IBM Education Assistance for z/OS V2R1

Item: Storage Tiers
Element/Component: DFSMShsm

Material is current as of June 2013

Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Presentation Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

Presentation Objectives

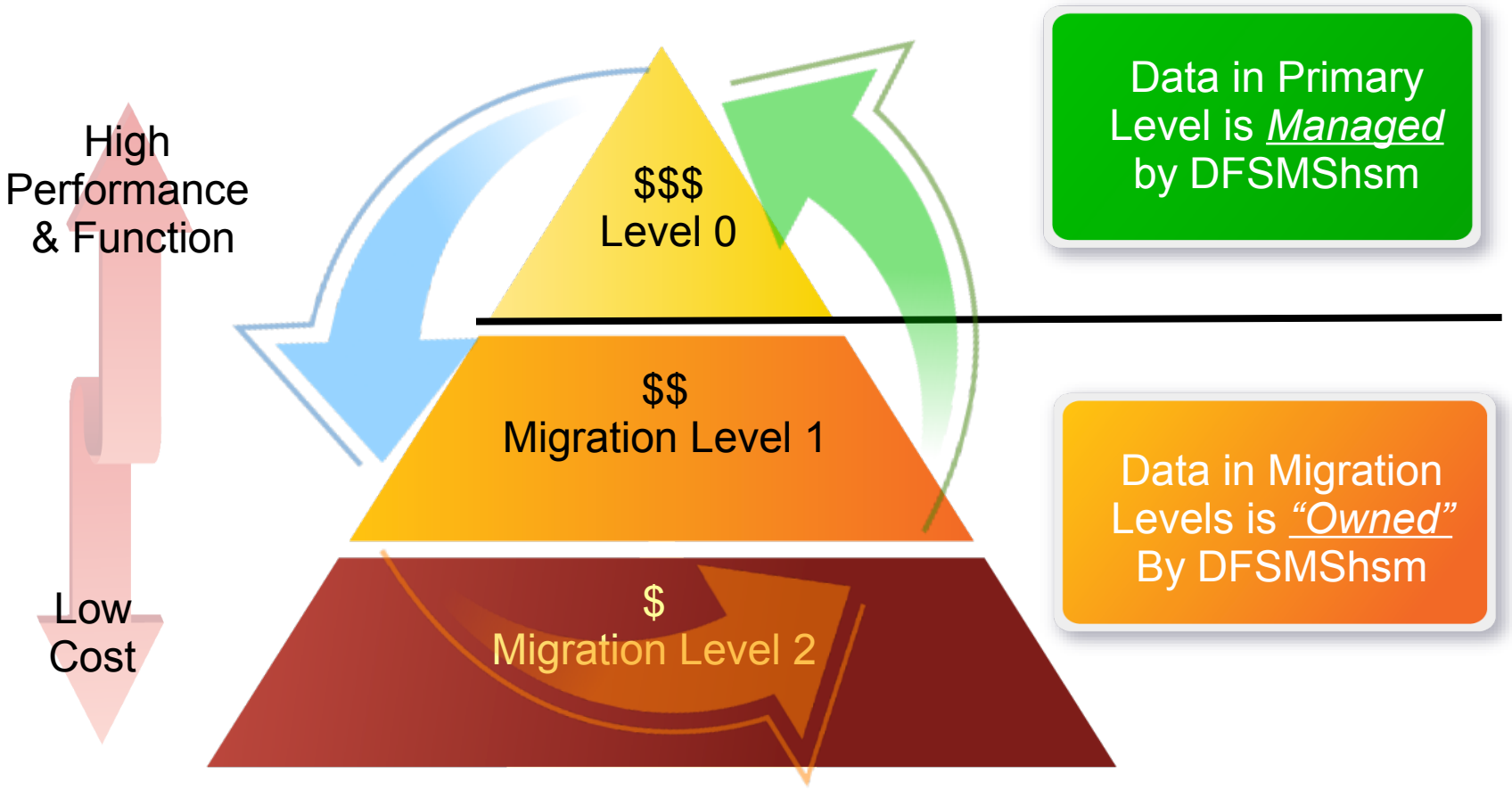
- Overview the role of DFSMShsm Space Management
- Describe how the new Storage Tiers function is integrated within the existing Space Management function
- Describe the types of data for which this function could be used
- Explain and give examples for invoking the function
- Describe the various options that are available

Overview

- Problem Statement / Need Addressed
  - We must align storage costs with a changing business value of information
  - No policy-based automation for moving data within the Primary Storage Hierarchy (Level 0)
  - No policy-based management of Active (open) data
- Solution
  - Storage Tiers - ***Automated, policy-based space management that moves data from tier to tier within the Primary (Level 0) Hierarchy***
- Benefit / Value
  - Better align storage costs with changing business value
  - Minimize the Total Cost of Ownership for System z data by actively managing data on the lowest cost storage that meets the business needs of the data

Background – DFSMS Today

The *classic* DFSMS storage hierarchy (30+ years old)



Background – DFSMS Today

- **Information life management is the process of managing information from creation through its useful life to its eventual deletion.**
  - **Today, DFSMS provides *policy-based*...**
    - Data Creation
    - Backup / Recovery Management
    - Space Management
    - Expiration
- **The Space Management Environment is evolving**
  - Typical configurations have changed to leave data on Level 0 longer and then migrate directly to ML2
    - When ML2 is a VTS, the VTS disk cache replaces the ML1 tier
      - Eliminates MIPS required for software compression to ML1
      - Eliminates DFSMShsm ML1->ML2 processing
- **Shortcomings of today's DFSMS functionality**
  - No policy-based automation for moving data within the Primary Storage Hierarchy (Level 0)
  - No policy-based management of Active (open) data

