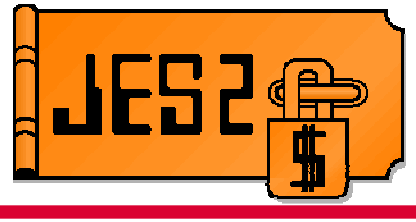
JES2 SPOOL: Defining, Managing, and Updating

SHARE Atlanta, 2012
Session 10844 – Thursday, March 15

Tom Wasik
JES2 Design/Development/Service
Rochester, MN
wasik@us.ibm.com

Trademarks



The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

IBM®
MVS
JES2
JES3
RACF®
z/OS®
zSeries®

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
UNIX is a registered trademark of The Open Group in the United States and other countries.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

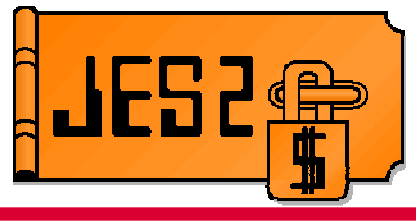
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

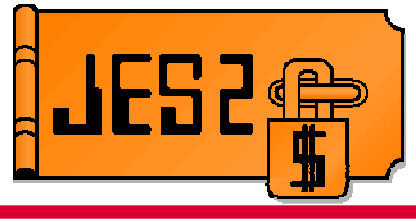
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Session Objectives



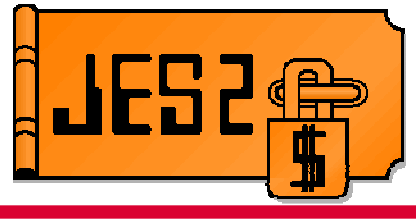
- **What is SPOOL**
- **How do you define it**
  - Flexibility on names and volumes
  - SPOOL commands and initialization statements
- **Managing and Updating SPOOL**
  - Extend SPOOL data set
  - SPOOL Migration

What is SPOOL?



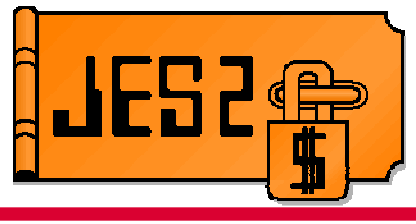
- **SPOOL is an acronym for *simultaneous peripheral operations on-line***
- **It is where JES2 stores bulk data**
  - SYSOUT/SYSIN/JCL data sets
  - Major control blocks (JCT, IOT, etc)
- **Comprised of 1 or more DASD volumes**
  - Each volume has one SPOOL data set on it
- **JES2 supports up to 252 SPOOL volumes**
- **Each volume can have up to 1M tracks**
  - Currently cannot use more than 1M track on a volume

Defining SPOOL



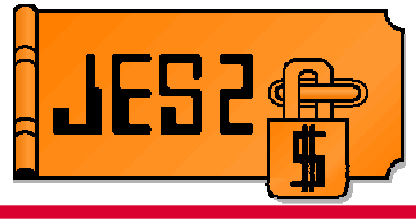
- **Keywords on SPOOLDEF define SPOOL**
- **Traditional method uses data set name and prefix**
  - VOLUME= is 4 or 5 character VOLSER prefix
  - DSNAME= is the 44 character data set name
  - All volume match the DSN and PREFIX
- **New method allows more flexability**
  - VOLUME= can have generics allowing more names
    - Can even be *
  - DSNAME= is just a default data set name
  - DSNMASK= specifies a generic pattern for data set
  - Any data set on any volume can be a SPOOL data set
    - Maintaining a naming convention is a good idea

How are SPOOLs located on a COLD start?



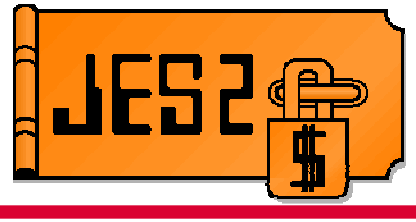
- **SPOOL initialization statement explicitly define SPOOL**
  - Volume serial must match PREFIX (with or without generics)
  - Data set name must match DSNMASK or it can default
  - Only specified volumes are used
- **JES2 discovers SPOOL volumes (traditional)**
  - Scans all online DASD UCBs for volume prefix
  - Attempts to allocate default data set name on each volume
  - If allocate works, it is a SPOOL volume
  - Used if no SPOOL init statements and VOLUME= has no generics

SPOOL Volumes on Warm Start?



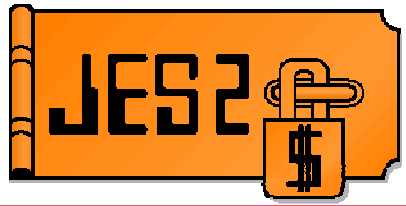
- **Always use same volumes as last time JES2 was up**
  - Volumes that were INACTIVE (halted) are not allocated to at warm start time
- **DSNMASK and PREFIX do not apply**
  - This allows the values to be changed for new SPOOL volumes without impacting existing volumes
- **Failure to allocate will cause start to fail**
  - UNLESS all member warm start, then volume can be set INACTIVE or deleted (losing all job on volume)

