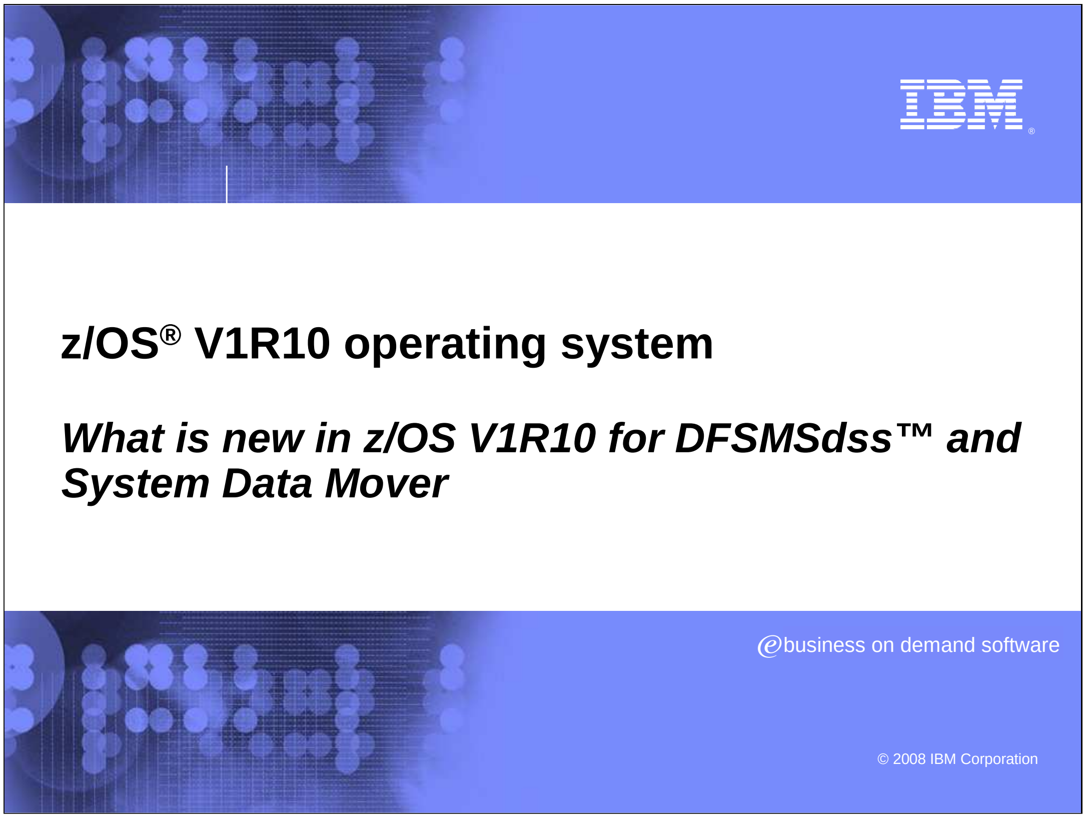
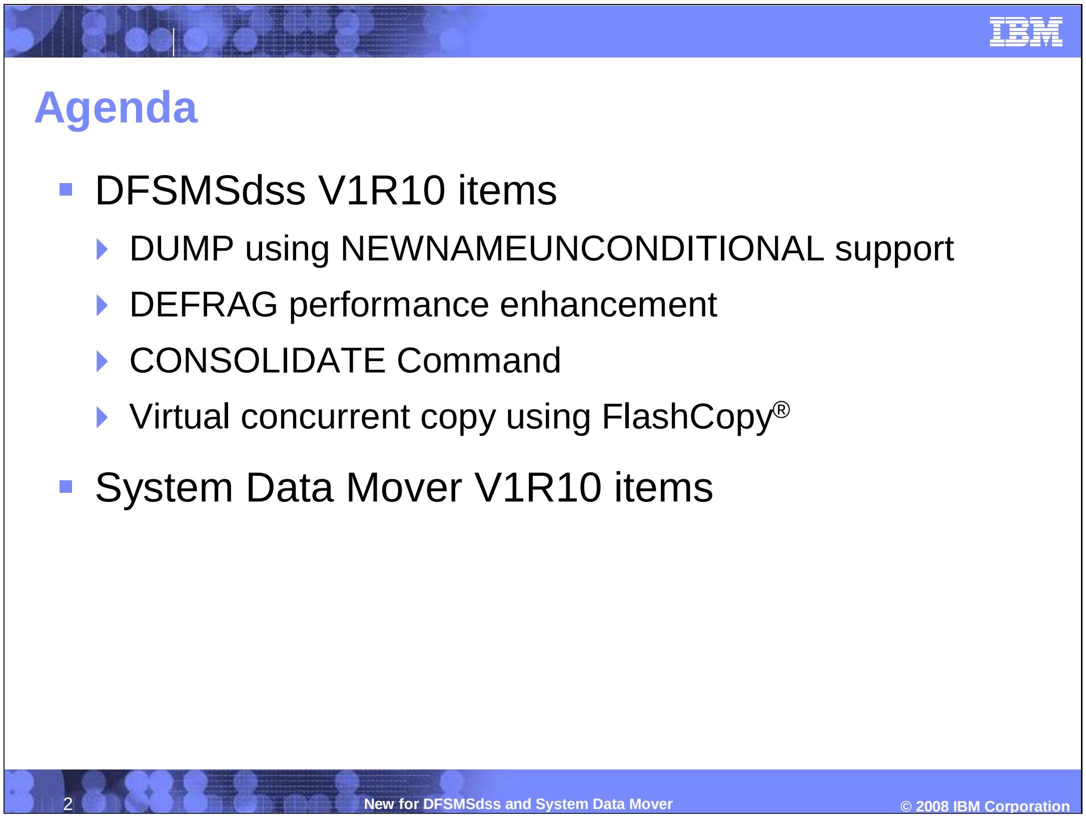


z/OS® V1R10 operating system

What is new in z/OS V1R10 for DFSMSdss™ and System Data Mover

*e*business on demand software

This presentation covers what is new in z/OS V1R10 for DFSMSdss and System Data Mover



Agenda

- DFSMSdss V1R10 items
  - DUMP using NEWNAMEUNCONDITIONAL support
  - DEFRAG performance enhancement
  - CONSOLIDATE Command
  - Virtual concurrent copy using FlashCopy®
- System Data Mover V1R10 items

The agenda for this presentation will cover recent items that have been made available in z/OS V1R10. The first part of the presentation will cover the following DFSMSdss items delivered in this release.

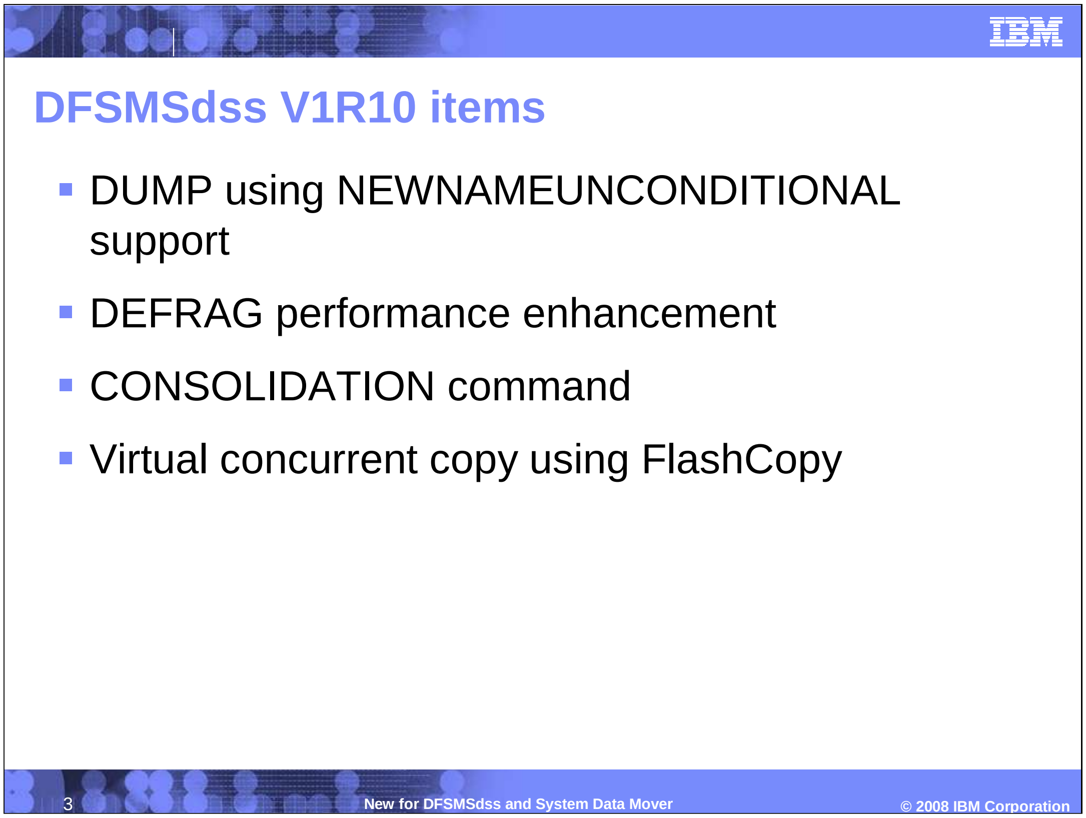
DUMP using NEWNAMEUNCONDITIONAL Support

DEFRAG Performance Enhancement

CONSOLIDATE Command

Virtual Concurrent Copy using FlashCopy

Following the DFSMSdss items will lead into the System Data Mover updates that were also made available in z/OS V1R10.



DFSMSdss V1R10 items

- DUMP using NEWNAMEUNCONDITIONAL support
- DEFRAG performance enhancement
- CONSOLIDATION command
- Virtual concurrent copy using FlashCopy

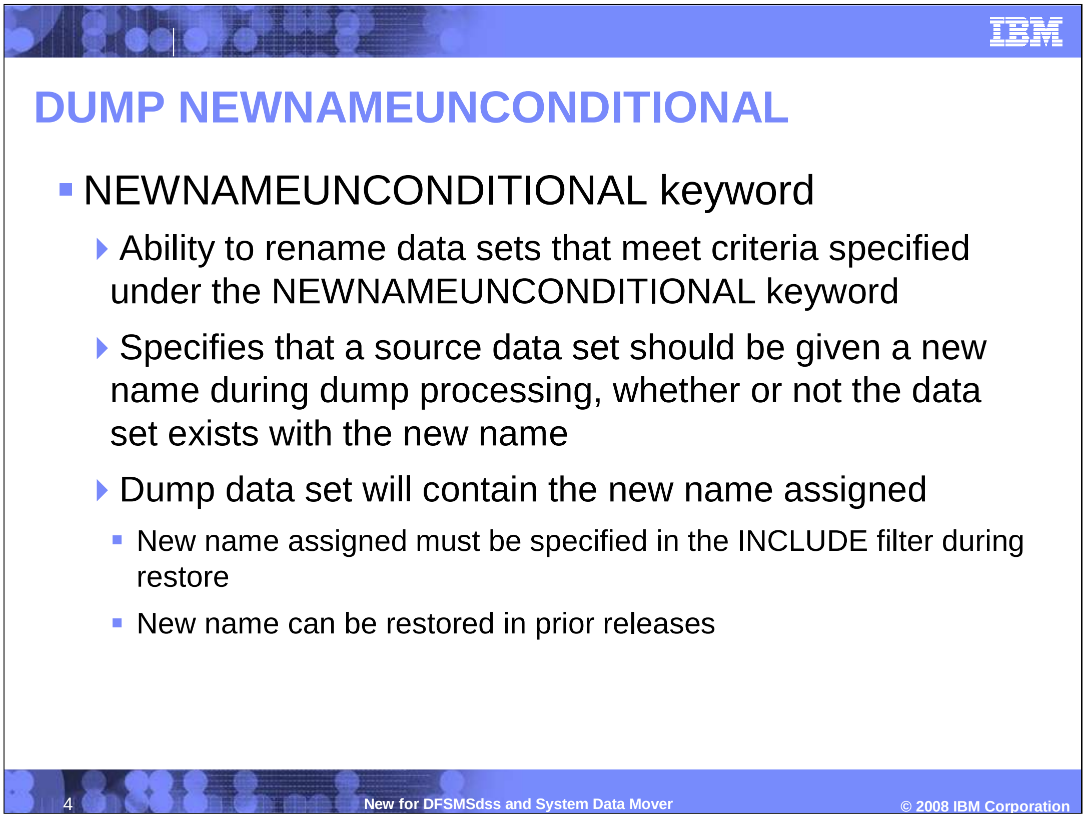
In z/OSV1R10 the items made available were

DUMP using NEWNAMEUNCONDITIONAL Support

DEFRAG Performance Enhancement

CONSOLIDATION Command

Virtual Concurrent Copy using FlashCopy



DUMP NEWNAMEUNCONDITIONAL

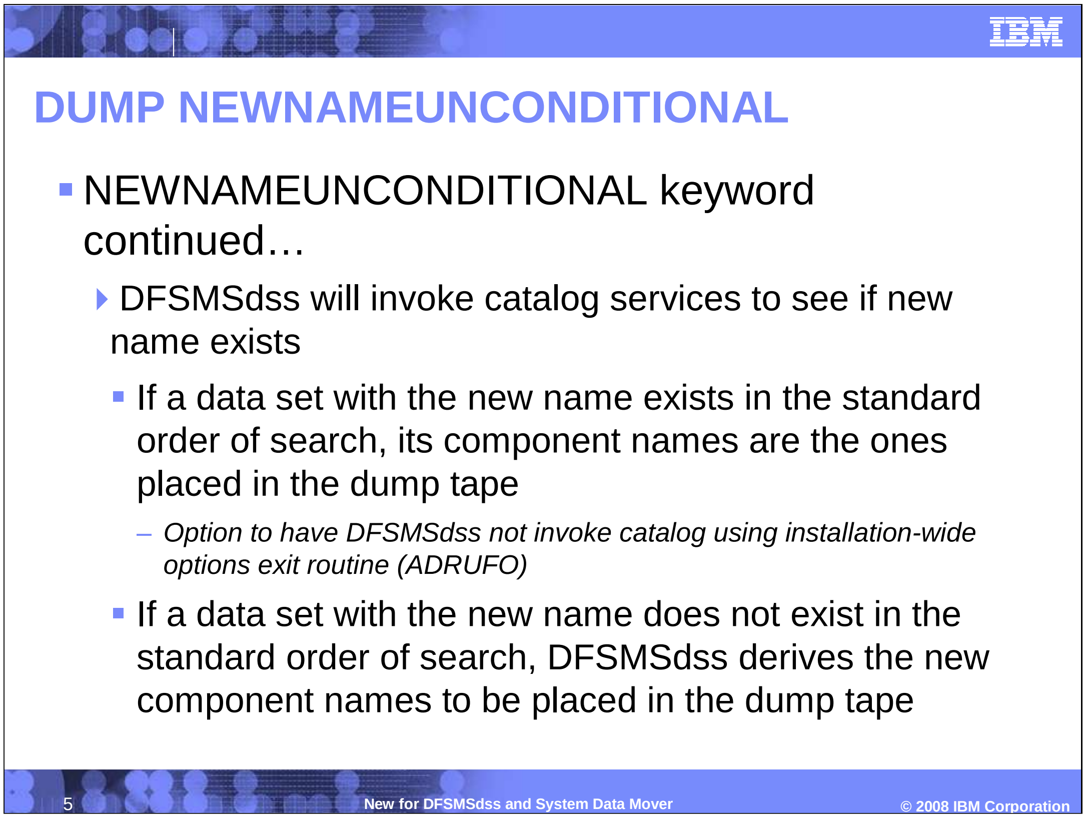
- NEWNAMEUNCONDITIONAL keyword
  - Ability to rename data sets that meet criteria specified under the NEWNAMEUNCONDITIONAL keyword
  - Specifies that a source data set should be given a new name during dump processing, whether or not the data set exists with the new name
  - Dump data set will contain the new name assigned
    - New name assigned must be specified in the INCLUDE filter during restore
    - New name can be restored in prior releases

4 New for DFSMSdss and System Data Mover

For the DUMP NEWNAMEUNCONDITIONAL item a new keyword is introduced NEWNAMEUNCONDITIONAL keyword.

