# Need Space? A How-to on EAV Planning and Best Practices

Barbara McDonald, IBM - bawhite@us.ibm.com

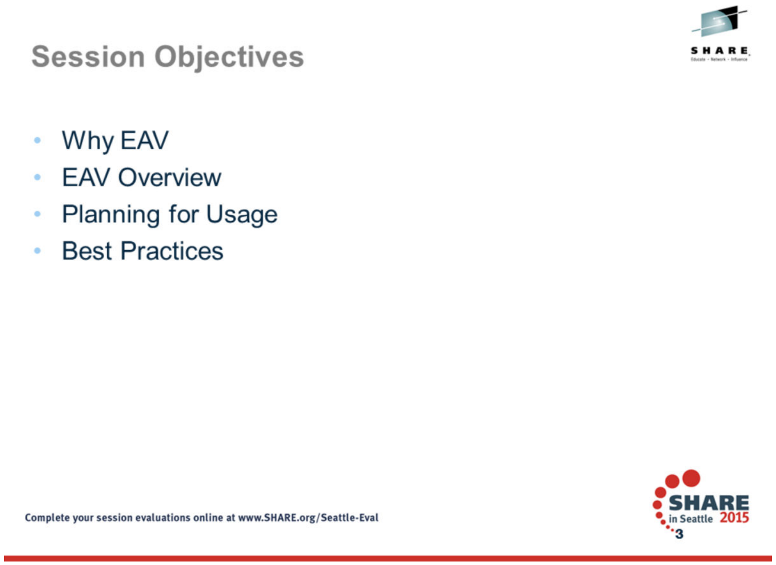
Cecilia Carranza Lewis, IBM carranc@us.ibm.com

March 2, 2015
Session 17111

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides **education**, professional **networking** and industry **influence.**

SHARE in Seattle 2015



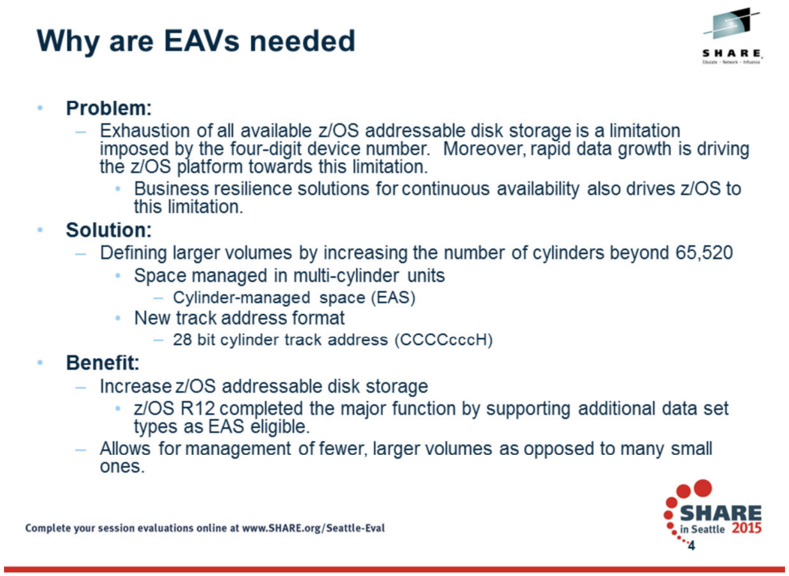
## Session Objectives

- Why EAV
- EAV Overview
- Planning for Usage
- Best Practices

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

3



## Why are EAVs needed

- **Problem:**
  - Exhaustion of all available z/OS addressable disk storage is a limitation imposed by the four-digit device number. Moreover, rapid data growth is driving the z/OS platform towards this limitation.
    - Business resilience solutions for continuous availability also drives z/OS to this limitation.
- **Solution:**
  - Defining larger volumes by increasing the number of cylinders beyond 65,520
    - Space managed in multi-cylinder units
      - Cylinder-managed space (EAS)
    - New track address format
      - 28 bit cylinder track address (CCCCcccH)
- **Benefit:**
  - Increase z/OS addressable disk storage
    - z/OS R12 completed the major function by supporting additional data set types as EAS eligible.
  - Allows for management of fewer, larger volumes as opposed to many small ones.

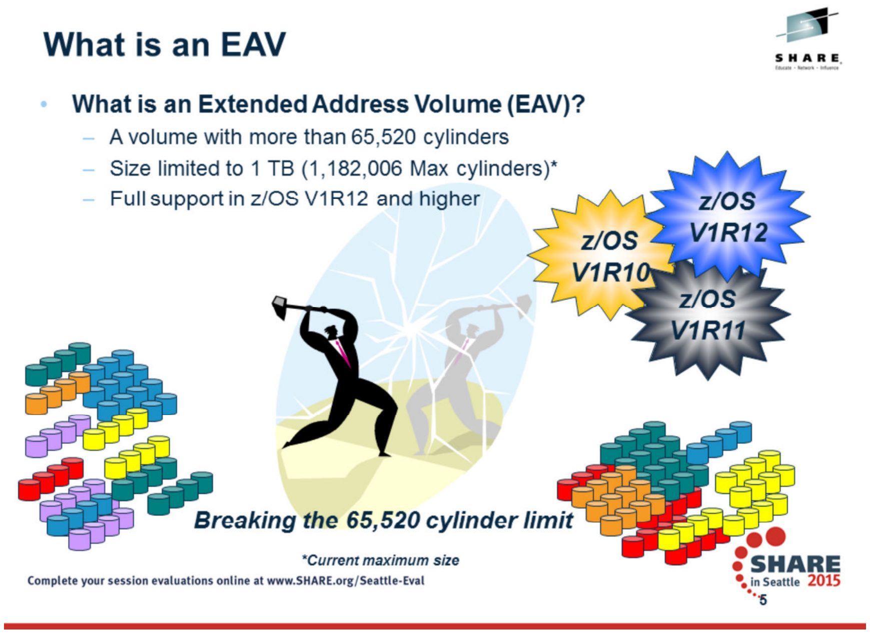
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

4

z/OS addressable disk storage is significantly increased with EAV and provides constraint relief for environments that are approaching the 4-digit device number limit in z/OS. EAV volumes are 3390 Model A volumes with an increased number of cylinders above what was previously supported as the maximum. This was 65,520 cylinders. This is accomplished with a new track address format, referred to as 28-bit cylinder track address, and a new way to manage this additional space as multi-cylinder units.

This enhanced support is intended to relieve storage constraints and help simplify storage management by providing the ability to manage fewer, larger volumes as opposed to many small volumes.


z/OS
V1R10
z/OS
V1R12
z/OS
V1R11

## What is an EAV

- **What is an Extended Address Volume (EAV)?**
  - A volume with more than 65,520 cylinders
  - Size limited to 1 TB (1,182,006 Max cylinders)*
  - Full support in z/OS V1R12 and higher

*Breaking the 65,520 cylinder limit*

**Current maximum size*

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

5

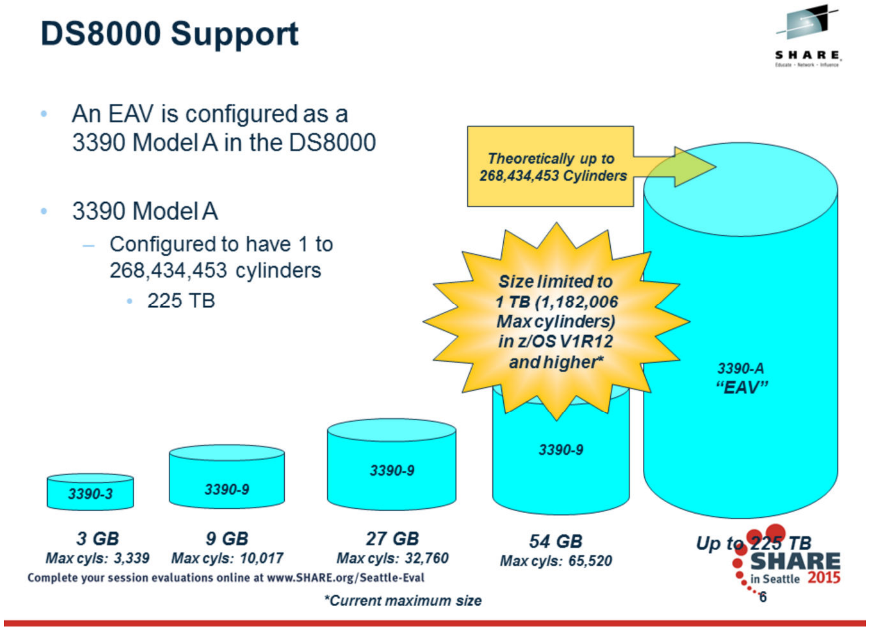
With z/OS V1R10 and higher releases, z/OS has added support for DASD volumes having more than 65,520 cylinders.

The capacity of storage volumes has increased steadily over the years within the constraints of the 3390 storage device architecture. To expand the capacity of DASD storage volumes beyond 65,520 cylinders, z/OS had to extend the track address format. Hence the name **Extended Address Volume** for a volume of more than 65,520 cylinders.

The name in z/OS publications for these larger volumes is Extended Address Volumes, or EAVs.

The new DASD volume maximum for V1R10 and V1R11 is 262,668 cylinders.

With z/OS V1R12 and higher the maximum volume size is 1 TB or 1,182,006 cylinders



## DS8000 Support

- An EAV is configured as a 3390 Model A in the DS8000
- 3390 Model A
  - Configured to have 1 to 268,434,453 cylinders
    - 225 TB

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

*Current maximum size

6

The Extended Address Volume (EAV) is the next step in providing larger volumes for z/OS. Z/OS provided this support in z/OS Version 1 release 10 of the operating system. Over the years we have grown volumes by increasing the number of cylinders and thus GB capacity. However, the existing track addressing architecture has limited our growth to relatively small GB capacity volumes which has put pressure on the 4-digit device number limit. (The largest available volume is one with 65,520 cylinders or approx 54 GB). z/OS V1R10 GA'd on 9/26/2008.

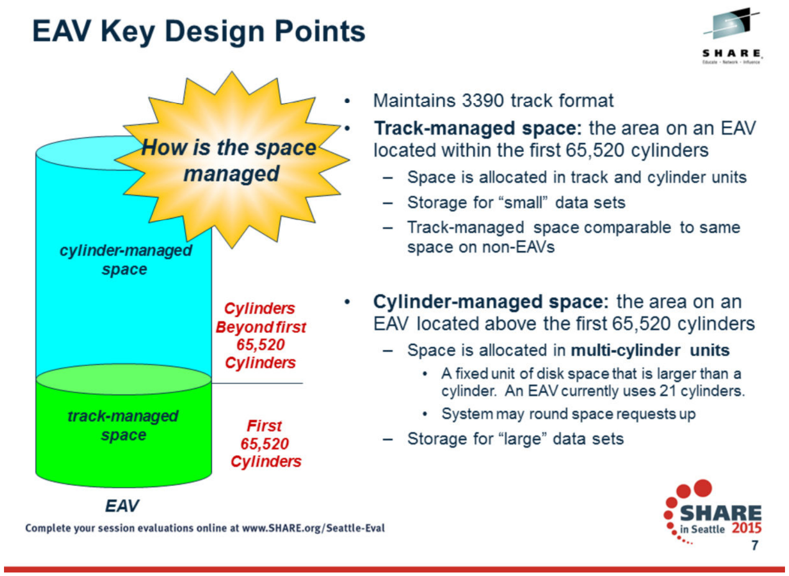
With EAV, we are implementing an architecture that will provide capacities of 100's of terabytes for a single volume. However, the first release was limited to a volume with 223 GB or 262,668 cylinders. This was done to help reduce our initial EAV testing requirements in order to get a solution out to the field. This will also allow us to understand and address possible performance concerns before very large volumes hit the field.

With follow on releases and starting with z/OS V1R12 and higher the new maximum is 1 TB volumes or 1,182,006 cylinders

An EAV is defined to be a volume with more than 65,520 cylinders. A volume of this size has to be configured in the DS8000 as a 3390 Model A. However a 3390 Model A is not always an EAV. A 3390 Model A is any device configured in the DS8000 to have from 1- 268,434,453 cylinders. This is an architectural maximum.

**The way an EAV is managed by the system allows it to be a general purpose volume. However, it will work especially well for applications with large files. PAV and HyperPAV technologies help in both these regards by allowing I/O rates to scale as a volume gets larger.**

**With the 3390 Model A, the model A refers to the model configured in the DS8000. It has no association with the 3390A notation in HCD that indicates a PAV-alias UCB in the zOS operating system. Model 'A' was chosen so that it did not imply a particular device size that previous models (3390-3, 3390-9) did. Some people adopted unofficial model numbers for larger devices. For example the 54 GB 3390 Model 9 is some times referred to as 3390 Model 54 even though that model does not exist.**



## EAV Key Design Points

- Maintains 3390 track format
- **Track-managed space:** the area on an EAV located within the first 65,520 cylinders
  - Space is allocated in track and cylinder units
  - Storage for "small" data sets
  - Track-managed space comparable to same space on non-EAVs
- **Cylinder-managed space:** the area on an EAV located above the first 65,520 cylinders
  - Space is allocated in **multi-cylinder units**
    - A fixed unit of disk space that is larger than a cylinder. An EAV currently uses 21 cylinders.
    - System may round space requests up
  - Storage for "large" data sets

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 7

One of the more important design points is that IBM will maintain its promise to customers that the 3390 track format and image size and tracks per cylinders will remain the same as previous 3390 models. An application using data sets on an EAV will be comparable to how it runs today.

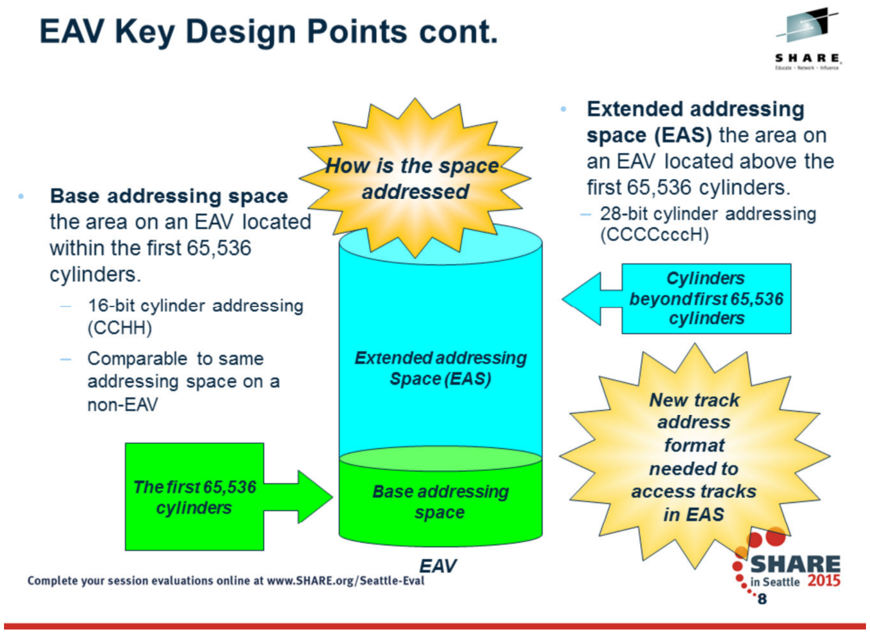
The track-managed space is the space on a volume that is managed in track and cylinder increments. All volumes today have track-managed space. Track-managed space ends at cylinder address 65,519. Each data set occupies an integral multiple of tracks. The track-managed space allows existing programs, allocations and physical migration products to continue to work.

The cylinder-managed space is the space on the volume that is managed only in multicylinder units. Cylinder-managed space begins at cylinder address 65,520. Each data set occupies an integral multiple of multicylinder units. Space requests targeted for the cylinder-managed space will be rounded up to the next multicylinder unit. The cylinder-managed space exists only on EAV volumes but IBM recommends not assuming that only EAVs have multicylinder units. A multicylinder unit (MCU) is a fixed unit of disk space that is larger than a cylinder. Currently, on an EAV, a multicylinder unit is 21 cylinders and the number of the first cylinder in each multicylinder unit is a multiple of 21.

We get many questions on why the multicylinder unit has 21 cylinders. 21 Cylinders value for the MCU is derived from being the smallest unit that can map out the largest possible EAV and stay within the index architecture (with a block size of 8192 bytes). It was a value that divides evenly into the 1 GB storage segments of a DS8000. These 1 GB segments are equivalent to 1113 cylinders. A data set allocated in cylinder-managed space may have its requested space quantity rounded up to the next MCU.

The track-managed space allows existing programs and physical migration products to continue to work. You can perform physical copies from a non-EAV to an EAV and have those data sets accessible.