Overview

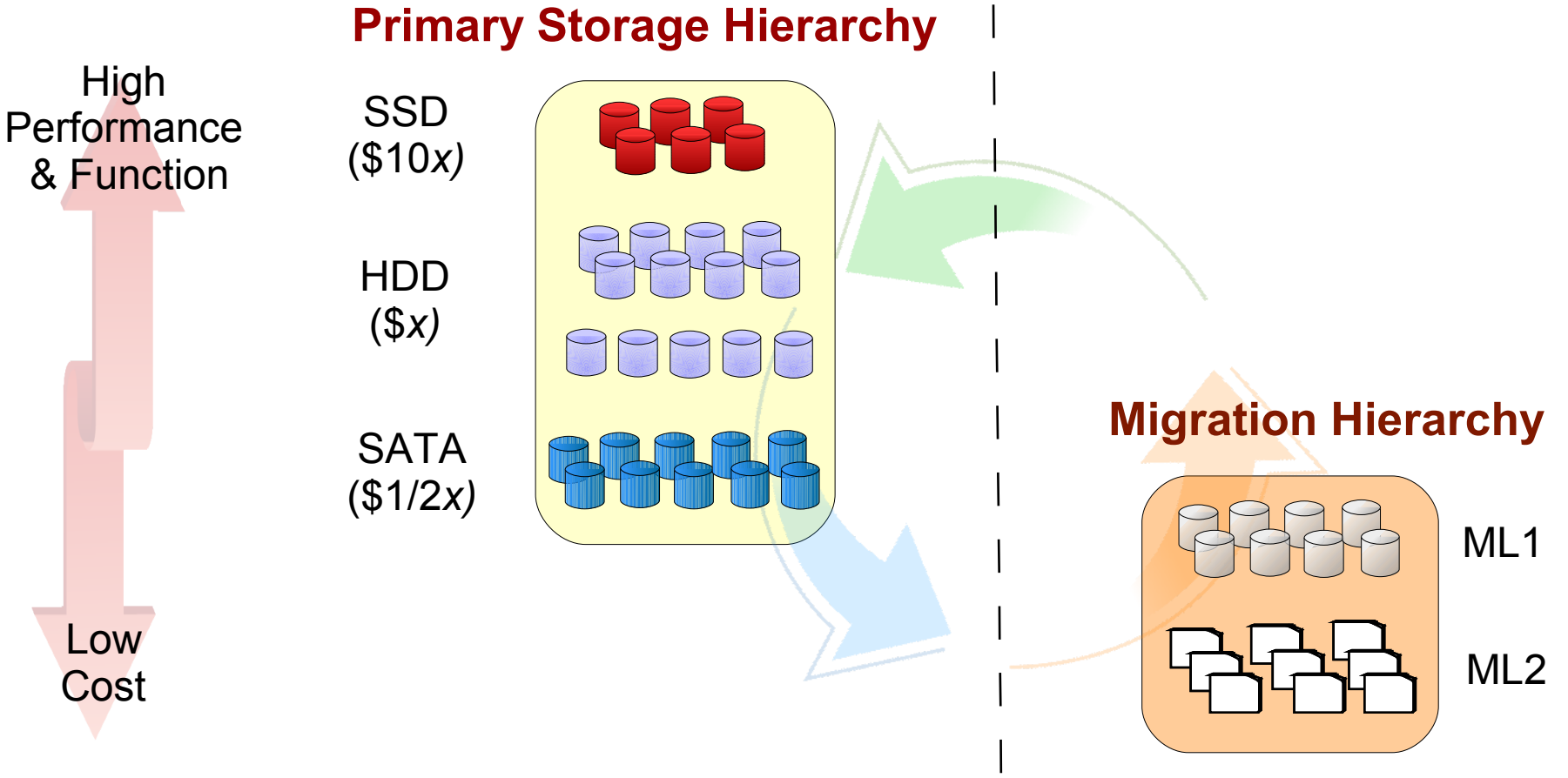
z/OS V2R1 DFSMS provides the Storage Tiers Solution

Automated, policy-based space management that moves data from tier to tier within the Primary (Level 0) Hierarchy

- Automated movement provided via the existing DFSMShsm Space Management function
  - *Movement is referred to as a 'Class Transition'*
  - Data remains in its original format and can be immediately accessed after the movement is complete
- Policies implemented via the existing Class Transition policies and updated Management Class policies
- Enhanced support for DB2, CICS and zFS data
  - Open data temporarily closed to enable movement

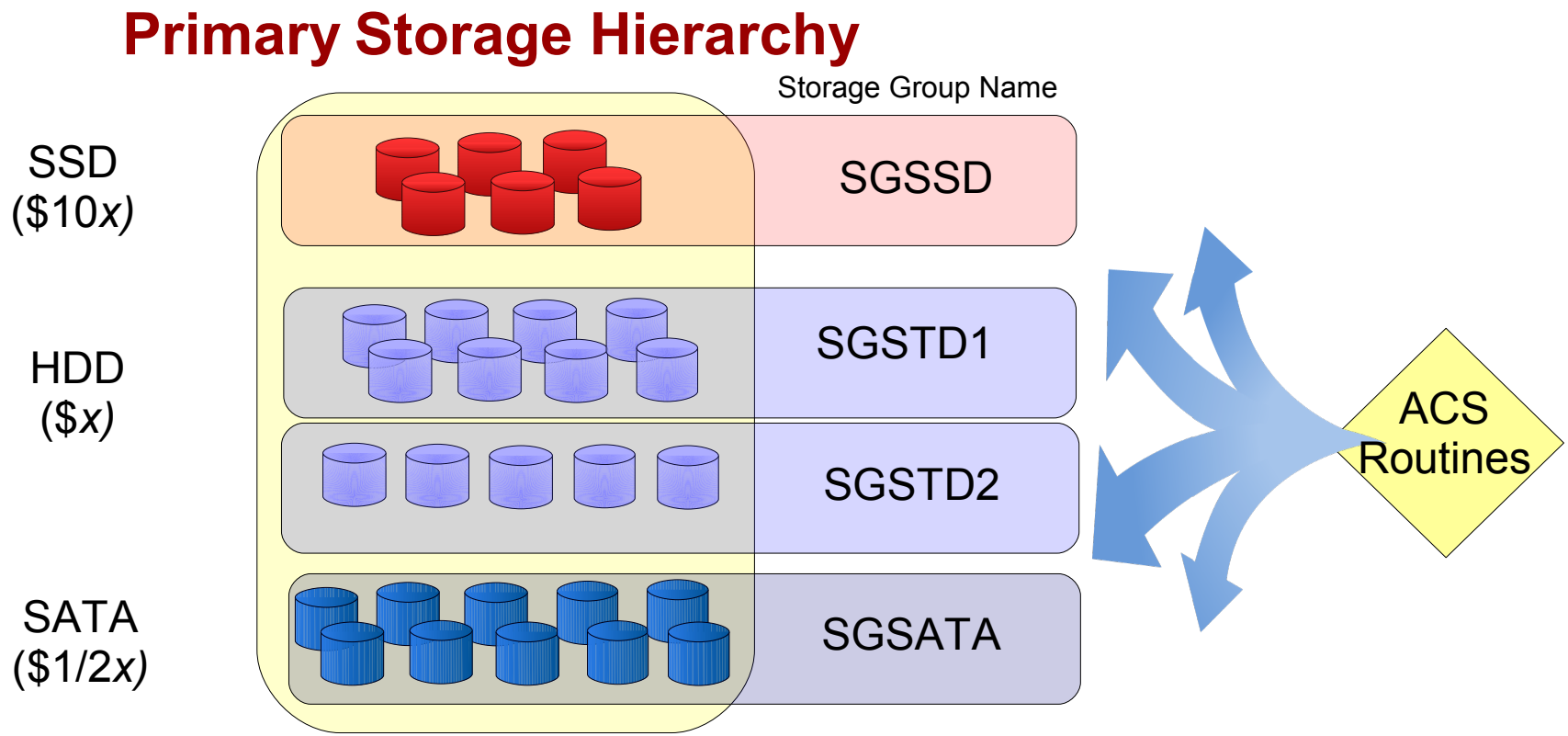
Overview

The *classic* DFSMS storage hierarchy is modified to represent *distinct* Primary and Migration Hierarchies...