It is very similar in behavior as with the existing RENAMEUNCONDITIONAL keyword.

- •Ability to rename data sets that meet criteria specified under the NEWNAMEUNCONDITIONAL keyword
- •Specifies that a source data set should be given a new name during dump processing, whether or not the data set exists with the new name
- •Dump data set will contain the new name assigned
  - •New name assigned must be specified in the INCLUDE filter during restore
  - •New name can be restored in prior releases



IBM

DUMP NEWNAMEUNCONDITIONAL

- NEWNAMEUNCONDITIONAL keyword continued…
  - DFSMSdss will invoke catalog services to see if new name exists
    - If a data set with the new name exists in the standard order of search, its component names are the ones placed in the dump tape
      - *Option to have DFSMSdss not invoke catalog using installation-wide options exit routine (ADRUFO)*
    - If a data set with the new name does not exist in the standard order of search, DFSMSdss derives the new component names to be placed in the dump tape

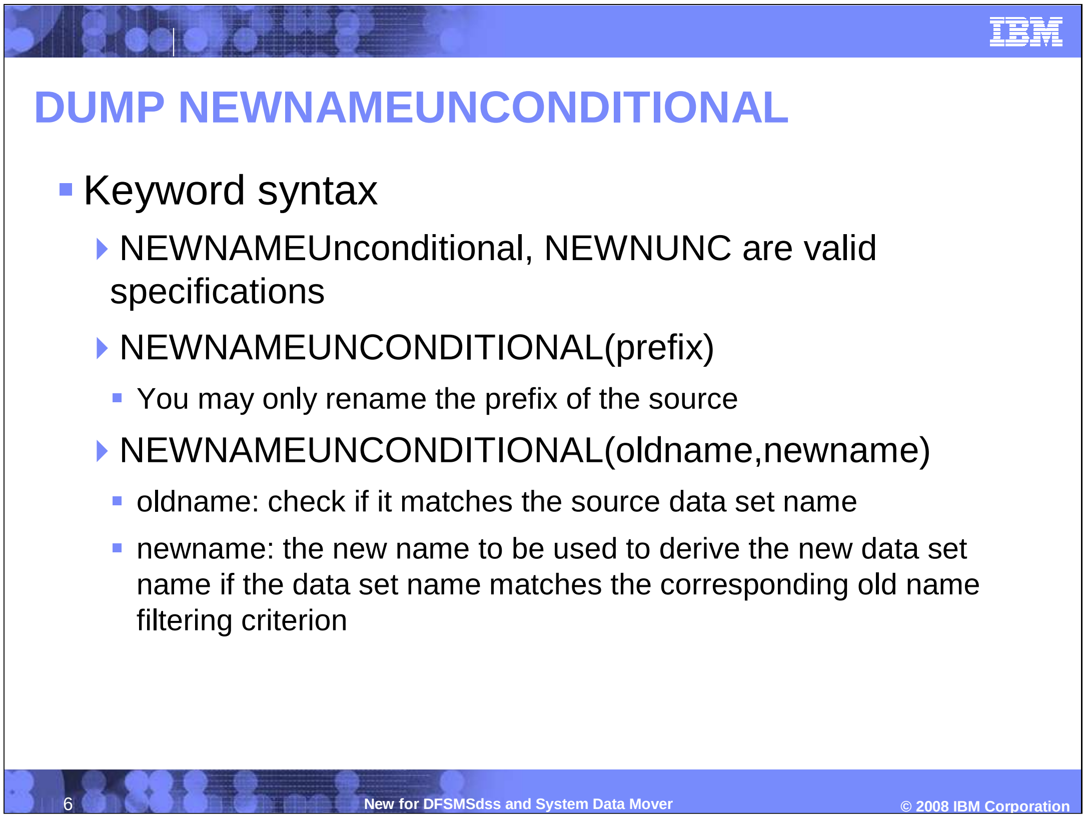
5 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

With the NEWNAMEUCONDITIONAL keyword DFSMSdss will invoke catalog services to see if new name exists

If a data set with the new name exists in the standard order of search, its component names are the ones placed in the dump tape

There is an option to have DFSMSdss not invoke Catalog using installation-wide options exit routine provided in the ADRUFO macro

If a data set with the new name does not exist in the standard order of search, DFSMSdss will derive the new component names to be placed in the dump tape



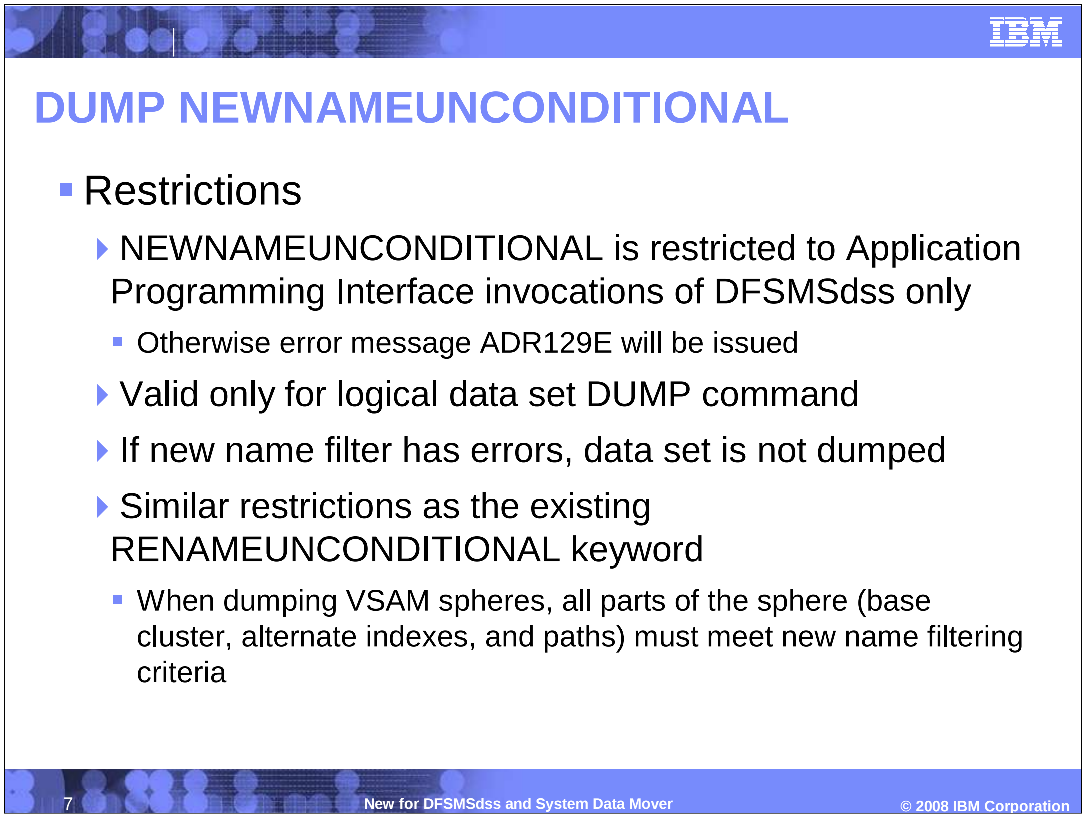
DUMP NEWNAMEUNCONDITIONAL

- Keyword syntax
  - NEWNAMEUnconditional, NEWNUNC are valid specifications
  - NEWNAMEUNCONDITIONAL(prefix)
    - You may only rename the prefix of the source
  - NEWNAMEUNCONDITIONAL(oldname,newname)
    - oldname: check if it matches the source data set name
    - newname: the new name to be used to derive the new data set name if the data set name matches the corresponding old name filtering criterion

6 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

The keyword syntax rules for NEWNAMEUNCONDITIONAL are

- •NEWNAMEUnconditional, NEWNUNC are valid specifications
- •NEWNAMEUNCONDITIONAL(prefix)
  - •You may only rename the prefix of the source
- •NEWNAMEUNCONDITIONAL(oldname,newname)
  - •oldname: check if it matches the source data set name
  - •newname: the new name to be used to derive the new data set name if the data set name matches the corresponding old name filtering criterion



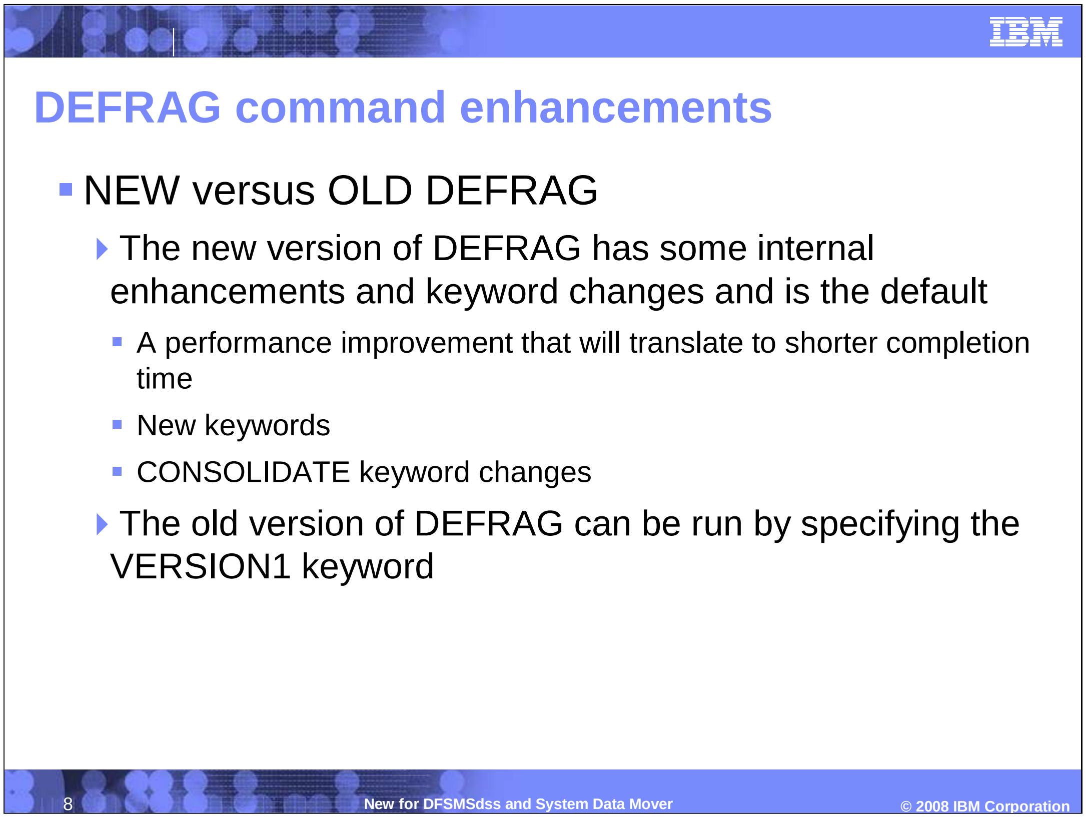
DUMP NEWNAMEUNCONDITIONAL

- Restrictions
  - NEWNAMEUNCONDITIONAL is restricted to Application Programming Interface invocations of DFSMSdss only
    - Otherwise error message ADR129E will be issued
  - Valid only for logical data set DUMP command
  - If new name filter has errors, data set is not dumped
  - Similar restrictions as the existing RENAMEUNCONDITIONAL keyword
    - When dumping VSAM spheres, all parts of the sphere (base cluster, alternate indexes, and paths) must meet new name filtering criteria

7 New for DFSMSdss and System Data Mover

The Restrictions for this items is that

- •NEWNAMEUNCONDITIONAL is restricted to Application Programming Interface invocations of DFSMSdss only
  - •Otherwise error message ADR129E will be issued
- •Valid only for logical data set DUMP command
- •If new name filter has errors, data set is not dumped
- •Similar restrictions as the existing RENAMEUNCONDITIONAL keyword
  - •When dumping VSAM spheres, all parts of the sphere (base cluster, alternate indexes, and paths) must meet new name filtering criteria



IBM

DEFRAG command enhancements

- NEW versus OLD DEFRAG
  - The new version of DEFRAG has some internal enhancements and keyword changes and is the default
    - A performance improvement that will translate to shorter completion time
    - New keywords
    - CONSOLIDATE keyword changes
  - The old version of DEFRAG can be run by specifying the VERSION1 keyword

8 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Another item that is part of z/OSV1R10 for DFSMSdss is the DEFRAG Command Enhancements item.

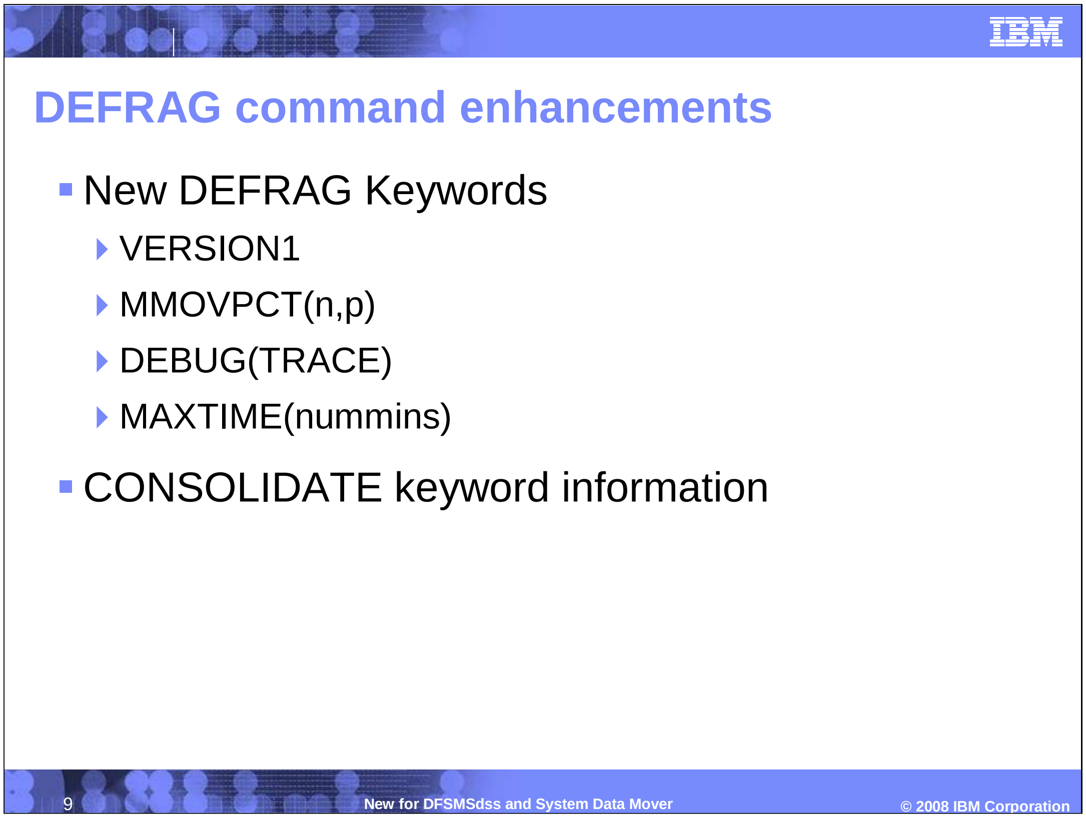
•The new version of DEFRAG has some internal enhancements and keyword changes and is the default