Data sets allocated in cylinder-managed space will be described with new type of  data set control blocks (DSCB) in the VTOC.  Tracks allocated in this space will also be addressed using the new track address.  Existing programs that are not changed will not recognize these new DSCBs and therefore will be protected from seeing how the tracks in cylinder-managed space are addressed.



## EAV Key Design Points cont.

- **Base addressing space** the area on an EAV located within the first 65,536 cylinders.
  - 16-bit cylinder addressing (CCHH)
  - Comparable to same addressing space on a non-EAV

- **Extended addressing space (EAS)** the area on an EAV located above the first 65,536 cylinders.
  - 28-bit cylinder addressing (CCCCcccH)

How is the space addressed

Extended addressing Space (EAS)

Base addressing space

EAV

The first 65,536 cylinders

Cylinders beyond first 65,536 cylinders

New track address format needed to access tracks in EAS

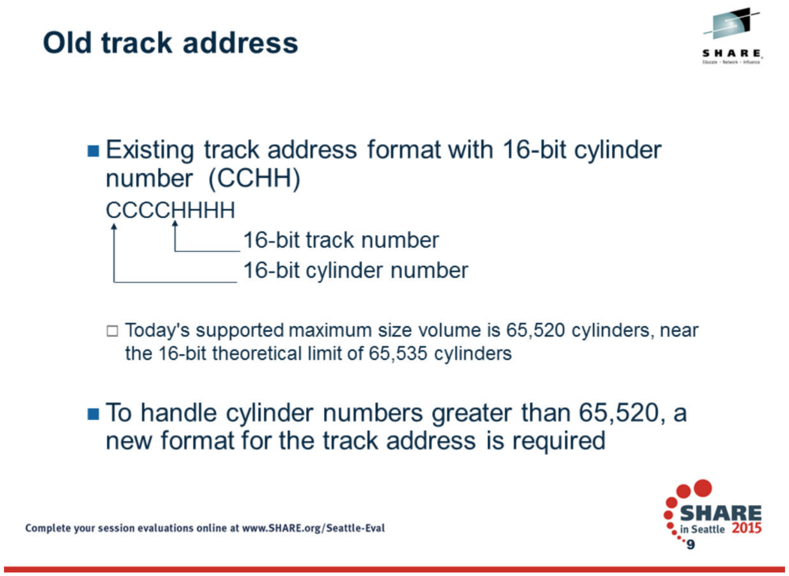
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

8

The EAS of an EAV is an abbreviation for extended addressing space (EAS) of an Extended Address Volume (EAV).  It refers to those cylinders of a volume beyond the first 65,535.

In this diagram of an EAV, cylinders up to but not including cylinder 65,536 are in the base addressing space of the EAV.  Cylinders starting with cylinder 65,536 are in the EAS of the EAV.



## Old track address

- Existing track address format with 16-bit cylinder number (CCHH)

```
CCCCHHHH
↑   ↑
|   └──── 16-bit track number
└──────── 16-bit cylinder number
```

  - □ Today's supported maximum size volume is 65,520 cylinders, near the 16-bit theoretical limit of 65,535 cylinders
- To handle cylinder numbers greater than 65,520, a new format for the track address is required

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

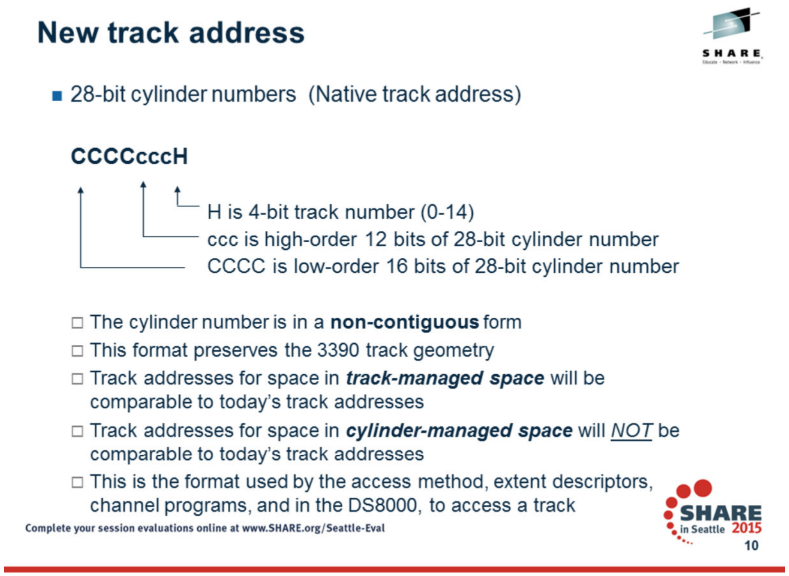
9

Today we generally refer to a track address using the CCHH notation.
However I show today's track address using hexadecimal nibbles. Thus the CCCCHHHH notation.

This track address is a 32 bit number that addresses each track within a volume. Each cylinder and track number uses a 16-bit number. However, for the track number only the low order 4-bits are used, where the high order 12-bits of the track number is not.

The maximum supported volume size today is 65,520 cylinders, which is near the 16-bit theoretical limit of 65535 cylinders.

Thus to handle cylinder numbers greater than 65,520, a new format for the track address is required….



## New track address

- 28-bit cylinder numbers (Native track address)

**CCCCcccH**

H is 4-bit track number (0-14)
ccc is high-order 12 bits of 28-bit cylinder number
CCCC is low-order 16 bits of 28-bit cylinder number

□ The cylinder number is in a **non-contiguous** form
□ This format preserves the 3390 track geometry
□ Track addresses for space in ***track-managed space*** will be comparable to today's track addresses
□ Track addresses for space in ***cylinder-managed space*** will *NOT* be comparable to today's track addresses
□ This is the format used by the access method, extent descriptors, channel programs, and in the DS8000, to access a track

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

For compatibility with older programs, the ccc portion is hexadecimal 000 for tracks in the base addressing space. This track address method is referred to as a 28-bit cylinder number.

**A few key points**

**The cylinder number is in a non-contiguous form. A program or human reading this track address needs to re-arrange the hex digits in the program or in their minds.**
**This format preserves the 3390 track geometry.**
**Track addresses for space in track-managed space will be comparable to today's track addresses.**
**Track addresses for space in cylinder-managed space will NOT be comparable to today's track addresses.**

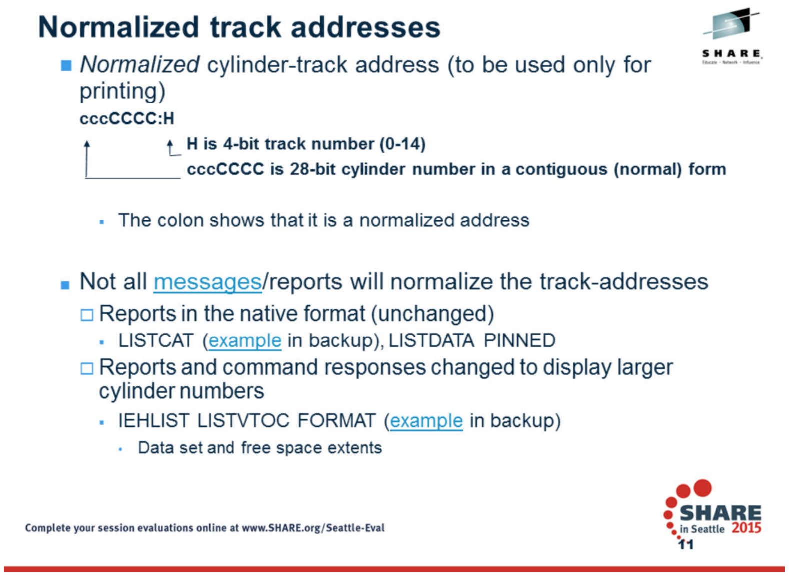
Reference Material/Terminology:
"Absolute track addresses" generally are represented in the documentation as cchh or bbcchh and the term generally is shortened to track addresses. Code that references these often must be changed.

Track addresses in existing channel programs and extent descriptors in DSCBs and elsewhere are in the form of CCHH, where CC is the 16-bit cylinder number and HH is the 16-bit track number in that cylinder. If the volume is an EAV, the cylinder number in these four CCHH bytes will be 28 bits and the track number will be four bits. For compatibility reasons, the 32 bits in each track address on an EAV will be in this format: CCCCcccH

**The 12 high order bits of the cylinder number are in the high order 12 bits of the two old HH bytes. This format might be written as CCCCcccH. You can compare two of these track addresses for equality but you cannot reliably use a simple comparison for greater than or less than. Any arithmetic must take this special format into consideration.**

An "absolute block address" is of the form cchhr or mbbcchhr and includes an absolute track address. Often they are written in capitals in the documentation as CCHHR or MBBCCHHR.

"Relative block addresses" are written as TTR or TTTR. The TT or TTT bytes count tracks relative to something. If they are relative to the beginning of a data set and inside the data set, they are unaffected by this project. If they are relative to the beginning of the volume, they might be affected by EAV.



## Normalized track addresses

- *Normalized* cylinder-track address (to be used only for printing)

  **cccCCCC:H**

  **H is 4-bit track number (0-14)**

  **cccCCCC is 28-bit cylinder number in a contiguous (normal) form**

  - The colon shows that it is a normalized address
- Not all messages/reports will normalize the track-addresses
  - □ Reports in the native format (unchanged)
    - LISTCAT (example in backup), LISTDATA PINNED
  - □ Reports and command responses changed to display larger cylinder numbers
    - IEHLIST LISTVTOC FORMAT (example in backup)
      - Data set and free space extents

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

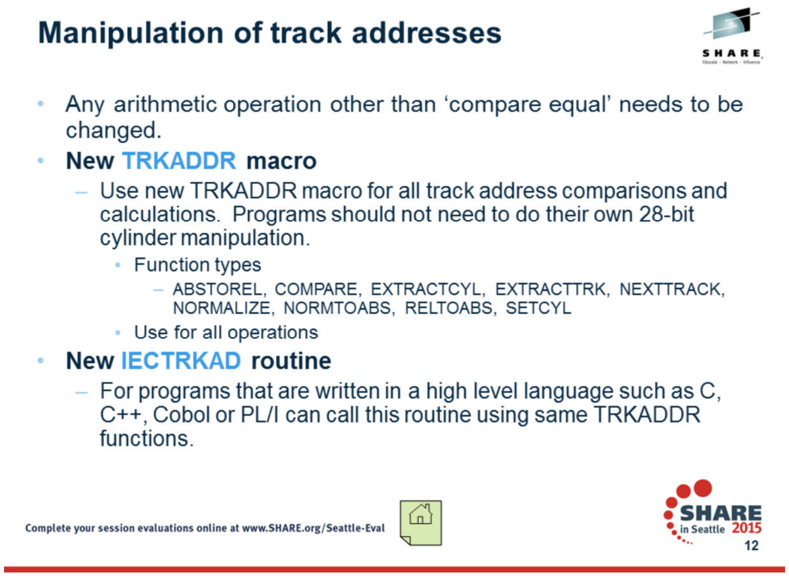
11

To assist in the understanding of 28-bit cylinder numbers a normalized cylinder-track address may be used for printing. This is where the bits are rearranged to a more readable format and a linear 28-bit cylinder number. The presence of a colon before the last hex nibble identifies the track address as being normalized.

Note that not all messages/reports will normalize the output. For example, IDCAMS LISTCAT reports in the native format. Others do the same.

Some reports have shifted their output in order to display the larger number of cylinders. IEHLIST LISTVTOC is an example of one.

Messages: ICKDSF, DSS (normalize)



## Manipulation of track addresses

- Any arithmetic operation other than 'compare equal' needs to be changed.
- **New TRKADDR macro**
  - Use new TRKADDR macro for all track address comparisons and calculations. Programs should not need to do their own 28-bit cylinder manipulation.
    - Function types
      - ABSTOREL, COMPARE, EXTRACTCYL, EXTRACTTRK, NEXTTRACK, NORMALIZE, NORMTOABS, RELTOABS, SETCYL
    - Use for all operations
- **New IECTRKAD routine**
  - For programs that are written in a high level language such as C, C++, Cobol or PL/I can call this routine using same TRKADDR functions.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

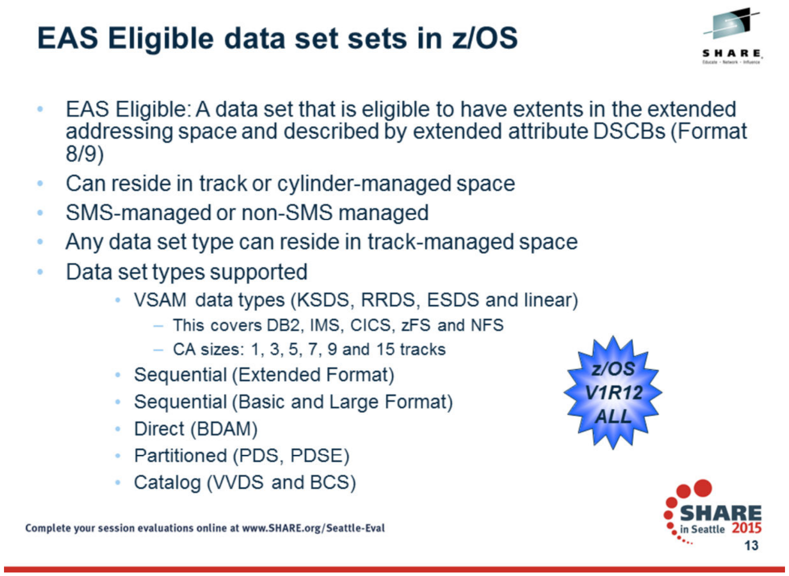
12

**The New *TRKADDR* macro**
IBM recommends using the new TRKADDR macro for all track address comparisons and calculations. Programs should not need to do 28-bit manipulation themselves. Use this for all track address computations even those not directly affected by this support for VSAM data sets using EAS. This prepares one for additional support. This macro supports these functions:
•ABSTOREL calculates the relative track number on the volume from the passed 'cchh' track address.
•COMPARE compares two track addresses in the new CCCCcccH format.
•EXTRACTCYL extracts the 28-bit cylinder number from the passed 'cchh' track address.
•EXTRACTTRK extracts the 4-bit track number from the passed 'cchh' track address.
•NEXTTRACK increments the track address by one track and increments the cylinder number if necessary from the passed 'cchh' track address.
•NORMALIZE reverses the 16-bit and 12-bit portions of the cylinder number from the passed 'cchh' track address. The CCCCcccH becomes cccCCCCH. This could be used to subsequently perform unsigned comparisons of track addresses.
•NORMTOABS reverses the 12-bit and 16-bit portions of the cylinder from the track address. The cccCCCCH becomes CCCCcccH. Use this to convert a normalized track address to an absolute 28-bit cylinder address.
•RELTOABS converts a relative track number on the volume (counting across cylinder boundaries) to a 28-bit cylinder address.
•SETCYL converts a cylinder number to a 28-bit cylinder address and sets the track portion to zero.

For details, see the documentation. This macro is available only in z/OS 1.10 but the expansion will run equally well on downlevel systems if the high order 12 bits of the track number are zero. Does not support running on pre-z/Architecture machines.

Conversions of relative track addresses to and from absolute format (MBBCCHHR) are supported automatically in the existing routines pointed to by CVTPRLTV and CVTPCNVT.

**New IECTRKAD routine**
Programs written in a high level language such as C, C++, Cobol, or PL/I can call a new routine named IECTRKAD that performs the same functions as the TRKADDR macro as described above. This routine is available only on z/OS 1.10, but programs linked with it can run on down-level releases.



