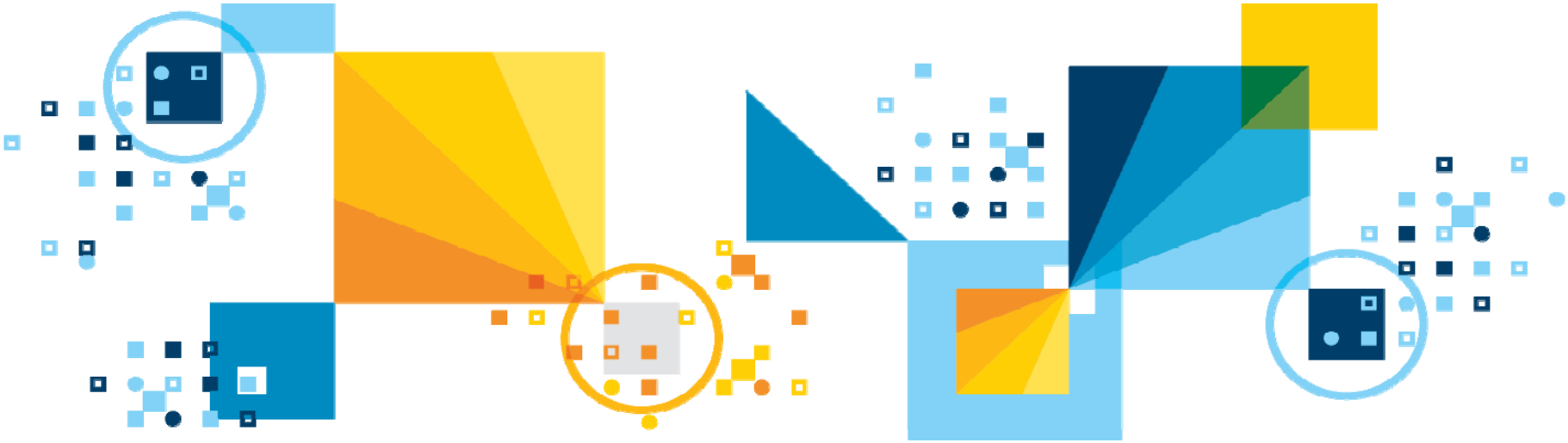
Storage Design

Module ID	10103
Length	1 hour + 1 hour Hands on Lab



For questions about this presentation contact askdata@ca.ibm.com

Disclaimer

Module Information

- You should have completed or acquired the necessary knowledge for the following modules in order to complete this module:
  - DB2 Fundamentals

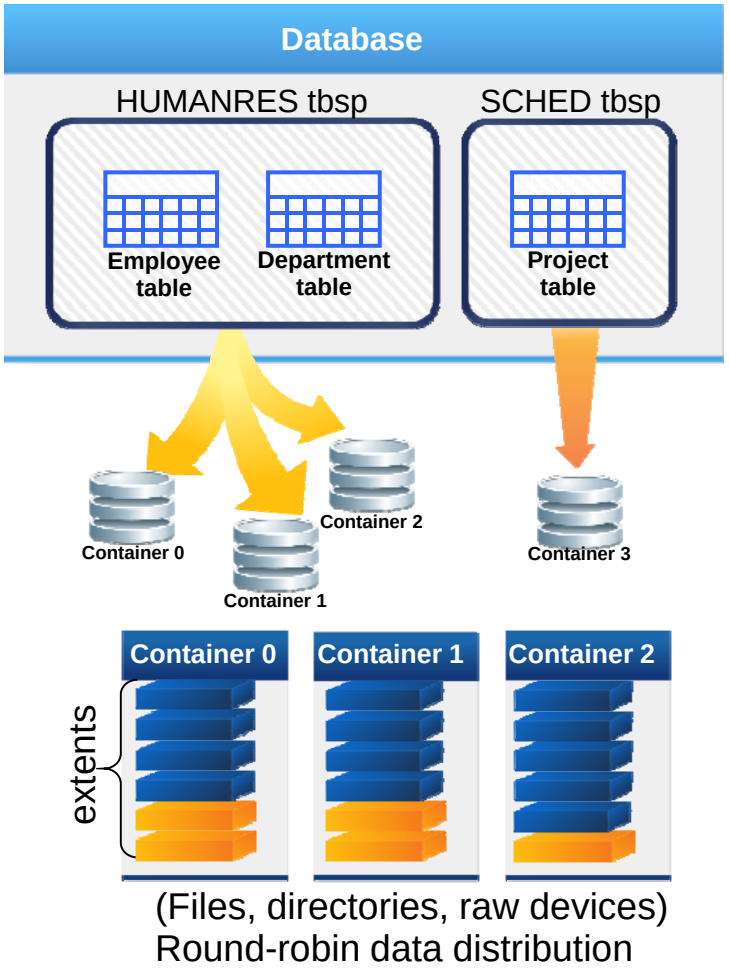
- After completing this module, you should be able to:
  - Describe DB2 data storage models
  - Explain the concepts of:
    - Tablespaces
    - Storage groups
    - Multi-temperature storage
  - Be able to perform the following tasks:
    - Creation of automatic storage databases
    - Enable multi-temperature storage

Module Content

- DB2 Storage Model
  - Auto-resize Feature
  - Automatic Storage
- Multi-Temperature Data Management
  - WLM Integration
- Summary

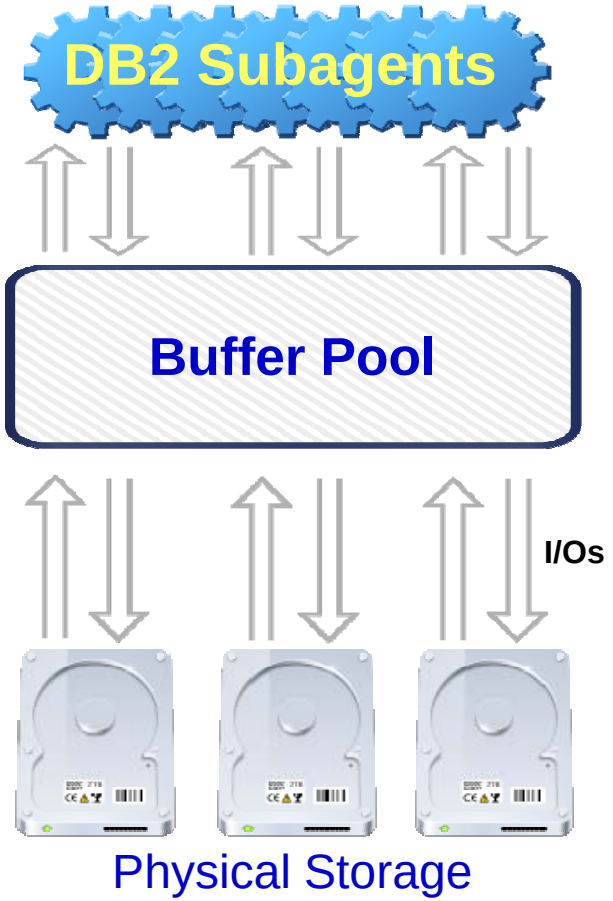
Table Space Review

- A layer of abstraction between logical and physical data
- Allows assignment of data to particular logical devices or portions thereof
- All tables, indexes, and other data are stored in a table space
- Associated to a specific buffer pool
- Managed in three different ways: SMS, DMS and Automatic Storage
- An Automatic Storage table space is associated to a Storage Group, that defines the set of containers