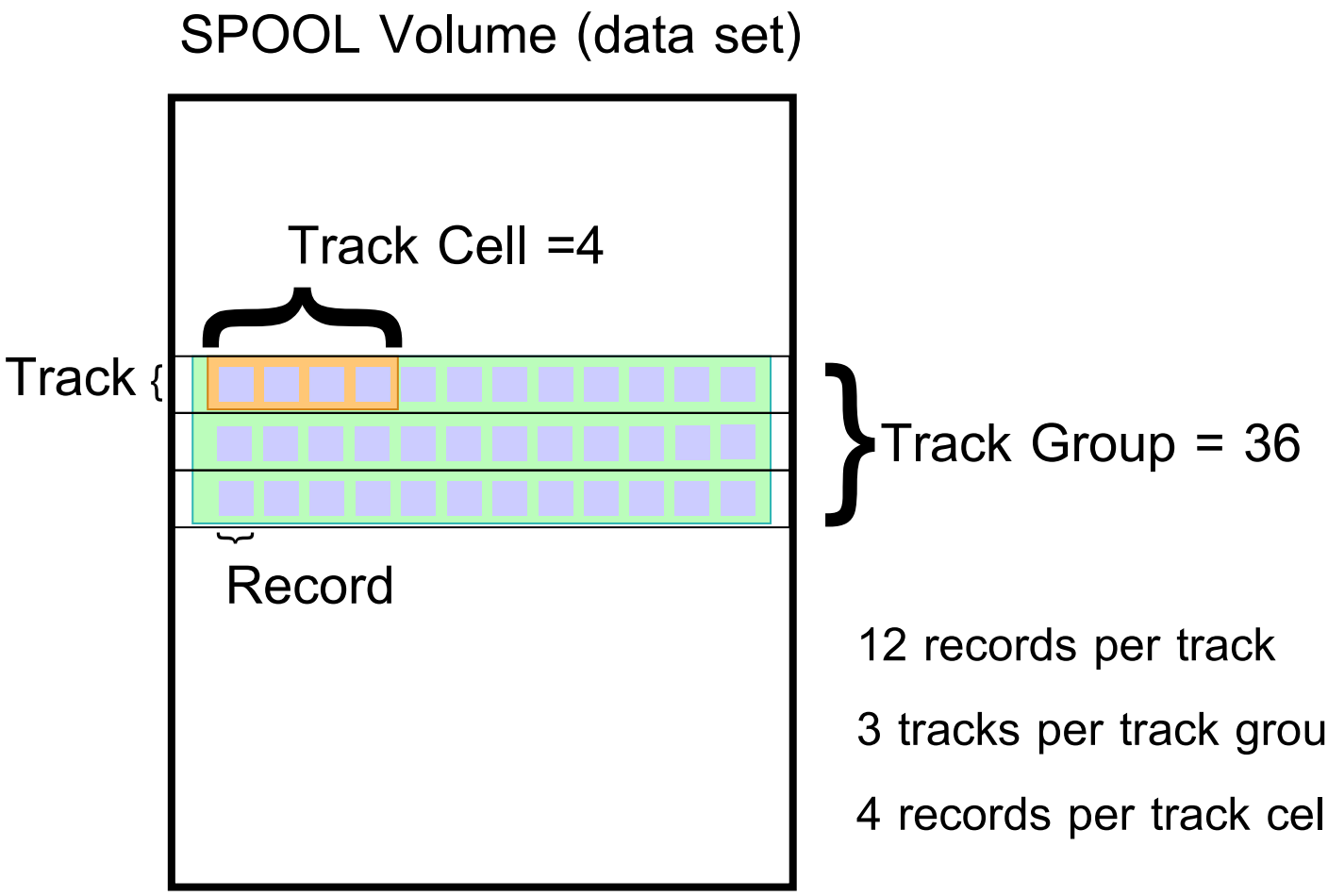
Other SPOOL Specifications (SPOOLDEF)



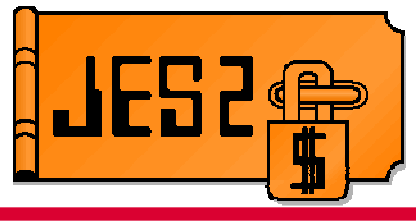
- **BUFSIZE controls size of record on SPOOL**
  - 1942-3992 valid range
  - Requires a cold start to alter
  - Recommend setting to 3992
- **TGSIZE controls SPOOL allocation size**
  - Track Group (TG) is the SPOOL space allocation unit
  - Expressed in records and rounded up to track
  - Can be changed via $T but binds to a volume at START
- **TRKCELL controls sub allocation unit size**
  - TRKCELLs allow better management of SPOOL space for non-SPIN data sets
  - Expressed in records (must be less than a track)
  - Can only be changed on a COLD start

Other SPOOL Specifications (SPOOLDEF)



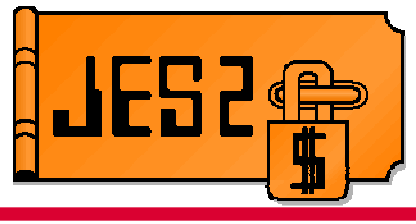


Other SPOOL Specifications (SPOOLDEF)



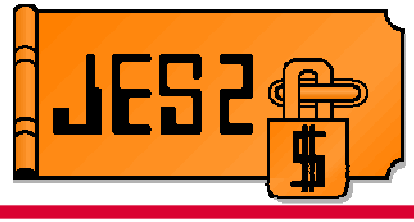
- **TGSPACE= defined the track group bit map**
  - MAX= specifies the total size of the map in bits
  - DEFINED= indicates how many bits are used by existing SPOOL volumes
  - ACTIVE= indicates bits (track groups) on volumes that are allocatable
  - FREE= number of bits (track groups) that are available on volumes athat ae allocatable
  - PERCENT= (ACTIVE-FREE)/ACTIVE