## EAS Eligible data set sets in z/OS

- EAS Eligible: A data set that is eligible to have extents in the extended addressing space and described by extended attribute DSCBs (Format 8/9)
- Can reside in track or cylinder-managed space
- SMS-managed or non-SMS managed
- Any data set type can reside in track-managed space
- Data set types supported
  - VSAM data types (KSDS, RRDS, ESDS and linear)
    - This covers DB2, IMS, CICS, zFS and NFS
    - CA sizes: 1, 3, 5, 7, 9 and 15 tracks
  - Sequential (Extended Format)
  - Sequential (Basic and Large Format)
  - Direct (BDAM)
  - Partitioned (PDS, PDSE)
  - Catalog (VVDS and BCS)

z/OS V1R12 ALL

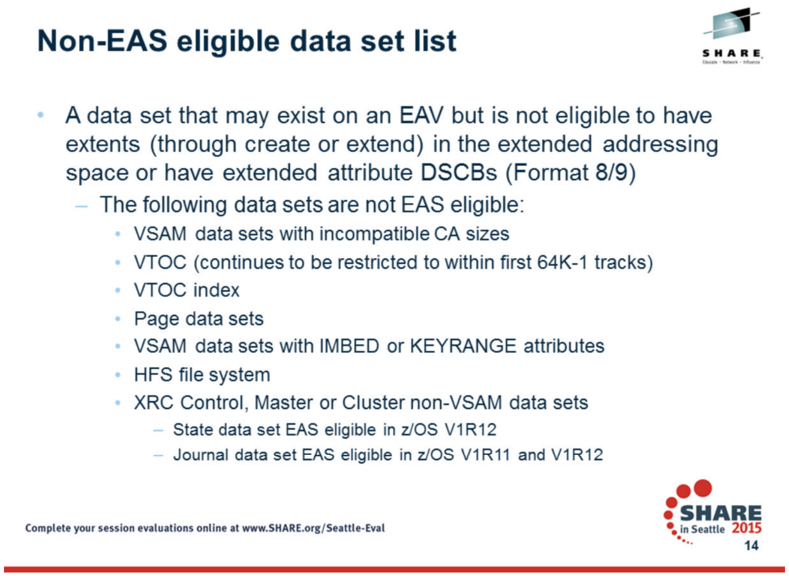
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

13

EAS eligible data sets are defined to be those that can be allocated in the extended addressing space. They can reside in track or cylinder-managed space and can be SMS-managed or non-SMS managed. This is sometimes referred to as cylinder-managed space.

In addition the following are EAS eligible:
•zFS data sets (which are VSAM)
•Data base (DB2, IMS) use of VSAM.
•VSAM data sets inherited from prior physical migrations or copies
•VSAM files that have compatible Control Area sizes. A compatible CA is one that divides evenly (a divisor) of the multi-cylinder value. This is 21 cylinders or 315 tracks.



## Non-EAS eligible data set list

- A data set that may exist on an EAV but is not eligible to have extents (through create or extend) in the extended addressing space or have extended attribute DSCBs (Format 8/9)
  - The following data sets are not EAS eligible:
    - VSAM data sets with incompatible CA sizes
    - VTOC (continues to be restricted to within first 64K-1 tracks)
    - VTOC index
    - Page data sets
    - VSAM data sets with IMBED or KEYRANGE attributes
    - HFS file system
    - XRC Control, Master or Cluster non-VSAM data sets
      - State data set EAS eligible in z/OS V1R12
      - Journal data set EAS eligible in z/OS V1R11 and V1R12

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

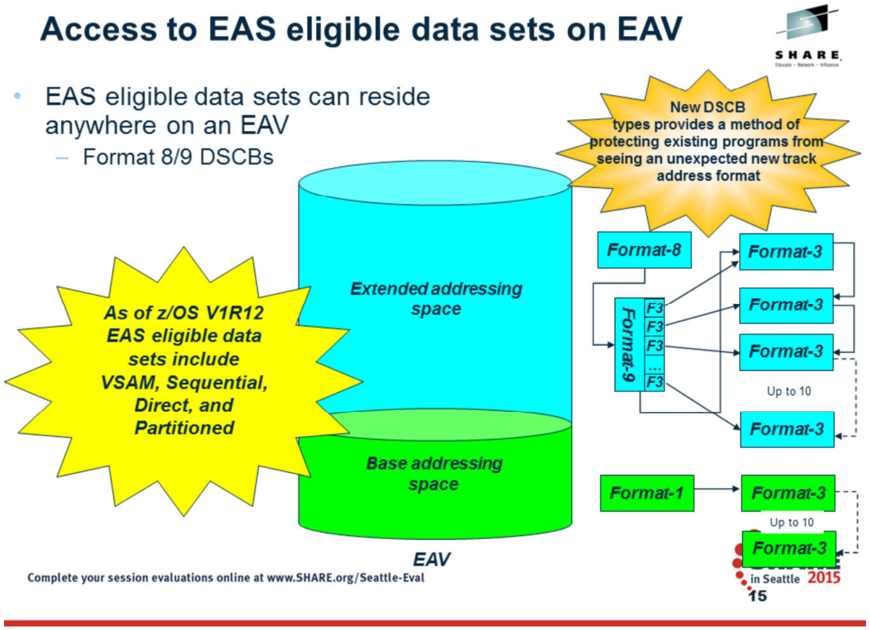
14

Please note that all data set types, even those listed here, can be allocated in the track-managed space on a device with cylinder-managed space (i.e. an EAV). Eligible EAS data sets can be created and extended anywhere on an EAV. Data sets that are not eligible for EAS processing can only be created or extended in the track-managed portions of the volume.

**EAS Eligible data set is a** data set that is eligible to have extents in the extended addressing space and described by extended attribute DSCBs (Format 8/9)

EXCP processing in XRC for the Control, Master, or Clustered non-VSAM data sets is not changed to support extended attribute DSCBs and 28-bit cylinder numbers. Therefore, attempts to access these data sets when allocated with extended attribute DSCBs (Format 8 and 9 DSCBs) will be prevented. The following table describes how the various XRC system data sets will be supported by the system.

XRC control, master and cluster data sets access will be prevented if allocated with extended attribute DSCBs.



## Access to EAS eligible data sets on EAV

- EAS eligible data sets can reside anywhere on an EAV
  - Format 8/9 DSCBs

As of z/OS V1R12 EAS eligible data sets include VSAM, Sequential, Direct, and Partitioned

New DSCB types provides a method of protecting existing programs from seeing an unexpected new track address format

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

15

Before V1R10, MVS data sets were described with a format-1 DSCB which optionally pointed to a format-3 DSCB. To allow a data set to reside in the EAS of an EAV, these types of data sets are allocated with format-8 DSCBs instead of format-1 DSCBs. The format-8 DSCB is virtually identical to a format-1 DSCB except in three respects: it has a control block ID of F8 instead of F1, it points to a format-9 DSCB instead of a format-3 DSCB and the cylinder numbers in the extent descriptions can include large cylinder numbers..

The format-9 DSCB is new as of V1R10 and has all the information z/OS needs to record the attributes of a data set in the EAS of an EAV. The format-9 DSCB can point to one or more format-3 DSCBs. Thus we see the format-8 DSCB exists strictly as a heads-up that a format-9 DSCB with all the EAS information has been inserted between the format-1 (aka format-8) DSCB and the format-3 DSCB chain.

The logical DSCB chain today for a data set today is: A format-1 and up to 10 possible format-3 DSCBs.
The logical DSCB chain for an EAS-eligible data set on an EAV will be: A format-8, one or more format-9s, and up to 10 possible format-3 DSCBs

In both cases the chain pointer in each DSCB will point to the next if one exists.

The format-9 DSCB is a place for additional attribute information and contains direct pointers to each possible format 3 DSCB. With this new services in the system (OBTAIN, CVAFDIR) can read the entire logical DSCB chain for a data set in one call. No more loops to read DSCBs until the chain pointer is zero . . . But it will still work.

DSCBs (data set control blocks) are VTOC (volume table of contents) entries that describe data set attributes and allocated extent information. This extent information describes allocated space using beginning and ending track addresses. These are called extent descriptors. These extent descriptors may contain 28-bit cylinder number for their track addresses. DSCBs also contain metadata in the format 1 DSCB that are the characteristics or attributes of the allocated data set. There is no more space available in the format 1 DSCB to add additional attributes.

Reference information:

Format 4 DSCB –

Number of cylinders on volume
Existing two-byte field DS4DSCYL in the F4 DSCB will contain the value of x'FFFE' (65534). This Identifies the volume as an EAV.
DS4EAV defined as constant value of X'FFFE'
New four-byte field DS4DCYL will contain the number of cylinders on the volume.

New allocation unit for cylinders above 65,520

- New two-byte field DS4LCYL will contain a code value of x'0010' to indicate that the cylinder-managed space after the first 65,520 cylinders must be allocated in units that are larger than one cylinder.
  - This value represents 65,520 cylinders divided by 4095. For a non-EAV this will be zero.
- New field DS4MCU ("minimum allocation unit") will contain the number of cylinders that each extent in the cylinder-managed area must be a multiple of.
  - For an EAV this will be 21. For a non-EAV this will be zero. It is valid only when the value in DS4DSCYL is DS4EAV.

Format 8 DSCB –

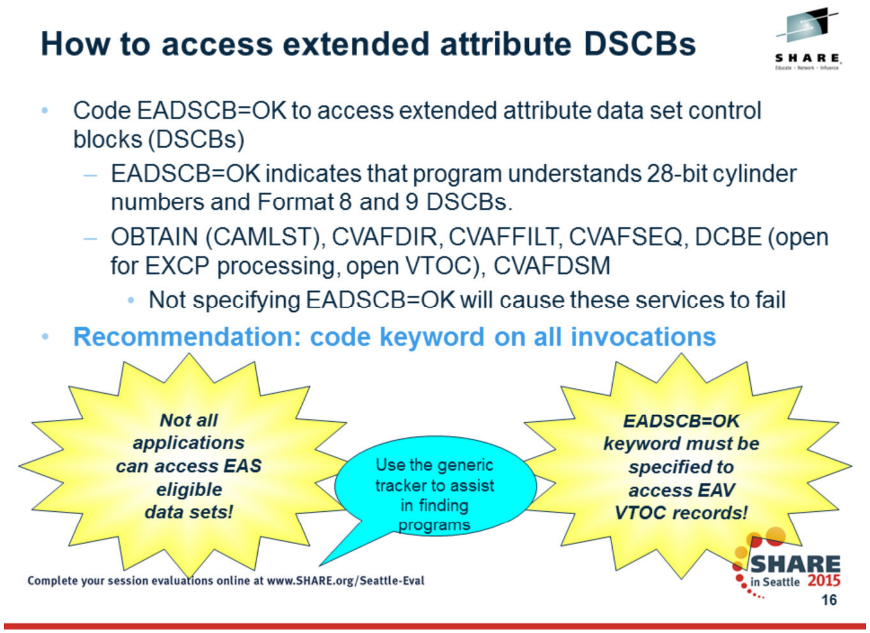
Identical to F1 DSCB with following exceptions

- The format identifier (DS1FMTID) will be x'F8' instead of x'F1' . New symbols defined:
  - DS1IDC constant value of X'F1' in DS1FMTID
  - DS8IDC constant value of X'F8' in DS1FMTID
- Track addresses in the extent descriptors starting in DS1EXT1 will use new track address format (that is, may contain 28-bit cylinder numbers)
- The 'next DSCB' address (DS1PTRDS) will always point to a F9 DSCB (new type of DSCB) instead of a possible first F3 DSCB.

Format 9 DSCB –
F9 DSCB will exist only for EAS eligible data sets (VSAM in first release)
Will contain the following

- The format identifier will be x'F9'
- A subtype field
  - In the first EAV release, the subtype will be 1.
  - In future releases, additional subtypes may be added.
- Track addresses which point directly to up to ten F3 DSCBs
  - All the format 3 DSCBs can be read with one channel program.
- A 20-byte field DS9ATRV1 that IBM is reserving for vendors. IBM will not specify or monitor its content.
- A 'next DSCB' address will point to a possible F3 DSCB. The F3 DSCBs will continue to be chained.
  - In future releases, additional F9 DSCBs may be chained between the subtype 1 and any F3 DSCBs

All are mapped in macro IECSDSL1.



## How to access extended attribute DSCBs

- Code EADSCB=OK to access extended attribute data set control blocks (DSCBs)
  - EADSCB=OK indicates that program understands 28-bit cylinder numbers and Format 8 and 9 DSCBs.
  - OBTAIN (CAMLST), CVAFDIR, CVAFFILT, CVAFSEQ, DCBE (open for EXCP processing, open VTOC), CVAFDSM
    - Not specifying EADSCB=OK will cause these services to fail
- **Recommendation: code keyword on all invocations**

*Not all applications can access EAS eligible data sets!*

Use the generic tracker to assist in finding programs

*EADSCB=OK keyword must be specified to access EAV VTOC records!*

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 16

Not all applications can access EAS eligible data sets! In general, applications that have awareness of the data set control blocks (DSCBs) that describe a data set cannot access EAS eligible data sets if such applications were coded at pre-V1R10 levels. V1R10 introduced new code options that must be specified explicitly to enable EAV support.

For more information about coding applications that support EAVs and EAS eligible data sets, see *DFSMSdfp Advanced Services*. For more information about installing and using EAVs, see *DFSMS Using the New Functions*.

Reference Material:

In order to access extended attribute DSCBs, the system will require the specification of a new permission keyword on system services that read DSCBs. By specifying EADSCB=OK, the invoking program is indicating to the system service that it understands extended attribute DSCBs and the 28-bit cylinder numbers that could be present in the data set's extent descriptors.

The EADSCB=OK keyword has been added to the following services:
•OBTAIN (CAMLST macro)
•CVAFDIR
•CVAFFILT
•CVAFDSM
•CVAFSEQ
•OPEN (DCBE macro) – opening VTOC or VSAM data set with EXCP access.

Not specifying EADSCB=OK will cause these services to fail if issued to a data set that supports extended attribute DSCBs or a volume that supports cylinder-managed space (CVAFDSM and OPEN). Code this keyword when your application supports EADSCB. IBM recommends specifying on all invocations of each service regardless if the application runs on pre 1.10 systems or accesses volumes that do not support extended attribute DSCBs. Macros on earlier releases do not recognize EADSCB but you can assemble with EADSCB=OK on 1.10 and run on downlevel releases, where EADSCB=OK will have no effect.

Reference Material:

Syntax, meanings and error codes for each service:

OBTAIN:

EADSCB=OK Specifies the support level for extended attribute DSCBs.
For OBTAIN SEARCH and SEEK requests, code this operand as shown when your program supports DSCBs that describe data sets with format-8 and format-9 DSCBs. The extent descriptors in DSCBs for a data set described with these formats may have track addresses that contain cylinder addresses 65,520 or larger.
Please note that the specification of this keyword is accepted for data sets described by all DSCB types, including format-1 DSCBs, regardless of the volume size where the data set resides. Your program can also run on an older level of the system that does not support this keyword. In these cases the EADSCB=OK is not applicable.
For a SEARCH request that does not have EADSCB=OK coded, OBTAIN will set return code 24 if the OBTAIN request ends up being for a format-8 DSCB.
For a SEEK request that does not have EADSCB=OK coded, OBTAIN will set return code 24 if the target of the OBTAIN request ends up being for Format 8 or 9 DSCB.
OBTAIN will not check format 3 DSCB extent ranges for track addresses that contain cylinder addreses 65,520 or larger.

CVAFDIR:

EADSCB=OK Specifies the support level for extended attribute DSCBs.
This specification indicates that the calling program supports DSCBs that describe data sets on volumes configured with more than 65,520 cylinders where returned DSCBs may contain Format 3/8/9 DSCBs which could describe 28-bit cylinder addresses.
For search calls where the data set name is passed (CVAFDIR ACCESS=READ, BFLEARG=0), a CVAFDIR request issued to an EAV volume will be failed if this new, EADSCB=OK, indicator is not set and the DSCB associated with this data set name is described by a Format 8 DSCB.
For seek calls where the record address is passed (CVAFDIR ACCESS=READ,BFLEARG=cchhr), a CVAFDIR request issued to an EAV volume will be failed if the new, EADSCB=OK, indicator is not set and the DSCB associated with this address is a Format 8 or 9 DSCB.
For seek calls where the record address is passed (CVAFDIR ACCESS=READ,BFLEARG=cchhr), and MULTIPLEDSCBS=NO is specified or defaulted to NO, a CVAFDIR request issued to an EAV volume will be failed if the new, EADSCB=OK, indicator is not set and the DSCB associated with this address is a Format 3 DSCB that contains track addresses above 65,520 cylinders.

The failing error code for this case will be reflected as follows:
•CVAF status code (CVSTAT) set to STAT082
•Return code 4

EADSCB=OK will resolve to a new indicator in the CVPL, CV4EADOK.

For all other calls the EADSCB=OK keyword is ignored.

CVAFFILT:

EADSCB=OK Specify the support level for extended attribute DSCBs.
This specification indicates that the calling program supports DSCBs that describe data sets on volumes configured with more than 65,520 cylinders where returned DSCBs may contain Format 8/9 DSCBs and may contain 28-bit cylinder addresses.

For fully qualified data set names in the filter criteria list, a CVAFFILT request issued to an EAV volume will be failed if this new EADSCB=OK indicator is not set and the DSCB associated with the fully qualified data set name is a format 8 DSCB.

For partially qualified data set names in the filter criteria list, a CVAFFILT request issued on an EAV volume will be failed if this new EADSCB=OK indicator is not set and a DSCB associated with a data set that matches the filter criteria list is a Format 8 DSCB.