Buffer Pool Review

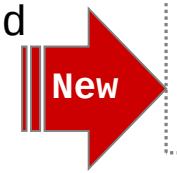
- Area of main memory used to cache table and index data
- Each database must have at least one buffer pool
  - By default IBMDEFAULTBP is used
  - SYSCAT.BUFFERPOOLS catalog view accesses the information for the buffer pools defined in the database
- Every table space is associated to a specific buffer pool of the same page size
  - Match buffer pool size with purpose of table to increase hit ratio
- Self-Tuning Memory Manager (STMM) optimizes BP utilization

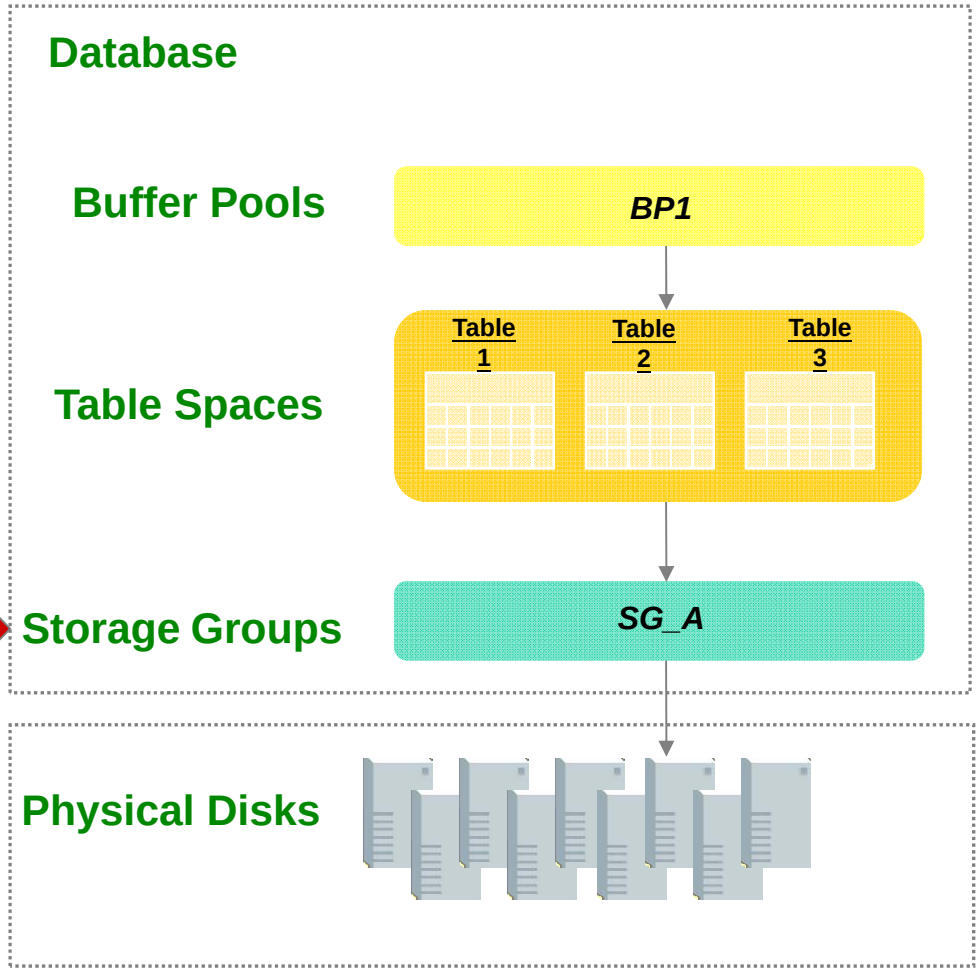



```
CREATE BUFFERPOOL bp4k PAGESIZE 4K
CREATE TABLESPACE tbsp1 PAGESIZE 4K BUFFERPOOL bp4k
```

DB2 Storage Model

- Database
  - Contains a set of objects used to store, manage, and access data
- Buffer Pool
  - Area of main memory for the purpose of caching data as it is read from disk
- Table Space
  - Logical space used to store data objects such as tables and indexes
- Storage Group
  - Set of storage paths configured to represent different classes of storage in the database system, where table spaces are stored
- Physical Disk
  - Physical location used to store data





System Managed Space (SMS)

- Highlights
  - Data stored in files on the filesystem
  - Space is allocated on demand
  - Access to data controlled using standard I/O functions of the OS
    - Pages might not be contiguous, which could have an impact on the performance of some queries
  - Starting in DB2 v10.1 user table spaces using SMS are deprecated
    - You can still specify the SMS type for catalog and temporary table spaces, however this is not recommended.

- How to create

```
CREATE TEMPORARY TABLESPACE tbsp1 MANAGED BY SYSTEM
USING ('d:\acc_tbsp', 'e:\acc_tbsp', 'f:\acc_tbsp')
```

Database Managed Space (DMS)

- Highlights
  - Data stored in pre-allocated files in a file system
  - Ideal for performance-sensitive applications as files are typically contiguous
  - Increased maintenance and monitoring compared to SMS
- How to create

```
CREATE TABLESPACE tbsp2 PAGESIZE 8K MANAGED BY DATABASE
USING (FILE ' /storage/dms1' 10 M)
```

- To increase the size of one or more containers in a DMS table space

```
ALTER TABLESPACE tbsp2 EXTEND (FILE 'file1', 200)
```

- To drop one or more containers in a DMS table space

```
ALTER TABLESPACE tbsp2 DROP (FILE 'file1', DEVICE '/dev/rdisk1')
```

- To decrease the size of one or more containers in a DMS table space

```
ALTER TABLESPACE tbsp1 REDUCE (FILE 'file1' 20M)
```

Database Managed Space – Auto-Resize Feature

- By default, the auto-resize feature is not enabled for DMS table spaces
- To enable the auto-resize feature, specify AUTORESIZE YES clause

```
CREATE TABLESPACE DMST1 MANAGED BY DATABASE USING
(FILE '/db2files/DMS1' 10 M) AUTORESIZE YES
```