Addressing SPOOL



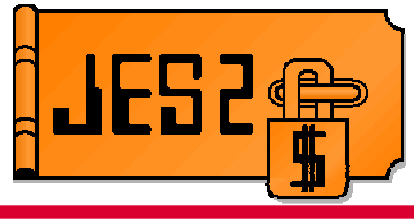
- **Basic addressing schemes**
  - MTTR – Extent (M) Track value (TT) and Record (R)
  - MQTR (MTTTTR) – Like MTTR with 4 bytes of T
  - MTTtr – Extension of MTTR with 20 bits of T and 4 bits for R
- **Addressing modes of SPOOL volumes**
  - Absolute – TT is from start of volume
  - Relative – TT is from start of data set
  - Large DS – 20 bit TT (MTTtr)

SPOOL Addressing History



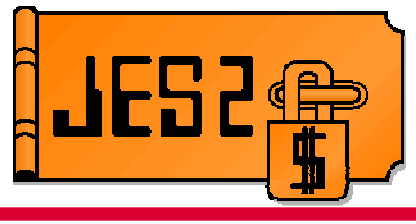
- **MTTR original addressing scheme**
  - TT was absolute track relative to start of data set
- **Relative track addressing added in z/OS1.2 (MTTR)**
  - SPOOLDEF RELADDR=
  - TT is relative to start of data set
  - Can place SPOOL anywhere on existing volumes
  - Size limit is 64K tracks
  - Rolled back to OS/390 release 10
- **Large data set support added in z/OS 1.7 (MTTtr)**
  - SPOOLDEF LARGEDS=
  - Implies relative addressing
  - 4 bits of R given to T so TTt can be 20 bits
  - Size limit is now 1M tracks
- **Cylinder managed support added in z/OS 1.12 (MTTtr)**
  - SPOOLDEF CYL_MANAGED
  - Implies relative addressing and large data set format
  - Data sets can exist beyond track managed storage
  - Size limit is still 1M tracks
- **Future is to move to all MQTRs to get past the 1M track limit**
  - Evolutionary change (MQTRs used in many places) with no timeframe for completion

SPOOL Addressing History



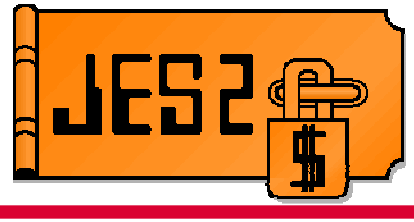
- **JES2 still supports accessing all format SPOOL volumes**
- **Volume format depends on settings when started**
  - $D SPOOL(x),UNITDATA display volume format (1.13)
- **Some commands restricted to newer formats**
  - Absolute addressed volumes cannot be extended
  - Must be LARGEDS format to go over 64K tracks

Formatting a SPOOL Volume



- **SPOOL volumes must be formatted**
  - Done by JES2 when they are first used
  - Time consuming process (faster in 1.13)
  - Can be done ahead of time
- **Pre-formatting makes starting a new volume faster**
  - Job to do it is in JES2 Initialization and Tuning Guide http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/has2a390/3.1.1.3

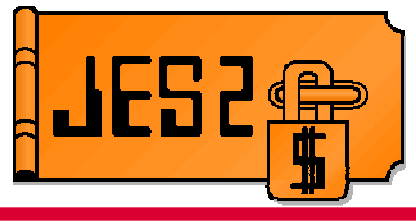
Formatting a SPOOL Volume



- **JCL to create and pre-format SPOOL**
  - Assuming 3992 byte BUFSIZE
  - Creates the SPOOL space on volume SPOOL2

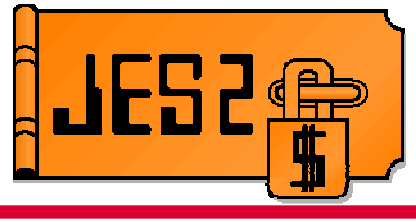
```
//ALLOCSPL JOB  (...),'SPOOL FORMAT',MSGLEVEL=1
//FORMAT   EXEC PGM=IEBDG
//*
//SPOOL    DD   DSN=SYS1.HASPACE,UNIT=3390,
//              VOL=SER=SPOOL2,DISP=(NEW,KEEP),
//              SPACE=(CYL,884,CONTIG),
//              DCB=(DSORG=PSU,RECFM=U,BLKSIZE=3992)
//*
//SYSPRINT DD   SYSOUT=A
//SYSIN    DD   *
  DSD OUTPUT=(SPOOL)
  FD NAME=SPOOL,FILL=X'FF',LENGTH=3992
  CREATE NAME=(SPOOL),QUANTITY=99999999
  END
/*
```

Key Properties of a SPOOL



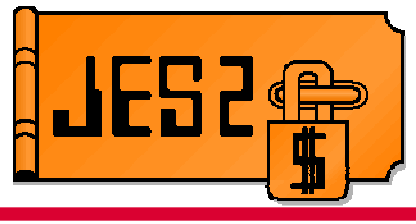
- **Selectable – Jobs that have space on the volume can be selected for processing**
  - Includes SYSOUT and JOBs
- **Allocatable – New SPOOL space can be handed out from the volume**
  - Track groups for this volume are in the BLOB

States a SPOOL Can Be In



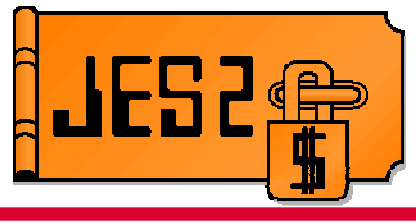
- **ACTIVE – Normal state of a volume**
  - Selectable and allocatable
- **STARTING – Volume transitioning to ACTIVE.**
  - Could be initial use of a volume
    - Not selectable or allocatable
  - Could be transitioning from a state below
    - Selectable/allocatable based on old state
- **HALTING – Transitioning to INACTIVE**
  - Not selectable and not allocatable
  - Waiting for active address processes to stop
- **INACTIVE – Not MVS allocated but still defined**
  - Not selectable and not allocatable
- **DRAINING – Transitioning to deleted (not exist)**
  - Selectable but not allocatable
  - Waiting for all jobs with space on the volume to go away