The failing error code for this case will be reflected as follows:
•Data set name status in the FCL (FCLDSNST) will be set to a status value of (x'06'). This status code indicates that a data set name is described by a Format 8 DSCB and the caller did not specify support for an EAV with the EADSCB=OK keyword.
•Set the no resume CVAF status code (CVSTAT) of STAT072
•Return code 4

EADSCB=OK will resolve to a new indicator in the CVPL, CV4EADOK.

CVAFSEQ:

EADSCB=OK Specifies the support level for extended attribute DSCBs.
This specification indicates that the calling program supports DSCBs that describe data sets on volumes configured with more than 65,520 cylinders. For calls that initiate physical sequential access (DSN=0 or omitted), a CVAFSEQ request issued to an EAV volume will be failed if this new, EADSCB=OK, indicator is not set.

For calls that initiate index order (DSN=address) where the BUFLIST=address keyword is specified, a CVAFSEQ request issued to an EAV volume will be failed if this new EADSCB=OK indicator is not set and the DSCB associated with this address is a Format 8 DSCB.

The failing error code for this case will be reflected as follows:
•CVAF status code (CVSTAT) set to STAT082
•Return code 4

EADSCB=OK will resolve to a new indicator in the CVPL, CV4EADOK.

All other calls to CVAFSEQ are allowed and EADSCB=OK will be ignored. That is CVAFSEQ calls with DSNONLY=YES and ARG=address specified.

DCBE Macro for OPEN:

EADSCB=OK Specifies the support level for extended attribute DSCBs.
This option allows you to specify whether your application program supports the following:

•Opening of the VTOC that describes a volume with more than 65,520 cylinders. The VTOC may contain Format 8 and 9 DSCBs. If you do not code this option, the OPEN function will issue ABEND 113-48 and message IEC142I. Code this option when your application program supports Format 8 and 9 DSCBs.

•Opening of a VSAM data set with a DCB specifying MACRF=E (EXCP) when the data set has Format 8 and 9 DSCBs. Track addresses in those DSCBs may contain cylinder addresses above 65,520. If you do not code this option, the OPEN function will issue ABEND 113-44 and message IEC142I. Code this option when your application program supports Format 8 and 9 DSCBs and such track addresses.

This resolves to the DCBEEADSCBOK (bit 2 of DCBEFLG3) flag in the DCBE.

New message for OPEN - IEC987I
IEC987I jjj,sss,ddn,dev,volser,dsn WARNING – DCB OPEN for a VSAM Data Set or VTOC but EADSCB=OK not specified on DCBE macro
Explanation: A DCB OPEN was issued for an EAS eligible (VSAM) data set or a VTOC on a volume volser but EADSCB=OK on the DCBE macro was not specified. This message will be issued only if the volume is not EAV. If this was an EAV volume, an ABEND would have been issued instead of this message
System Action: OPEN processing of the data set is allowed to continue
Operator Response: None
Systems Programmer Response: None

CVAFDSM:

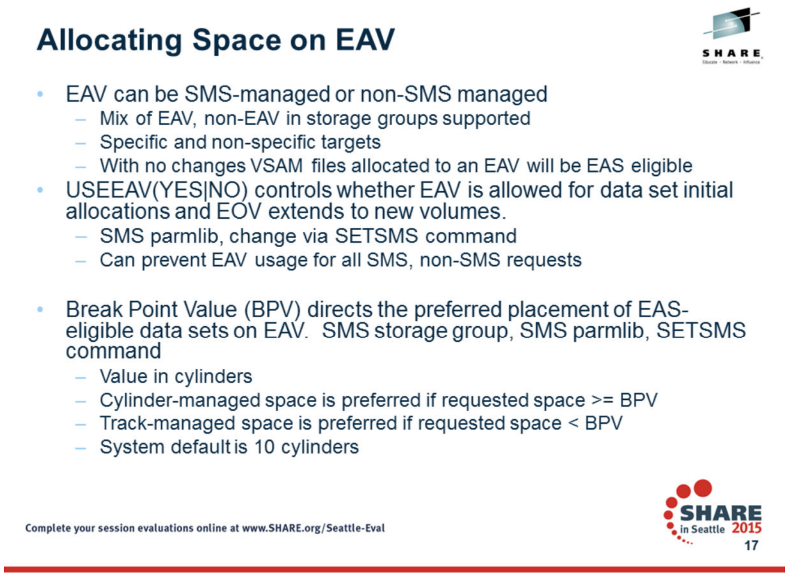
EADSCB=OK Specifies the support level for extended attribute DSCBs.

This specification indicates that the calling program supports relative track addresses (RTAs) that could contain tracks from a volume configured with more than 65,520 cylinders. For calls that request unallocated space (ACCESS=MAPDATA, MAP=VOLUME, RTA4BYTE=YES and EXTENTS=address), a CVAFDSM request issued to an

EAV volume will be failed if this new EADSCB=OK indicator is not set.

The failing error code for this case will be reflected as follows:
- CVAF status code (CVSTAT) set to STAT082.
- Return code 4

EADSCB=OK will resolve to a new indicator in the CVPL, CV4EADOK.
All other calls to CVAFDSM are allowed and EADSCB=OK will be ignored.



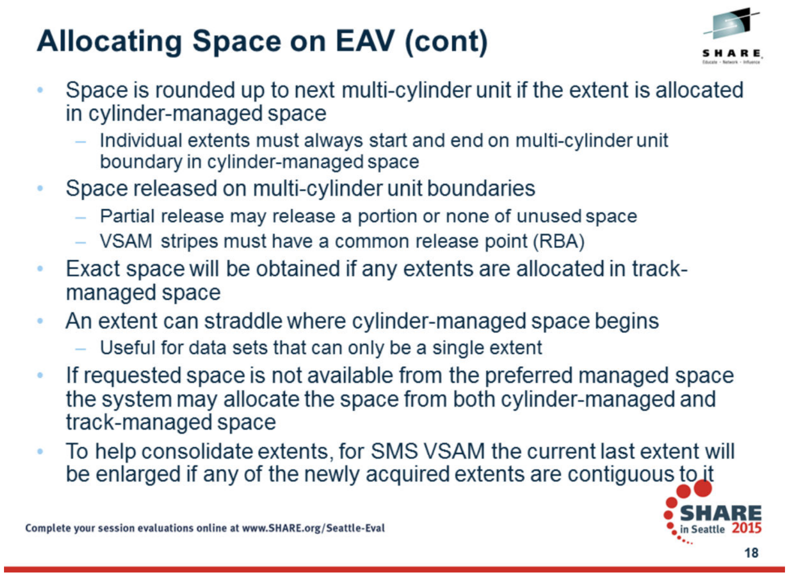
## Allocating Space on EAV

- EAV can be SMS-managed or non-SMS managed
  - Mix of EAV, non-EAV in storage groups supported
  - Specific and non-specific targets
  - With no changes VSAM files allocated to an EAV will be EAS eligible
- USEEAV(YES|NO) controls whether EAV is allowed for data set initial allocations and EOV extends to new volumes.
  - SMS parmlib, change via SETSMS command
  - Can prevent EAV usage for all SMS, non-SMS requests
- Break Point Value (BPV) directs the preferred placement of EAS-eligible data sets on EAV. SMS storage group, SMS parmlib, SETSMS command
  - Value in cylinders
  - Cylinder-managed space is preferred if requested space >= BPV
  - Track-managed space is preferred if requested space < BPV
  - System default is 10 cylinders

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 17

Break Point Value also discussed in Best Practices



## Allocating Space on EAV (cont)

- Space is rounded up to next multi-cylinder unit if the extent is allocated in cylinder-managed space
  - Individual extents must always start and end on multi-cylinder unit boundary in cylinder-managed space
- Space released on multi-cylinder unit boundaries
  - Partial release may release a portion or none of unused space
  - VSAM stripes must have a common release point (RBA)
- Exact space will be obtained if any extents are allocated in track-managed space
- An extent can straddle where cylinder-managed space begins
  - Useful for data sets that can only be a single extent
- If requested space is not available from the preferred managed space the system may allocate the space from both cylinder-managed and track-managed space
- To help consolidate extents, for SMS VSAM the current last extent will be enlarged if any of the newly acquired extents are contiguous to it

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

18

Space is rounded up to the next MCU if the entire extent(s) is allocated in cylinder-managed space.

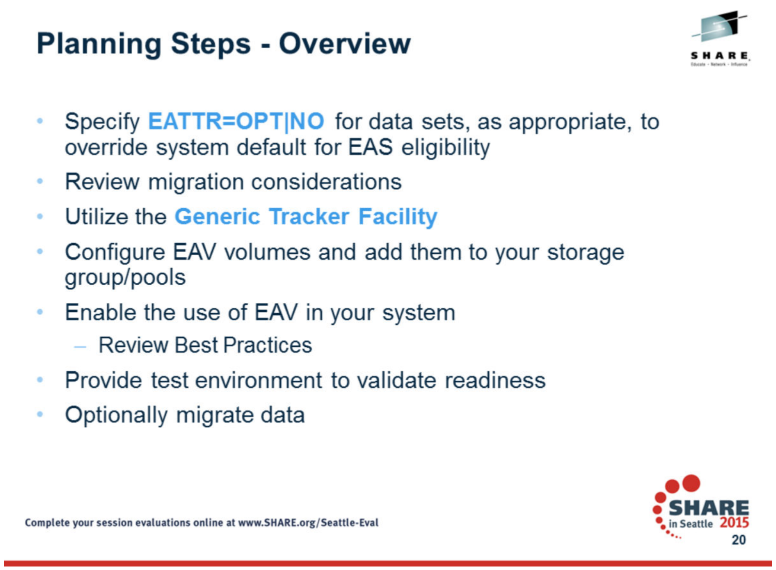
Individual extents always start/end on MCU boundary. This applies to the creation of data sets, extending them with more space and also releasing unused space (partial release). For partial release, some or all of the unused space may not be released because extents must end on a MCU boundary. For VSAM striping, all stripes must have a common 'release point' - - common RBA based on MCU or CA boundaries.

Exact space will be obtained if some portion of the space is allocated in track-managed space.

Current algorithms in the search for space continue to apply. The system will select the first free space extent from the start of the preferred managed-space that can satisfy the requested quantity. When the space cannot be satisfied in one extent, additional extents may be used to satisfy the request where the used extents are ranked from the largest to smallest in the preferred managed space. When the request cannot be satisfied from the preferred managed space, the available free space extents from the entire volume are ranked and used. Space could be from the non-preferred area or from both track-managed and cylinder-managed spaces.

For VSAM extended format the system looks at all newly acquired extents for an extent contiguous to the current last extent. Today's code only looks at the first returned extent from the extend of space.

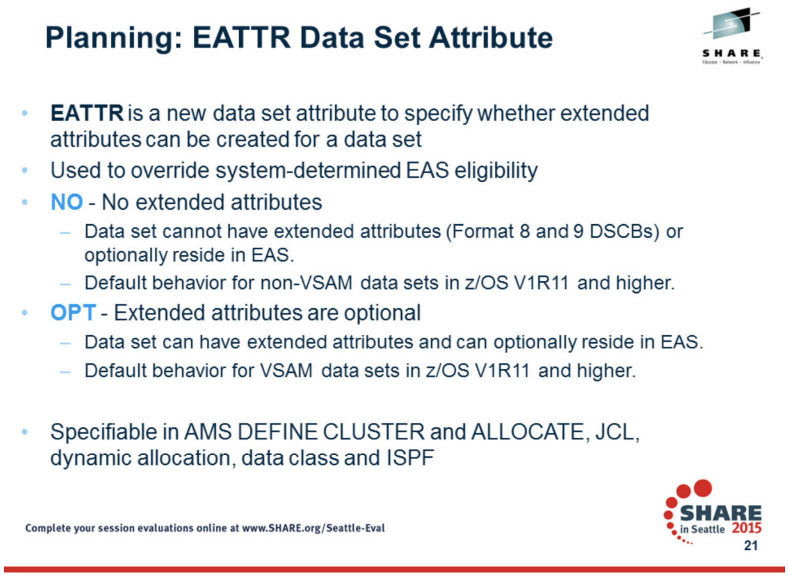
## Planning and Best Practices



## Planning Steps - Overview

- Specify **EATTR=OPT|NO** for data sets, as appropriate, to override system default for EAS eligibility
- Review migration considerations
- Utilize the **Generic Tracker Facility**
- Configure EAV volumes and add them to your storage group/pools
- Enable the use of EAV in your system
  - Review Best Practices
- Provide test environment to validate readiness
- Optionally migrate data

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



## Planning: EATTR Data Set Attribute

- **EATTR** is a new data set attribute to specify whether extended attributes can be created for a data set
- Used to override system-determined EAS eligibility
- **NO** - No extended attributes
  - Data set cannot have extended attributes (Format 8 and 9 DSCBs) or optionally reside in EAS.
  - Default behavior for non-VSAM data sets in z/OS V1R11 and higher.
- **OPT** - Extended attributes are optional
  - Data set can have extended attributes and can optionally reside in EAS.
  - Default behavior for VSAM data sets in z/OS V1R11 and higher.
- Specifiable in AMS DEFINE CLUSTER and ALLOCATE, JCL, dynamic allocation, data class and ISPF

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 21

EATTR, was added to allow a user to control whether a data set can have extended attribute DSCBs and thus control whether it can be allocated in EAS.

•EATTR of NO indicates that the data set can not have extended attributes or reside in EAS. This is the default for non-VSAM data sets

•EATTR of OPT indicates that the data set can have extended attributes and can optionally reside in EAS. This is the default for VSAM data sets

EATTR is specifiable for <u>all data set types</u> and could be specified in the list of these interfaces. The EATTR value will be encoded to a value and written in the format **1 or 8** DSCB for all data set types and in the VVDS for VSAM clusters. The EATTR value recorded for a data set type that is not supported as being EAS-eligible will have no affect until a future time when the system might begin supporting that data set type for EAS.

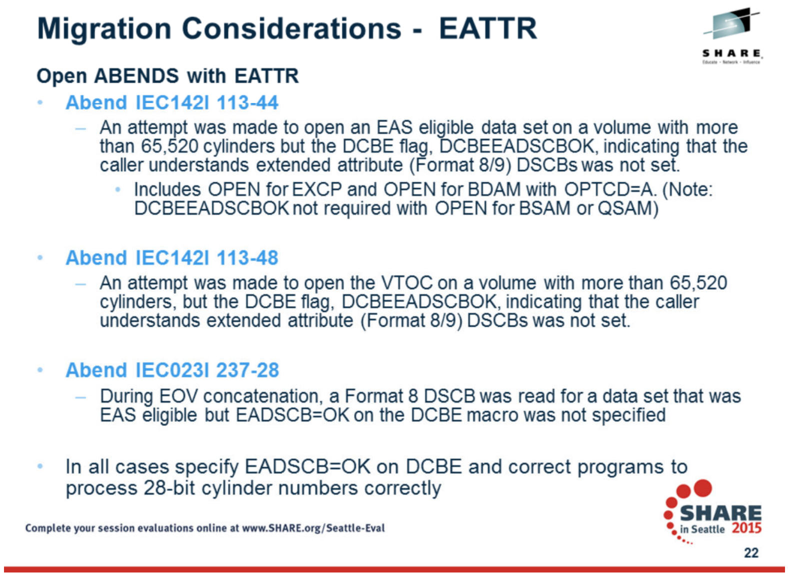
Displaying the EATTR value will be provided in the list of reports specified including DCOLLECT.