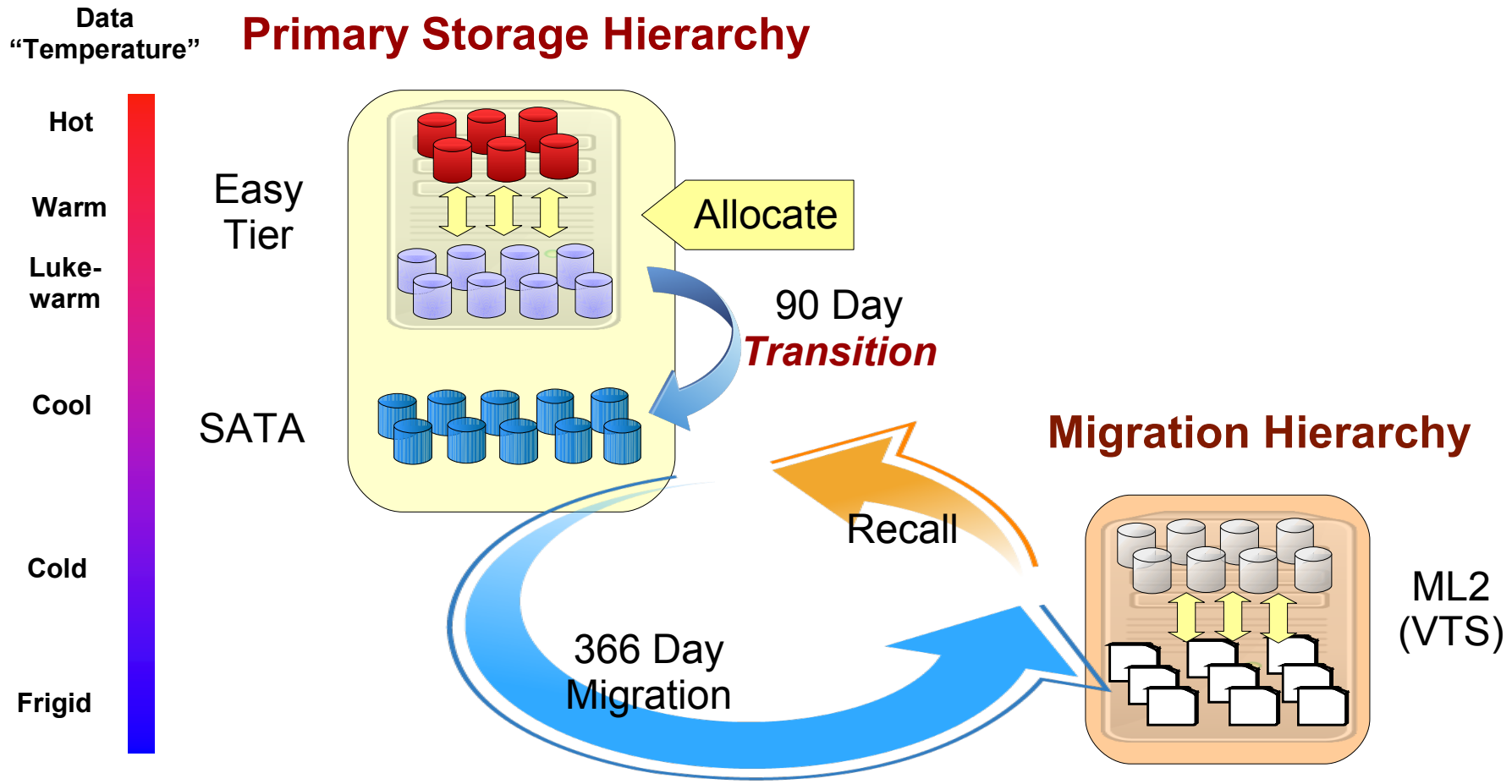


Storage Tiers Overview

The Primary Storage Hierarchy is not new. It is the existing hierarchy of storage as defined by Storage Class and Storage Group policies and SMS Automatic Class Selection (ACS) routines.



This example environment shows data being “transitioned” from an Easy Tier device to standard SATA devices after 90 days, and then becoming eligible for migration after 366 days of inactivity.



Usage

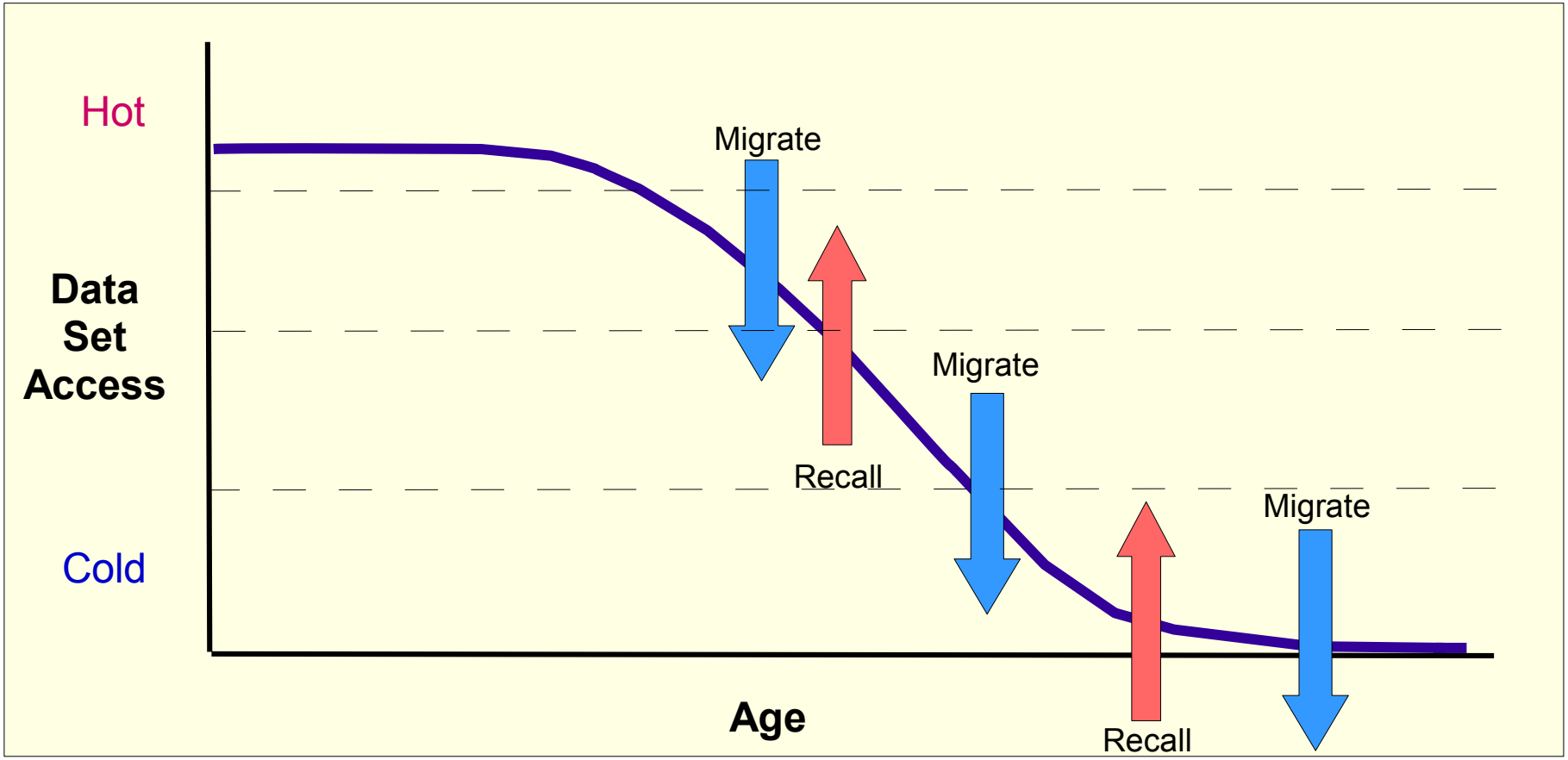
- **What are examples of data sets that would benefit from this solution?**