States a SPOOL Can Be In



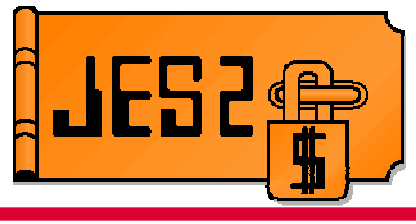
- **Reserved – Property of an ACTIVE volume**
  - Not allocatable but selectable
  - Not waiting or transitioning to a new state
  - Set/reset by $T or $S command
- **Migrating – Active migration moving data**
  - Not selectable and not allocatable
  - Will transition to MAPPED when migration completes
    - DRAINING if migration fails
- **Mapped – Physically exists on a target volume**
  - Not allocatable, selectable inherited from target volume
  - Waiting for all jobs with space on volume to go away
  - Will transition to does not exist (like DRAINING)
  - No commands allowed against volume (cannot change state)
- **Extending – Size of volume being increased**
  - Selectable and allocatable
  - Returns to ACTIVE when process completes

SPOOL Commands



- **$S – Start a new volume or restart exiting**
  - Places volume in STARTING state
- **$Z – Halt a SPOOL volume**
  - Places volume in HALTING state
- **$P – Drain a SPOOL volume**
  - Places volume in DRAINING state
- **$M – Migrate a SPOOL to a new volume**
  - Starts migration process eventually placing volume in MIGRATING state
- **$T – Alter the attributes of a SPOOL volume**
- **$T SIZE= - Extend a SPOOL into adjacent free space**
  - Places volume in EXTENDING state

Starting a SPOOL volume

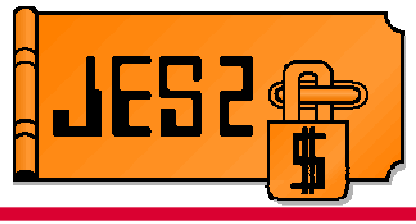


- **Command to start a new volume is**

  $S SPOOL(*volser*),DSN=*dsname*

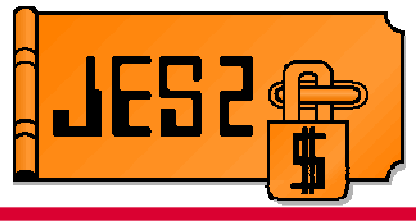
- ***volser* must match VOLUME=**
  - Either as a prefix or a generic match
- ***dsname* must match either**
  - Default data set name (SPOOLDEF DSN=)
  - Generic match of SPOOLDEF DSNMASK=
- **Can specify SPACE= to create data set**
  - Specify number of cylinders, tracks, or MAX

Starting a SPOOL volume



- **Other operands on $S**
  - FORMAT forces JES2 to format the volume
  - RESERVED= specifies to set the reserved property once volume is started
  - HALT or Z specifies to place the volume in HALTING state once it is started
  - DRAIN or P specifies to place the volume in DRAINING state once it is started
  - CANCEL when specified with DRAIN or P will issue a $CJ,P for all jobs on the volume after the volume is placed in DRAINING state

Starting a SPOOL volume



- **What is a "Mini-Format"?**
  - When starting a new volume you often see

```
$HASP423 SPOOL1 IS BEING MINI-FORMATTED
```

  - This occurs even when the volume was pre-formatted
- **Mini format resets the record 0 for the volume**
  - Record 0 (R0) is an 8 byte record at the start of every track that is owned by the access method using the data set
  - JES2 uses it to record ownership of each track group on SPOOL
  - When a new volume is started, R0 on every track is set to a known value
    - If the volume needs to be formatted, that process resets R0
    - If the volume is pre-formatted, then a mini-format just resets R0

Extending a SPOOL Volume

- **Command to extend SPOOL into adjacent free space**
  - $TSPOOL(*xxxxxx*),SPACE=
    - Syntax for SPACE= same as $S SPOOL
      - MAX, (TRK,xxxx), (CYL,xxxx)
- **SPACE= is the NEW TOTAL size of the data set**
  - It is NOT the increment
- **Extend occurs without impacting running jobs**
  - New space is always formatted by JES2
- **Message $HASP740 indicates Extend is successful**
- **$DSPOOL displays the results of the extend**
  - $DSPOOL,TGNUM displays the number of track groups in the data set
  - $DSPL,UNITDATA displays the track range (TRKRANGE) of the data set

Extending a SPOOL Volume

- **Sample commands:**

```
$TSPOOL(SPOOLX),SPACE=MAX
 $HASP893 VOLUME(SPOOLX)
 $HASP893 VOLUME(SPOOLX)  STATUS=ACTIVE,AWAITING(EXTEND),
 $HASP893                 COMMAND=(EXTEND),PERCENT=0
 $HASP646 3.4074 PERCENT SPOOL UTILIZATION
 $HASP630 VOLUME SPOOLX ACTIVE    42 PERCENT UTILIZATION
 $HASP740 Volume SPOOLX Extend successful.

$TSPOOL(SPOOLX),SPACE=(CYL,200)
 $HASP893 VOLUME(SPOOLX)
 $HASP893 VOLUME(SPOOLX)  STATUS=ACTIVE,AWAITING(EXTEND),
 $HASP893                 COMMAND=(EXTEND),PERCENT=1
 $HASP646 3.5151 PERCENT SPOOL UTILIZATION
 $HASP443 SPOOLX DATASET SYS1.HASPACE NOT EXTENDED
EXTEND SPOOL UNSUCCESSFUL RC=20
 $HASP741 Volume SPOOLX Extend unsuccessful. Error Code = 60,
 Insufficient space.
```