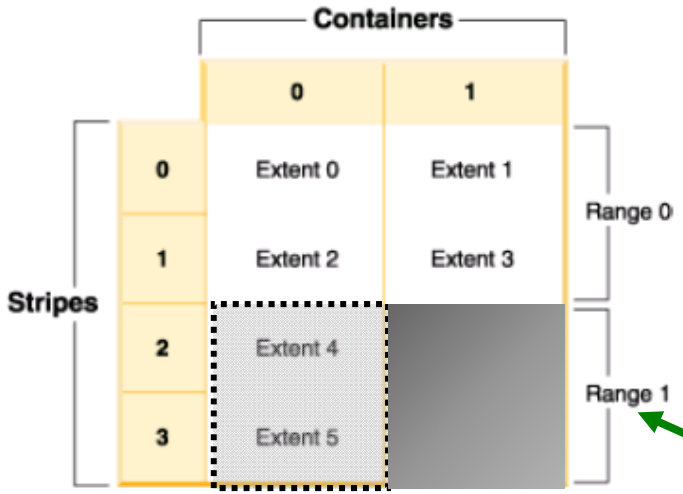
```
ALTER TABLESPACE DMTS1 AUTORESIZE YES
```

- Options to be used with AUTORESIZE:
  - MAXSIZE defines the maximum size for the table space
  - INCREASESIZE defines the amount of space to increase the table space when there are no free more free extents available

```
CREATE TABLESPACE DMS1 MANAGED BY DATABASE USING
(FILE '/db2files/DMS1' 50 M)
AUTORESIZE YES
INCREASESIZE 50 PERCENT
MAXSIZE 1G
```

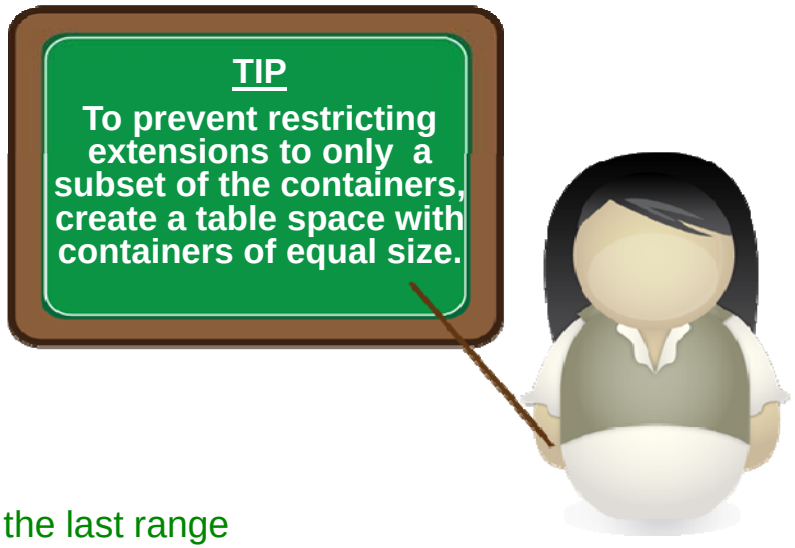
Database Managed Space – Auto-Resize Feature

- How DB2 decides when to increase the size of a table space with the Auto Resize feature:
  - DB2 decides which containers can be extended so that no rebalancing occurs
  - Only containers present within the last range of the table space map are extended
  - All those containers are extended by an equal amount



Container 0 will be extended

Range 1 is the last range and it only has **container 0** in it.

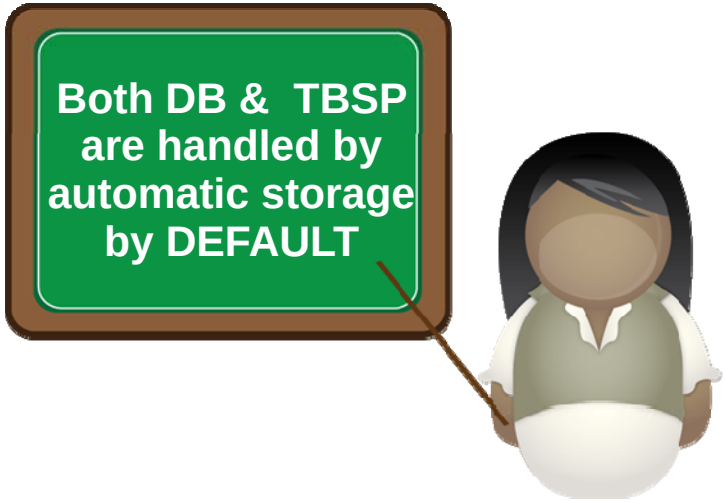


Automatic Storage

- Highlights
  - Low maintenance, containers are created and extended as needed
  - Creates a DMS table space for regular/large table spaces
  - Creates a SMS table space for temporary table spaces
  - Leverage the new storage groups – a new layer of abstraction between logical (table spaces) and physical storage (containers) configured to represent different classes of storage available to your database

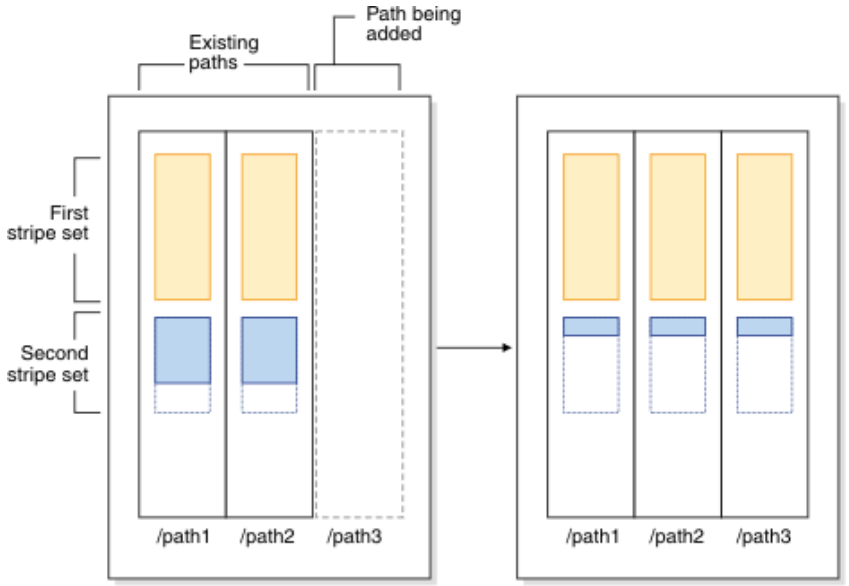
- How to create

```
CREATE DATABASE db1
CREATE TABLESPACE tbsp1
```