Volume selection uses the EATTR values for SMS and HSM non-SMS volumes processing. EATTR will be determined as follows:

- In the JCL interface by merging EATTR from the JCL, LIKE= and data class in that order

Your programs can read DSCBs by issuing OBTAIN, CVAFDIR or CVAFFILT macros or by reading a VTOC. In a format 1 or 8 DSCB the EATTR value is recorded in two bits at offset 61:

```
DS1FLAG1
.... ..00                    EATTR not specified.
.... ..01   DS1EATTR_NO      EATTR=NO.
.... ..10   DS1EATTR_OPT   EATTR=OPT.
```

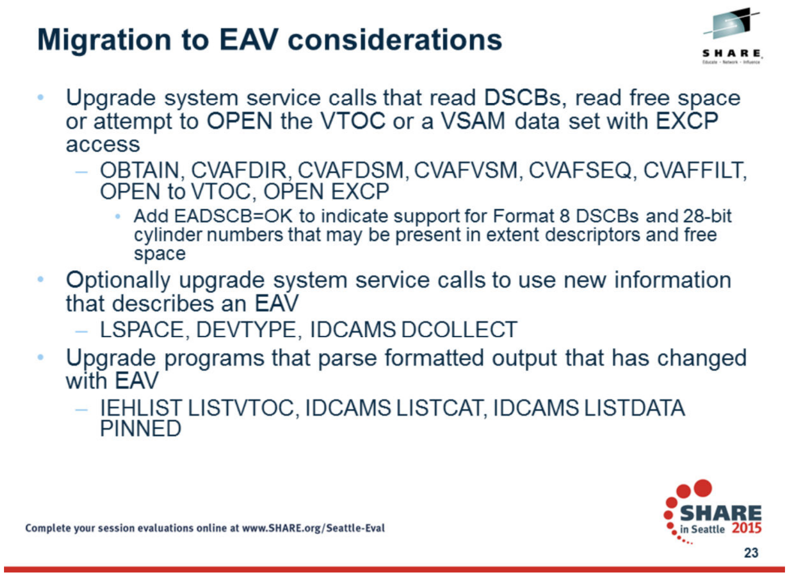


## Migration Considerations - EATTR

**Open ABENDS with EATTR**

- **Abend IEC142I 113-44**
  - An attempt was made to open an EAS eligible data set on a volume with more than 65,520 cylinders but the DCBE flag, DCBEEADSCBOK, indicating that the caller understands extended attribute (Format 8/9) DSCBs was not set.
    - Includes OPEN for EXCP and OPEN for BDAM with OPTCD=A. (Note: DCBEEADSCBOK not required with OPEN for BSAM or QSAM)
- **Abend IEC142I 113-48**
  - An attempt was made to open the VTOC on a volume with more than 65,520 cylinders, but the DCBE flag, DCBEEADSCBOK, indicating that the caller understands extended attribute (Format 8/9) DSCBs was not set.
- **Abend IEC023I 237-28**
  - During EOV concatenation, a Format 8 DSCB was read for a data set that was EAS eligible but EADSCB=OK on the DCBE macro was not specified
- In all cases specify EADSCB=OK on DCBE and correct programs to process 28-bit cylinder numbers correctly

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



## Migration to EAV considerations

- Upgrade system service calls that read DSCBs, read free space or attempt to OPEN the VTOC or a VSAM data set with EXCP access
  - OBTAIN, CVAFDIR, CVAFDSM, CVAFVSM, CVAFSEQ, CVAFFILT, OPEN to VTOC, OPEN EXCP
    - Add EADSCB=OK to indicate support for Format 8 DSCBs and 28-bit cylinder numbers that may be present in extent descriptors and free space
- Optionally upgrade system service calls to use new information that describes an EAV
  - LSPACE, DEVTYPE, IDCAMS DCOLLECT
- Upgrade programs that parse formatted output that has changed with EAV
  - IEHLIST LISTVTOC, IDCAMS LISTCAT, IDCAMS LISTDATA PINNED

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

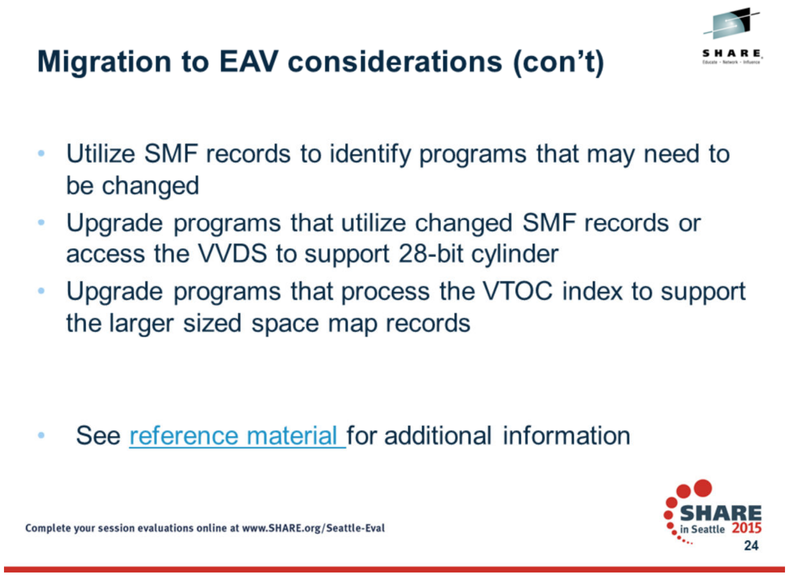
SHARE in Seattle 2015

23

The tracking facility can be used to help identify the first three migration considerations list.

Recommend actions for migration:

•Look for OBTAIN, REALLOC, CVAFDIR, CVAFSEQ, CVAFDSM and CVAFFILT macro calls.
•Look for programs that calculate volume or data set size by any means, including reading a VTOC or VTOC index directly with a BSAM or EXCP DCB. The system cannot distinguish between programs that read DSCBs for space information and those that read them for metadata. Both kinds of program require a new option.
•Look for EXCP, XDAP and STARTIO macros for DASD channel programs and other programs that examine DASD channel programs or track addresses.
•Look for programs that examine any of the many operator messages that contain a DASD track or block address or data set or volume size. Such track or block addresses generally are represented in the documentation as *cchh*, *cchhr, mbbcchhr or track-address*.
•Look for programs that access the VTOC index (“SYS1.VTOCIX.*volser*”) and VVDS data sets (“SYS1.VVDS.Vvolser”)



## Migration to EAV considerations (con't)

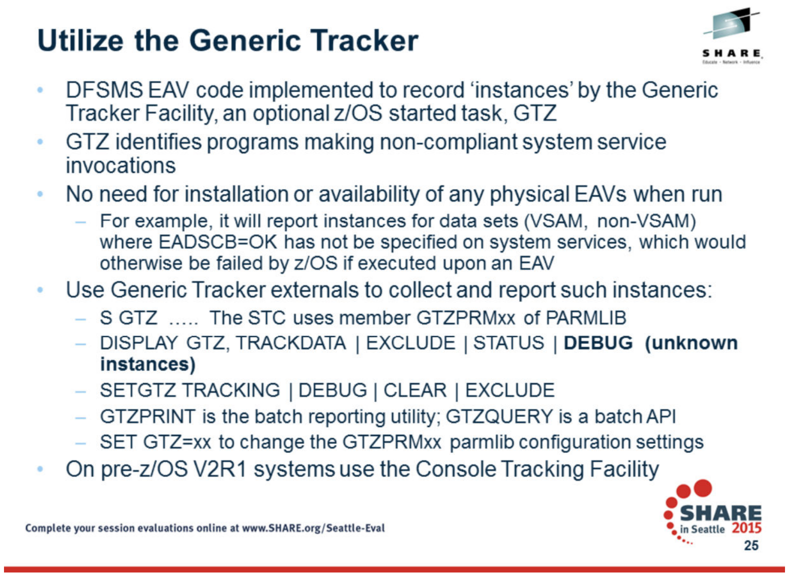
- Utilize SMF records to identify programs that may need to be changed
- Upgrade programs that utilize changed SMF records or access the VVDS to support 28-bit cylinder
- Upgrade programs that process the VTOC index to support the larger sized space map records

- See reference material for additional information

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

24



## Utilize the Generic Tracker

- DFSMS EAV code implemented to record 'instances' by the Generic Tracker Facility, an optional z/OS started task, GTZ
- GTZ identifies programs making non-compliant system service invocations
- No need for installation or availability of any physical EAVs when run
  - For example, it will report instances for data sets (VSAM, non-VSAM) where EADSCB=OK has not be specified on system services, which would otherwise be failed by z/OS if executed upon an EAV
- Use Generic Tracker externals to collect and report such instances:
  - S GTZ ..... The STC uses member GTZPRMxx of PARMLIB
  - DISPLAY GTZ, TRACKDATA | EXCLUDE | STATUS | **DEBUG (unknown instances)**
  - SETGTZ TRACKING | DEBUG | CLEAR | EXCLUDE
  - GTZPRINT is the batch reporting utility; GTZQUERY is a batch API
  - SET GTZ=xx to change the GTZPRMxx parmlib configuration settings
- On pre-z/OS V2R1 systems use the Console Tracking Facility

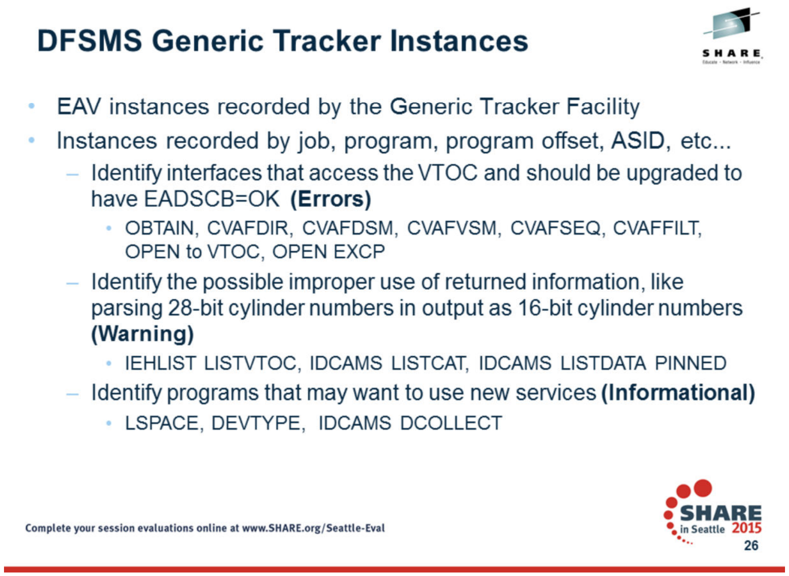
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

25

Tracking Facility
The objective of the EAV migration assistance tracker is three-fold.
•The first is to identify executions of select systems services by job and program name where the invoking programs may require analysis for changes to use new services provided by the system service. These program calls and the reported output are not considered in error as valid information is returned. These will be considered as informational instances.
•The second is to identify the possible improper use of returned information, like parsing 28-bit cylinder numbers in output as 16-bit cylinder numbers. These will be considered as warning instances.
•The third is to identify instances that will either be failed or identified with an informational message if they are run on an extended address volume. These will be considered as error instances. These errors apply to the following functions when the target volume of the operation is a non-EAV and the function invoked did not specify the EADSCB=OK keyword:

- DADSM OBTAIN
- CVAFDIR
- CVAFSEQ
- CVAFFILT
- EXCP OPEN of an EAS eligible data set
- The other instances are volume oriented and are not affected by additional data set types becoming eligible for EAS. They include:
- CVAFVSM
- CVAFDSM
- LISTCAT
- DCOLLECT
- IEHLIST
- DEVTYPE
- LSPACE
- LISTDATA PINNED
- EXCP OPEN VTOC



## DFSMS Generic Tracker Instances

- EAV instances recorded by the Generic Tracker Facility
- Instances recorded by job, program, program offset, ASID, etc...
  - Identify interfaces that access the VTOC and should be upgraded to have EADSCB=OK **(Errors)**
    - OBTAIN, CVAFDIR, CVAFDSM, CVAFVSM, CVAFSEQ, CVAFFILT, OPEN to VTOC, OPEN EXCP
  - Identify the possible improper use of returned information, like parsing 28-bit cylinder numbers in output as 16-bit cylinder numbers **(Warning)**
    - IEHLIST LISTVTOC, IDCAMS LISTCAT, IDCAMS LISTDATA PINNED
  - Identify programs that may want to use new services **(Informational)**
    - LSPACE, DEVTYPE, IDCAMS DCOLLECT

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

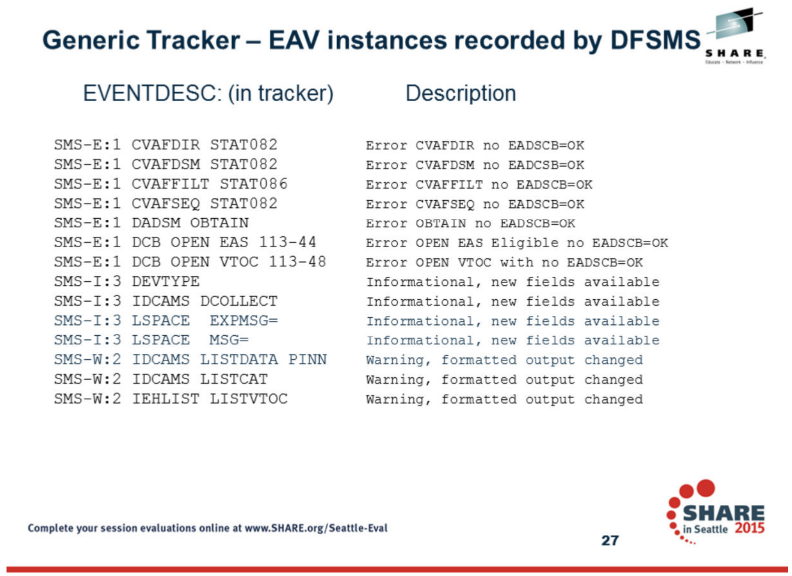
26

Identify instances of events running in the system by job, program name (+offset). These instances could be errors, warnings or informational instances of certain functions running in the system. Errors are instances that would be failed if actually run on an EAV. The warning and informational instances would not fail the function. All would be collected in the tracking facility without the actual use of an EAV.

Keep in mind that being in the tracking facility report does not mean there is a problem. It is just a way to help people find what they should examine to see whether it should be changed.

Reference material:

- See GENERIC TRACKER information in IBM publications.



## Generic Tracker – EAV instances recorded by DFSMS

EVENTDESC: (in tracker) Description

```
SMS-E:1 CVAFDIR STAT082          Error CVAFDIR no EADSCB=OK
SMS-E:1 CVAFDSM STAT082          Error CVAFDSM no EADCSB=OK
SMS-E:1 CVAFFILT STAT086         Error CVAFFILT no EADSCB=OK
SMS-E:1 CVAFSEQ STAT082          Error CVAFSEQ no EADSCB=OK
SMS-E:1 DADSM OBTAIN             Error OBTAIN no EADSCB=OK
SMS-E:1 DCB OPEN EAS 113-44      Error OPEN EAS Eligible no EADSCB=OK
SMS-E:1 DCB OPEN VTOC 113-48     Error OPEN VTOC with no EADSCB=OK
SMS-I:3 DEVTYPE                  Informational, new fields available
SMS-I:3 IDCAMS DCOLLECT          Informational, new fields available
SMS-I:3 LSPACE  EXPMSG=          Informational, new fields available
SMS-I:3 LSPACE  MSG=             Informational, new fields available
SMS-W:2 IDCAMS LISTDATA PINN     Warning, formatted output changed
SMS-W:2 IDCAMS LISTCAT           Warning, formatted output changed
SMS-W:2 IEHLIST LISTVTOC         Warning, formatted output changed
```

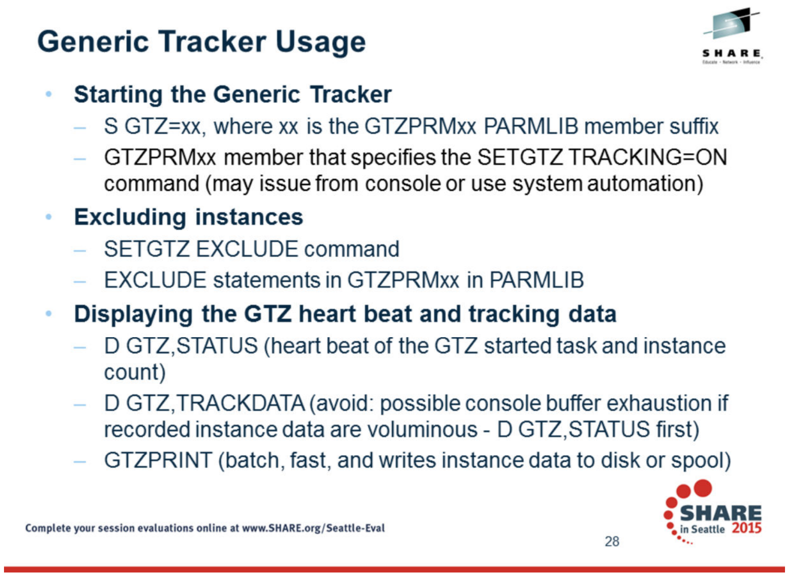
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

27

This is a sample display of instances that could be recorded in the tracking facility.

- The tracking information column identifies the instance as being DFSMS related with the ‘SMS’ prefix.
- The additional ‘I’, ‘E’, or ‘W’ appended to SMS identifies the instance as being an informational, error or warning event.
- The remaining text in the tracking information describe the event that was recorded or for error events the type of error that would have occurred if the function were executed on an EAV. The tracking value is a value unique to the error being recorded.
- The JOBNAME, PROGRAM+OFF and ASID identifies what was being run at the time of the instance.