Extending a SPOOL Volume

- **Extension of the data set is limited by:**
  - SPOOL volume must be
    - STATUS=ACTIVE
    - No commands or migration active or pending against it
    - Using relative addressing
  - Available free space contiguous (after) to the JES2 SPOOL extent
  - Total size limited to architecture
    - JES2 limit is based on LARGEDS on SPOOLDEF
      - Allowed/Always – limit is 1M tracks
      - Fail – limit is 64K
    - DSCB format limits expansion into EAS storage
      - Should migrate to CYL_MANAGED=ALLOWED
      - Allocate SPOOL using DD EATTR = OPT to build format 8/9 DSCB
  - All members of the MAS must be at JES2 z/OS V1R13
- **After extend completes, down level members can join the MAS and use the extended data set**
- **Single JES2 SPOOL extent per volume restriction still applies**

SPOOL Migration

- **$M SPOOL command to move data off volume**
  - Faster than $P SPOOL (Minutes not days)
  - Function enabled with OA36158 (PTF UA64366)
- **Command works with active address spaces using volume**
  - Less activity is better/faster but no need to IPL to stop active jobs
- **Goal of SPOOL migration is to stop using SPOOL data set**
  - It is NOT to eliminate the internal representation of the volume
  - Old data set can be deleted and SPOOL volume taken offline
- **After a successful SPOOL migration**
  - $DSPOOL still shows volume
  - $DJQ,SPOOL= still displays volume
  - New status is MAPPED

Migrating a SPOOL Volume

Key Terms and Definitions:

- ***Source Volume*** – The SPOOL to be migrated.
- ***Target Volume*** – The SPOOL to receive the migrated data.
- ***MERGE Migration*** – Copy a *Source Volume* to free space on an existing *Target Volume*.
- ***MOVE Migration*** – Copy an inactive *Source Volume* to a new *Target Volume*.
- ***Active Migration*** – A migration that is currently being processed
- ***Migrator*** – The member that coordinates the migration.
- ***Migration Phase*** – The current 'step' of the migration process

Migrating a SPOOL Volume

Key Terms and Definitions:

- ***Mapped Volume:***
  - When a *Merge Migration* completes, the *Source Volume* becomes *Mapped.*
  - *Mapped Volumes* are deleted when all Jobs with space on it have been purged.
  - *Mapped Volumes* are no longer allocated to the SPOOL data set
    - The SPOOL data set on the volume can be deleted
    - The physical device can be removed.
- ***Mapped Target:***
  - A volume with at least one *Mapped Volume* mapped onto it.
- ***Reserved*** **:**
  - Attribute of any SPOOL volume set via $T SPOOL(volser),RESERVED=YES|NO
  - Indicates if the SPOOL volume is selectable but not allocatable.
  - Can be used to Reserve a volume for future *Merge Migration(s)*.
  - Reserved volumes have no entries in the BLOB.

Migrating a SPOOL Volume

- **Phases of a SPOOL migration**
  - PENDING – Command issued and queued for processing
  - INITIALIZING – Create data structures and subtasks.
  - SETUP – Prepare source and target data set
  - COPY – First pass copy of all data from source to target
  - CATCHUP – Second pass copy of tracks updated by active applications
  - CANCEL – Error phase that synchronizes stopping migration
  - BACKOUT – Error phase to undo any work done in migration
  - CLEANUP – Delete data structures and end active migration
- **Cancel can be requested up until start of catchup phase**
  - Internal cancel can occur later in error recovery cases
- **Phase start/end messages issued to SYSLOG**
  - DEBUG VERBOSE=YES sends messages to console
- **Some source volume state changes occur before the INITIALIZING phase and after the CLEANUP phase**

Migrating a SPOOL Volume

- **Two forms of SPOOL migration, MOVE and MERGE**
  - Move takes all data on an existing volume and moves it to a new one
    - Source must be INACTIVE ($Z SPOOL done)
      - No active jobs on the volume
    - Target cannot be currently an active SPOOL volume
    - Can specify space to use to create data set on target
    - At the end of move, old (source) volume does not exist
    - Target after a move is active
  - Merge takes all data on one volume and merges it onto free space on another volume
    - Most flexible migration option
    - Source can be in any state with active jobs/address spaces
      - Less activity is good
    - Results is a mapped volume that goes away when all jobs using it are deleted
      - Similar to $P SPOOL but device is no longer in use

Migrating a SPOOL Volume

- **Command syntaxes**
- **$M SPOOL command syntax (merge)**

  $M SPOOL(*volser*),TARGET=*target*
- **$M SPOOL command syntax (move)**

  $M SPOOL(*volser*),TARGET=*target*

  [,SPACE=(CYL|TRK|MAX,*size*)]

  [,DSNAME=*dsname*]

  [,RESERVED]
- **$M SPOOL cancel command**

  $M SPOOL(*volser*),CANCEL

- **Multi-source move command is also supported**

  $M SPOOL(*volsr1,volsr2,volsr3…*),TARGET=*volser*
- **1st volume can be a move or a merge, remainder are merges**
- **Migration happens 1 volume at a time (one per target)**