•A Performance improvement that will translate to shorter completion time

•New keywords

•CONSOLIDATE keyword changes

•The old version of DEFRAG can be run by specifying the VERSION1 keyword



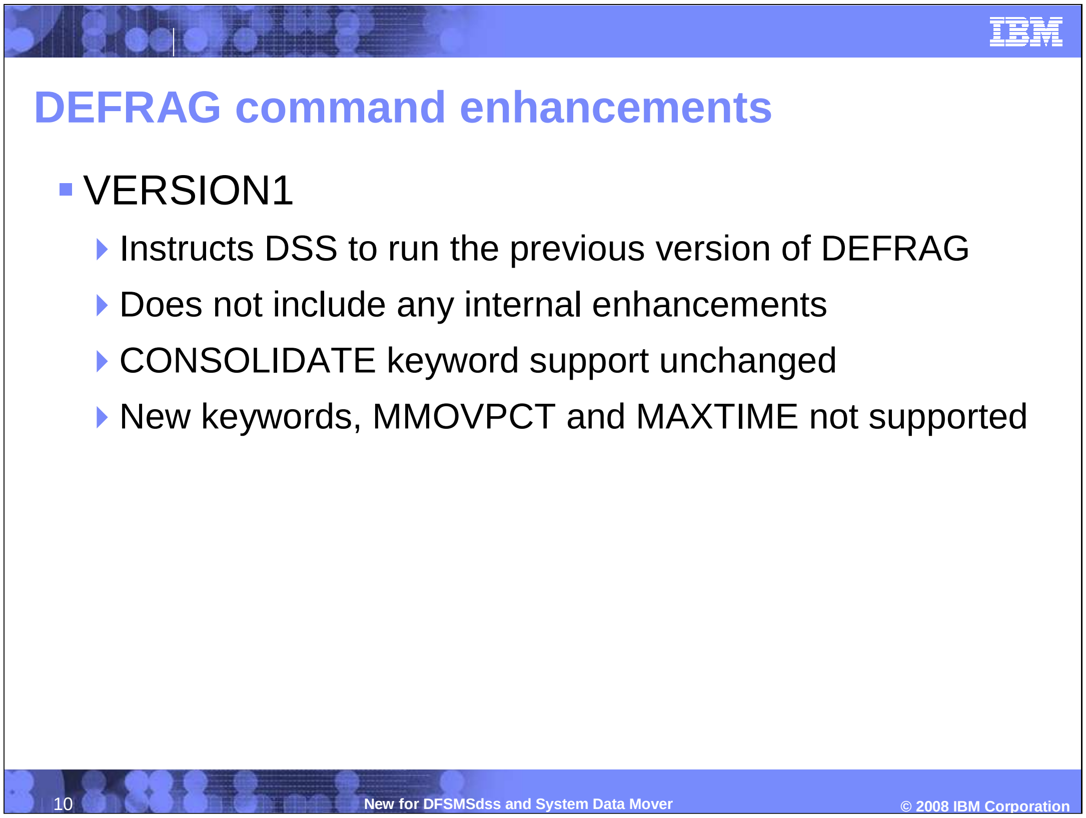
DEFRAG command enhancements

- New DEFRAG Keywords
  - VERSION1
  - MMOVPCT(n,p)
  - DEBUG(TRACE)
  - MAXTIME(nummins)
- CONSOLIDATE keyword information

New DEFRAG Keywords that are being introduced are

- VERSION1
- MMOVPCT(n,p)
- DEBUG(TRACE)
- MAXTIME(nummins)

The some information is covered regarding the CONSOLIDATE keyword.



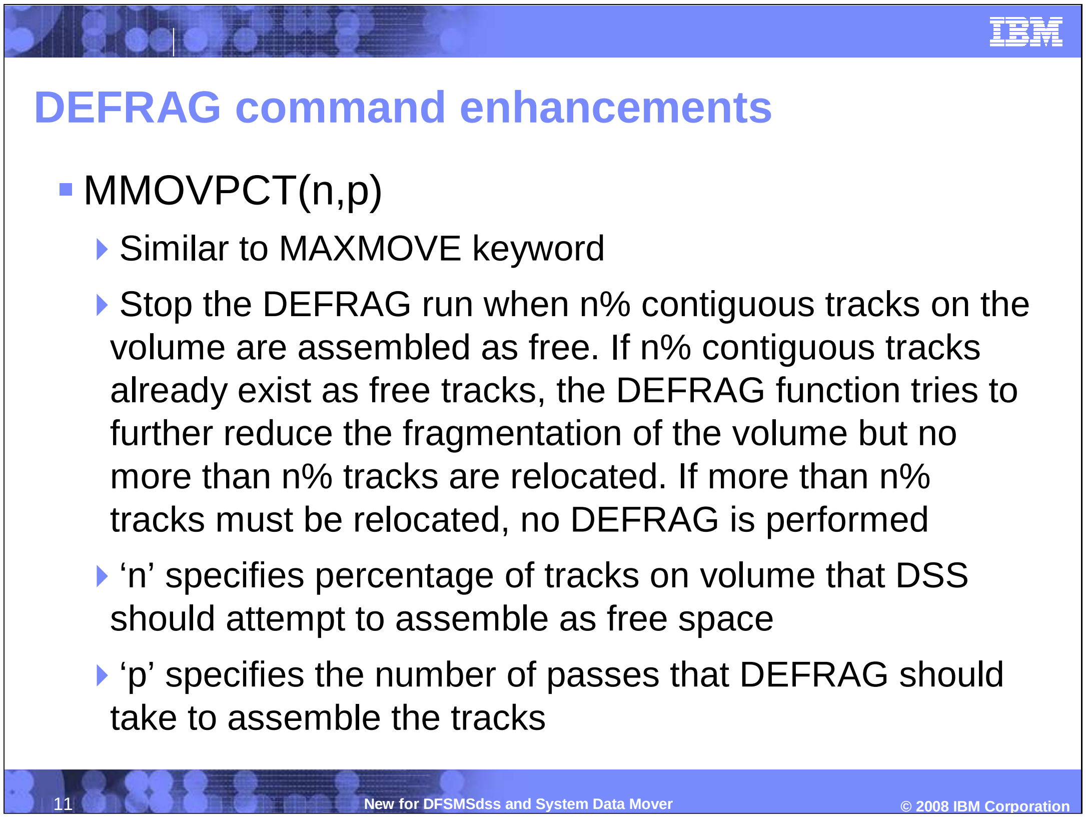
DEFRAG command enhancements

- VERSION1
  - Instructs DSS to run the previous version of DEFRAG
  - Does not include any internal enhancements
  - CONSOLIDATE keyword support unchanged
  - New keywords, MMOVPCT and MAXTIME not supported

10 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

VERSION1

•Instructs DSS to run the previous version of DEFRAG.
•Does not include any internal enhancements.
•CONSOLIDATE keyword support unchanged.
•New keywords, MMOVPCT and MAXTIME not supported.



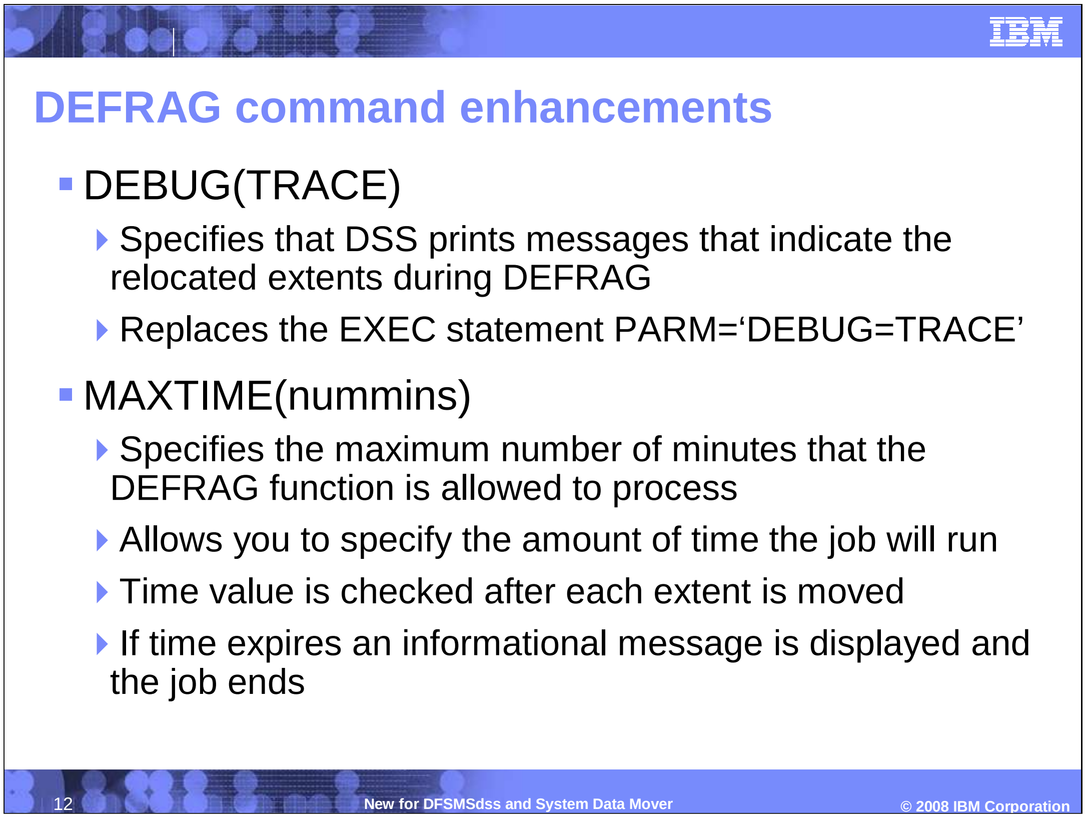
DEFRAG command enhancements

- MMOVPCT(n,p)
  - Similar to MAXMOVE keyword
  - Stop the DEFRAG run when n% contiguous tracks on the volume are assembled as free. If n% contiguous tracks already exist as free tracks, the DEFRAG function tries to further reduce the fragmentation of the volume but no more than n% tracks are relocated. If more than n% tracks must be relocated, no DEFRAG is performed
  - ‘n’ specifies percentage of tracks on volume that DSS should attempt to assemble as free space
  - ‘p’ specifies the number of passes that DEFRAG should take to assemble the tracks

11 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

MMOVPCT(n,p)

•Similar to MAXMOVE keyword.
•Stop the DEFRAG run when n% contiguous tracks on the volume are assembled as free. If n% contiguous tracks already exist as free tracks, the DEFRAG function tries to further reduce the fragmentation of the volume but no more than n% tracks are relocated. If more than n% tracks must be relocated, no DEFRAG is performed.
•‘n’ specifies percentage of tracks on volume that DSS should attempt to assemble as free space.
•‘p’ specifies the number of passes that DEFRAG should take to assemble the tracks.



DEFRAG command enhancements

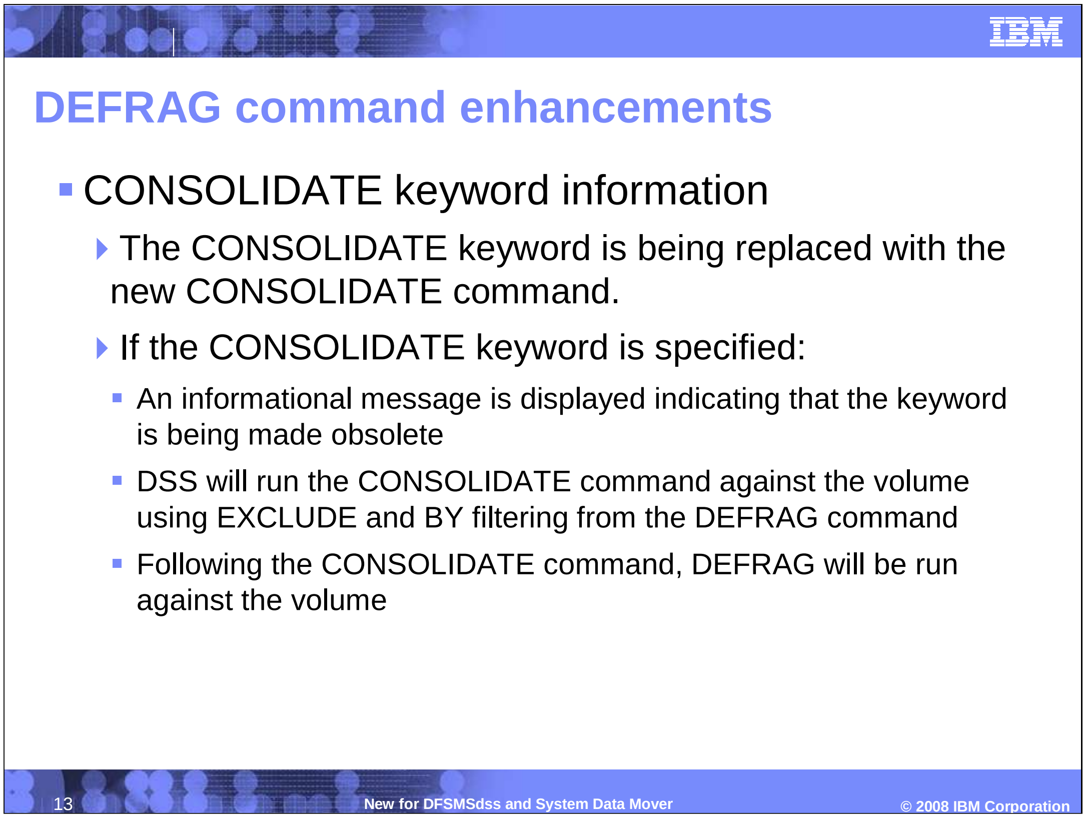
- DEBUG(TRACE)
  - Specifies that DSS prints messages that indicate the relocated extents during DEFRAG
  - Replaces the EXEC statement PARM=‘DEBUG=TRACE’
- MAXTIME(nummins)
  - Specifies the maximum number of minutes that the DEFRAG function is allowed to process
  - Allows you to specify the amount of time the job will run
  - Time value is checked after each extent is moved
  - If time expires an informational message is displayed and the job ends

DEBUG(TRACE)

•Specifies that DSS prints messages that indicate the relocated extents during DEFRAG.
•Replaces the EXEC statement PARM=‘DEBUG=TRACE’.