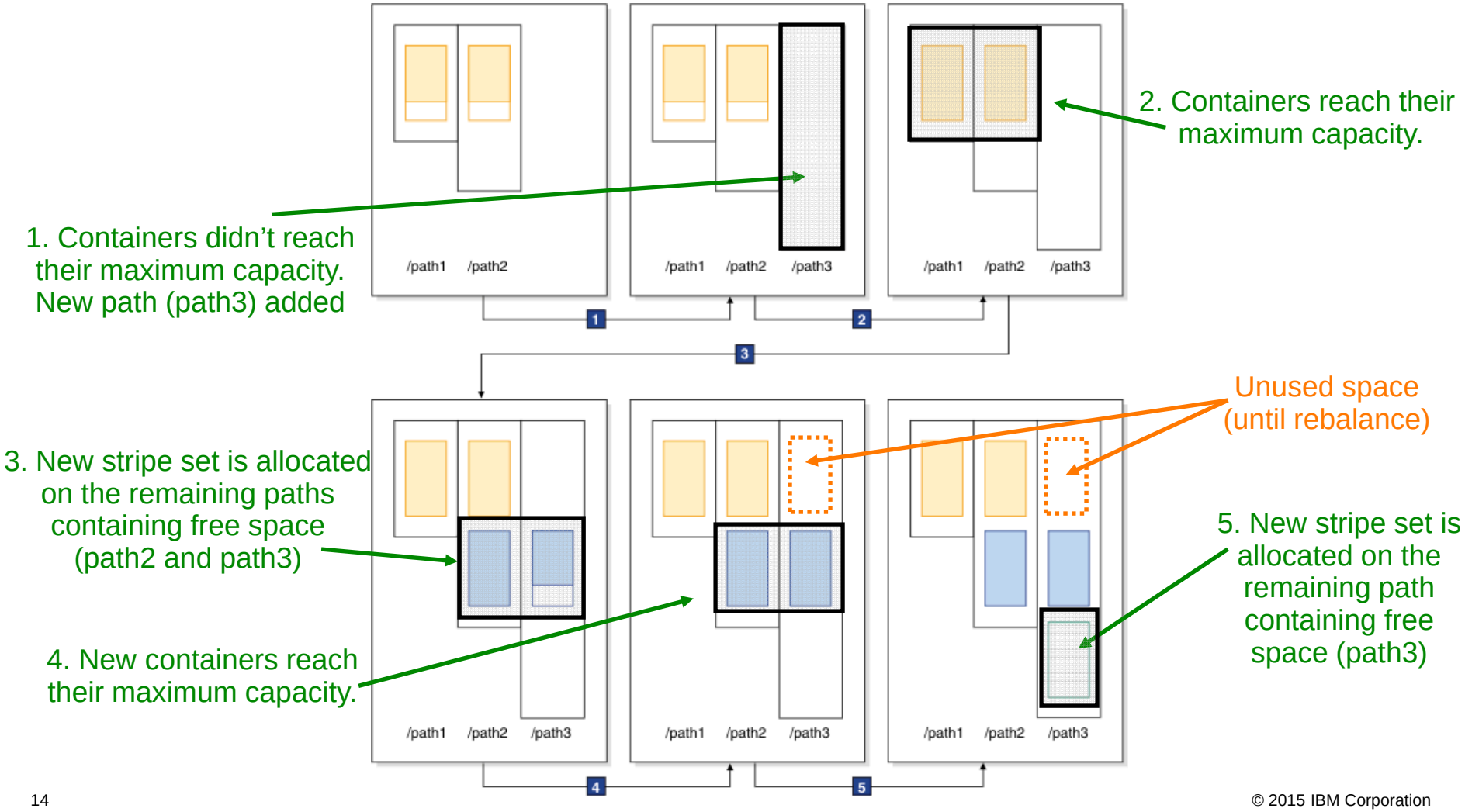
Automatic Storage

- Storage for automatic table spaces is managed at the storage group level
  - Storage is added to the storage group used by the table spaces
- DB2 automatically creates and grows the underlying containers
  - By extending existing containers
  - By adding a new stripe set of containers
- For added storage, table spaces that already exist do not start consuming storage on the new paths immediately
  - Rebalance Operation extends the existing stripe set and rebalances the data along all containers



Automatic Storage

- Adding a new stripe set of containers (without REBALANCING)



Automatic Storage – Storage Groups

- IBMSTOGROUP created immediately and contains: SYSCATSPACE, TEMPSPACE1, USERSPACE1

```
CREATE DATABASE TESTDB7 ON '/dbpath1','/dbpath2';
```

IBMSTOGROUP is the DEFAULT storage group containing paths /dbpath1 and /dbpath2

- Leverage storage groups by creating or adding paths to groups

```
CREATE STOGROUP sg_1 ON '/mnt/ssd1', '/mnt/ssd2'
ALTER STOGROUP sg_1 ADD '/mnt/ssd3'
```

- To specify another storage group as DEFAULT, create or alter an existing one

```
CREATE STOGROUP sg_2 ON '/path1', '/path2' SET AS DEFAULT
ALTER STOGROUP sg_1 SET AS DEFAULT
```

  - NO impact on existing table spaces, used for newly created table spaces
  - Can move existing table spaces using ALTER TABLESPACE

- Storage Path
  - Can be added to one or more storage group
  - Can be across all database partitions
  - Use ADMIN_GET_STORAGE_PATHS administrative view to see the list of storage paths