Migrating a SPOOL Volume

- **General restriction (for move and merge migrations):**
  - The *Source Volume* cannot be a *Mapped Target*
    - Cannot merge A to B and then move or merge B to C
      - Once A no longer exists then B can be migrated to C
  - The *Source Volume* cannot be actively migrating or extending.
  - The track size of the *Target Volume* cannot be less than the *Source Volume*
  - The *Source Volume* cannot be stunted
  - All MAS members must be at JES2 z/OS 1.13
  - Must be at z11 checkpoint mode.
- **Each SPOOL migration requires a separate XCF group**
  - Used to manage messages for each unique migration
  - JES2 limits migration to 5 concurrent migrations per MAS
  - Group name is SYSMG*xxx*
    - *xxx* is the decimal source SPOOL extent
  - Use D XCF,COUPLE to display MAXGROUP formatted in CDS

Migrating a SPOOL Volume

- **Move migration moves an INACTIVE volume to a new volume**
- **Upon successful completion**
  - The *Source Volume* no longer exists
  - The *Target Volume* exists and is active
    - Could be RESERVED if requested on $M SPOOL command
- ***Source Volume* STATUS= values:**
  - INACTIVE ->MIGRATING ->does not exist
- ***Target Volume* STATUS= values:**
  - Does not exist ->ACTIVE
- **Additional move migration restrictions :**
  - The *Source Volume* must be INACTIVE
  - *Source Volume* cannot be in Absolute format (instead, do a merge).
  - The Target Volume will inherit the Source Volume Tracks per Track Group value.

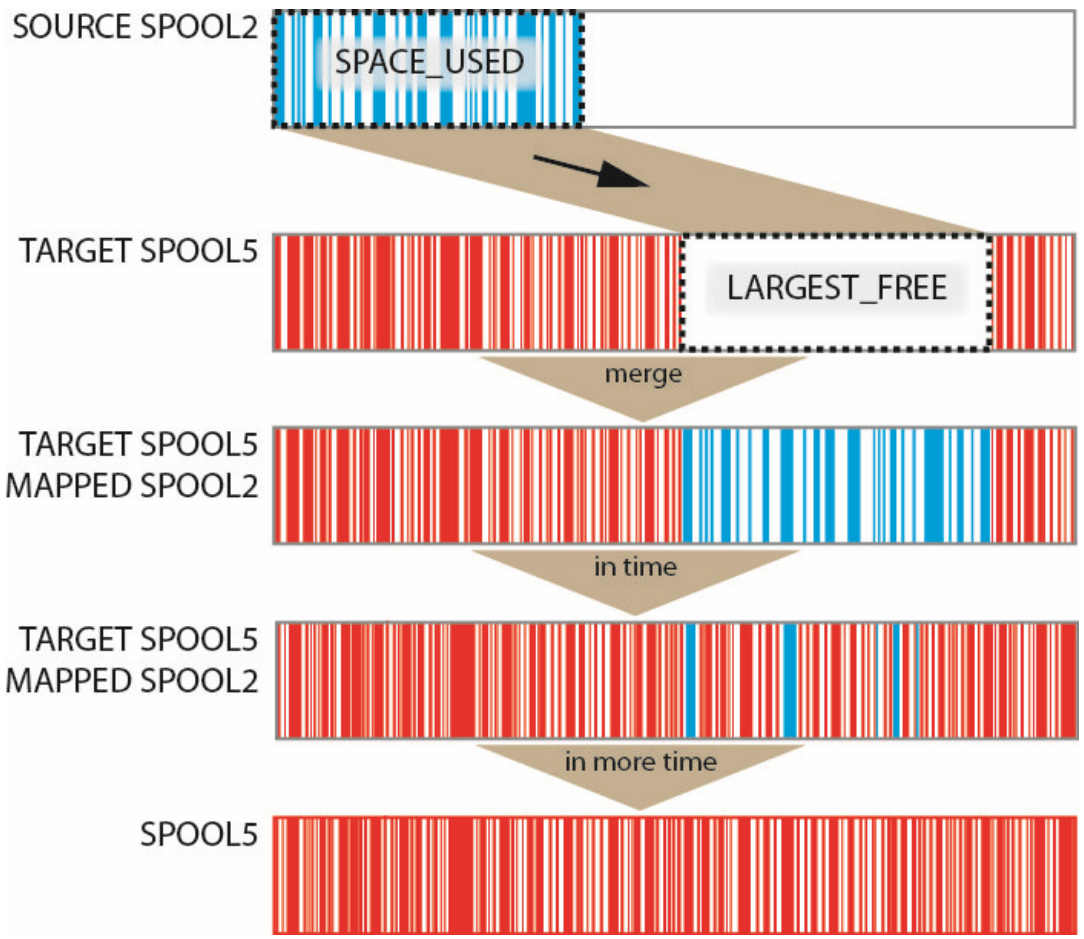
Migrating a SPOOL Volume

- **Merge migration moves a S*ource Volume* to an free space on an active *Target Volume***
- **Upon successful completion**
  - The *Source Volume* still exists but is STATUS=MAPPED
    - Still displays in $DSPOOL and in $DJQ,SPOOL lists
  - The *Target Volume* is a mapped on volume
- ***Source Volume* STATUS= values:**
  - INACTIVE ->MIGRATING ->MAPPED
- **Additional merge migration restrictions**
  - The *Target Volume* must be *Active* (can be *Reserved*).
  - The *Target Volume* cannot be stunted.
  - The *Target Volume* must use relative addressing..