MAXTIME(nummins)

•Specifies the maximum number of minutes that the DEFRAG function is allowed to process.
•Allows you to specify the amount of time the job will run.
•Time value is checked after each extent is moved.
•If time expires an informational message is displayed and the job ends



IBM

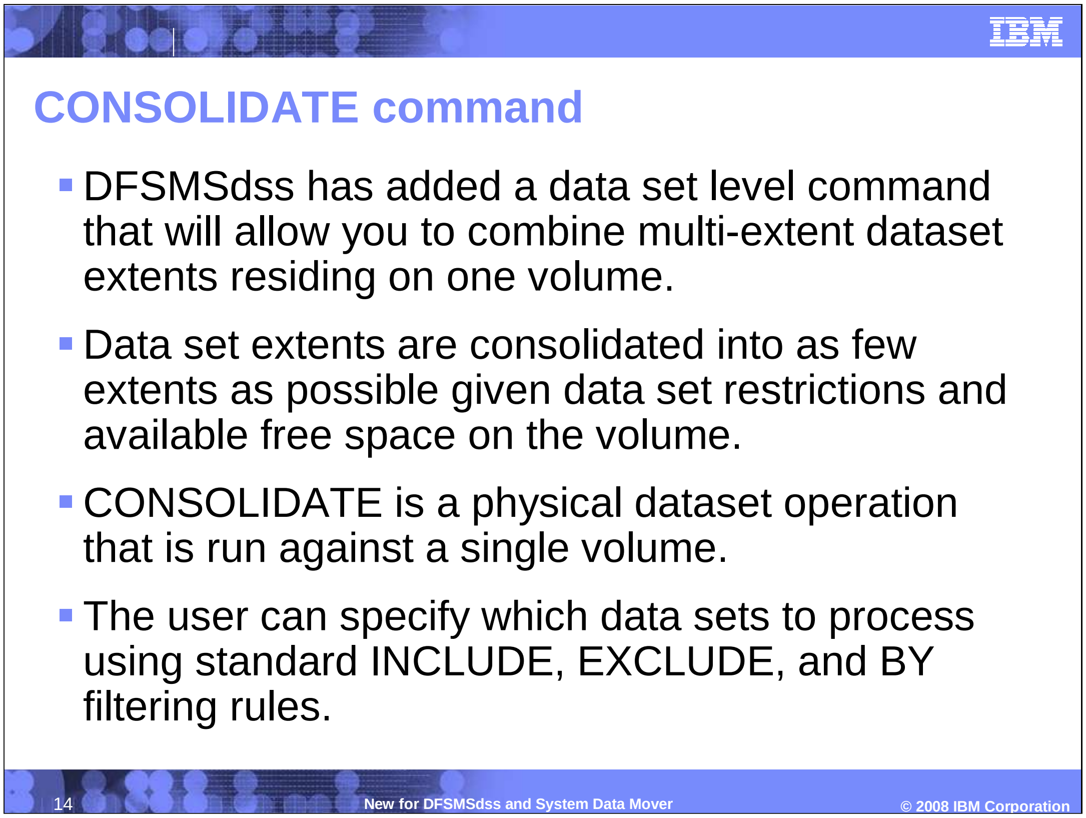
DEFRAG command enhancements

- CONSOLIDATE keyword information
  - The CONSOLIDATE keyword is being replaced with the new CONSOLIDATE command.
  - If the CONSOLIDATE keyword is specified:
    - An informational message is displayed indicating that the keyword is being made obsolete
    - DSS will run the CONSOLIDATE command against the volume using EXCLUDE and BY filtering from the DEFRAG command
    - Following the CONSOLIDATE command, DEFRAG will be run against the volume

13 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Regarding the CONSOLIDATE Keyword.

- •The CONSOLIDATE keyword is being replaced with the new CONSOLIDATE command.
- •If the CONSOLIDATE keyword is specified:
  - •An informational message is displayed indicating that the keyword is being made obsolete.
  - •DSS will run the CONSOLIDATE command against the volume using EXCLUDE and BY filtering from the DEFRAG command.
  - •Following the CONSOLIDATE command, DEFRAG will be run against the volume.



CONSOLIDATE command

- DFSMSdss has added a data set level command that will allow you to combine multi-extent dataset extents residing on one volume.
- Data set extents are consolidated into as few extents as possible given data set restrictions and available free space on the volume.
- CONSOLIDATE is a physical dataset operation that is run against a single volume.
- The user can specify which data sets to process using standard INCLUDE, EXCLUDE, and BY filtering rules.

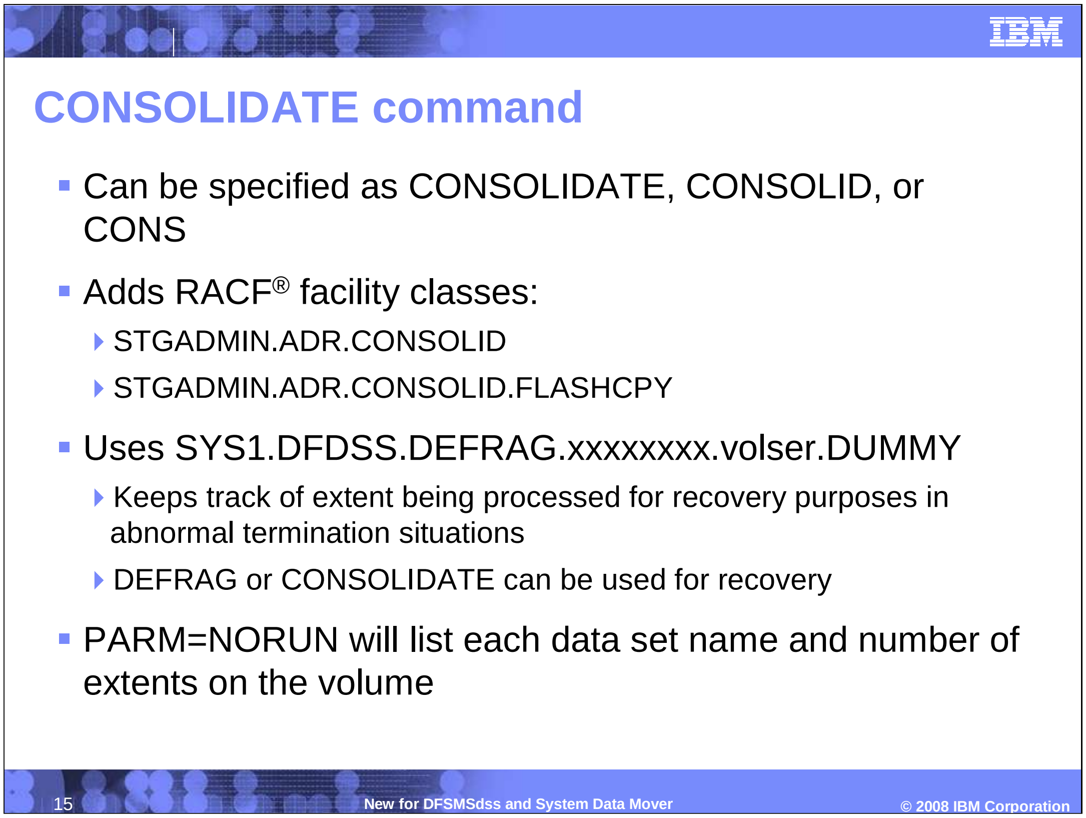
This leads into the CONSOLIDATE Command.

DFSMSdss has added a data set level command that will allow you to combine multi-extent dataset extents residing on one volume.

Data set extents are consolidated into as few extents as possible given data set restrictions and available free space on the volume.

CONSOLIDATE is a physical dataset operation that is run against a single volume.

The user can specify which data sets to process using standard INCLUDE, EXCLUDE, and BY filtering rules.



CONSOLIDATE command

- Can be specified as CONSOLIDATE, CONSOLID, or CONS
- Adds RACF® facility classes:
  - STGADMIN.ADR.CONSOLID
  - STGADMIN.ADR.CONSOLID.FLASHCPY
- Uses SYS1.DFDSS.DEFRAG.xxxxxxxx.volser.DUMMY
  - Keeps track of extent being processed for recovery purposes in abnormal termination situations
  - DEFRAG or CONSOLIDATE can be used for recovery
- PARM=NORUN will list each data set name and number of extents on the volume

15 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Can be specified as CONSOLIDATE, CONSOLID, or CONS

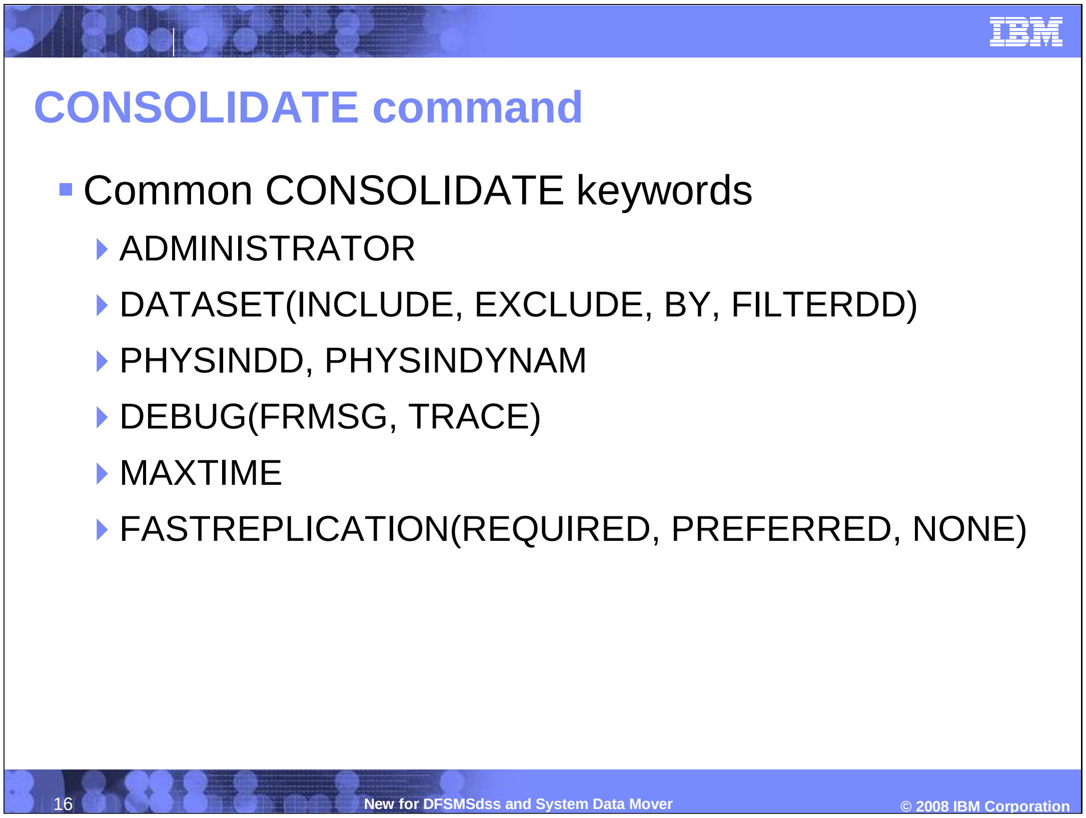
Adds RACF Facility classes:

- •STGADMIN.ADR.CONSOLID
- •STGADMIN.ADR.CONSOLID.FLASHCPY

Uses SYS1.DFDSS.DEFRAG.xxxxxxxx.volser.DUMMY

- •Keeps track of extent being processed for recovery purposes in abnormal termination situations
- •DEFRAG or CONSOLIDATE can be used for recovery

PARM=NORUN will list each data set name and number of extents on the volume



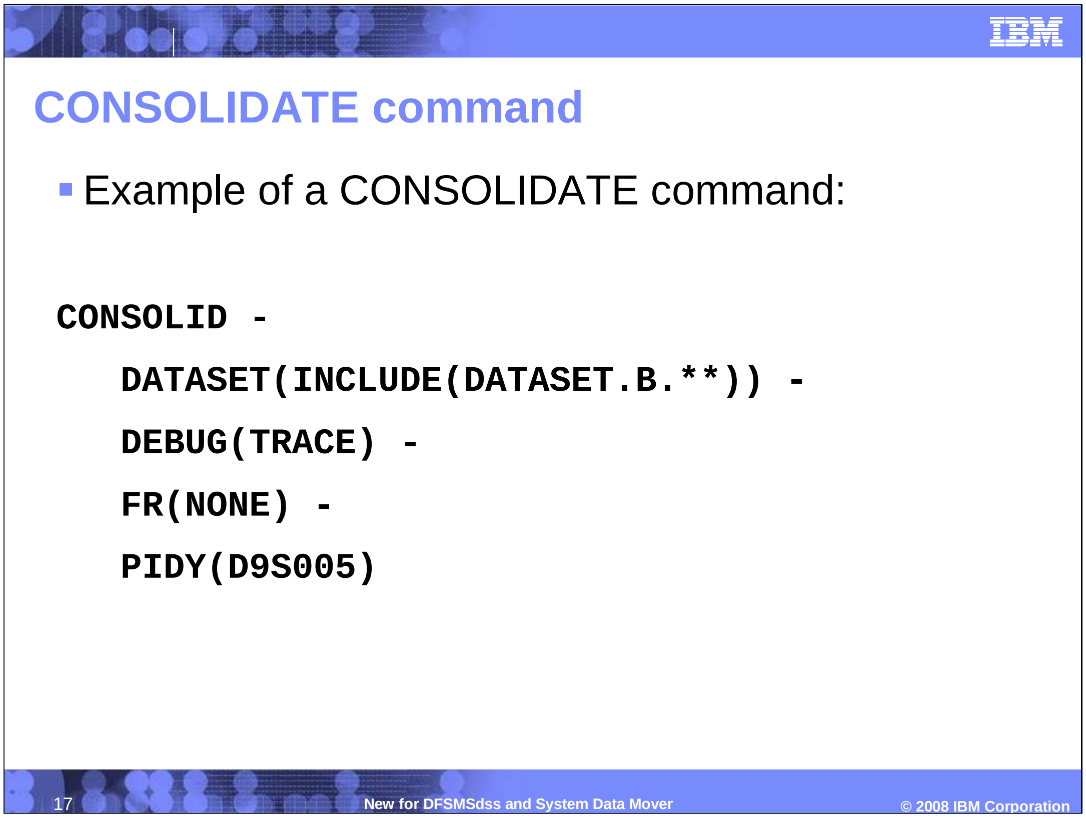
CONSOLIDATE command

- Common CONSOLIDATE keywords
  - ADMINISTRATOR
  - DATASET(INCLUDE, EXCLUDE, BY, FILTERDD)
  - PHYSINDD, PHYSINDYNAM
  - DEBUG(FRMSG, TRACE)
  - MAXTIME
  - FASTREPLICATION(REQUIRED, PREFERRED, NONE)

16 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Some Common CONSOLIDATE keywords are

- •ADMINISTRATOR
- •DATASET(INCLUDE, EXCLUDE, BY, FILTERDD)
- •PHYSINDD, PHYSINDYNAM
- •DEBUG(FRMSG, TRACE)
- •MAXTIME
- •FASTREPLICATION(REQUIRED, PREFERRED, NONE)



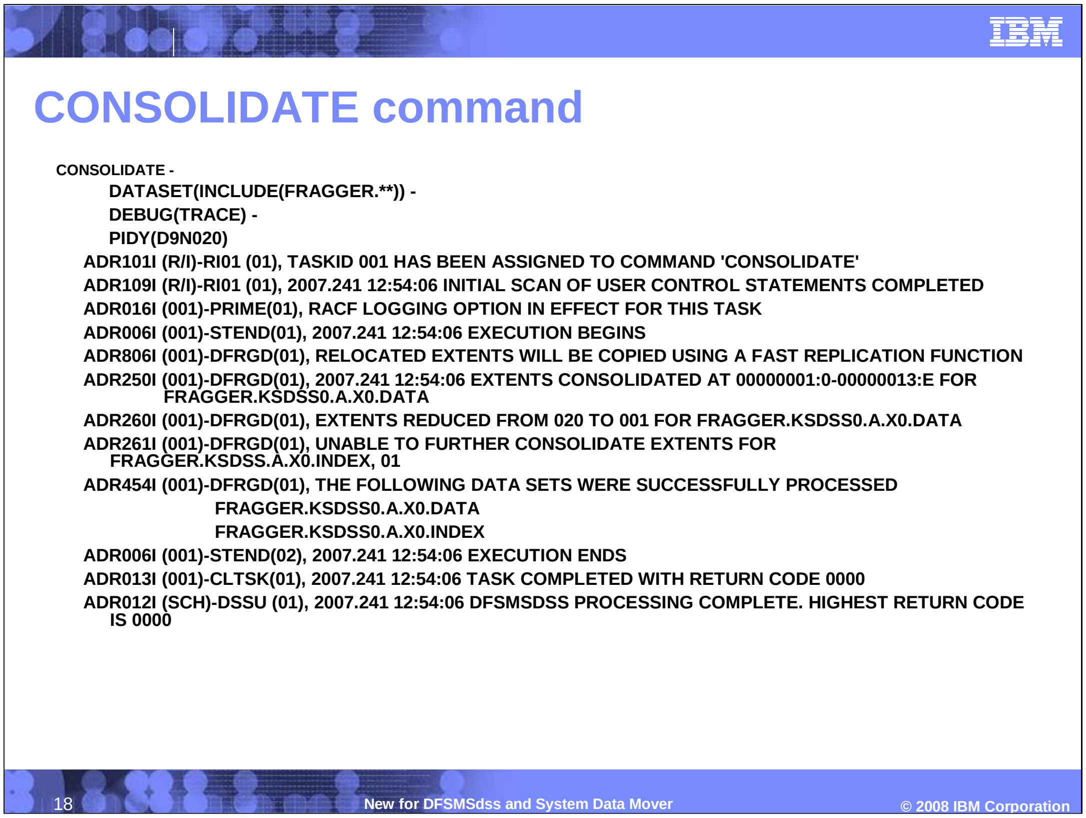
CONSOLIDATE command

- Example of a CONSOLIDATE command:

```
CONSOLID -
   DATASET(INCLUDE(DATASET.B.**)) -
   DEBUG(TRACE) -
   FR(NONE) -
   PIDY(D9S005)
```

17 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

This is an example of the JCL on how to invoke the CONSOLIDATE command.