  - Data sets currently not eligible for migration because they always need to be immediately accessible
    - Recall delay is unacceptable
    - Data sets could be allocated on a particular class of storage and then later *transitioned* to a less expensive class of storage for permanent retention

  - Data sets that are eligible for migration today, but there would be a benefit to keep them online for a longer period of time.
    - Convert the migration of data sets to *transition* to a lower cost storage.
    - Increase the number of days that the data set must be unreferenced before migrating directly to ML2

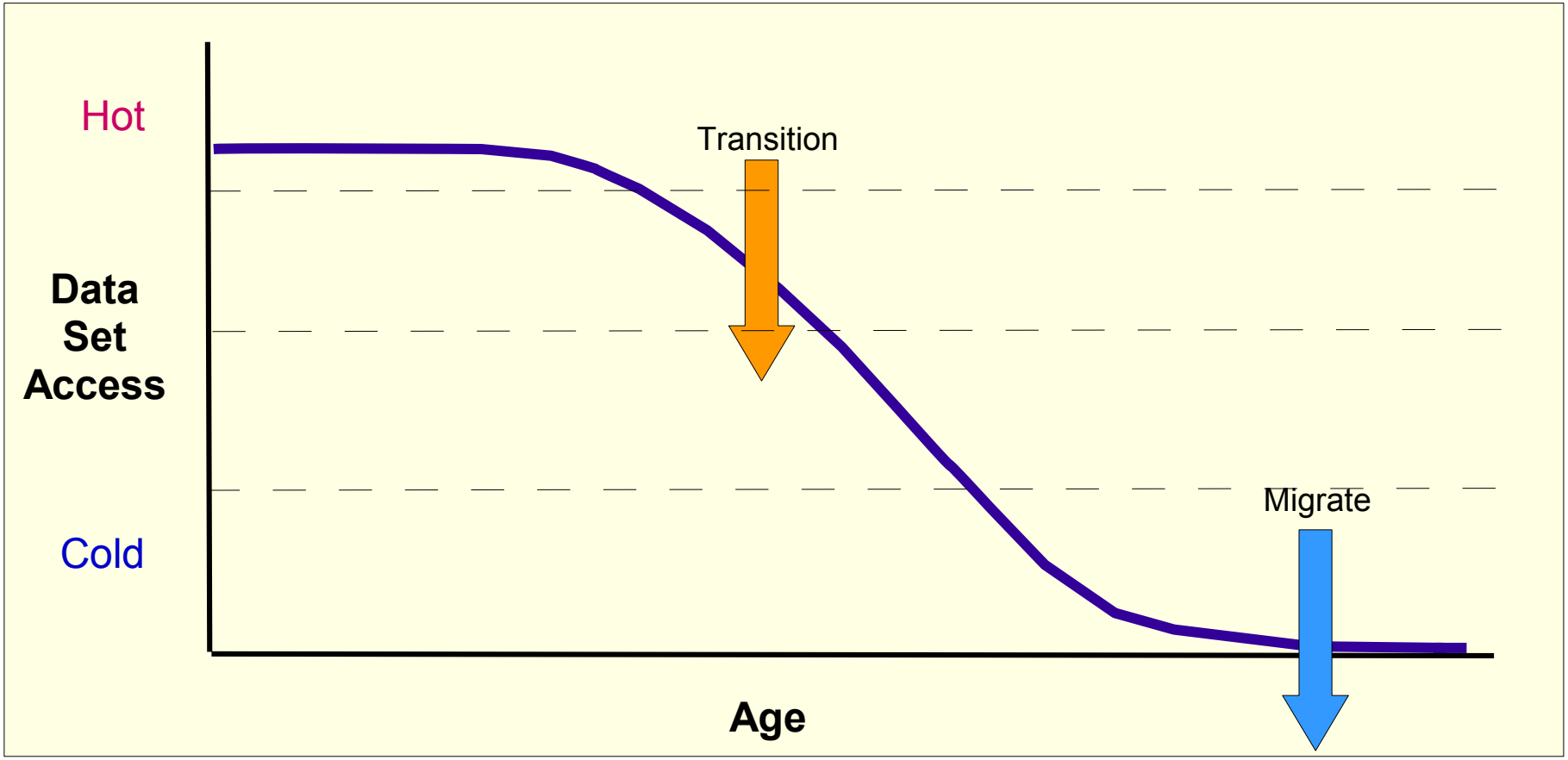
Usage

As a data set ages and goes through cycles of activity and inactivity, it can go through many migration and recall iterations.