Migrating a SPOOL Volume

MERGE Migration :

- **Copies an existing *Source Volume* to free space on a *Target Volume* :**
  - Upon completion, the *Source Volume* becomes a *Mapped Volume.*
- **Remains *MAPPED* until all jobs and SYSOUT that have space on the *Source Volume* are purged. It then goes away (no longer exists).**
- **Notice free space can be allocated through target volume**
  - No new allocations through source view



Migrating a SPOOL Volume

- **$D SPOOL..,MIGDATA helps determine migration requirements**
  - SPACE_USED is high water mark of used space on volume
  - LARGEST_FREE is largest contiguous free space on the volume

```
$D SPOOL,MIGDATA
$HASP893 VOLUME(SPOOL2)  MIGDATA=(SPACE_USED=433410,
$HASP893                 LARGEST_FREE=16590)
$HASP893 VOLUME(SPOOL5)  MIGDATA=(SPACE_USED=418215,
$HASP893                 LARGEST_FREE=31785)
```

  - Display all volumes having contiguous free space greater than 17000 tracks**:**

```
$D SPOOL,MIGDATA=LARGEST_FREE>17000,MIGDATA
$HASP893 VOLUME(SPOOL5)  MIGDATA=(SPACE_USED=418215,
$HASP893                 LARGEST_FREE=31785)
```

- **Note:** Track groups in the BLOB are considered to be used (not free)

Planning a Migration

- **Map out your new SPOOL configuration**
  - How many SPOOLs and what volumes are moving where.
  - Are you consolidating volumes? Just moving them?
- **Determine MOVE or MERGE migration.**
  - Merge is the preferred method due to the reduced number of restrictions
  - Merge is assumed for this discussion
- **Consider issuing a drain ($P) command for source**
  - Ensure there is enough free space on other volumes
  - Do this before as soon as practical before a merge
  - This reduces the time and impact of the merge has on the system

Planning a Migration

- **This is the time to adjust your TGSIZE if needed.**
  - TGSIZE is bound to a volume when it is started
  - Now is a good time to evaluate that
- **Ensure there are enough track groups for the existing volume PLUS any volume that will be added for the merge**
  - $DSPOOLDEF,TGSPACE and compare MAX (the current limit) to DEFINED (those used by existing volumes)
  - Difference needs to cover the space needed by new target volumes
  - You can use $T SPOOLDEF to update the values but may need to expand the CKPT to accommodate a new limit.
- **If new volumes are to be used for target volumes, start them now**
  - $S SPOOL with the reserved attribute so they are ready to use.
- **These previous steps could be done the day before the migration**
  - These have little to no impact on the running system

Planning a Migration

- **Perform the needed migration(s)**
  - Presumably during a quieter time on the system
- **Once all migrations to a specific target volume are complete, reset the reserved attribute of the target**
  - Allows target to resume normal operations.
- **At this point the source volume(s) are marked as STATUS MAPPED.**
  - Once all jobs age off the system, the mapped volume will go away
  - You may need to do a rolling IPL to finally get all jobs off old volume

Migrating a SPOOL Volume

$M SPOOL to move an inactive Source Volume to a new Target Volume

```
$mspool(spool2),target=spool3,reserved
$HASP808 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=9 -- Migration INITIALIZING phase started.
IXZ0001I CONNECTION TO JESXCF COMPONENT ESTABLISHED,
         GROUP SYSMG001 MEMBER POK$IBM1
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=49 -- Migration phase INITIALIZING is complete. Migrator
         and spool assistant subtasks have been attached.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=32 -- Migration phase SETUP-MOVE is starting.
$HASP423 SPOOL3 IS BEING MINI-FORMATTED
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=5 -- Completed allocation of target volume.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=26 -- Migration phase COPY-MOVE is starting.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=27 -- Migration phase COPY-MOVE is complete.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=28 -- Migration phase CATCHUP-MOVE is starting.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=29 -- Migration phase CATCHUP-MOVE is complete.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=47 -- Migration phase CLEANUP-MOVE is starting.
IXZ0002I CONNECTION TO JESXCF COMPONENT DISABLED,
         GROUP SYSMG001 MEMBER POK$IBM1
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=31 -- Migration phase CLEANUP-MOVE is complete.
$HASP808 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=33 -- Migration processing completed. Migration was
         successful.
```

Migrating a SPOOL Volume

$M SPOOL to merge Source Volume(s) to an existing Target Volume:

```
$mspool(spool2),target=spool3
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=4 -- Initiated drain of source volume.
$HASP808 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=9 -- Migration INITIALIZING phase started.
IXZ0001I CONNECTION TO JESXCF COMPONENT ESTABLISHED,
         GROUP SYSMG001 MEMBER POK$IBM1
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=49 -- Migration phase INITIALIZING is complete. Migrator
         and spool assistant subtasks have been attached.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=24 -- Migration phase SETUP-MERGE is starting.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=25 -- Migration phase SETUP-MERGE is complete.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=26 -- Migration phase COPY-MERGE is starting.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=27 -- Migration phase COPY-MERGE is complete.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=28 -- Migration phase CATCHUP-MERGE is starting.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=29 -- Migration phase CATCHUP-MERGE is complete.
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=47 -- Migration phase CLEANUP-MERGE is starting.
IXZ0002I CONNECTION TO JESXCF COMPONENT DISABLED,
         GROUP SYSMG001 MEMBER POK$IBM1
$HASP809 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=31 -- Migration phase CLEANUP-MERGE is complete.
$HASP808 Migration of SOURCE=SPOOL2 volume to TARGET=SPOOL3 volume
         RC=33 -- Migration processing completed. Migration was
         successful
```