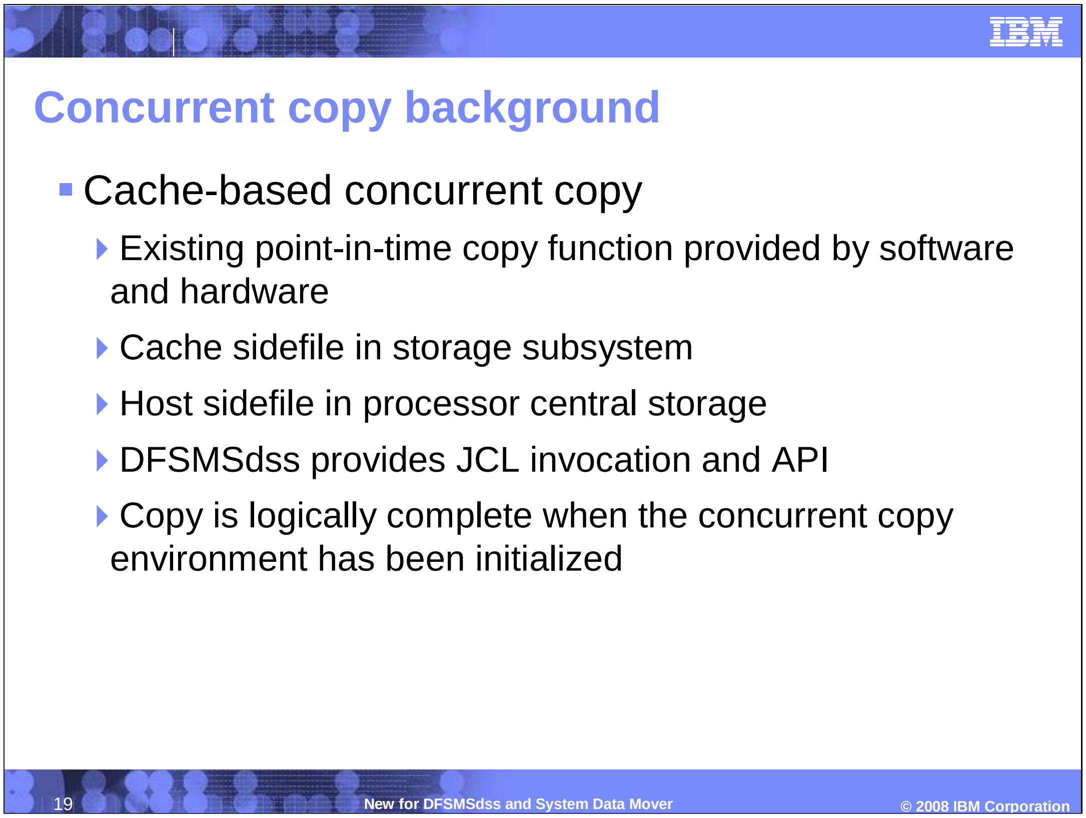


CONSOLIDATE command

```
CONSOLIDATE -
        DATASET(INCLUDE(FRAGGER.**)) -
        DEBUG(TRACE) -
        PIDY(D9N020)
    ADR101I (R/I)-RI01 (01), TASKID 001 HAS BEEN ASSIGNED TO COMMAND 'CONSOLIDATE'
    ADR109I (R/I)-RI01 (01), 2007.241 12:54:06 INITIAL SCAN OF USER CONTROL STATEMENTS COMPLETED
    ADR016I (001)-PRIME(01), RACF LOGGING OPTION IN EFFECT FOR THIS TASK
    ADR006I (001)-STEND(01), 2007.241 12:54:06 EXECUTION BEGINS
    ADR806I (001)-DFRGD(01), RELOCATED EXTENTS WILL BE COPIED USING A FAST REPLICATION FUNCTION
    ADR250I (001)-DFRGD(01), 2007.241 12:54:06 EXTENTS CONSOLIDATED AT 00000001:0-00000013:E FOR
               FRAGGER.KSDSS0.A.X0.DATA
    ADR260I (001)-DFRGD(01), EXTENTS REDUCED FROM 020 TO 001 FOR FRAGGER.KSDSS0.A.X0.DATA
    ADR261I (001)-DFRGD(01), UNABLE TO FURTHER CONSOLIDATE EXTENTS FOR
        FRAGGER.KSDSS.A.X0.INDEX, 01
    ADR454I (001)-DFRGD(01), THE FOLLOWING DATA SETS WERE SUCCESSFULLY PROCESSED
                        FRAGGER.KSDSS0.A.X0.DATA
                        FRAGGER.KSDSS0.A.X0.INDEX
    ADR006I (001)-STEND(02), 2007.241 12:54:06 EXECUTION ENDS
    ADR013I (001)-CLTSK(01), 2007.241 12:54:06 TASK COMPLETED WITH RETURN CODE 0000
    ADR012I (SCH)-DSSU (01), 2007.241 12:54:06 DFSMSDSS PROCESSING COMPLETE. HIGHEST RETURN CODE
        IS 0000
```

18 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Here is an example of the joblog when a Consolidate command is run.



Concurrent copy background

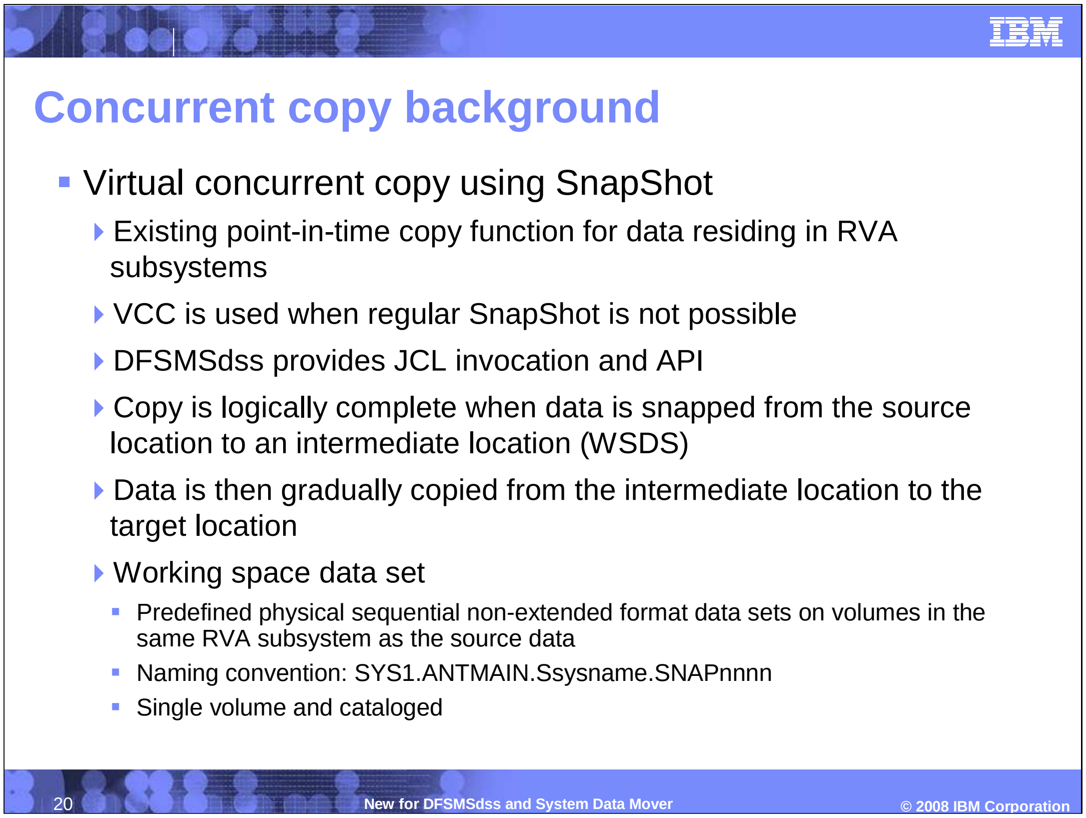
- Cache-based concurrent copy
  - Existing point-in-time copy function provided by software and hardware
  - Cache sidefile in storage subsystem
  - Host sidefile in processor central storage
  - DFSMSdss provides JCL invocation and API
  - Copy is logically complete when the concurrent copy environment has been initialized

19 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Now for the Virtual Concurrent Copy item. First, some brief background information on Concurrent Copy.

Cache-based Concurrent Copy

- •Existing point-in-time copy function provided by software and hardware.
- •Cache sidefile in storage subsystem.
- •Host sidefile in processor central storage.
- •DFSMSdss provides JCL invocation and API.
- •Copy is logically complete when the concurrent copy environment has been initialized.



Concurrent copy background

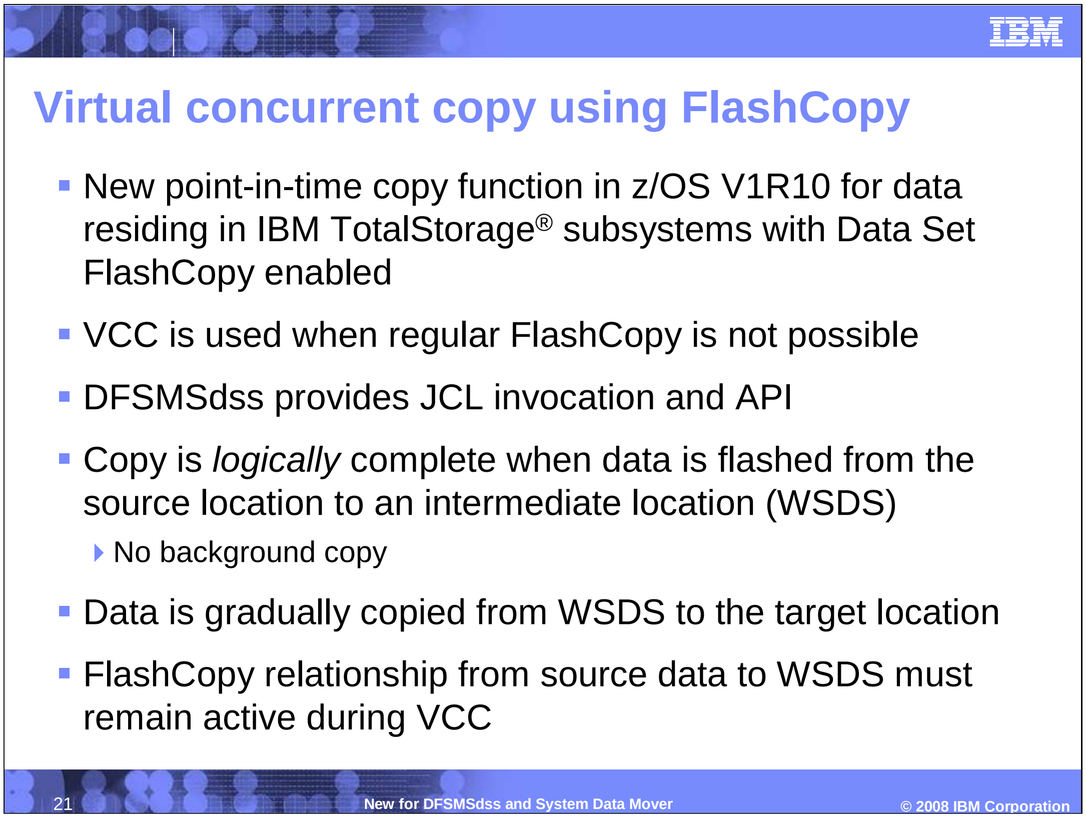
- Virtual concurrent copy using SnapShot
  - Existing point-in-time copy function for data residing in RVA subsystems
  - VCC is used when regular SnapShot is not possible
  - DFSMSdss provides JCL invocation and API
  - Copy is logically complete when data is snapped from the source location to an intermediate location (WSDS)
  - Data is then gradually copied from the intermediate location to the target location
  - Working space data set
    - Predefined physical sequential non-extended format data sets on volumes in the same RVA subsystem as the source data
    - Naming convention: SYS1.ANTMAIN.Ssysname.SNAPnnnn
    - Single volume and cataloged

20 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

Now with Virtual Concurrent Copy using Snapshot.

•Existing point-in-time copy function for data residing in RVA subsystems.
•VCC is used when regular SnapShot is not possible.
•DFSMSdss provides JCL invocation and API.
•Copy is logically complete when data is snapped from the source location to an intermediate location (WSDS).
•Data is then gradually copied from the intermediate location to the target location.
•Working Space Data Set characteristics are the following:

•They are predefined physical sequential non-extended format data sets on volumes in the same RVA subsystem as the source data.
•Naming convention: SYS1.ANTMAIN.Ssysname.SNAPnnnn
•Must be single volume and cataloged.



IBM

Virtual concurrent copy using FlashCopy

- New point-in-time copy function in z/OS V1R10 for data residing in IBM TotalStorage® subsystems with Data Set FlashCopy enabled
- VCC is used when regular FlashCopy is not possible
- DFSMSdss provides JCL invocation and API
- Copy is *logically* complete when data is flashed from the source location to an intermediate location (WSDS)
  - No background copy
- Data is gradually copied from WSDS to the target location
- FlashCopy relationship from source data to WSDS must remain active during VCC

21 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

In z/OSV1R10 DFSMSdss is introducing Virtual Concurrent Copy Using FlashCopy.

New point-in-time copy function in z/OS V1R10 for data residing in IBM TotalStorage subsystems with Data Set FlashCopy enabled

VCC is used when regular FlashCopy is not possible.

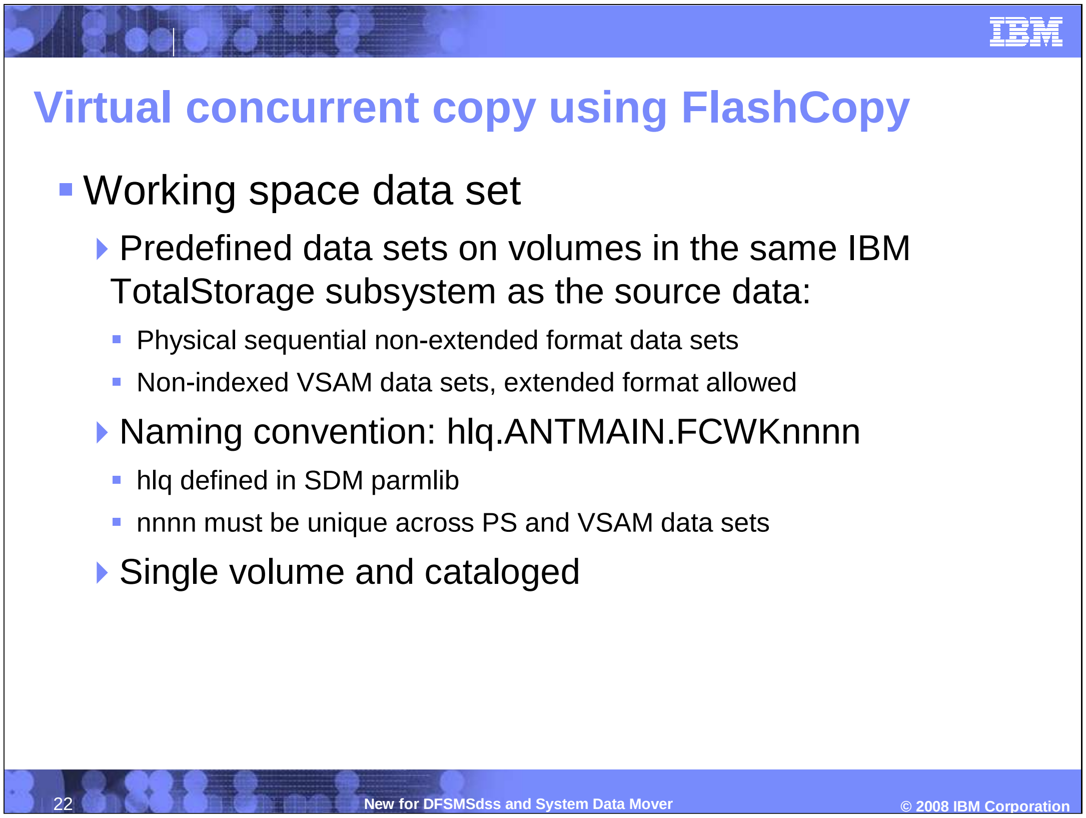
DFSMSdss provides JCL invocation and API.