Usage

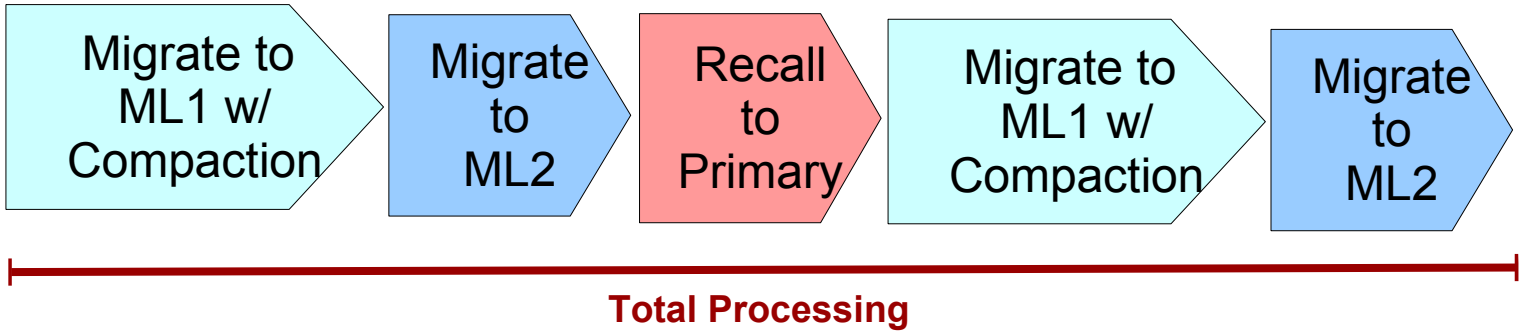
The migration / recall iterations can be replaced with a single class transition and potentially single migration.



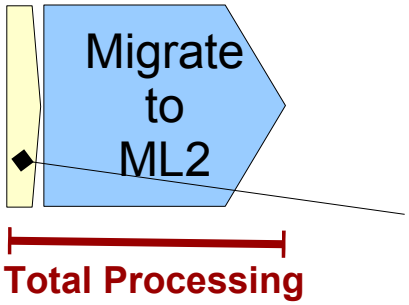
Usage

Contrast the DFSMShsm processing required when ...

- A data set is migrated to ML1 → ML2, recalled and remigrated just once



- A data set is transitioned using FlashCopy and then migrated directly to ML2



•**Transition** with FlashCopy uses minimal host MIPS. Data movement is done within the storage controller.

Usage Notes

- **There is a distinction between using Migrate/Recall and Class Transitions**
  - When a data set is recalled, it will be returned to the class of storage as directed by the ACS routines, which would typically be higher than where a data set would be transitioned to
  - When a data set transitions to a lower class of storage, it will remain there until it is transitioned again or until it migrates

- **In order for FlashCopy to be used for a transition,** the movement must be within the same storage controller. This may be difficult to achieve.
  - FlashCopy will be discussed in detail later in the presentation

Invocation

- The function of *DFSMShsm Space Management* processing is to use *policy-based automation* to ensure that volumes within the Primary Storage Hierarchy have enough *free space* for new data and to ensure that data is stored at the *lowest acceptable tier in the Storage Hierarchy*
  - This is accomplished through
    - Data set expiration
    - Migration of unreferenced data to the Migration Hierarchy
  - ✔ "Class Transitions" within the Primary Hierarchy
- "Class Transition" processing is integrated into the existing DFSMShsm *Space Management* functions
  - Primary Space Management
  - On Demand Migration
    - *New function introduced in V1R13. Performs space management on a volume as soon as it goes over high threshold. Replacement for on-the-hour Interval Migration processing.*
  - Interval Migration

Invocation

- When a volume is selected for *space management processing* due to being over threshold, in addition to existing expiration and migration checking, space management functions will determine if a data set is eligible to be *transitioned*, based on *management class criteria*
- *Phases of space management processing*
  - **Phase 1** *(non-movement processing)*
    - Expiration processing
    - Release unused space
    - Fast Subsequent Migration
  - **Phase 2** *(if volume is still over threshold, data is moved)*
    - Class Transitions
      - Data sets eligible for both a transition and migration will be migrated
      - Otherwise, transitions will be processed before migrations
    - Migrations

★As today, volume processing ends when the low threshold is met or no more data sets are eligible to be processed