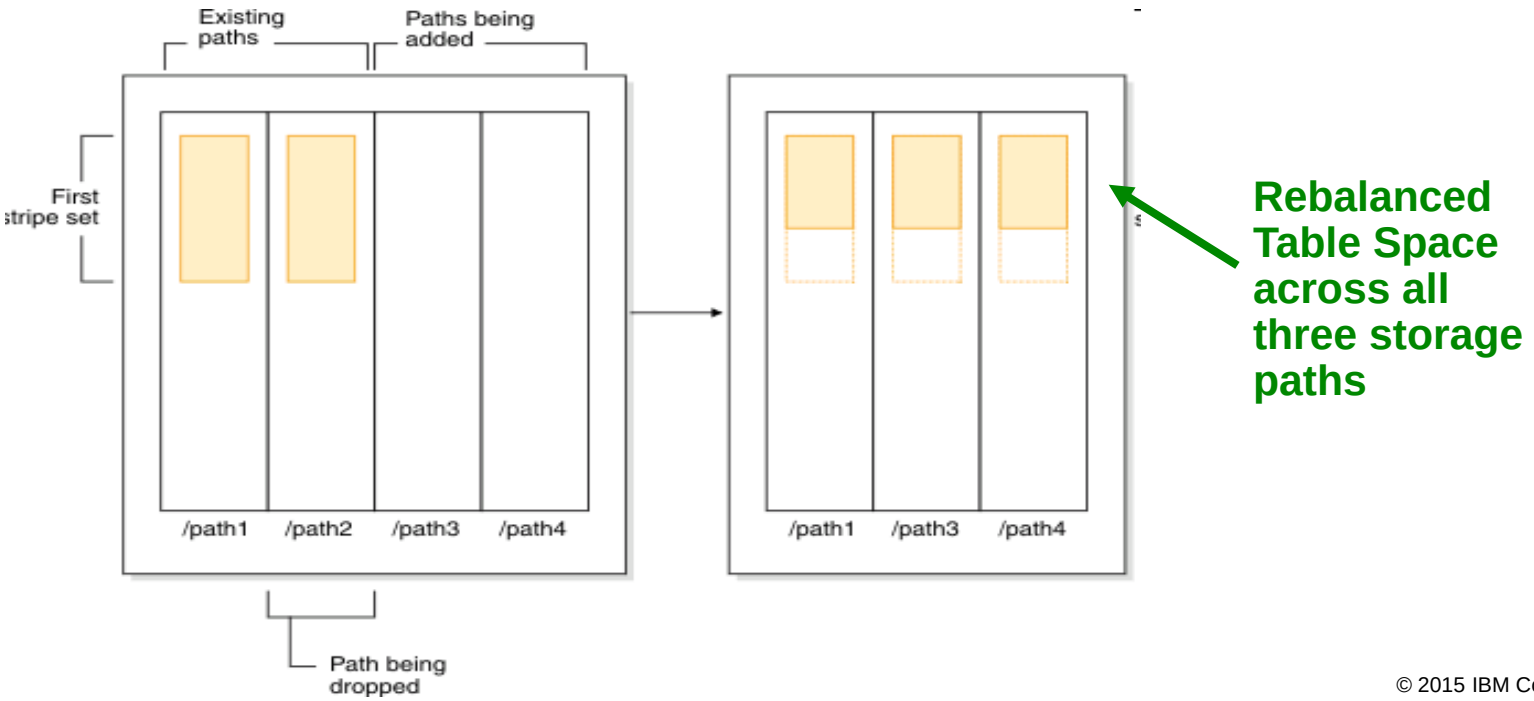
Add / Drop Storage Paths to Storage Group

- Scenario: Add '/path3', '/path4', and drop '/path2' from storage group sg_2

```
ALTER STOGROUP sg_2 ADD '/path3', '/path4'
ALTER STOGROUP sg_2 DROP '/path2'
```

- Explicitly execute REBALANCE operation
  - To start using added path(s) immediately
  - To initiate data movement off path(s) marked for drop pending

```
ALTER TABLESPACE tpsp1 REBALANCE
```



Dropping a Storage Group

- Storage groups can be dropped when they do not contain any table spaces
- Step 1: Find the table spaces using the storage group

```
SELECT TBSP_NAME, TBSP_CONTENT_TYPE FROM table (MON_GET_TABLESPACE(' ', -2))
WHERE TBSP_USING_AUTO_STORAGE = 1 AND STORAGE_GROUP_NAME = 'STO_GROUP_OLD'
ORDER BY TBSP_ID
```

- Step 2: Assign table spaces to another storage group

```
ALTER TABLESPACE tbsp_2009 USING STOGROUP STO_GROUP_NEW
```

- Step 3: Monitor rebalance activity for the storage group to be dropped

```
SELECT * from table (MON_GET_REBALANCE_STATUS( ' ', -2))
WHERE REBALANCER_SOURCE_STORAGE_GROUP_NAME = 'STO_GROUP_OLD'
```

Empty result state indicates move complete

- Step 4: Drop the storage group

```
DROP STOGROUP STO_GROUP_OLD
```

Automatic Storage – Table Space

- Specify a storage group on the CREATE TABLESPACE command

```
CREATE TABLESPACE mytbspc USING STOGROUP sg_1
```

- Dynamically inherits media attributes from the storage group
  - using the INHERIT clause (DATA TAG, TRANSFER RATE, OVERHEAD are inherited)
  - Can override inherited media attributes storage group
- Change the storage group for a table space using the ALTER TABLESPACE command

```
ALTER TABLESPACE mytbspc USING STOGROUP sg_2
```

  - By default, when changing the storage group, the table space inherits attributes from the target storage group
  - An implicit REBALANCE will move data from the source storage group to the target one
  - DMS table spaces can be converted to Automatic Storage in order to use storage groups

Convert DMS Table Spaces to use Automatic Storage

- Use ALTER TABLESPACE to modify the DMS table space to use automatic storage
  1. Specify the new table space to convert and indicate the storage group to be used

```
ALTER TABLESPACE tbsp_2011q1 MANAGED BY
AUTOMATIC STORAGE USING STOGROUP sg_2
```

  2. Move the data from the old containers to the storage paths in the storage group

```
ALTER TABLESPACE tbsp_2011q1 REBALANCE
```

- Asynchronous operation ✅
- Does not affect data availability ✅

  3. Monitor the progress using the monitoring table function MON_GET_REBALANCE_STATUS

DB2 Storage Model – Summary

- Automatic Storage Table Space
  - Containers are created and extended as needed
  - Automatically handles resizing table spaces
  - Creates a DMS table space for regular/large table spaces
  - Creates a SMS table space for user or system temporary table spaces
  - Leverage Storage Groups – a new layer of abstraction between logical (table spaces) and physical storage (containers)

- Database Managed Space (DMS)
  - Data stored in pre-allocated files in a file system
  - Ideal for performance-sensitive applications as files are typically contiguous
  - Increased maintenance and monitoring compared to SMS
  - Auto-resize feature maintains the table space sizes by automatically extending existing containers
- System Managed Space (SMS)
  - Data stored in files, space is allocated on demand
  - Access to data controlled using standard I/O functions of the OS
  - Low maintenance