Copy is *logically* complete when data is flashed from the source location to an intermediate location (WSDS)

•There is no background copy.

Data is gradually copied from WSDS to the target location.

FlashCopy relationship from source data to WSDS must remain active during VCC.



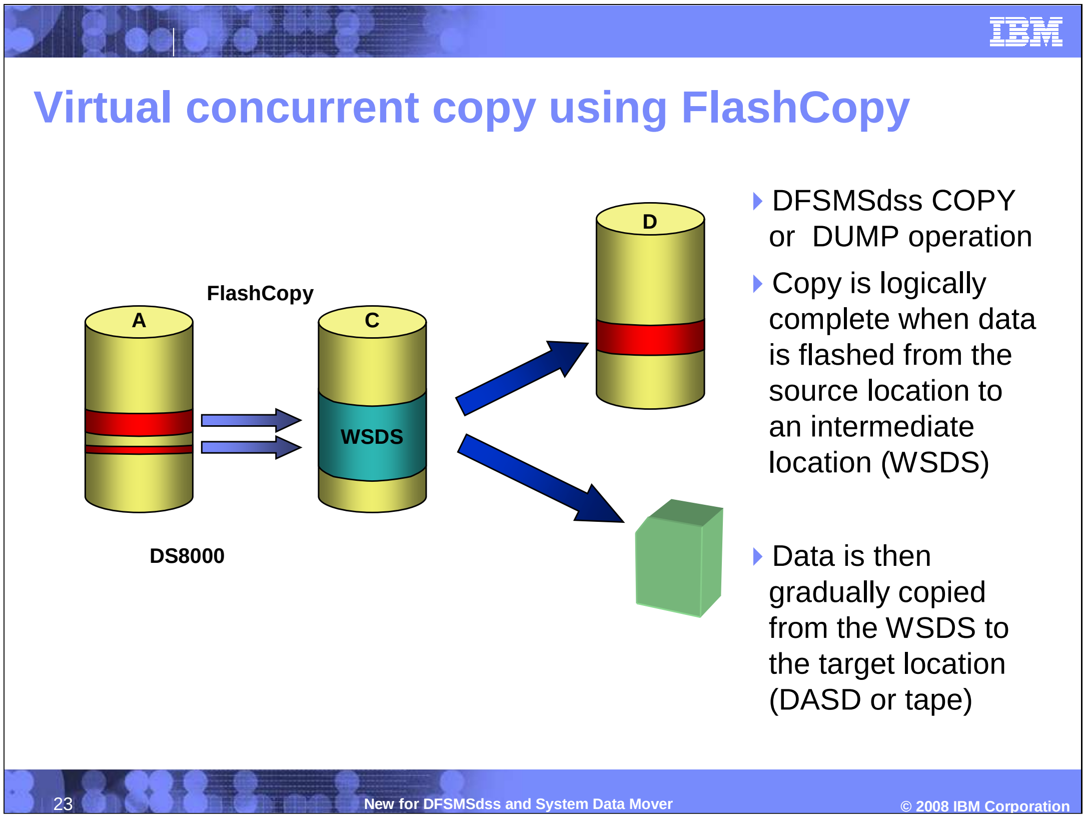
Virtual concurrent copy using FlashCopy

- Working space data set
  - Predefined data sets on volumes in the same IBM TotalStorage subsystem as the source data:
    - Physical sequential non-extended format data sets
    - Non-indexed VSAM data sets, extended format allowed
  - Naming convention: hlq.ANTMAIN.FCWKnnnn
    - hlq defined in SDM parmlib
    - nnnn must be unique across PS and VSAM data sets
  - Single volume and cataloged

Some characteristics of the Working Space Data set that are specific to the VCC function using FlashCopy.

The Working Space Data Set must be:

- •Predefined data sets on volumes in the same IBM TotalStorage subsystem as the source data:
  - •Physical sequential non-extended format data sets.
  - •Non-indexed VSAM data sets, extended format allowed.
- •Naming convention: hlq.ANTMAIN.FCWKnnnn.
  - •High Level Qualifier is defined in SDM parmlib.
  - •nnnn must be unique across PS and VSAM data sets.
- •The data set must be Single volume and cataloged.



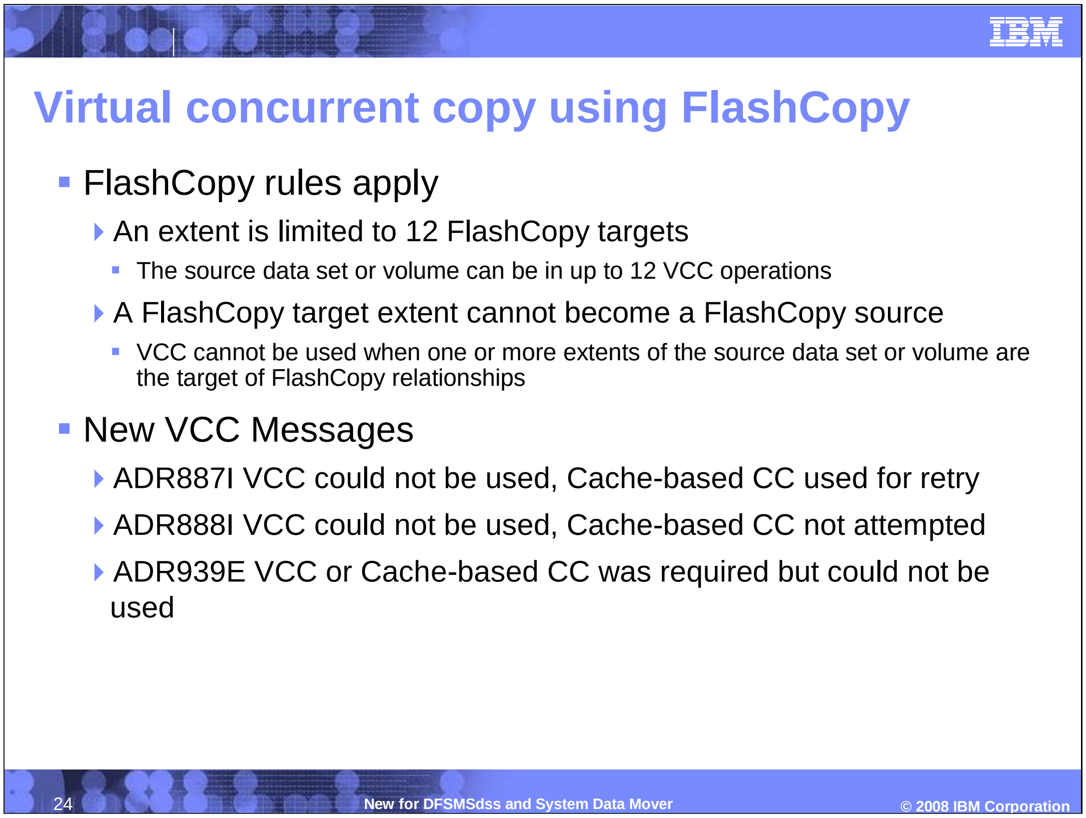
IBM

Virtual concurrent copy using FlashCopy

- DFSMSdss COPY or DUMP operation
- Copy is logically complete when data is flashed from the source location to an intermediate location (WSDS)
- Data is then gradually copied from the WSDS to the target location (DASD or tape)

23 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

This slide provides a visual representation of VCC using FlashCopy



IBM

Virtual concurrent copy using FlashCopy

- FlashCopy rules apply
  - An extent is limited to 12 FlashCopy targets
    - The source data set or volume can be in up to 12 VCC operations
  - A FlashCopy target extent cannot become a FlashCopy source
    - VCC cannot be used when one or more extents of the source data set or volume are the target of FlashCopy relationships
- New VCC Messages
  - ADR887I VCC could not be used, Cache-based CC used for retry
  - ADR888I VCC could not be used, Cache-based CC not attempted
  - ADR939E VCC or Cache-based CC was required but could not be used

24 New for DFSMSdss and System Data Mover

Some additional important point is that since FlashCopy technology is being used FlashCopy rules still apply.

FlashCopy rules apply

•An extent is limited to 12 FlashCopy targets

•The source data set or volume can be in up to 12 VCC operations

•A FlashCopy target extent cannot become a FlashCopy source

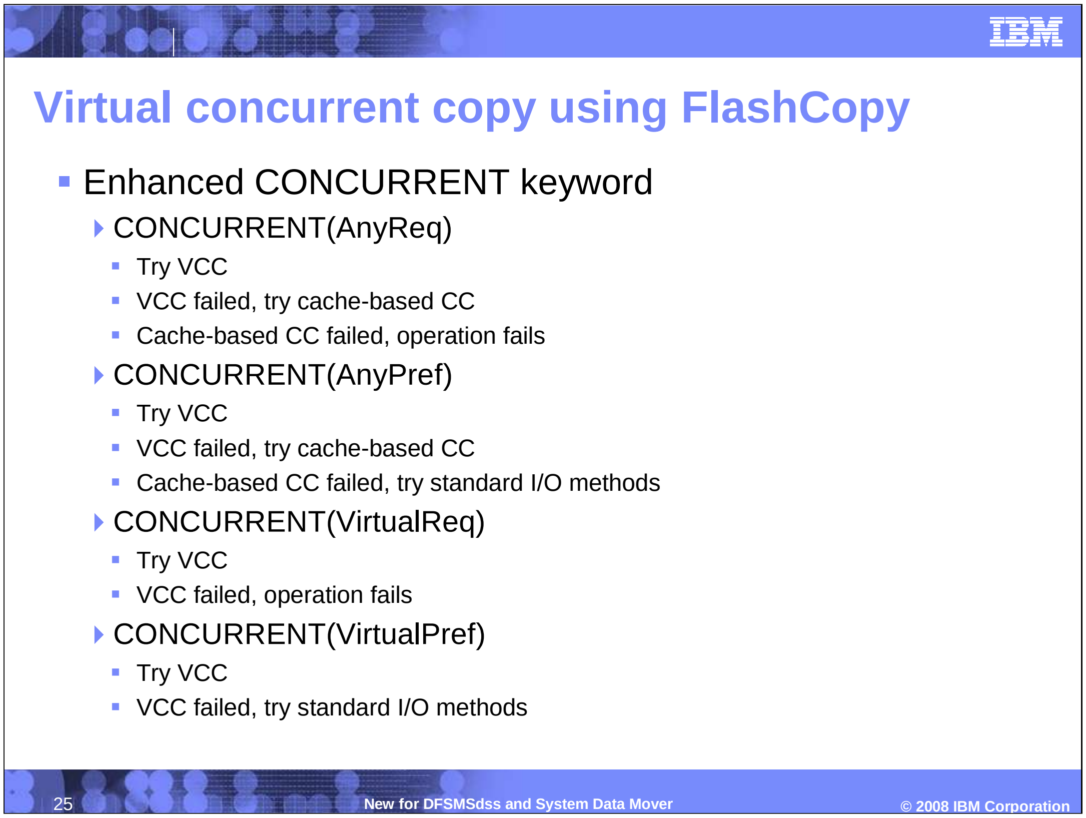
•VCC cannot be used when one or more extents of the source data set or volume are the target of FlashCopy relationships

New VCC Messages introduced are:

•ADR887I states that VCC could not be used, so Cache-based CC will be used for to retry.

•ADR888I states that VCC could not be used, but Cache-based CC will not be attempted.

•ADR939E states that VCC or Cache-based CC was required but could not be used.



IBM

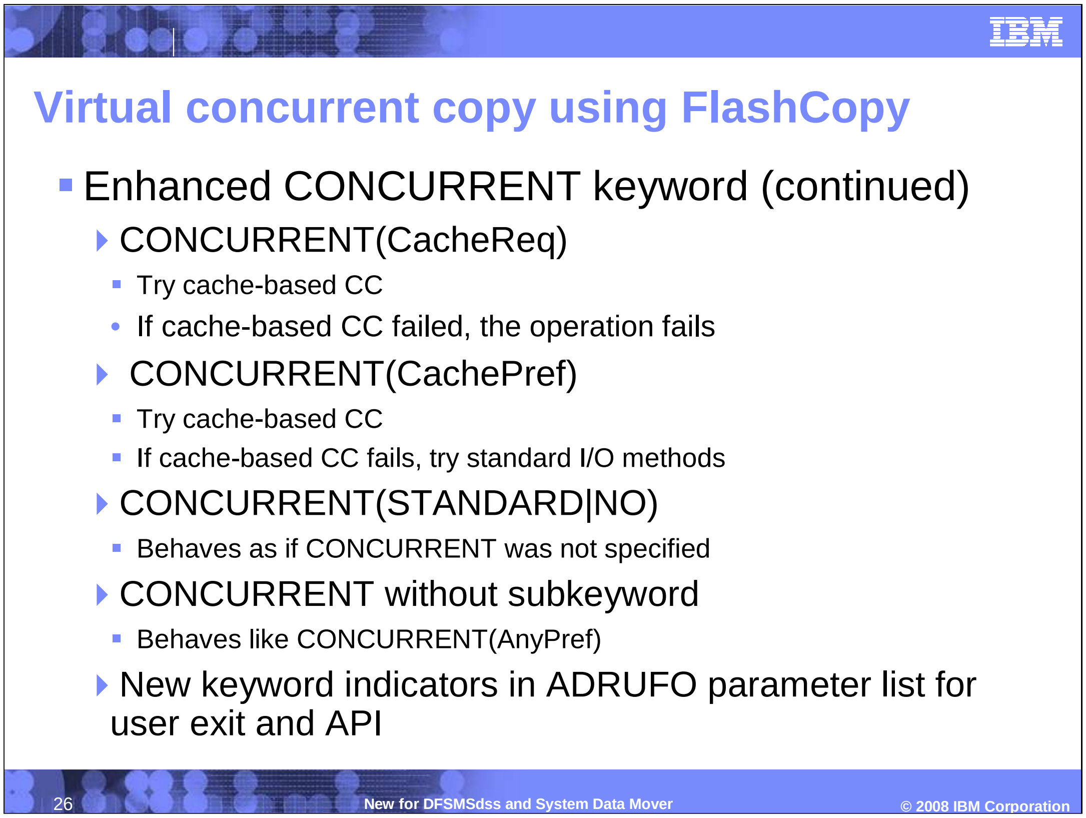
Virtual concurrent copy using FlashCopy

- Enhanced CONCURRENT keyword
  - CONCURRENT(AnyReq)
    - Try VCC
    - VCC failed, try cache-based CC
    - Cache-based CC failed, operation fails
  - CONCURRENT(AnyPref)
    - Try VCC
    - VCC failed, try cache-based CC
    - Cache-based CC failed, try standard I/O methods
  - CONCURRENT(VirtualReq)
    - Try VCC
    - VCC failed, operation fails
  - CONCURRENT(VirtualPref)
    - Try VCC
    - VCC failed, try standard I/O methods

The existing CONCURRENT keyword is enhanced with the following:

- Specifying CONCURRENT(AnyReq) indicates that
  - VCC is first attempted.
  - If VCC fails, cache-based CC is attempted.
  - If Cache-based CC fails, the operation fails.
- Specifying CONCURRENT(AnyPref) indicates that
  - VCC is first attempted.
  - If VCC fails, cache-based CC is attempted.
  - If Cache-based CC fails, standard I/O is attempted.