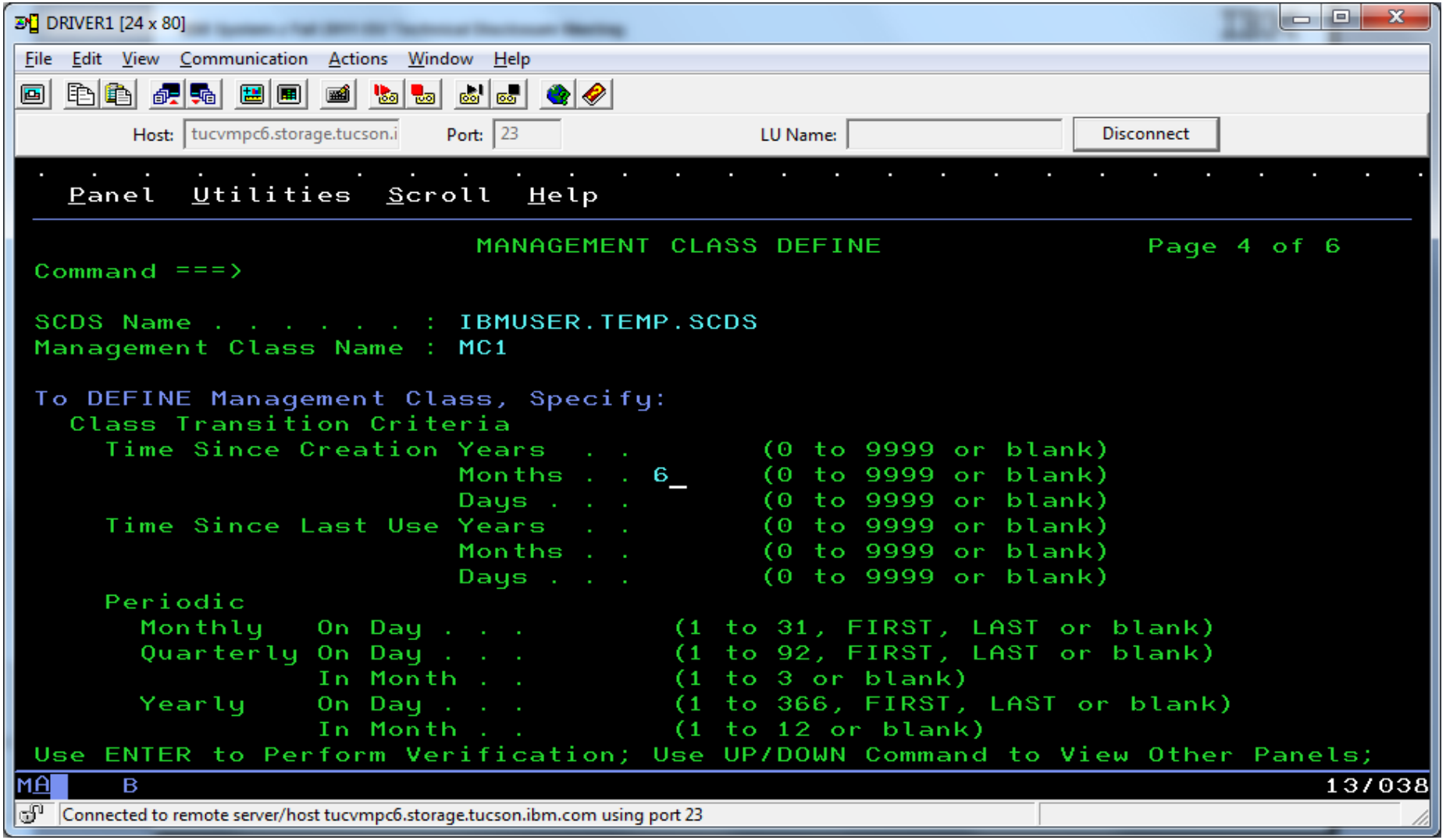
Invocation

- The SMS Management Class provides the Class Transition policies:
  - **Class Transition Criteria:** If and when a data set should be transitioned
  - **Serialization Error Exit:** Indicates what type of special processing should occur if the data set cannot be serialized
  - **Transition Copy Technique:** Which copy technique should be used to move the data set

Invocation

- The Class Transition Criteria specifies if and when a data set should be transitioned.
  - **Default:** Class transitions are *not* performed
  - **Time since Creation:** Data set is eligible for a transition on or after this time.
    - This is a *subjective* setting. It indicates that regardless of the usage of the data set, it should be transitioned.
  - **Time since Last Use:** Data set is eligible for a transition on or after this time.
    - This is an *objective* setting. It indicates that a data set should not be transitioned until it has not been referenced for a certain period of time.
  - **Periodic:** Data set is eligible for a transition on a specific date.
    - This is a *subjective* setting. It indicates that regardless of the usage of the data set, it should be transitioned.
- Only one criterion may be specified.

Class Transition Criteria



Invocation

- The Serialization Error Exit indicates what type of special processing should occur if the data set cannot be exclusively serialized
  - Since **'Time since Creation'** and **'Periodic'** may specify that a transition occur during a period of time when a data set is being accessed, this setting specifies what to do if a data set cannot be exclusively serialized for data movement
  - For database data, it may be expected that the data is always open and special processing must be done to transition the data at any time

- Since it is expected that data sets may be open, the default is to not issue an error message if a data set cannot be exclusively serialized, it is just skipped *(similar to migration processing)*
  - To issue a DFSMShsm message for this condition, issue PATCH .MGCB.+EF BITS(. . .1 . . . .)
  - An FSR record is created to enable reporting for this condition

Invocation

- Serialization Error Exit settings
  - **NONE:** (Default) If a data set cannot be exclusively serialized, the data set is not transitioned
  - **DB2:** Data assigned to this management class are DB2 objects
    - Invoke DB2 to close and unallocate the object. If this is successful, the object is serialized and moved and DB2 is invoked to reopen the object
  - **CICS:** Data assigned to this management class are CICS objects
    - Invoke CICS to take the object offline. If this is successful, the object is serialized and moved and CICS is invoked to reopen the object
  - **zFS:** Data assigned to this management class are zFS data sets
    - Invoke zFS to unmount the data set. If this is successful, the data set is serialized and moved and zFS is invoked to remount the data set
  - **EXIT:** User exit is invoked to preprocess and post-process the data set
    - Enables users / ISVs to provide an exit that will be invoked before and after transitioning an allocated data set. The data set is only transitioned if serialization is obtained after the first invocation of the exit
    - Documented in *DFSMS Installation Exits (SC23-6850-0)*

Invocation

- The Transition Copy Technique indicates what type of copy technique should be used to move the data set
  - **Standard:** (Default) Use standard I/O
  - **Fast Replication Preferred:** Prefer Fast Replication. If it cannot be used, then use standard I/O.
  - **Fast Replication Required:** Require Fast Replication. If it cannot be used, fail the data movement.
    - Requires the target volume to be in the same storage controller.
  - **Preserve Mirror Preferred:** Prefer Preserve Mirror. This indicates that a Metro Mirror primary volume is allowed to become a FlashCopy target volume. If Preserve Mirror cannot be used, FlashCopy or standard I/O may be used.
  - **Preserve Mirror Required:** Require Preserve Mirror. The transition is only performed if the Metro Mirror primary target volume will not go duplex pending. This parameter has no affect if the target volume is not a Metro Mirror primary volume.

Invocation

- Transition Copy Technique
  - If a copy technique *other than* 'Standard' is specified, then a valid backup copy must exist before the data set is transitioned
    - This is required because DFSMShsm receives control immediately after the FlashCopy relationship is established.
    - DFSMShsm is not notified if there is a physical error within the storage controller during the background physical copy
    - While very unlikely, this ensures that if any physical error occurs while transitioning the data, that a backup copy is available to recover the data set

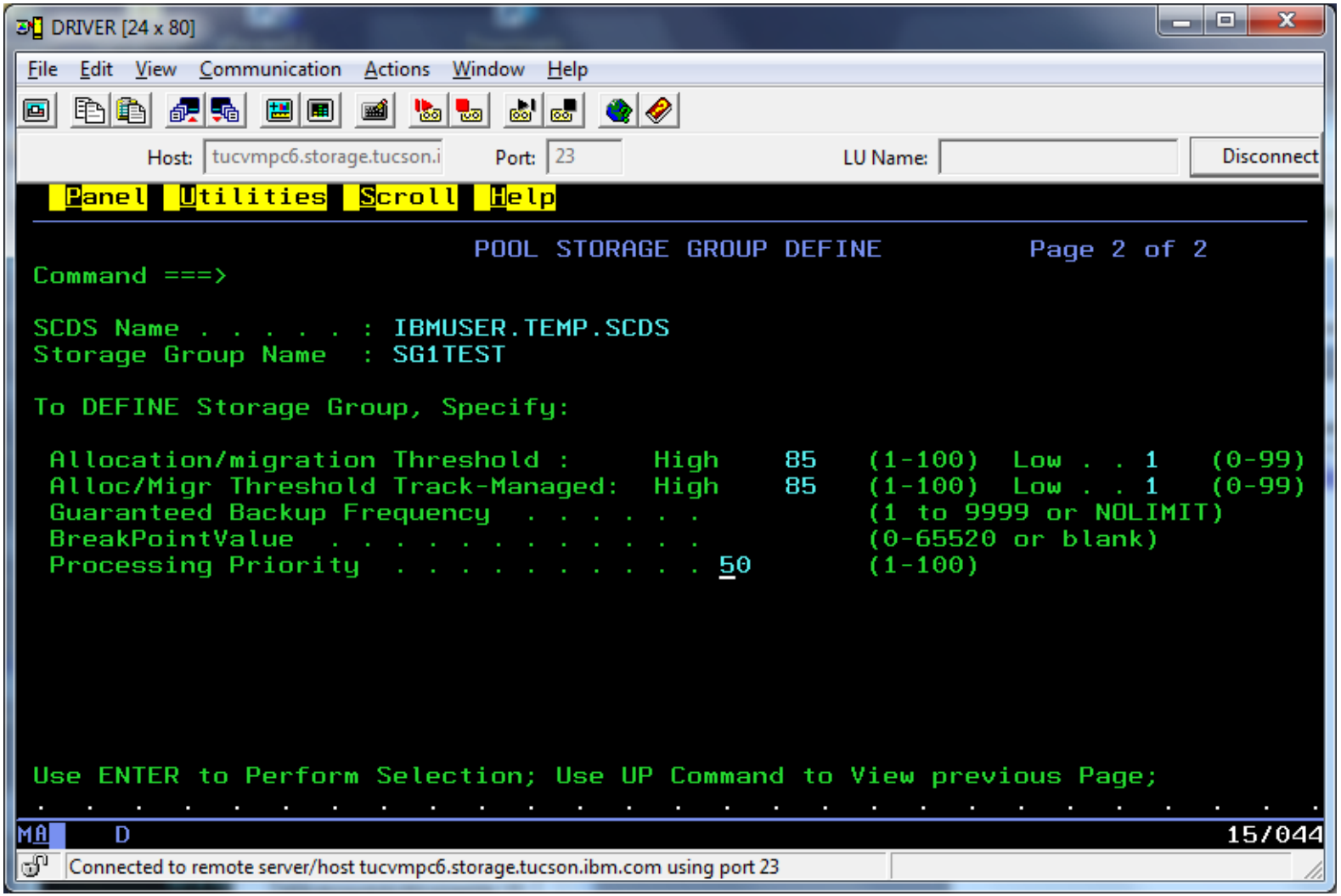
      This is critical because the data set is Deleted after the logical completion