Content

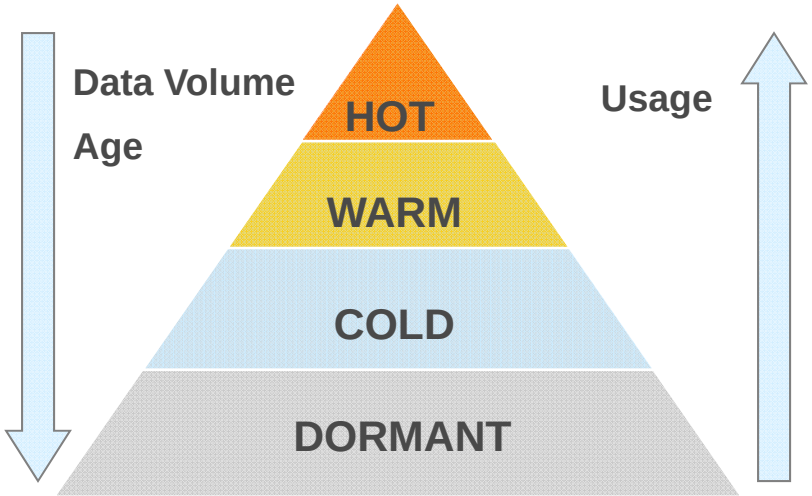
- DB2 Storage Model
  - Auto-resize Feature
  - Automatic Storage
- Multi-Temperature Data Management
  - DB2 workload manager (WLM) Integration
- Summary

Multi-Temperature Storage – Business Motivation

- There is an explosive growth in storage requirements due to
  - technology shifts
  - unstructured data
  - regulatory mandates
- Data Warehouses retain large volumes of data for real-time business analytics
  - 100+ TB of data stored
  - Require expensive high speed storage for BI + DSS
- IT budgets cannot support the growth necessary without a change in storage practices.
- Difficult tradeoff between performance and cost
  - Data in frequent use vs. less frequent/ historical data
  - Small subset of data benefits from high-speed storage
  - High-speed storage (e.g. SSD) = Higher costs
- Access to various groups of data can have different performance and reliability requirements – Tiered storage is needed
- Multi-temperature Storage provides a solution

Multi-Temperature Data Management

- Provides the ability to assign priority to data (hot, warm, cool, cold) and dynamically assign it to different classes of storage
  - Data temperature signifies priority of the data defined by business
  - Data temperature is inversely proportional to volume
    - Small portion of hot data vs. large portion of warm/cold data
- Data can change temperature
  - As data ages
  - As business criteria behind temperature changes



Sales data of this month = **most frequent**

Sales data of this quarter = **less frequent**

Sales data of previous quarters = **rarely accessed**

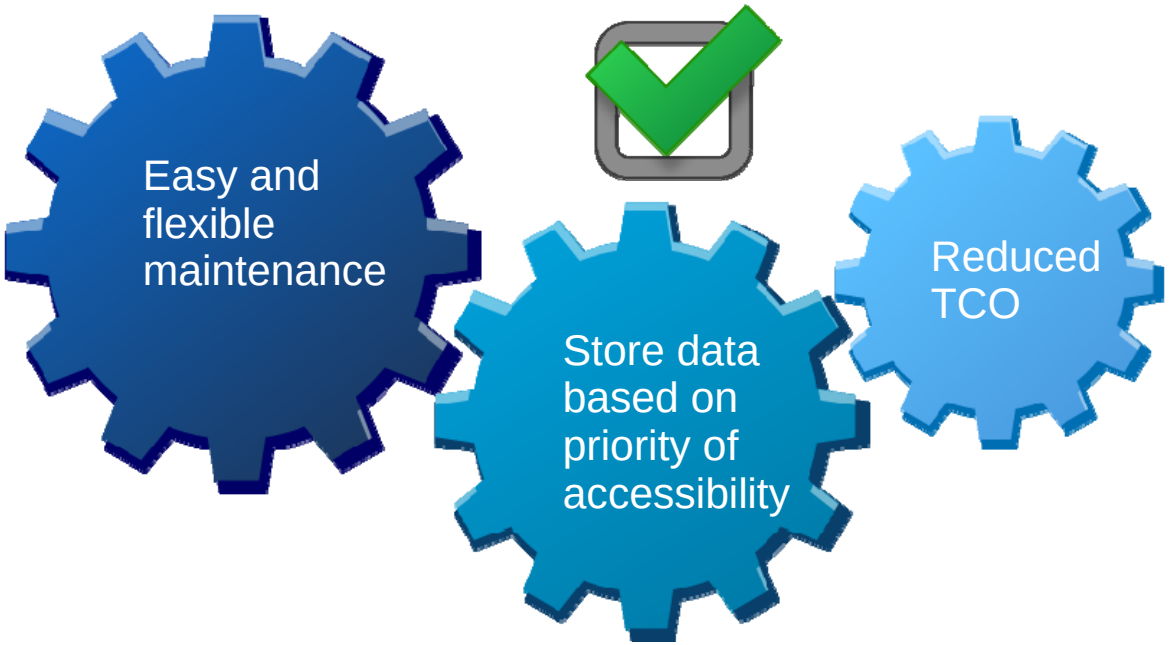
Sales data of past years = **historical data**