  - Specifying CONCURRENT(VirtualReq) indicates that
  - VCC is first attempted.
  - If VCC fails the operation fails.
- Specifying CONCURRENT(VirtualPref) indicates that
  - VCC is first attempted.
  - If VCC fails the standard I/O is attempted.



IBM

Virtual concurrent copy using FlashCopy

- Enhanced CONCURRENT keyword (continued)
  - CONCURRENT(CacheReq)
    - Try cache-based CC
    - If cache-based CC failed, the operation fails
  - CONCURRENT(CachePref)
    - Try cache-based CC
    - If cache-based CC fails, try standard I/O methods
  - CONCURRENT(STANDARD|NO)
    - Behaves as if CONCURRENT was not specified
  - CONCURRENT without subkeyword
    - Behaves like CONCURRENT(AnyPref)
  - New keyword indicators in ADRUFO parameter list for user exit and API

26 New for DFSMSdss and System Data Mover

•Specifying CONCURRENT(CacheReq) indicates that

•cache-based CC is first attempted.

•If cache-based CC fails then the operation fails.

• Specifying CONCURRENT(CachePref) indicates that

•cache-based CC is first attempted.

•If cache-based CC fails standard I/O method is attempted.

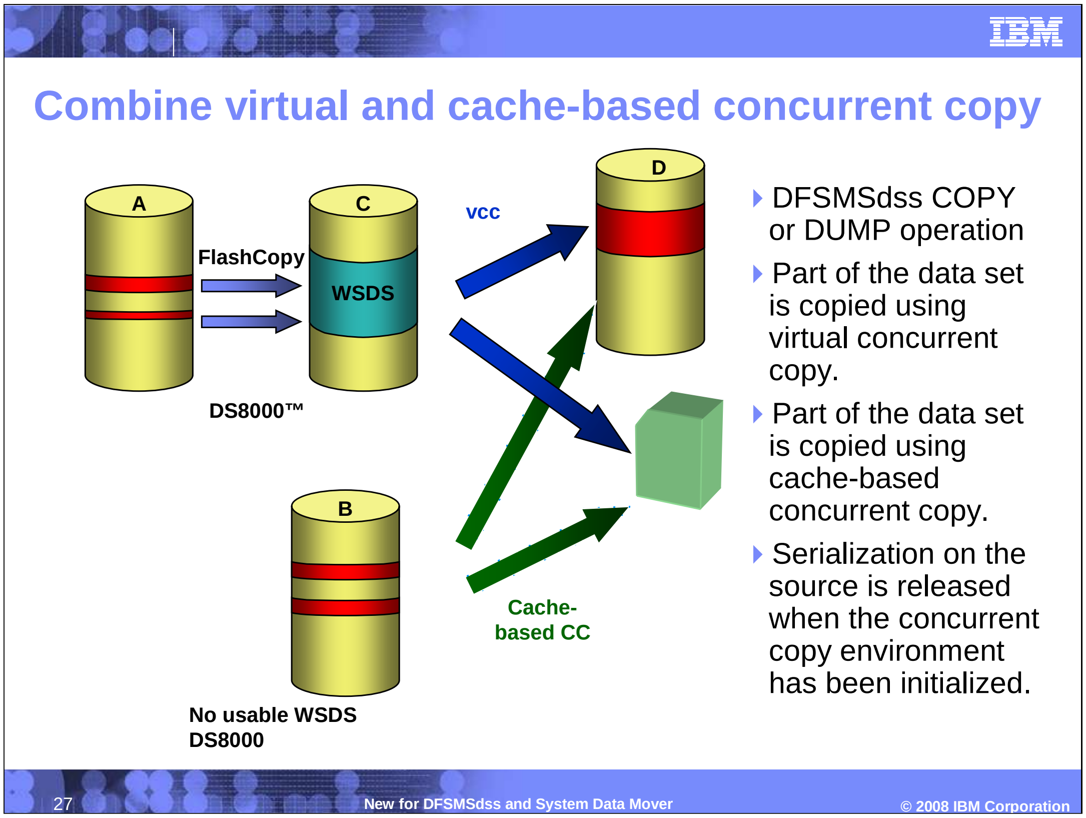
•Specifying CONCURRENT(STANDARD or NO) indicates

•that the behaves is as if CONCURRENT was not specified.

•Specifying CONCURRENT without a subkeyword

•Defaults to as if CONCURRENT(AnyPref) is specified.

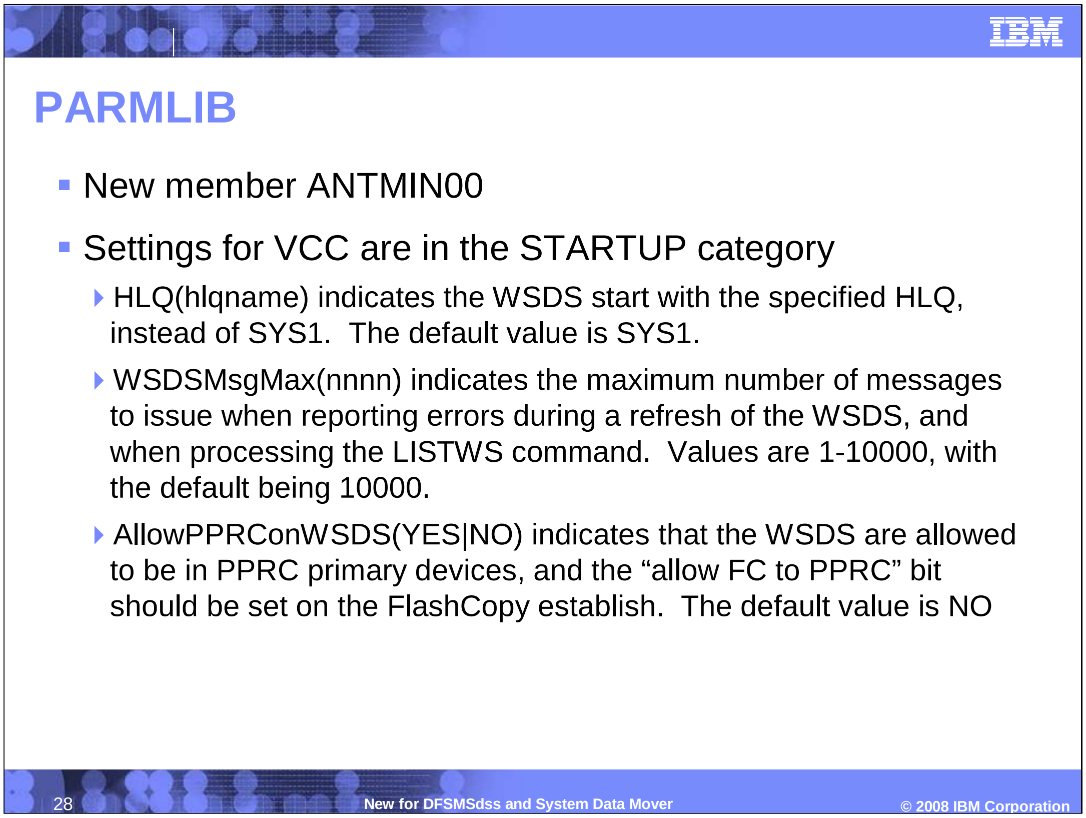
•The various options of CONCURRENT are also available to be specified in the ADRUFO parameter list for user exit and API invocations.



Combine virtual and cache-based concurrent copy

- DFSMSdss COPY or DUMP operation
- Part of the data set is copied using virtual concurrent copy.
- Part of the data set is copied using cache-based concurrent copy.
- Serialization on the source is released when the concurrent copy environment has been initialized.

This slide provides a visual representation of VCC using FlashCopy and Cache-based CC.



PARMLIB

- New member ANTMIN00
- Settings for VCC are in the STARTUP category
  - HLQ(hlqname) indicates the WSDS start with the specified HLQ, instead of SYS1. The default value is SYS1.
  - WSDSMsgMax(nnnn) indicates the maximum number of messages to issue when reporting errors during a refresh of the WSDS, and when processing the LISTWS command. Values are 1-10000, with the default being 10000.
  - AllowPPRConWSDS(YES|NO) indicates that the WSDS are allowed to be in PPRC primary devices, and the “allow FC to PPRC” bit should be set on the FlashCopy establish. The default value is NO

28 New for DFSMSdss and System Data Mover © 2008 IBM Corporation

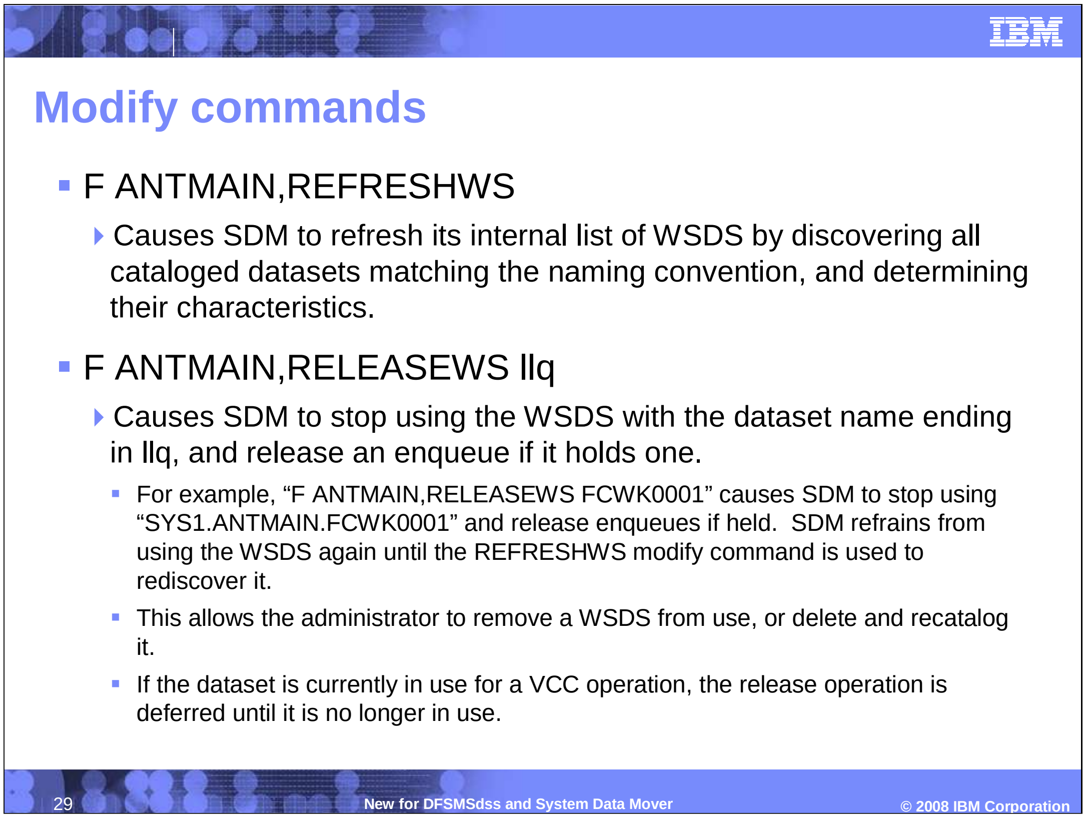
A new PARMLIB member has been added, to control VCC operations. The member name is ANTMIN00. Only one member is supported at this time.

Changes to the PARMLIB values are activated by cancelling the ANTMAIN address space at a time when there is no HSM, DSS or ANTRQST activity, or at IPL.

The HLQ parameter allows any valid data set qualifier to be substituted for SYS1.

The WSDSMsgMax parameter can be used to limit excessive messages when many working space data sets are allocated.

The AllowPPRConWSDS parameter can be used in an environment where the volumes containing the working space data sets must be PPRC primary volumes. When this parameter is set to YES, and the PPRC pairs were established with the parameter which allows a FlashCopy to be established, the PPRC pair will go into “Duplex Pending” mode. Use this option sparingly, only in environments where use of VCC is required and all volumes are in PPRC relationships, and then only by allocating working space data sets on volumes which are not critical for disaster recovery.



Modify commands

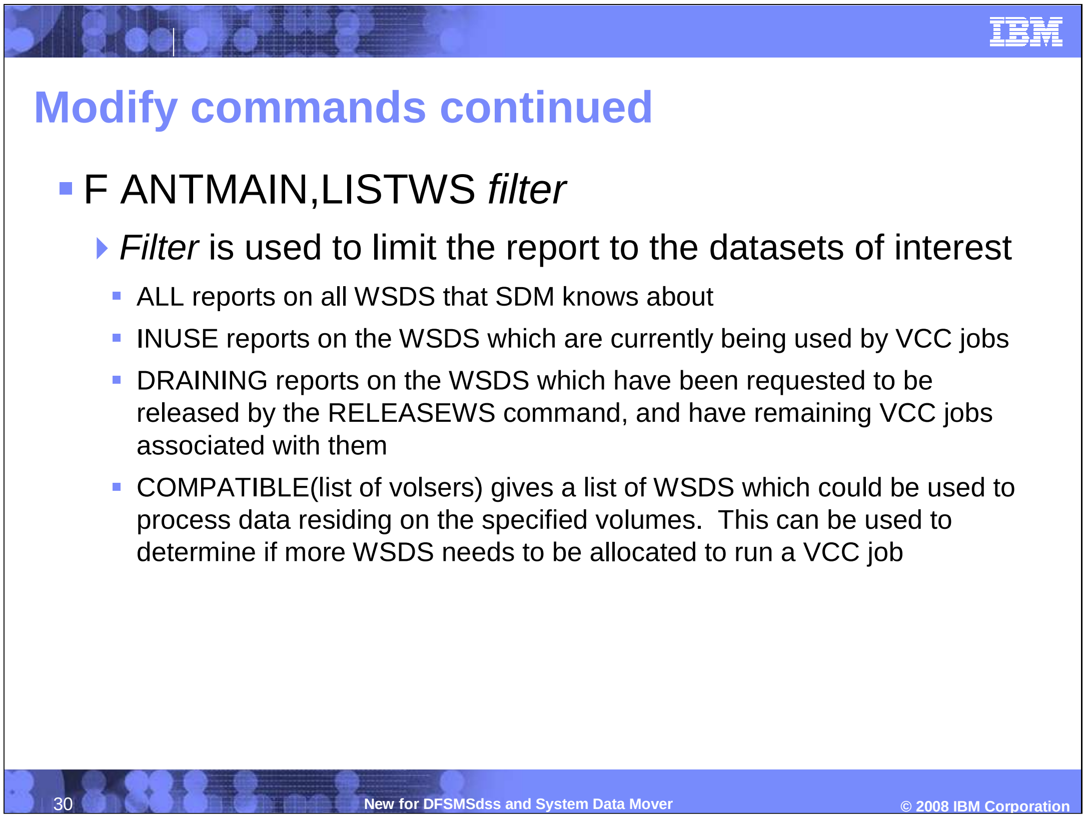
- F ANTMAIN,REFRESHWS
  - Causes SDM to refresh its internal list of WSDS by discovering all cataloged datasets matching the naming convention, and determining their characteristics.
- F ANTMAIN,RELEASEWS llq
  - Causes SDM to stop using the WSDS with the dataset name ending in llq, and release an enqueue if it holds one.
    - For example, “F ANTMAIN,RELEASEWS FCWK0001” causes SDM to stop using “SYS1.ANTMAIN.FCWK0001” and release enqueues if held. SDM refrains from using the WSDS again until the REFRESHWS modify command is used to rediscover it.
    - This allows the administrator to remove a WSDS from use, or delete and recatalog it.
    - If the dataset is currently in use for a VCC operation, the release operation is deferred until it is no longer in use.