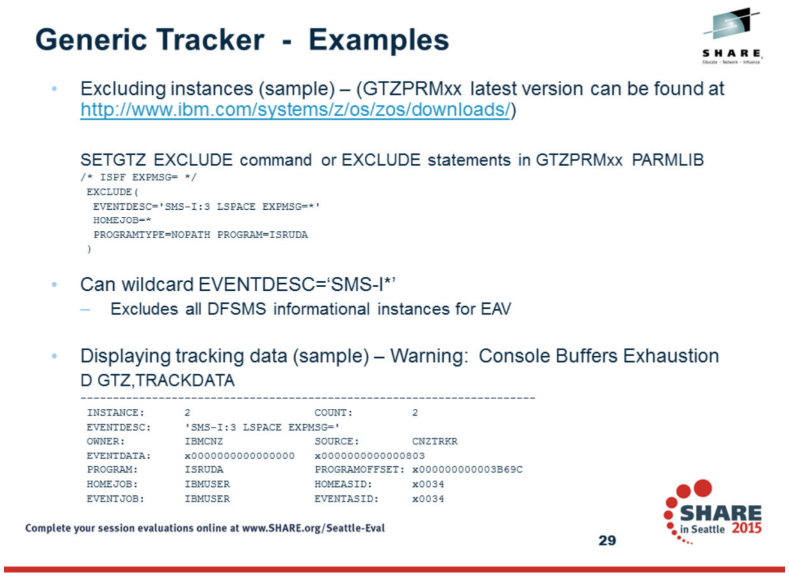
Only unique instances are recorded. Duplicates are kept tracked by the NUM column being incremented. The tracking information, jobname and program name fields make up a unique instance.



## Generic Tracker Usage

- **Starting the Generic Tracker**
  - S GTZ=xx, where xx is the GTZPRMxx PARMLIB member suffix
  - GTZPRMxx member that specifies the SETGTZ TRACKING=ON command (may issue from console or use system automation)
- **Excluding instances**
  - SETGTZ EXCLUDE command
  - EXCLUDE statements in GTZPRMxx in PARMLIB
- **Displaying the GTZ heart beat and tracking data**
  - D GTZ,STATUS (heart beat of the GTZ started task and instance count)
  - D GTZ,TRACKDATA (avoid: possible console buffer exhaustion if recorded instance data are voluminous - D GTZ,STATUS first)
  - GTZPRINT (batch, fast, and writes instance data to disk or spool)

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



## Generic Tracker - Examples

- Excluding instances (sample) – (GTZPRMxx latest version can be found at http://www.ibm.com/systems/z/os/zos/downloads/)

  SETGTZ EXCLUDE command or EXCLUDE statements in GTZPRMxx PARMLIB

```
/* ISPF EXPMSG= */
 EXCLUDE(
  EVENTDESC='SMS-I:3 LSPACE EXPMSG=*'
  HOMEJOB=*
  PROGRAMTYPE=NOPATH PROGRAM=ISRUDA
 )
```

- Can wildcard EVENTDESC='SMS-I*'
  - Excludes all DFSMS informational instances for EAV

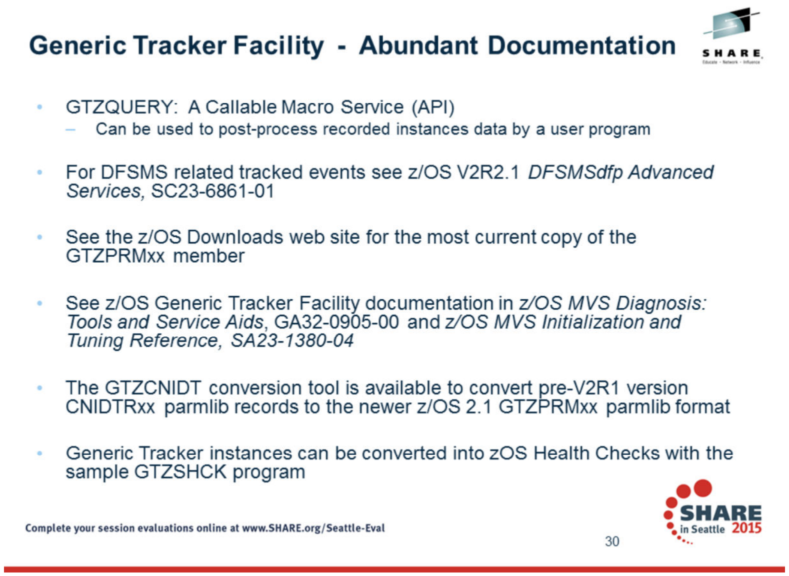
- Displaying tracking data (sample) – Warning: Console Buffers Exhaustion

  D GTZ,TRACKDATA

```
----------------------------------------------------------------------
 INSTANCE:     2                   COUNT:        2
 EVENTDESC:    'SMS-I:3 LSPACE EXPMSG='
 OWNER:        IBMCNZ              SOURCE:       CNZTRKR
 EVENTDATA:    x0000000000000000   x0000000000000803
 PROGRAM:      ISRUDA              PROGRAMOFFSET: x000000000003B69C
 HOMEJOB:      IBMUSER             HOMEASID:     x0034
 EVENTJOB:     IBMUSER             EVENTASID:    x0034
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

29



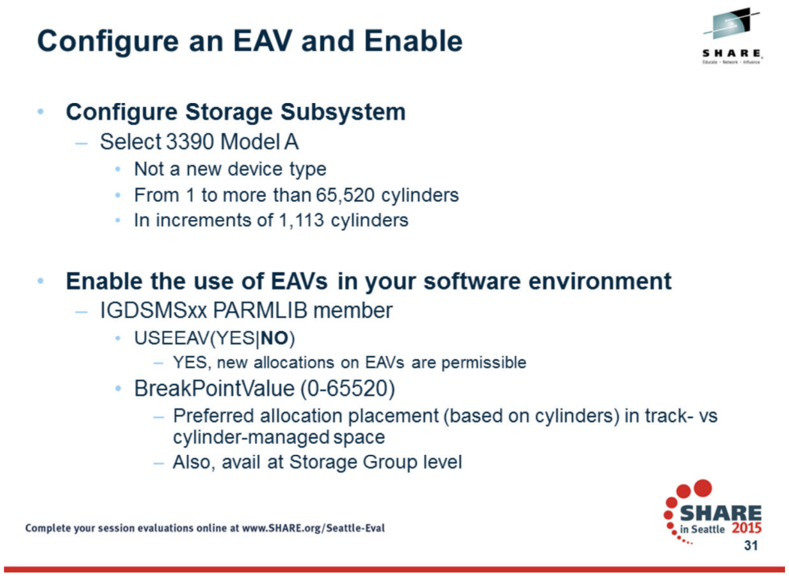
## Generic Tracker Facility - Abundant Documentation

- GTZQUERY: A Callable Macro Service (API)
  - Can be used to post-process recorded instances data by a user program
- For DFSMS related tracked events see z/OS V2R2.1 *DFSMSdfp Advanced Services,* SC23-6861-01
- See the z/OS Downloads web site for the most current copy of the GTZPRMxx member
- See z/OS Generic Tracker Facility documentation in *z/OS MVS Diagnosis: Tools and Service Aids*, GA32-0905-00 and *z/OS MVS Initialization and Tuning Reference, SA23-1380-04*
- The GTZCNIDT conversion tool is available to convert pre-V2R1 version CNIDTRxx parmlib records to the newer z/OS 2.1 GTZPRMxx parmlib format
- Generic Tracker instances can be converted into zOS Health Checks with the sample GTZSHCK program

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

30

The conversion tool GTZCNIDT, with sample JCL in GTZCNIDJ, can be used to create a GTZPRMxx parmlib member from an existing CNIDTRxx parmlib member if used on a pre-z/OS 2.1



## Configure an EAV and Enable

- **Configure Storage Subsystem**
  - Select 3390 Model A
    - Not a new device type
    - From 1 to more than 65,520 cylinders
    - In increments of 1,113 cylinders
- **Enable the use of EAVs in your software environment**
  - IGDSMSxx PARMLIB member
    - USEEAV(YES|**NO**)
      - YES, new allocations on EAVs are permissible
    - BreakPointValue (0-65520)
      - Preferred allocation placement (based on cylinders) in track- vs cylinder-managed space
      - Also, avail at Storage Group level

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 31

An EAV can be a new one configured in the DS8000 or an existing volume that can be expanded in place without disruption. When a volume is configured with more than 65,520 cylinders the total size of the volume will be restricted to an integral multiple of 1113 cylinders. This is equivalent to a 1 GB segment in the DS8000; thus with this rule, space in the backing storage is fully utilized.

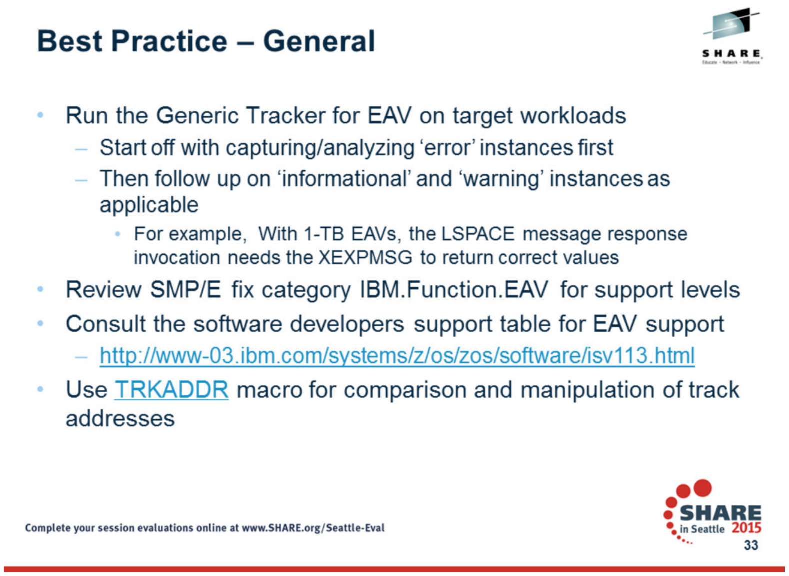
*Example: Creating CKD volumes using mkckdvol*

```
dscli> mkckdvol -extpool p2 -cap 262668 0080
Date/Time: 08 April 2007 17:04:31 IBM DSCLI Version: 5.1.0.204 DS: IBM.2107-7503461
CMUC00021I mkckdvol: CKD Volume 0000 successfully created.
```

## Best Practices

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

32



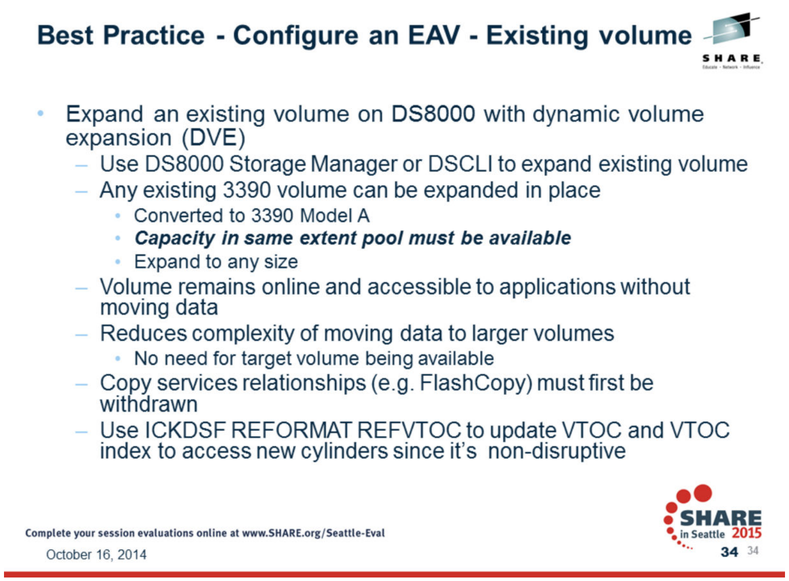
## Best Practice – General

- Run the Generic Tracker for EAV on target workloads
  - Start off with capturing/analyzing 'error' instances first
  - Then follow up on 'informational' and 'warning' instances as applicable
    - For example, With 1-TB EAVs, the LSPACE message response invocation needs the XEXPMSG to return correct values
- Review SMP/E fix category IBM.Function.EAV for support levels
- Consult the software developers support table for EAV support
  - http://www-03.ibm.com/systems/z/os/zos/software/isv113.html
- Use TRKADDR macro for comparison and manipulation of track addresses

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

33



## Best Practice - Configure an EAV - Existing volume

- Expand an existing volume on DS8000 with dynamic volume expansion (DVE)
  - Use DS8000 Storage Manager or DSCLI to expand existing volume
  - Any existing 3390 volume can be expanded in place
    - Converted to 3390 Model A
    - ***Capacity in same extent pool must be available***
    - Expand to any size
  - Volume remains online and accessible to applications without moving data
  - Reduces complexity of moving data to larger volumes
    - No need for target volume being available
  - Copy services relationships (e.g. FlashCopy) must first be withdrawn
  - Use ICKDSF REFORMAT REFVTOC to update VTOC and VTOC index to access new cylinders since it's non-disruptive

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

October 16, 2014

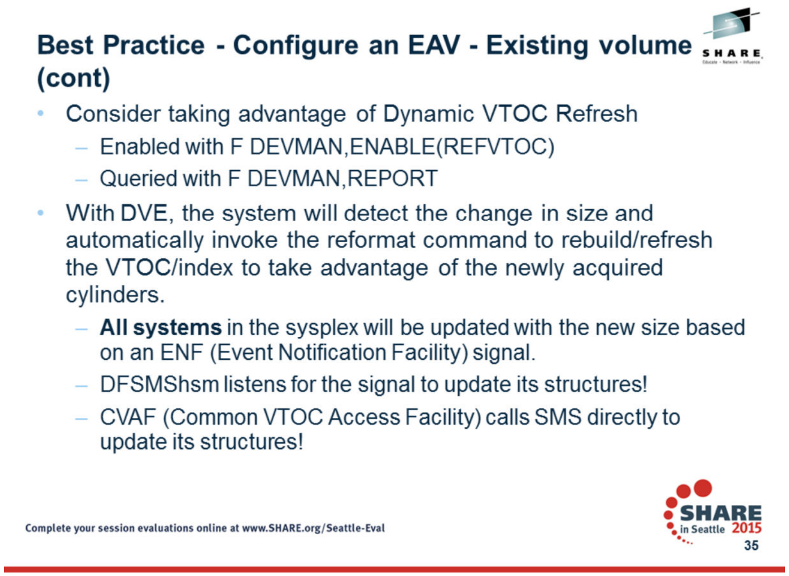
SHARE in Seattle 2015

34 34

An EAV can be a new one configured in the DS8000 or an existing volume that can be expanded in place without disruption. When a volume is configured with more than 65,520 cylinders the total size of the volume will be restricted to an integral multiple of 1113 cylinders. This is equivalent to a 1 GB segment in the DS8000, thus with this rule space in the back storage is fully utilized.

*Example: Expanding a CKD volume*