Migrating a SPOOL Volume

- **Use $D SPOOL to monitor a migration and check results :**
  - MANY ways to view and filter. Here are a few examples :
- **Display all spool volumes that are currently migrating:**

```
$D SPOOL(*),STATUS=MIGRATING
$HASP893 VOLUME(SPOL7) STATUS=MIGRATING-MOVE,TARGET=SPOL2
$HASP893 VOLUME(SPOL8) STATUS=MIGRATING-MOVE,TARGET=SPOL2
$HASP646 75.0000 PERCENT SPOOL UTILIZATION
```

- **Display all spool volumes that have a Target (are Mapped) :**

```
$D SPOOL,TARGET^=' ',TARGET
$HASP893 VOLUME(SPOL4) TARGET=SPOL11
$HASP646 80.0000 PERCENT SPOOL UTILIZATION
```

- **Display all spool volumes that are reserved :**

```
$DSPL,RESERVED=YES
$HASP893 VOLUME(SPOL2) STATUS=RESERVED,PERCENT=20
$HASP893 VOLUME(SPOL3) STATUS=RESERVED,PERCENT=40
$HASP646 30.0000 PERCENT SPOOL UTILIZATION
```

- **$D SPOOL,PHASE – Displays current migration phase**
- **$D SPOOL,MPERCENT – Displays percent of migration that is complete**

Migrating a SPOOL Volume

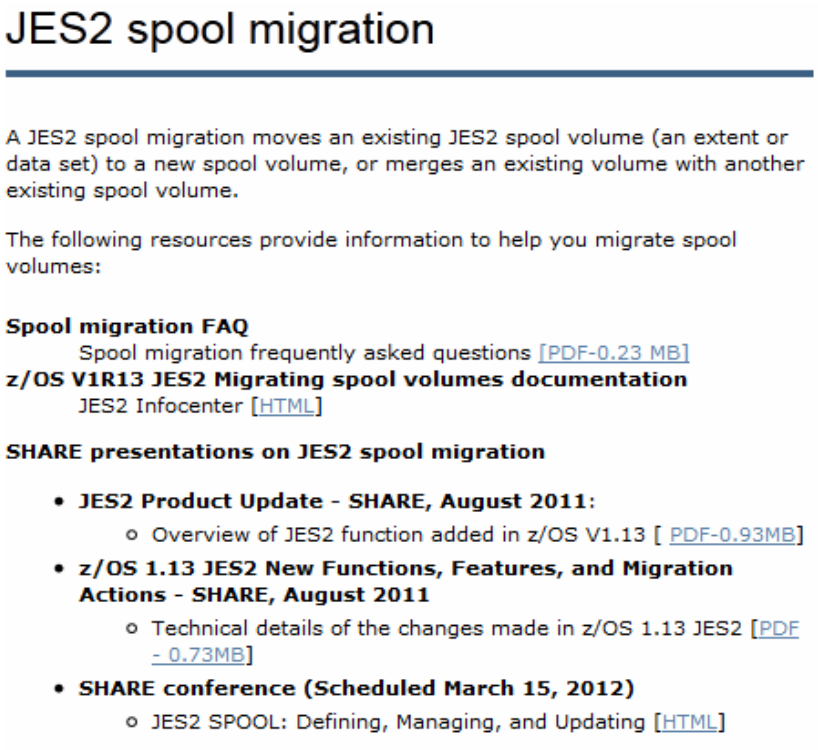
- **$DPERFDATA(MIGRSTAT) displays migration statistics**
  - Information on migrator for successful migrations

```
$HASP660 SPOOL MIGRATION STATISTICS C09,
$HASP660 MERGE OF VOLUME SPLX4Y TO SPLX4Z AT 2011.166,10:24:23
$HASP660  INIT           1.506416
$HASP660  SETUP          0.613308
$HASP660  COPY       20:06.822199 TRKS   951999 MSGS   538850
$HASP660  CATCHUP     1:16.766202 TRKS      170 MSGS     1268
$HASP660  CLEANUP        0.728138
$HASP660  TOTAL      21:26.437014
```

- **Note that CATCHUP time includes a 1 minute cool down timer.**

SPOOL Migration Enabled

- **JES2 SPOOL migration function has been enabled**
  - APAR OA36158 (PTF UA64366) closed February 24, 2012
- **New SPOOL migration page on the web**
  - http://www-03.ibm.com/systems/z/os/zos/jes2_spoolmigration.html
- **SHARE session**
  - **10844: JES2 SPOOL: Defining, Managing, and Updating**
    - Thursday 8:00AM



JES2 spool migration

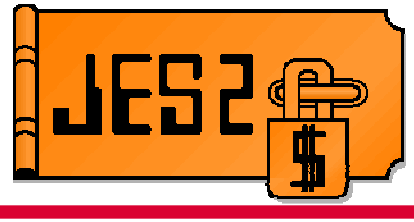
A JES2 spool migration moves an existing JES2 spool volume (an extent or data set) to a new spool volume, or merges an existing volume with another existing spool volume.

The following resources provide information to help you migrate spool volumes:

Spool migration FAQ
Spool migration frequently asked questions [PDF-0.23 MB]
z/OS V1R13 JES2 Migrating spool volumes documentation
JES2 Infocenter [HTML]

SHARE presentations on JES2 spool migration

- **JES2 Product Update - SHARE, August 2011:**
  - Overview of JES2 function added in z/OS V1.13 [PDF-0.93MB]
- **z/OS 1.13 JES2 New Functions, Features, and Migration Actions - SHARE, August 2011**
  - Technical details of the changes made in z/OS 1.13 JES2 [PDF - 0.73MB]
- **SHARE conference (Scheduled March 15, 2012)**
  - JES2 SPOOL: Defining, Managing, and Updating [HTML]



Questions?

Session 10844