SDM provides three modify commands to the ANTMAIN address space, to control VCC operations.

The REFRESHWS command causes SDM to refresh its internal list of working space data sets, by discovering all data sets which match the WSDS naming convention.

It is also used to inform SDM that it is OK to start using a data set which was released using the RELEASEWS command.

The RELEASEWS command causes SDM to release control of the specified WSDS. This allows the data set to be deleted, and allocated on another volume. It can also be used to force SDM to disregard a WSDS when processing VCC. The release operation occurs immediately if the data set is not in use, and is deferred if a VCC operation is currently using it.



Modify commands continued

- F ANTMAIN,LISTWS *filter*
  - *Filter* is used to limit the report to the datasets of interest
    - ALL reports on all WSDS that SDM knows about
    - INUSE reports on the WSDS which are currently being used by VCC jobs
    - DRAINING reports on the WSDS which have been requested to be released by the RELEASEWS command, and have remaining VCC jobs associated with them
    - COMPATIBLE(list of volsers) gives a list of WSDS which could be used to process data residing on the specified volumes. This can be used to determine if more WSDS needs to be allocated to run a VCC job

The LISTWS command produces a list of working space data sets. The filter parameter is required, and can be used to limit the amount of output produced by the list.

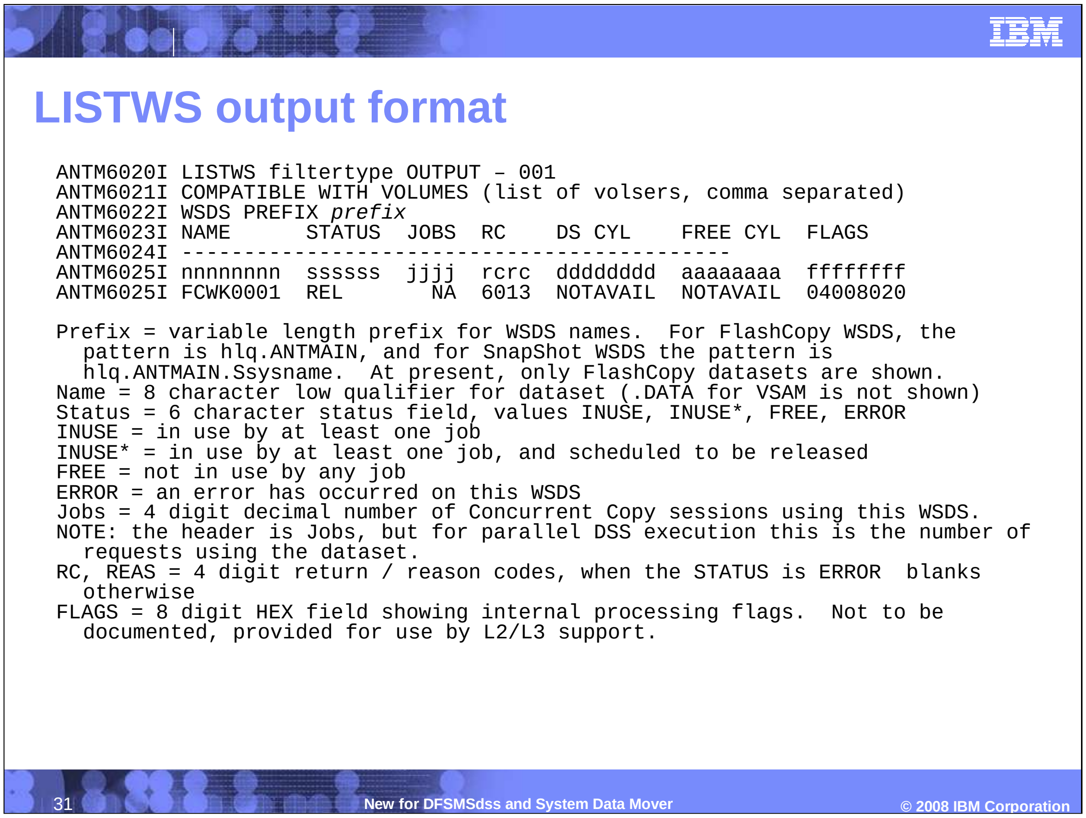
The PARMLIB setting WSDSMsgMax can also be used to limit the number of output lines.

The ALL filter is useful to determine if ANTMAIN can access all the expected WSDS

The INUSE filter could be used to find out the high water mark of WSDS usage at a given point in time

The DRAINING filter is used to determine the current status of WSDS which were released using the RELEASEWS command

The COMPATIBLE filter can be used to determine if there are data sets allocated on volumes which are able to be FlashCopied from the specified source volumes.

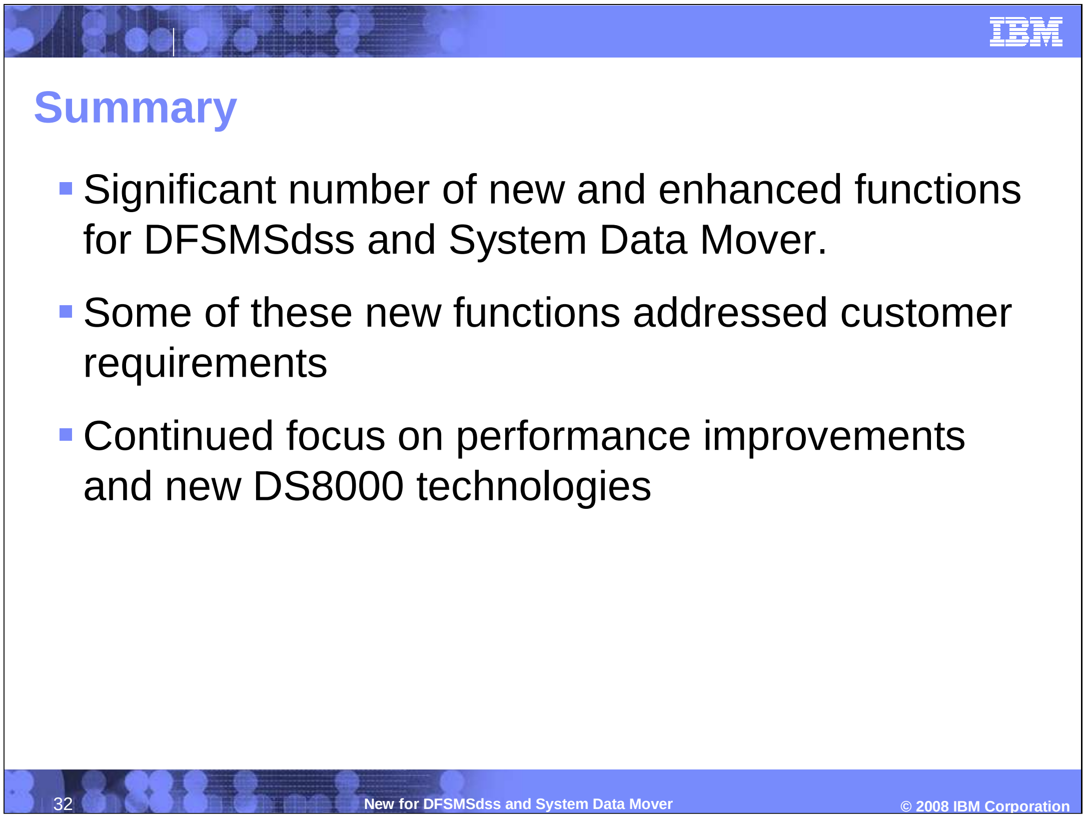


LISTWS output format

```
ANTM6020I LISTWS filtertype OUTPUT – 001
ANTM6021I COMPATIBLE WITH VOLUMES (list of volsers, comma separated)
ANTM6022I WSDS PREFIX prefix
ANTM6023I NAME      STATUS  JOBS  RC    DS CYL    FREE CYL  FLAGS
ANTM6024I --------------------------------------------
ANTM6025I nnnnnnnn  ssssss  jjjj  rcrc  dddddddd  aaaaaaaa  ffffffff
ANTM6025I FCWK0001  REL       NA  6013  NOTAVAIL  NOTAVAIL  04008020

Prefix = variable length prefix for WSDS names.  For FlashCopy WSDS, the
  pattern is hlq.ANTMAIN, and for SnapShot WSDS the pattern is
  hlq.ANTMAIN.Ssysname.  At present, only FlashCopy datasets are shown.
Name = 8 character low qualifier for dataset (.DATA for VSAM is not shown)
Status = 6 character status field, values INUSE, INUSE*, FREE, ERROR
INUSE = in use by at least one job
INUSE* = in use by at least one job, and scheduled to be released
FREE = not in use by any job
ERROR = an error has occurred on this WSDS
Jobs = 4 digit decimal number of Concurrent Copy sessions using this WSDS.
NOTE: the header is Jobs, but for parallel DSS execution this is the number of
  requests using the dataset.
RC, REAS = 4 digit return / reason codes, when the STATUS is ERROR  blanks
  otherwise
FLAGS = 8 digit HEX field showing internal processing flags.  Not to be
  documented, provided for use by L2/L3 support.
```

This chart shows the output format for the LISTWS command.



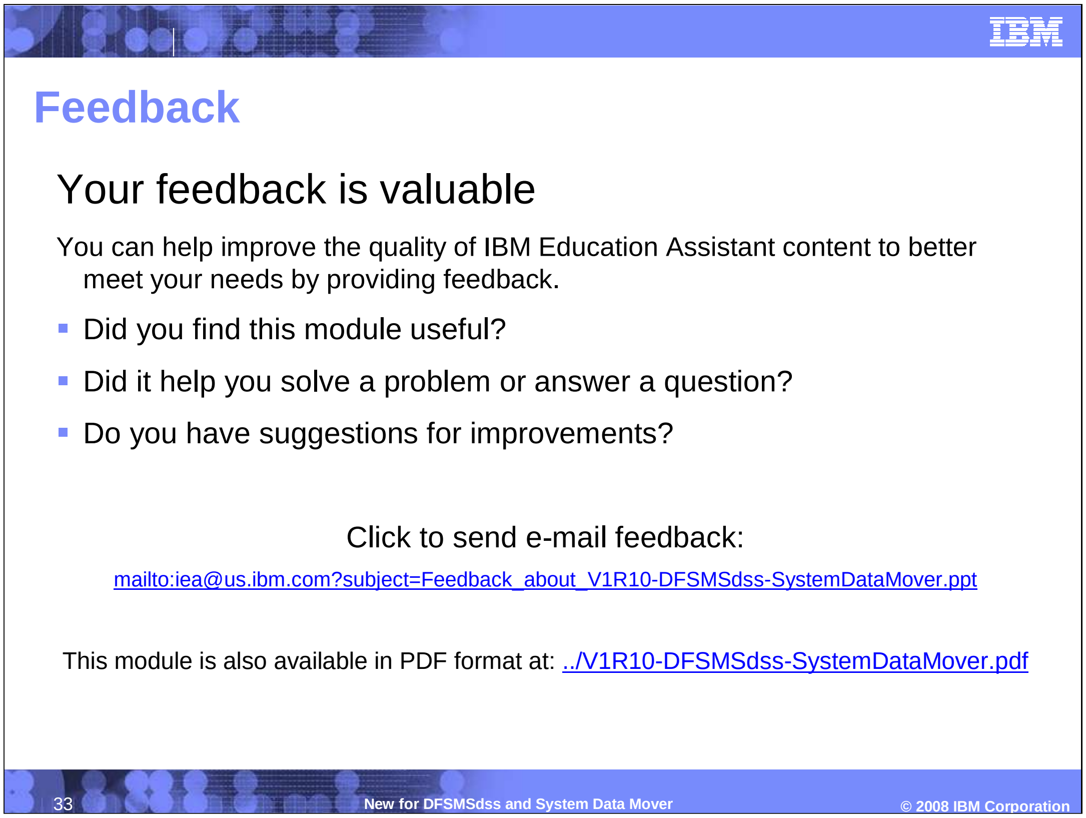
Summary

- Significant number of new and enhanced functions for DFSMSdss and System Data Mover.
- Some of these new functions addressed customer requirements
- Continued focus on performance improvements and new DS8000 technologies

This concludes the presentation. The presentation covered a significant number of new and enhanced functions for DFSMSdss and System Data Mover.

Some of these new functions addressed were to fulfill customer requirements.

These items demonstrate a continued focus on performance improvements and new DS8000 technologies.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

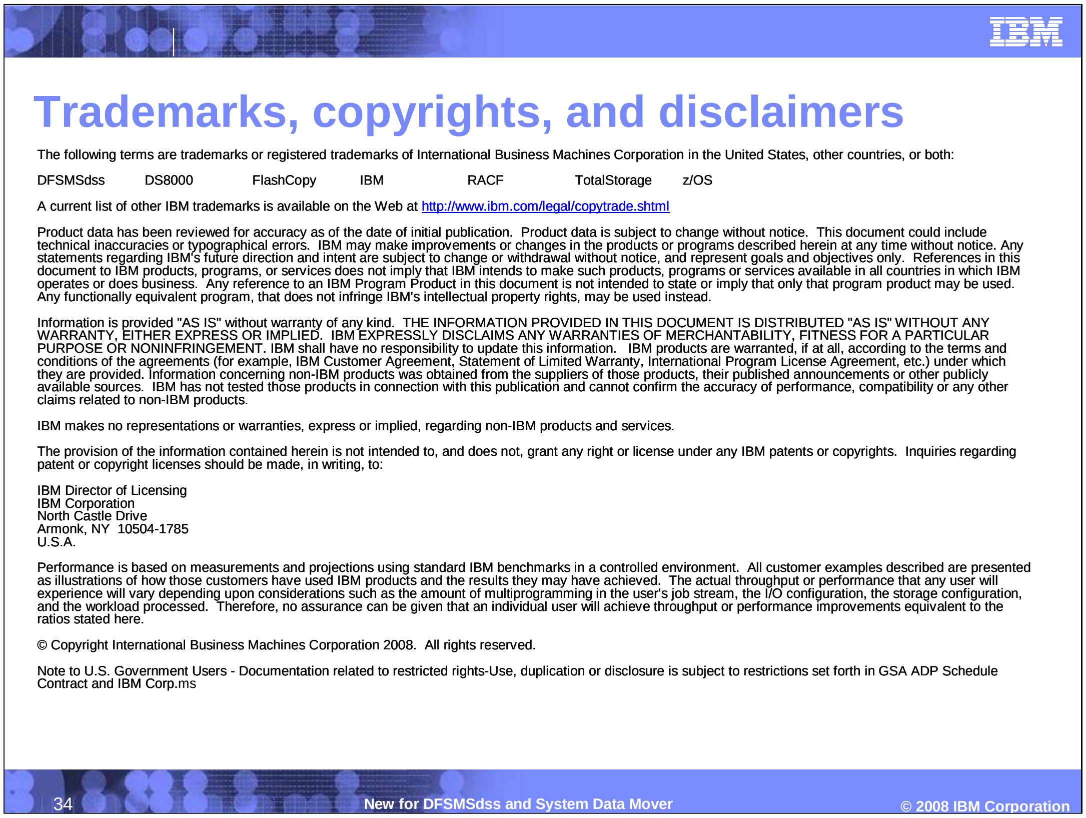
Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_V1R10-DFSMSdss-SystemDataMover.ppt

This module is also available in PDF format at: ../V1R10-DFSMSdss-SystemDataMover.pdf

33 New for DFSMSdss and System Data Mover

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

DFSMSdss DS8000 FlashCopy IBM RACF TotalStorage z/OS

A current list of other IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.ms