Multi-Temperature Data Classification

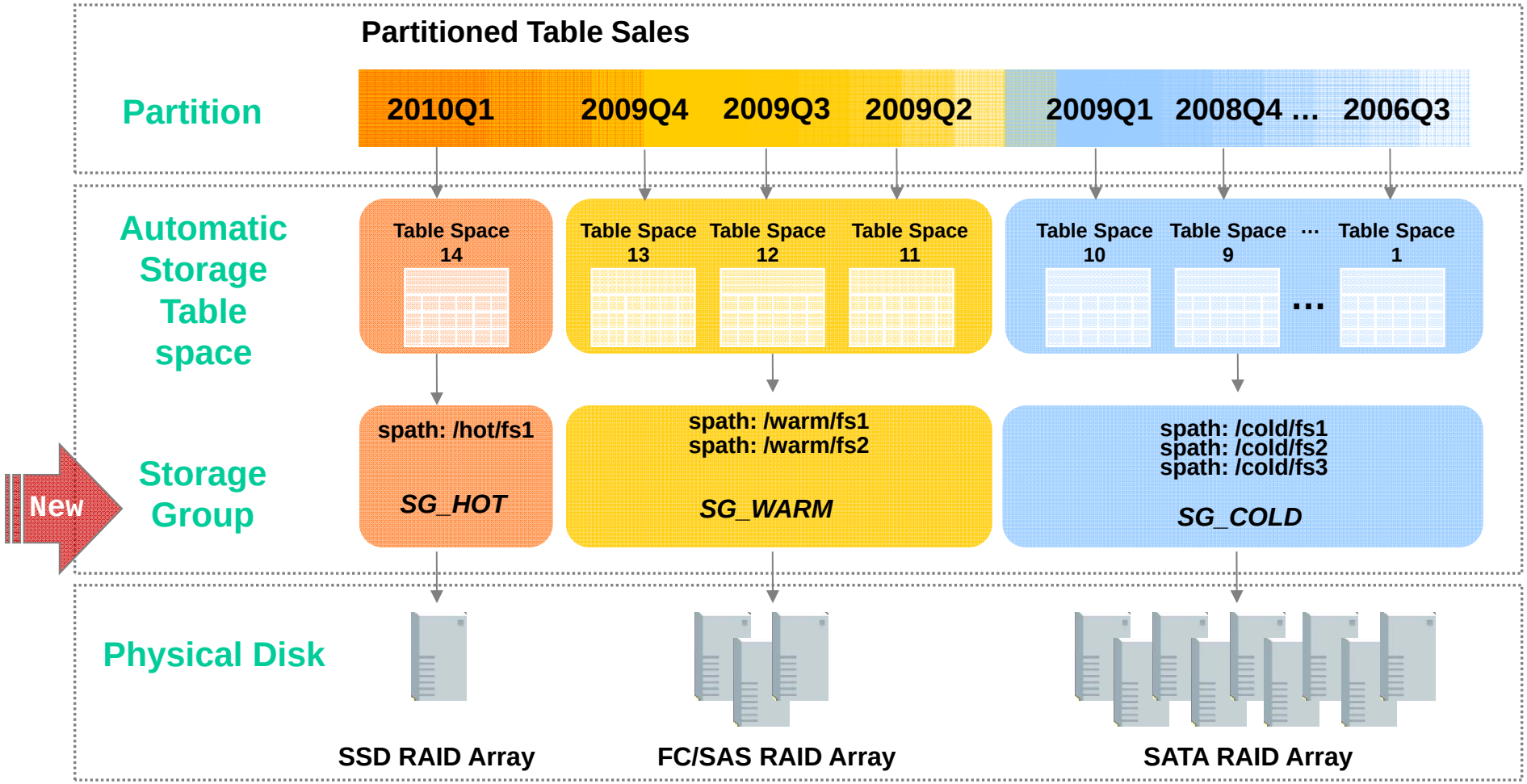
- Optimizing data into classes of storage becomes increasingly important in managing storage costs as a data warehouse grows in its amount of storage.
- Data priority can be based on:
  - Frequency of access
  - Acceptable access time, defined by QoS specified in Service Level Agreement (SLA)
  - Volatility of the data
  - Application requirements
- Quality of service is the ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow

Storage Groups

- Storage Groups allow the flexibility to implement Multi-temperature Data Management in Automatic Storage table spaces
- Different Storage Groups can represent different classes of storage
  - Hot data assigned to storage groups with fast devices
  - Warm or Cold data assigned to slower devices
- Easy maintenance when data ages and needs to be moved to a different storage class



Leveraging Storage Groups

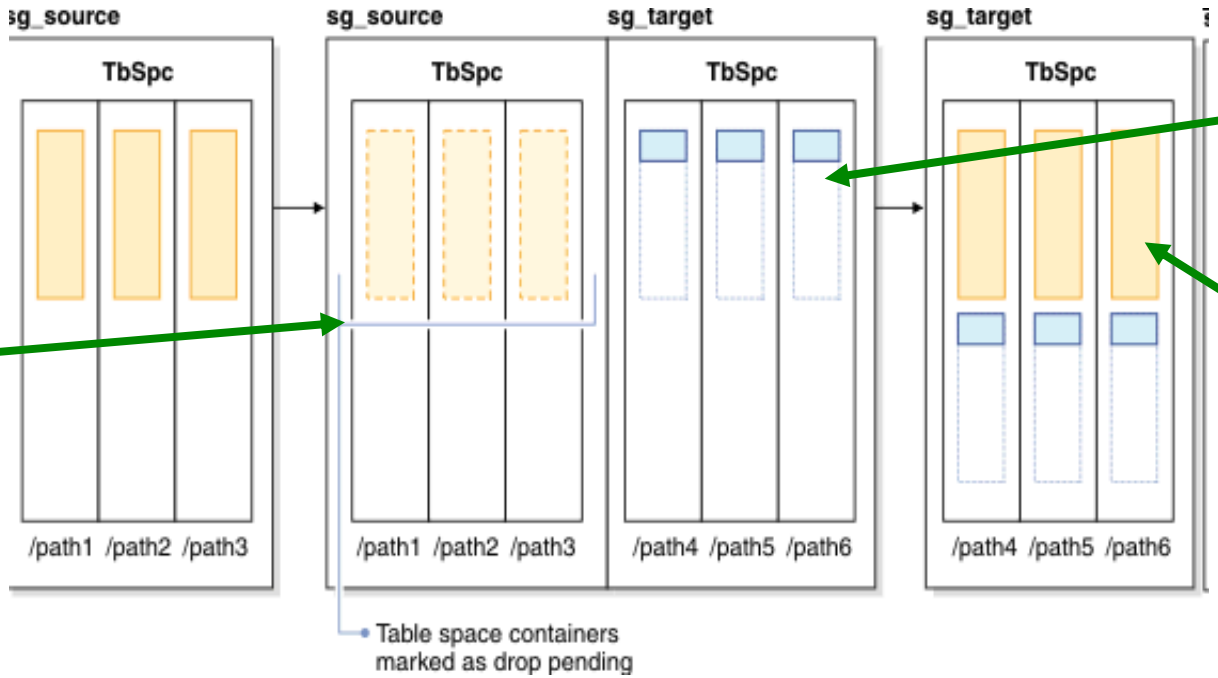


How to Change the Temperature of Your Data

- Use ALTER TABLESPACE to change the data temperature of table space

```
ALTER TABLESPACE tbSpc USING STOGROUP sg_target
```

- Implicit REBALANCE occurs when a table space moved between storage groups



(1) sg_source containers are marked dropped pending

(2) sg_target containers store all new data

(3) Reverse balance moves all data to sg_target

(4) Old containers are dropped

- Monitor the progress of REBALANCE by using the new monitoring table function MON_GET_REBALANCE_STATUS

Multi-temperature Storage Integrates with DB2 WLM

- Existing WLM perspectives are user-centric (who) and request-centric (what)
- Introducing a new perspective “data-centric” (where)
  - New data tag attribute
    - For storage group or table space
    - Priority can be given to requests based on what data is accessed [Values 1 (high) – 9 (low)]
- WLM work class and threshold DDL have been extended to support the new data tag attribute
  - DB2 optimizer can provide an estimated list of data tags for data touched by a query at compile
    - The data tag can influence the initial placement of the activity into a service class
- New Data tag threshold DATATAGINSC uses information that is available at runtime to remap an activity to a different service subclass

Multi-temperature Storage – A Sample Scenario

- GOAL: Reduce warehouse storage costs while meeting the desired Quality of Service requirements for access to last 3 quarters of data
- Step 1: Create two storage groups to reflect the 2 tiers of storage This would result in transfer rate, overhead, etc being programmatically computed at the storage group level.

```
CREATE STOGROUP sg_hot ON '/ssd/path1', '/ssd/path2' DATA TAG 1
CREATE STOGROUP sg_warm ON '/hdd/path1', '/hdd/path2' DATA TAG 5
```