```
dscli> chckdvol -cap 262668 0080
Date/Time: April 06, 2008 4:54:09 IBM DSCLI Version: 5.3.0.977 DS: IBM.2107-7520781
CMUC00332W chckdvol: Some host operating systems do not support changing the volume size.
Are you sure that you want to resize the volume? [y/n]: y
CMUC00022I chckdvol: CKD Volume 0080 successfully modified.
```

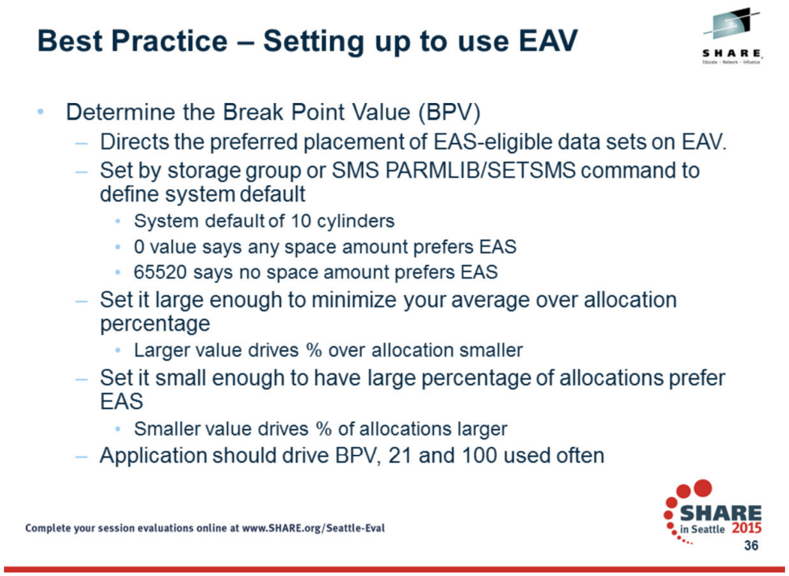


## Best Practice - Configure an EAV - Existing volume (cont)

- Consider taking advantage of Dynamic VTOC Refresh
  - Enabled with F DEVMAN,ENABLE(REFVTOC)
  - Queried with F DEVMAN,REPORT
- With DVE, the system will detect the change in size and automatically invoke the reformat command to rebuild/refresh the VTOC/index to take advantage of the newly acquired cylinders.
  - **All systems** in the sysplex will be updated with the new size based on an ENF (Event Notification Facility) signal.
  - DFSMShsm listens for the signal to update its structures!
  - CVAF (Common VTOC Access Facility) calls SMS directly to update its structures!

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 35



## Best Practice – Setting up to use EAV

- Determine the Break Point Value (BPV)
  - Directs the preferred placement of EAS-eligible data sets on EAV.
  - Set by storage group or SMS PARMLIB/SETSMS command to define system default
    - System default of 10 cylinders
    - 0 value says any space amount prefers EAS
    - 65520 says no space amount prefers EAS
  - Set it large enough to minimize your average over allocation percentage
    - Larger value drives % over allocation smaller
  - Set it small enough to have large percentage of allocations prefer EAS
    - Smaller value drives % of allocations larger
  - Application should drive BPV, 21 and 100 used often

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

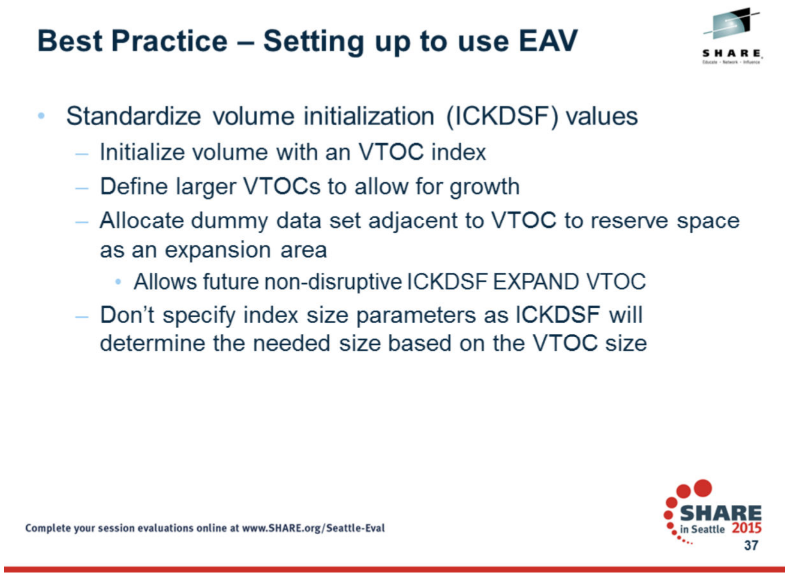
SHARE in Seattle 2015

36

Specifics on USEEAV parmlib<br>
USEEAV(YES|NO) controls whether EAV is allowed for data set initial allocations

- •Specified in SMS PARMLIB member IGDSMSxx
- •Defaults to NO if SMS is active
- •Defaults to YES if SMS is not active
- •May be changed via SETSMS USEEAV(YES|NO) when SMS is active
- •Has no effect for data set extend to same volume

EAS eligible data sets will always on creation have Format 8/9 DSCBs regardless of where the actual extents where allocated from. Extend may convert a format 1 DSCB for an EAS eligible data set to a Format 8/9 DSCB pair

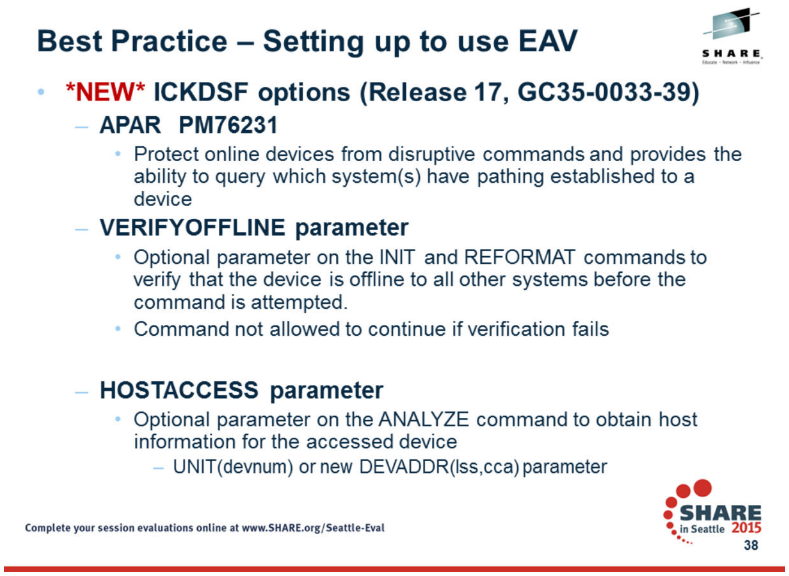


## Best Practice – Setting up to use EAV

- Standardize volume initialization (ICKDSF) values
  - Initialize volume with an VTOC index
  - Define larger VTOCs to allow for growth
  - Allocate dummy data set adjacent to VTOC to reserve space as an expansion area
    - Allows future non-disruptive ICKDSF EXPAND VTOC
  - Don’t specify index size parameters as ICKDSF will determine the needed size based on the VTOC size

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

Allocating dummy data sets that are adjacent to the VTOC prior to allocating the index data set and/or allocating an index data set much larger than normal that is adjacent to the VTOC or these dummy data sets are ways to ensure a VTOC has a place to be expanded into. Another option is to create a VTOC that is much larger than normal allowing it to have the capacity in the future to support a larger number of data sets.



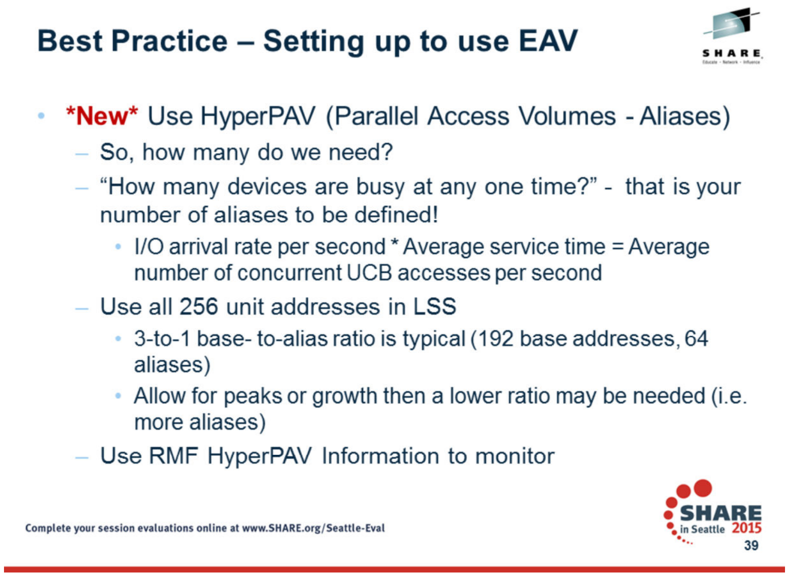
## Best Practice – Setting up to use EAV

- *NEW* ICKDSF options (Release 17, GC35-0033-39)
  - APAR PM76231
    - Protect online devices from disruptive commands and provides the ability to query which system(s) have pathing established to a device
  - VERIFYOFFLINE parameter
    - Optional parameter on the INIT and REFORMAT commands to verify that the device is offline to all other systems before the command is attempted.
    - Command not allowed to continue if verification fails
  - HOSTACCESS parameter
    - Optional parameter on the ANALYZE command to obtain host information for the accessed device
      - UNIT(devnum) or new DEVADDR(lss,cca) parameter

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

38



## Best Practice – Setting up to use EAV

- ***New*** Use HyperPAV (Parallel Access Volumes - Aliases)
  - So, how many do we need?
  - “How many devices are busy at any one time?” - that is your number of aliases to be defined!
    - I/O arrival rate per second * Average service time = Average number of concurrent UCB accesses per second
  - Use all 256 unit addresses in LSS
    - 3-to-1 base- to-alias ratio is typical (192 base addresses, 64 aliases)
    - Allow for peaks or growth then a lower ratio may be needed (i.e. more aliases)
  - Use RMF HyperPAV Information to monitor

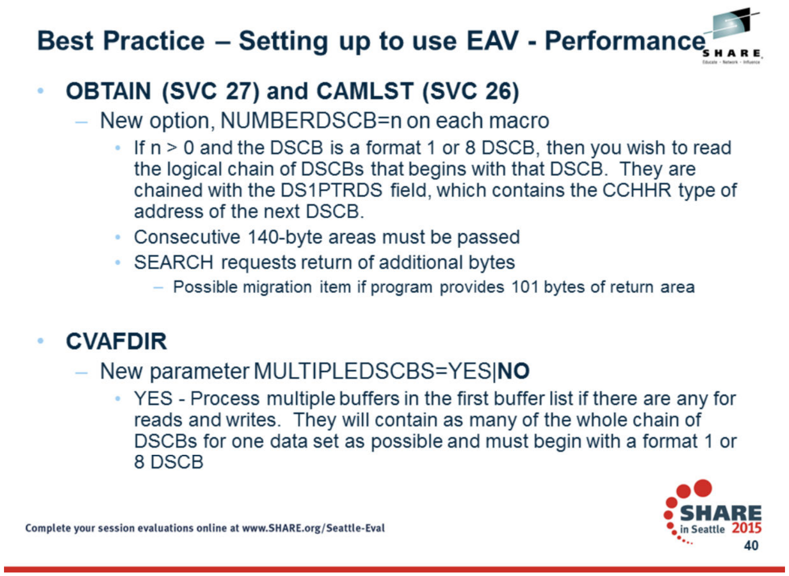
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

39

PAV aliases pool for base volumes in an LSS
Aliases assigned to base as needed
Eliminates/reduces IOSQ time on the UCB

Allows I/Os to scale with larger volumes



## Best Practice – Setting up to use EAV - Performance

- **OBTAIN (SVC 27) and CAMLST (SVC 26)**
  - New option, NUMBERDSCB=n on each macro
    - If n > 0 and the DSCB is a format 1 or 8 DSCB, then you wish to read the logical chain of DSCBs that begins with that DSCB. They are chained with the DS1PTRDS field, which contains the CCHHR type of address of the next DSCB.
    - Consecutive 140-byte areas must be passed
    - SEARCH requests return of additional bytes
      - Possible migration item if program provides 101 bytes of return area
- **CVAFDIR**
  - New parameter MULTIPLEDSCBS=YES|**NO**
    - YES - Process multiple buffers in the first buffer list if there are any for reads and writes. They will contain as many of the whole chain of DSCBs for one data set as possible and must begin with a format 1 or 8 DSCB

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

40

Access to the VTOC records has been enhanced to allow a program to read the logical DSCB chain for a data set in one invocation of a service. OBTAIN and CAMLST have a new option to specify the number of DSCBs you wish to read. CVAFDIR provides a new keyword, MULTIPLEDSCB=YES, to indicate to CVAFDIR processing to use the multiple buffers passed in the buffer list.

OBTAIN search processing will store an additional two-bytes in the callers return area right after the 101 bytes that is set by OBTAIN on prior releases. Please ensure your program provides the minimum 140-byte return area. These two bytes are set to the total number of consecutive 140-bytes areas that are needed to read all the DSCBs for the data set.

**OBTAIN:**
NUMBERDSCB=number_dscbs

For SEARCH and SEEK requests, number_dscbs is an absolute expression with a value between 0 and 255 that designates the number of consecutive 140-byte return areas that are provided in wkarea_relexp. The system treats a value of 0 as a 1. Currently the system does not support a chain of more than 12 DSCBs for one data set but it is valid for you to provide an area that is longer than currently needed. The system verifies that the provided area is valid. When you provide an area that is long enough to contain more than one DSCB, obtain processing will return DSCBs for the requested data set name in logical VTOC order until all the 140-byte return areas are used. The logical VTOC order is a format-1 DSCB, followed by zero or more format-3 DSCBs or a format-8 DSCB, followed by one or more format-9 DSCBs, followed by zero or more format-3 DSCBs. No absolute maximum number of DSCBs for a data set should be assumed. For SEARCH requests, the actual number of DSCBs will be returned in a field located in the first 140-byte return area. For SEEK requests where the target of the seek operation is not a format-1 or format-8 DSCB, the NUMBERDSCB value will be treated as if it were 1 and only that single DSCB will be returned.

Please note that for programs run on an older level of the system that does not support this keyword, the NUMBERDSCB value will be treated as if it were 1.

**CVAFDIR:**

MULTIPLEDSCBS=YES

This specification indicates that the calling program requests to read/write multiple DSCBs to/from a buffer list that contains more than one buffer list entry. This flag will resolve to a new indicator in the CVPL, CV4MULTD, to be set on. Multiple DSCB processing for reads and writes is requested by specifying the MULTIPLEDSCBS=YES keyword and providing a buffer list that contains more than one buffer list entry (BFLHNOE>1).

MULTIPLEDSCBS=NO

This specification indicates that the calling program requests that only one DSCB should be processed. This is the default for MF=L and MF=I forms of the CVAFDIR marco. When the MULTIPLEDSCBS keyword is not specified on the MF=E form, the existing setting of CV4MULTD is left unchanged. When MULTIPLEDSCBS=NO is specified or defaulted, only the first available buffer list entry is processed.

**More info on MultipleDSCB. CVAFDIR reads:**

Only the first buffer list entry, seek or search, argument will be used. This will provide orientation to the data set from which the subsequent data set DSCBs will be read. As each one is read the DSCB argument (BFLEARG) in each buffer list entry will be set in the format specified by the caller in each buffer list entry. The buffer list argument will be indicated as updated with the flag, BFLEAUPD, set on. Reading the data set DSCBs will be in the logical VTOC order and will continue as long as buffer list entries are available to return the DSCB. A new field in the buffer list header (BFLHNOEN) will be set by CVAFDIR read processing to indicate the number of buffer list entries that are needed to read the entire set of DSCBs for the data set. This number is set in the header of the first buffer list and its setting is independent of the specification of the MULTIPLEDSCBS= keyword, the target volume type and whether the number of provided buffer list entries, BFLHNOE, is short.

The logical VTOC order would be either:

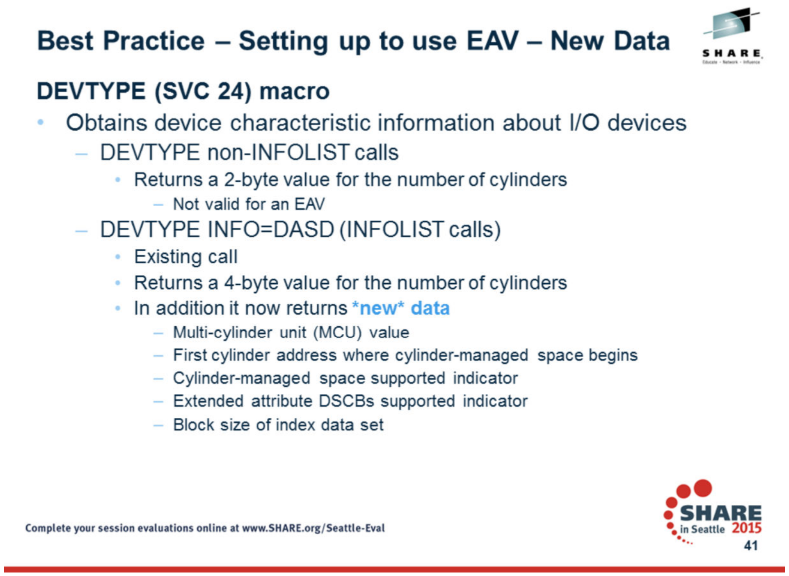
- Format-1 -> Format-3s
- Format -8, -> Format-9s -> Format-3s

Buffer list entries other than the first must provide a 140-byte buffer. The first buffer list entry buffer size will follow the same rules as today. That is, with the seek option, provide either a 96-byte or 140-byte buffer and for the search option provide a 96-byte buffer. All other buffer list entry processing flags as described in the buffer list entry flag byte (BFLEFL) should continue to be supported. They include data in buffer modified, skip, I/O error, no key verify and argument format qualifiers. The DSCB argument (BFLEARG) returned in each buffer list entry is in the format determined by the argument format qualifiers (BFLECHR or BFLETTR) in each buffer list entry.

**More info on MULTIPLEDSCB CVAFDIR writes:**

The buffer list header must indicate the number of buffer list entries passed. Only buffer list entries without the skip flag on will be processed. The order in which the DSCBs are passed in the buffer list entries must correspond to the logical VTOC order as described previously. A new flag in the buffer list header can be set by the caller to indicate that the logical order in which the DSCBs appear in the buffer list must be written in reverse order (BFLHWREV).