Invocation

- The new Storage Group Processing Priority specifies the relative order in which storage groups should be processed during Primary Space Management
  - *Class Transition processing will move data from one storage group to another*
  - In order to help ensure that the 'receiving' storage groups have enough space for the data sets that will be moved to them, a new storage group Processing Priority is provided
    - These storage groups should be assigned a higher priority
  - Storage Groups will be processed in the order of their priority
    - A higher value means a higher priority
    - Valid values are 1-100
    - Default value is 50

✓ Only applies to Primary Space Management processing

Storage Group Processing Priority



Invocation

- Once DFSMShsm determines that a data set has met the Class Transition criteria specified by the Management Class, it invokes the ACS routines to determine what the transition should be
  - ACS Routines are invoked with new ACS environment (&ACSENVIR) of SPMGCLTR, for 'space management class transition'
  - The following routines are invoked
    - Management Class
    - Storage Class
    - Storage Group
  - Any or all can be *transitioned*

- If the classes and storage group returned match the existing classes and storage group, *then no transition occurs*

Invocation

- Management Class
  - When a new management class is assigned, DFSMShsm will begin using the newly assigned policies to manage the data set
  - If only the management class changes, then the data set is altered to assign it to the new management class and *no data movement is performed*

  - *Example*
    - Upon creation, a data set is assigned to a management class for which the data set is only eligible to migrate to ML1 (not ML2) and 2 backup copies are maintained

    - After 120 days from creation, the data set is transitioned to a different management class for which the data set is eligible to migrate to ML2 and only 1 backup copy is maintained.

Invocation

- Management Class

```
IF &ACSENVIR = 'SPMGCLTR' THEN
/*  SPACE MANAGEMENT CLASS TRANSITION  */

   SELECT (&MGMTCLAS)
      WHEN ('NOML2')    SET &MGMTCLAS = 'ML2OK'
      WHEN ('DB2NEW') SET &MGMTCLAS = 'DB2OLD'
      OTHERWISE         SET &MGMTCLAS = &MGMTCLAS
   END
ELSE
   ...
```

Invocation

- Storage Class
  - Storage Class indicates the *'preferred'* class of storage to which the data set should be allocated
  - If storage class changes, but storage group is the same, *and a device matching the new storage class attributes cannot be selected*, the data set is *not* moved

```
IF &ACSENVIR = 'SPMGCLTR' THEN
   SELECT (&STORCLAS)
       WHEN ('SSD')         SET &STORCLAS = 'EASYTIER'
       WHEN ('EASYTIER') SET &STORCLAS = 'SATA'
       OTHERWISE            SET &STORCLAS = &STORCLAS
   END
 ELSE ...
```

Invocation

- Storage Group
  - From 1 to 15 storage groups may be returned
    - It is the administrator's responsibility to ensure that a different storage group name provides a meaningful transition

```
IF &ACSENVIR = 'SPMGCLTR' THEN
   SELECT (&STORCLAS)
      WHEN ('EASYTIER') SET &STORGRP = 'EASYTIER'
      WHEN ('SCSATA')   SET &STORGRP = 'SGSATA'
      OTHERWISE         SET &STORGRP = &STORGRP
END
ELSE ...
```

Invocation

- When DFSMShsm determines that a data set should be moved for a Class Transition, DFSMSdss is invoked to perform a Logical COPY with DELETE
  - DFSMSdss is the full data mover
    - Unlike migrate/recall and backup/recover where DSS is only the half data mover
    - DFSMSdss handles Copy Technique and Exit processing
    - After the movement, the data set retains all existing attributes and can be immediately accessed

      The catalog is updated as a part of the movement
  - *No* new DFSMShsm control data set records created for transitions
    - New FSR record type created for reporting purposes

      FSRTYPE = 24

      New DFSMS Report Generator *sample report* provided
    - DSR and VSR records are updated for DFSMShsm REPORT

Invocation

- REPORT FUNCTION(TRANSITION) SYSOUT(A)
  - 'TRANSITION' is a new option

```
--DFSMSHSM STATISTICS REPORT ------- AT 08:05:07 ON 2011/09/13 FOR SYSTEM=2094
DAILY STATISTICS REPORT FOR 11/09/13
STARTUPS=000, SHUTDOWNS=000, ABENDS=000, WORK ELEMENTS PROCESSED=000005, BKUP VO
DATA SET MIGRATIONS BY VOLUME REQUEST= 0000000, DATA SET MIGRATIONS BY DATA SET
EXTENT REDUCTIONS= 0000000 RECALL MOUNTS AVOIDED= 00000 RECOVER MOUNTS AVOIDED=
DATA SET CLASS TRANSITION = 00000085 REQUESTED, 00000002 FAILED
                      NUMBER  ------READ--------  -----WRITTEN------  ------REQUES
   HSM FUNCTION     DATASETS TRK/BLK    BYTES    TRK/BLK    BYTES    SYSTEM USER
CLASS TRANSITION
 PRIMARY - PRIMARY 0000083  00053161 000632776K 00053161 000632776K 000085 00000
```