Data tags represent business priority of the data and is used by the optimizer

- Step 2: Assign table spaces to storage groups

```
CREATE TABLESPACE q1_2011_tbsp USING STOGROUP sg_warm
CREATE TABLESPACE q2_2011_tbsp USING STOGROUP sg_warm  DATA TAG 3
CREATE TABLESPACE q3_2011_tbsp USING STOGROUP sg_hot
```

Multi-temperature Storage – A Sample Scenario

… A New Quarter Begins

- Create a new table space and change storage group for Q3 table space
  - Q4 table space will reside on hot storage
  - Q3 data will be moved and rebalanced across slower storage

```
CREATE TABLESPACE q4_2011_tbsp USING STOGROUP sg_hot
ALTER TABLESPACE q3_2011_tbsp USING STOGROUP sg_warm DATA TAG 3
```

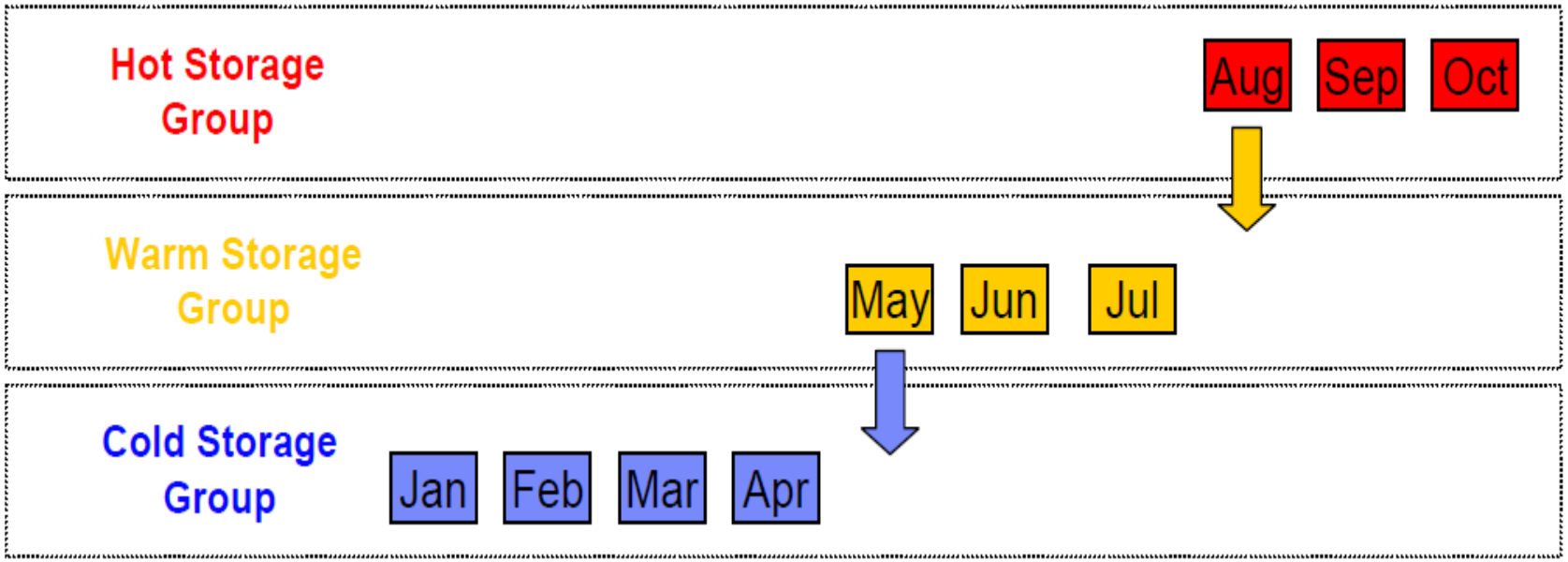
- Data Tag changed to allow optimizer to consider the changed data priority

```
ALTER TABLESPACE q2_2011_tbsp DATA TAG 5
```

• Only the most frequently accessed data resides on high-end expensive storage and meets the QoS requirements for that data access
• The bulk of the data resides on less expensive storage.
• Provides easy management by DBA's

Automate Data Aging Across Storage Tiers

- Using Optim Configuration Manager you can define a data migration job for your multiple storage tiers
  - Optimize the use of your storage by configuring a multi-temperature storage scheme
  - Data migration aging policies can be designed for one, some or all your partitioned tables with range partitions.



Oracle Storage Comparison

- **DB2**
  - Utilizes Automatic Storage and Table Partitioning to easily move table partitions between storage groups. (Free in Enterprise and AE Edition)
  - Simple, on-line operation to move data from one temperature storage to another
  - Utilizes standard OS file systems
  - Integrated with WLM to provide work load priority based on the data being accessed
- **Oracle**
  - Requires Oracle Partitioning (extra cost option for Enterprise Edition)
  - More script driven versus statement driven, not as simple as DB2 Multi-Temperature Storage
  - Requires Automatic Storage Management (ASM), does not use standard OS file systems
  - Oracle Database has no integration with Database Resource Manager and NO ability within it to prioritize queries based on data temperature

Summary

- Each database must have a minimal set of storage areas that are used for storing system, user and temporary data.
- A database must contain at least three storage areas:
  - A catalog area
  - One or more user areas
  - One or more temporary areas
- User table spaces managed by system are **deprecated**
  - You can still specify the SMS type for catalog table spaces and temporary table spaces
- Logical grouping introduced in DB2 10.1 – Storage Groups
- Multi-temperature data management with storage groups
  - Improves manageability of storage
    - Saves on storage costs while satisfying performance requirements
  - Tiered storage system
    - Degree of data access can correspond to the relative speed of the storage device
  - Ease of use
    - Storage groups as database objects are easy to implement and maintain
  - Improve performance
    - Fine tune the workload environment using data tag attribute for business priorities

The Next Steps…

The Next Steps…

- Complete the Hands on Lab for this module
  - Log onto SKI, go to “My Learning” page, and select the “In Progress” tab.
  - Find the module
  - Download the workbook and the virtual machine image
  - Follow the instructions in the workbook to complete the lab
- Complete the online quiz for this module
  - Log onto SKI, go to “My Learning” page, and select the “In Progress” tab.
  - Find the module and select the quiz
- Provide feedback on the module
  - Log onto SKI, go to “My Learning” page
  - Find the module and select the “Leave Feedback” button to leave your comments

Questions?
askdata@ca.ibm.com