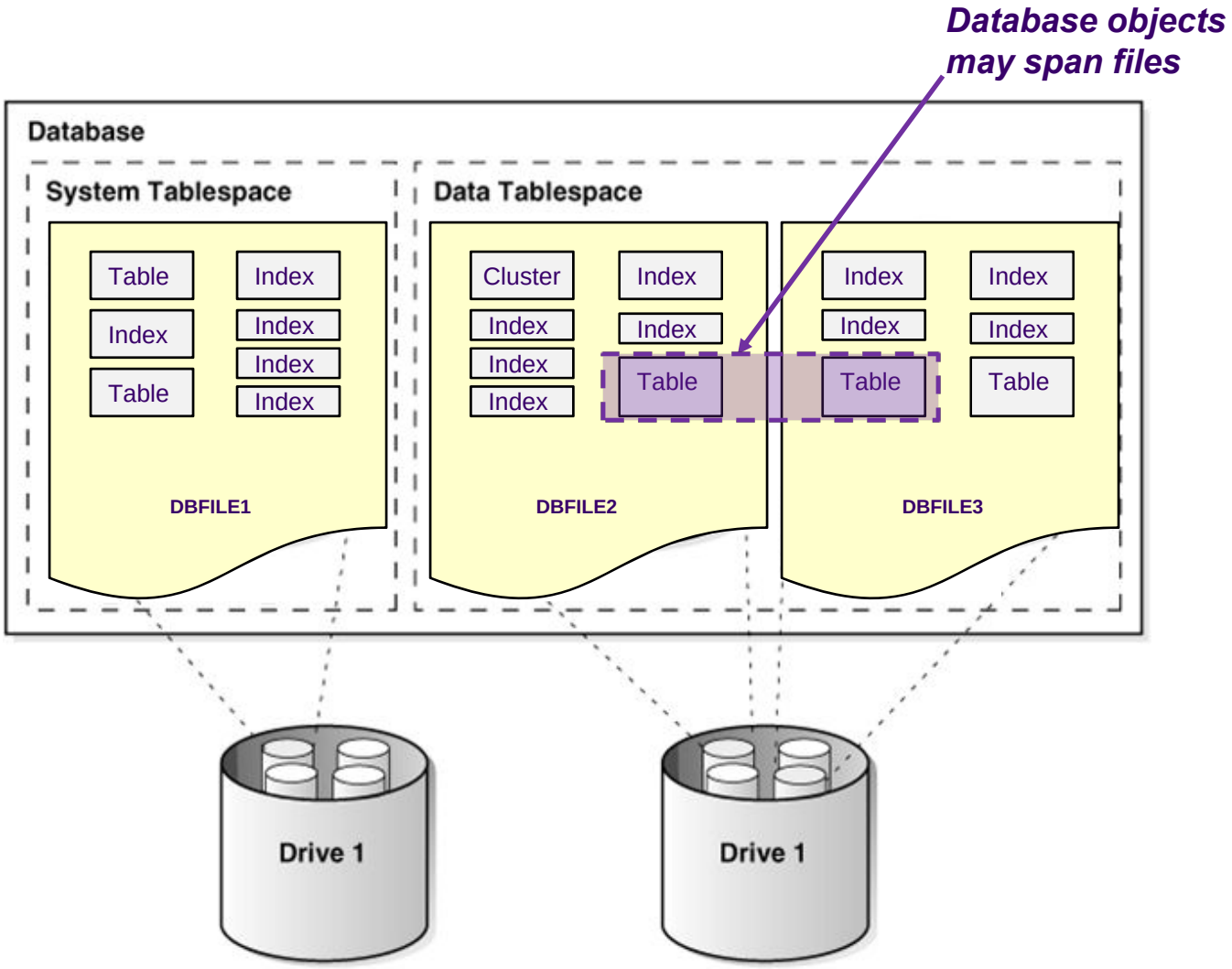
How Databases store data



Logical view

How databases store data

Physical view



Properties of datafiles

- Datafile size can be up to 2^{32} database blocks
  - Database block size: 2-32KB (8KB is most used)
- Backup/restore unit is a datafile
  - Large datafile restore takes longer time
  - Popular datafile size: 2-8GB
- Tablespace size can be increased by adding new datafiles or increasing size of existing datafiles
- Datafile headers are updated at every database checkpoint
  - Use access temperature policy or IO temperature based policy

Tiering for tables : Partitions

- Most databases support partitioned table
- Example:

```
order_entry table
partition key → order_date
```

- Each partition maps to distinct set of datafiles
  - Name-based DST policy works well
  - Example: In Q1 of 2009, move Q1 of 2008 to tier-2
- Wish list item:
  - To improve database checkpoint performance, keep datafile header (1-4 database blocks) in tier-1

Tiering for Tables : Cost Savings

- OLTP database
  ORDERS → growth per quarter = 1TB
- Savings for 5 Years

	Storage cost/TB	DST policy	Cost of Storage	Savings
One-tier Strategy	Tier1 → $7500	None	$150,000	
Two-tier Strategy	Tier1 → $7500 Tier2 → $2000	Current Year (Q1,Q2,Q3,Q4) Previous Years	$30,000 $32,000 ---------- $62,000	$88,000 (~60%)

I/O activity-based cyclic tiering

- Example: database for sporting goods
  - Requirement: busy data on tier-1; idle data on tier-2
- Catalog table is partitioned by season
  summer items : **`summer_part`**
  winter items : **`winter_part`**
- Set IO temp-based DST policy to exchange partitions between tier-1 and tier-2
  - When access temperature > 30, for 3 days relocate to tier-1
  - When access temperature < 10, for 7 days relocate to tier-2

Tiering for un-partitioned tables
Use application provided Statistics

- VxFS provides file level statistics
- Databases provide datafiles statistics in its catalog
- Datafiles are added to tablespace on demand
- Older datafiles have older data
- IO activity on older datafiles decreases over time
  - Not necessarily zero
- IO temperature based DST policy is best for this case

Future Work

- Implement SSD as a tier
- Below-file-level placement and relocation

Additional Information

- DST Yellow Book
  - http://eval.symantec.com/mktginfo/enterprise/yellowbooks/dynamic_storage_tiering_03_2006.en-us.pdf
- DST Analyzer
  - http://www.symantec.com/connect/downloads/dynamic-storage-tiering-analyzer-new-tool-quantify-benefits-dst-your-environment

Thank you for your attention

Q&A / Feedback