Invocation

- A new field was created in the NVR/VVR (catalog entry) called 'Last Successful Class Transition Date' (LSCTD)
  - When a data set is successfully transitioned, LSCTD is...
    - set to zero when the management class was changed

      -or-
    - set to the current date if the management class was not changed
  - LSCTD is used by DFSMShsm to know when a data set has already been successfully transitioned
  - DFSMShsm will attempt to transition a data set if it has met the transition criteria *AND* the LSCTD is zero
  - Exception for PERIODIC transitions. Data set will transition if the last transition was before the specified period

Invocation

- 'Last Successful Class Transition Date' (LSCTD) shown in LISTCAT output:

```
NONVSAM ------- STORTIER.M01.S01.D01.N01.PSFB
    IN-CAT --- STRTRFVT.USERCAT
    HISTORY
      DATASET-OWNER-----(NULL)     CREATION--------2012.001
      RELEASE----------------2     EXPIRATION------0000.000
      ACCOUNT-INFO-----------------------------------(NULL)
    SMSDATA
      STORAGECLASS ---SCLASS22     MANAGEMENTCLASS-MCLASS01
      DATACLASS ------DCLASS01     LBACKUP ---2012.001.0701
      LAST TRANSITION-2012.013
    VOLUMES
      VOLSER------------LSMS12     DEVTYPE------X'3010200F'
    ASSOCIATIONS--------(NULL)
    ATTRIBUTES
```

- Field is also available via DCOLLECT

Invocation

- Basic support was added to the **MD Exit** to prevent a data set from being processed
  - MD Exit will be referred to as the 'Space Management' Exit
  - New flag MDQEFCTR will be passed to indicate that the exit is being invoked for a data set that is Transitioning
  - New RC52 indicates that a data set should not be transitioned or migrated
    - Data set movement will be failed with ARC1245I RC92

Invocation

- RECALL Processing
  - When a data set is recalled, DFSMShsm will determine if a data set missed one or more transitions while the data set was migrated
    - If a transition was missed, then DFSMShsm will invoke the ACS routines for each transition that was missed, to determine the appropriate management class, storage class and storage group to which the data set should be recalled
  - If a migrated data set was transitioned before migration, the DFSMShsm will invoke the ACS routines with SPMGCLTR to ensure that the recall is to the correct device

Invocation

- Since Class Transitions are a part of the existing space management function...
  - The same tasking level that controls data set migrations controls data set transitions
    - If you expect more work during an existing space management window, then you may consider increasing your tasking level
  - The same HOLD, RELEASE, CANCEL, etc commands for space management control Class Transition processing also
  - ARC0734I (standard space management message) issued for class transitions
    - Insert updated to show 'ACTION=CLASS-TR'
    - Indicates From volume, To volume, RC, RSC, etc

Invocation

- *Thinking through processing DB2, CICS and zFS data...*
  - Ensure that only DB2, CICS or zFS data sets are assigned to a management class with a Serialization Exit specified to one of these types
  - It may be desirable to only perform Serialization Exit processing for these data set types during Primary Space Management (after hours processing)
    - Disable On Demand Migration and Interval Migration for these data set types
      - or -
    - Use new SETSYS *(described on next slide)* to indicate that ODM and IM should not perform Serialization Exit processing

Invocation

SETSYS CLASSTRANSITIONS(EVENTDRIVENMIGRATION(Y|N SERIALIZATIONEXIT(Y|N)))

- This SETSYS indicates if Event Driven Migration (On Demand Migration and Interval Migration) should process class transitions
  - The default value is 'Y'es
- If 'Y'es is specified (or defaulted), then SERIALIZATIONEXIT indicates if Serialization Exit processing should be performed
  - The default is 'N'o
  - This means that any open data sets will be skipped during ODM and IM if exclusive access cannot be obtained for data movement
- Aliases:
  SETSYS CLTR(EDM(Y|N SERL(Y|N)))

Invocation

- NaviQuest sample jobs and clists were modified to support the updates to management class and storage group settings in ISMF
  - This includes the following jobs/clists:
    - ACBJBAIO – MC list and report
    - ACBJBAIQ – MC report
    - ACBJBAIX – SG list and report
    - ACBJBAIZ – SG report
    - ACBJBAJ1 – MC define/alter/display
    - ACBJBAJ2 – SG define/alter/display

Use Case

Monthly customer statements are allocated on standard HDD disk. After 45 days, the statements should remain online, but should be transitioned to SATA disk in order to minimize the cost of storing the statements online.

Implementation Steps

- Ensure one or more storage groups comprised of SATA disk are setup/available to receive the transitioned statements
- Update the storage group ACS Routines with the appropriate class transition logic.
  - For example, when the ACS Environment is a class transition, if the management class is 'ONLNSTMT', then assign to a storage group comprised of SATA disks.
- Update the management class to which these statements are assigned to have a class transition occur 45 days after creation
- During Space Management, DFSMShsm will transition eligible statements from standard HDD to SATA
- Be Careful, don't transition too much at once!

Interactions & Dependencies

- **Software Dependencies**
  - No DB2, CICS or zFS dependencies
  - Existing interfaces for these products are used for the Serialization Exit processing
- **Hardware Dependencies**
  - None
- **Exploiters**
  - The Serialization Exit provides special processing for open DB2, CICS and zFS data
  - Serialization Exit (EXIT) enables ISVs to provide specific pre/post processing for their unique data

Migration & Coexistence Considerations

- **Migration Actions**
  - None
- **Coexistence**
  - Coexistence APARs OA36576 (HSM) and OA37582 (DSS) are required
  - Pre-V2R1 DFSMShsm will not process transitions
  - ** Migrated data sets that have gone through a transition cannot be recalled on a preV2R1 release because the LSCTD would be lost.
    - In a CRQ environment, these Recalls will only be selected by V2R1 hosts
    - In a nonCRQ environment, transitioned data sets should not be migrated until all hosts are at V2R1 or higher
  - Recovering a transitioned data set (with HSM or DSS) on a pre-V2R1 release will cause the LSCTD to be lost (Message ADR556W)
  - MCDS record size must be a minimum of 2040 bytes
  - EIREC22 structure in ADREID0 is larger

Appendix

- **Publications for z/OS V2R1.0**
  - *DFSMS Using New Functions* (SC23-6857-0)
  - *DFSMShsm Storage Administration* (SC23-6871-0)
  - *DFSMShsm Implementation and Customization Guide* (SC23-6869-0)
  - *DFSMSdss Storage Administration* (SC23-6868-0)
  - *DFSMS Installation Exits* (SC23-6850-0)
  - *DFSMSdfp Storage Administration* (SC23-6860-0)
  - *DFSMSdfp Diagnosis* (SC23-6863-0)
  - *MVS System Messages, Vol 2 (ARC-ASA)* (SA38-0669-0)