For write processing the first buffer list entry, like today, can be a 96-byte buffer if the DSCB to write is a Format 1. The same will hold true for a Format 8 DSCB. A 140-byte buffer can also be provided for these DSCBs as long as the BFLEARG points to the actual DSCB that needs to be written.  Buffer list entries that do not describe Format 1 or 8 DSCBs must provide a 140-byte buffer and its buffer address (BFLEARG) must point to the actual DSCB that needs to be written.  The caller must also specify the buffer list entry argument (BFLEARG) as a CCHHR for these buffer list entries.  A new flag in the buffer list entry could be set for entries where a format 0 DSCB verify before a write is not needed (BFLENVER).  This will override the VERIFY=YES setting.



## Best Practice – Setting up to use EAV – New Data

### DEVTYPE (SVC 24) macro

- Obtains device characteristic information about I/O devices
  - DEVTYPE non-INFOLIST calls
    - Returns a 2-byte value for the number of cylinders
      - Not valid for an EAV
  - DEVTYPE INFO=DASD (INFOLIST calls)
    - Existing call
    - Returns a 4-byte value for the number of cylinders
    - In addition it now returns *new* data
      - Multi-cylinder unit (MCU) value
      - First cylinder address where cylinder-managed space begins
      - Cylinder-managed space supported indicator
      - Extended attribute DSCBs supported indicator
      - Block size of index data set

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

41

Use the DEVTYPE macro to obtain device characteristics information about I/O devices. The information includes the number of cylinders among others. The DEVTYPE macro issued without the INFOLIST parameter returns a 2-byte value for the number of cylinders. The DEVTYPE macro issued with the INFOLIST parameter (INFO=DASD) returns a different format of the device characteristics information. This includes a 4-byte value for the number of cylinders and is mapped to field DVAICYL in shipped mapping macro, IHADVA. The 2-byte field is mapped by DVACYL. In addition, the listed five new fields will be returned.

DVAIXVLD BIT DVACYLMG, DVAEADSCB, DVAVIRSZ valid.

DVACYLMG BIT

- Cylinder-managed space exists on this volume and begins at DVALCYL in multicylinder units of DVAMCU. DVAEADSCB is also set with this flag on. Valid when DVAIXVLD is set.

DVAEADSCB BIT

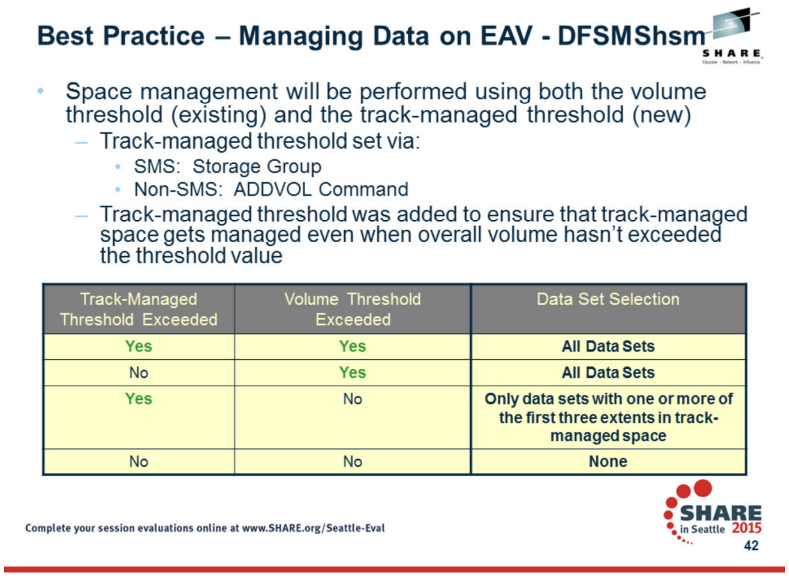
- Extended attribute DSCBs, Format 8 and 9 DSCBs, are allowed on this volume. Valid when DVAIXVLD is set.

DVAMCU 8-bit integer.

- Minimum allocation size in cylinders for cylinder-managed space. Each extent in this space must be a multiple of this value. Also referred to as the multi-cylinder unit (MCU). This is the smallest unit of disk space in cylinders that can be allocated in cylinder-managed space. Valid when DVACYLMG is set. This field is zero on releases before z/OS 1.10 or if the status is not yet known. In these two cases DVAIXVLD is not set.

DVALCYL 16-bit integer.

- First cylinder address divided by 4095 where space is managed in multi-cylinder units. Cylinder-managed space begins at this address. Valid when DVACYLMG is set. This field is zero on releases before z/OS 1.10 or if the status is not yet known. In these two cases DVAIXVLD is not set. DVAVIRSZ 16-bit integer. Block size of the index data set. Valid when DAVIXVLD is on. When valid and zero the volume has no working VTOC index. This field is zero on releases before z/OS 1.10 or if the status is not yet known. In these cases DVAIXVLD is not set.



## Best Practice – Managing Data on EAV - DFSMShsm

- Space management will be performed using both the volume threshold (existing) and the track-managed threshold (new)
  - Track-managed threshold set via:
    - SMS: Storage Group
    - Non-SMS: ADDVOL Command
  - Track-managed threshold was added to ensure that track-managed space gets managed even when overall volume hasn't exceeded the threshold value

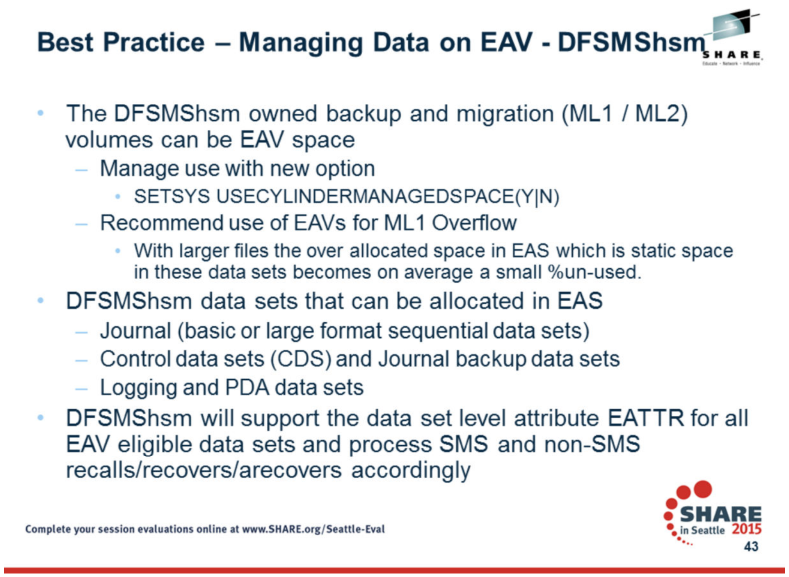
| Track-Managed Threshold Exceeded | Volume Threshold Exceeded | Data Set Selection |
|---|---|---|
| Yes | Yes | **All Data Sets** |
| No | Yes | **All Data Sets** |
| Yes | No | **Only data sets with one or more of the first three extents in track-managed space** |
| No | No | **None** |

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

42

Space management will be performed on both the total volume and track-managed space. When the track-managed space threshold is exceeded and the total volume threshold is not, only data sets with one or more of its first three extents in track-managed space will be processed. The purpose of this support is to ensure that track-managed space gets space-managed even when the overall volume has not exceeded its volume threshold.



## Best Practice – Managing Data on EAV - DFSMShsm

- The DFSMShsm owned backup and migration (ML1 / ML2) volumes can be EAV space
  - Manage use with new option
    - SETSYS USECYLINDERMANAGEDSPACE(Y|N)
  - Recommend use of EAVs for ML1 Overflow
    - With larger files the over allocated space in EAS which is static space in these data sets becomes on average a small %un-used.
- DFSMShsm data sets that can be allocated in EAS
  - Journal (basic or large format sequential data sets)
  - Control data sets (CDS) and Journal backup data sets
  - Logging and PDA data sets
- DFSMShsm will support the data set level attribute EATTR for all EAV eligible data sets and process SMS and non-SMS recalls/recovers/arecovers accordingly

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

43

DFSMShsm will be enhanced to support the additional non-VSAM data set types in the EAS. DFSMShsm will enable all the EAV volume space to be used when they are defined as DFSMShsm-owned disk volumes (ML1, ML2, and backup volumes). Migration copies, backup versions, VTOC and catalog copies can be allocated in the EAS. A special flag will be added to the MCC and MCD records respectively to indicate that the migration copy or backup version is located in the EAS. The DFSMShsm Journal (basic or large format SAM) will now be allowed to be allocated in the EAS. Additionally, all of the backup data sets for the HSM CDS's and journal will now be able to be allocated in the EAS. All other DFSMShsm sequential data sets (PDA, LOG) can now be allocated in the EAS.

**ML1, ML2 and Backup EAV Support**

DFSMShsm will allow use of the cylinder-managed space for migration copies and/or backup versions. A new SETSYS value will be introduced to manage allocation of ML1, ML2 or Backup data sets. The new command is SETSYS USECYLINDERMANAGEDSPACE(Y|N).

**SETSYS USECYLINDERMANAGEDSPACE(Y|N)**

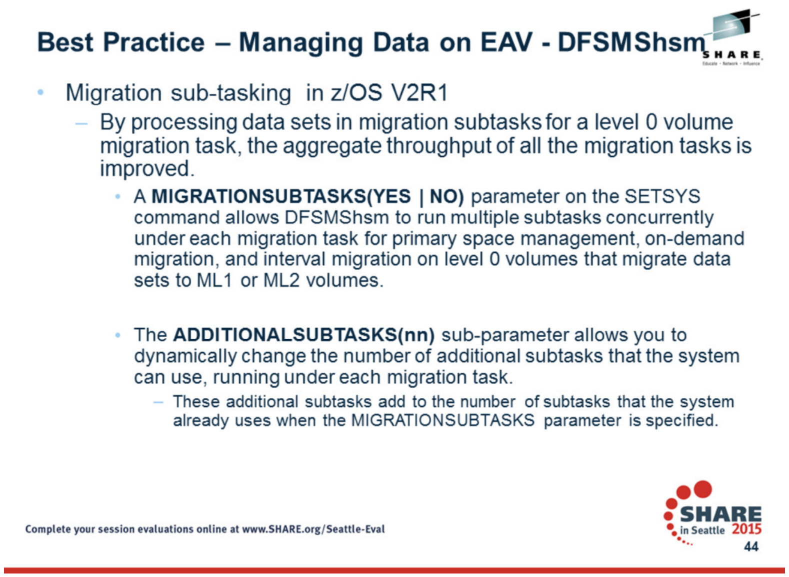
USECYLINDERMANAGEDSPACE(Y) – migration copies and backup versions can reside in EAS. For allocation DFSMShsm will use EATTR=OPT for its migration copy or backup version.

USECYLINDERMANAGEDSPACE(N) – migration copies and backup versions can not reside in EAS. For allocation DFSMShsm will not specify an EATTR value. The default is N. The command can be abbreviated as USECMS.

**Notes:**

1.For the FREEVOL command, if SETSYS USECYLINDERMANAGEDSPACE(Y) command is issued and migration copies, backup versions and/or VTOC copies are going to be moved, then EATTR = OPT will be used. Otherwise migration copies, backup versions and/or VTOC copies being moved by the FREEVOL command will not specify an EATTR value.

2.For the ARECOVER command, the value of the USECYLINDERMANAGEDSPACE parameter (Y|N) will manage allocation of migrated data sets specified in the INCLUDE list.



## Best Practice – Managing Data on EAV - DFSMShsm

- Migration sub-tasking in z/OS V2R1
  - By processing data sets in migration subtasks for a level 0 volume migration task, the aggregate throughput of all the migration tasks is improved.
    - A **MIGRATIONSUBTASKS(YES | NO)** parameter on the SETSYS command allows DFSMShsm to run multiple subtasks concurrently under each migration task for primary space management, on-demand migration, and interval migration on level 0 volumes that migrate data sets to ML1 or ML2 volumes.
    - The **ADDITIONALSUBTASKS(nn)** sub-parameter allows you to dynamically change the number of additional subtasks that the system can use, running under each migration task.
      - These additional subtasks add to the number of subtasks that the system already uses when the MIGRATIONSUBTASKS parameter is specified.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

44

- A **MIGRATIONSUBTASKS(YES |NO)** parameter has been added to the SETSYS command. It allows DFSMShsm to run multiple subtasks concurrently under each migration task for primary space management, on-demand migration, and interval migration on level 0 volumes that migrate data sets to ML1 or ML2 volumes.
- The **ADDITIONALSUBTASKS(*nn*)** sub-parameter allows you to dynamically change the number of additional subtasks that the system can use, running under each migration task. These additional subtasks add to the number of subtasks that the system already uses when the MIGRATIONSUBTASKS parameter is specified. Note that the actual number of total subtasks used can vary. In general, the total migration subtasks used will be lower if a large value is specified with the maximum migration tasks (MAXMIGRATIONTASKS) parameter. Conversely, the total migration subtasks used will be higher, up to 15, if a smaller value is specified with the maximum migration tasks (MAXMIGRATIONTASKS) parameter.

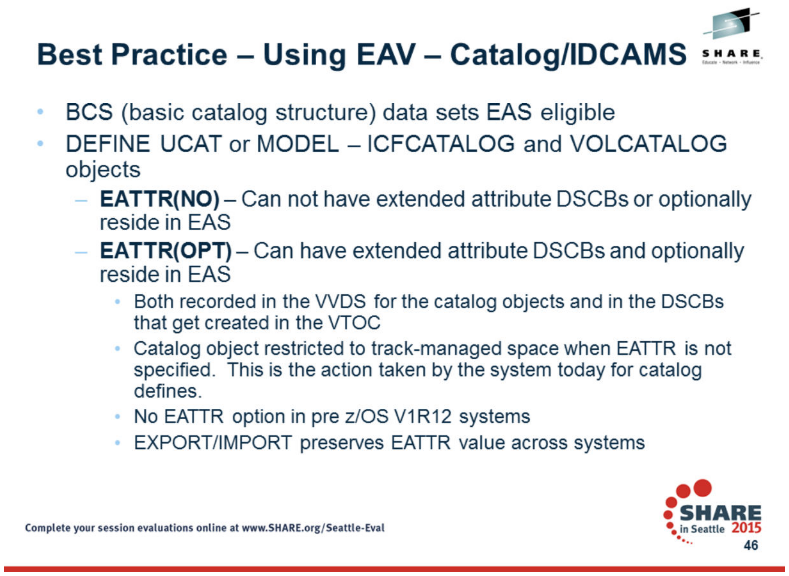
- **By processing data sets in migration subtasks for a level 0 volume migration task, the aggregate throughput of all the migration tasks is improved**.

# Best Practice – Managing Data on EAV - DFSMShsm

- **DFSMS Storage Tiers Enhancements**
  - Today, Moving data to newly defined disk volumes within a storage environment can be manually intensive and cumbersome.
  - ***New enhancement in V2.2***: Enable the DFSMS Class Transition function to move data laterally to the same tier of storage, in addition to the ability of moving data to different tiers storage.
    - MOVE keyword added to the MIGRATE DSNAME, VOLUME and STORAGEGROUP commands
      - Every data set will be processed, regardless of management class policy or threshold, and ACS routines will be invoked to determine the new storage class and/or storage group.
      - **Use Case:** Move DB2 data from existing smaller volumes to the new larger, newly defined EAVs.

  ? **Why it Matters:** Simplify the task of migrating data to newly defined disk volumes.

- **For more details, see Session 17102: Transitioning to Transitions, Wednesday 3:15PM**



## Best Practice – Using EAV – Catalog/IDCAMS

- BCS (basic catalog structure) data sets EAS eligible
- DEFINE UCAT or MODEL – ICFCATALOG and VOLCATALOG objects
  - **EATTR(NO)** – Can not have extended attribute DSCBs or optionally reside in EAS
  - **EATTR(OPT)** – Can have extended attribute DSCBs and optionally reside in EAS
    - Both recorded in the VVDS for the catalog objects and in the DSCBs that get created in the VTOC
    - Catalog object restricted to track-managed space when EATTR is not specified. This is the action taken by the system today for catalog defines.
    - No EATTR option in pre z/OS V1R12 systems
    - EXPORT/IMPORT preserves EATTR value across systems

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

46

AMS will be changed to support EATTR as a keyword on the DEFINE UCAT as well as on DEFINE UCAT MODEL. This applies to both ICFCATALOG and VOLCATALOG objects.

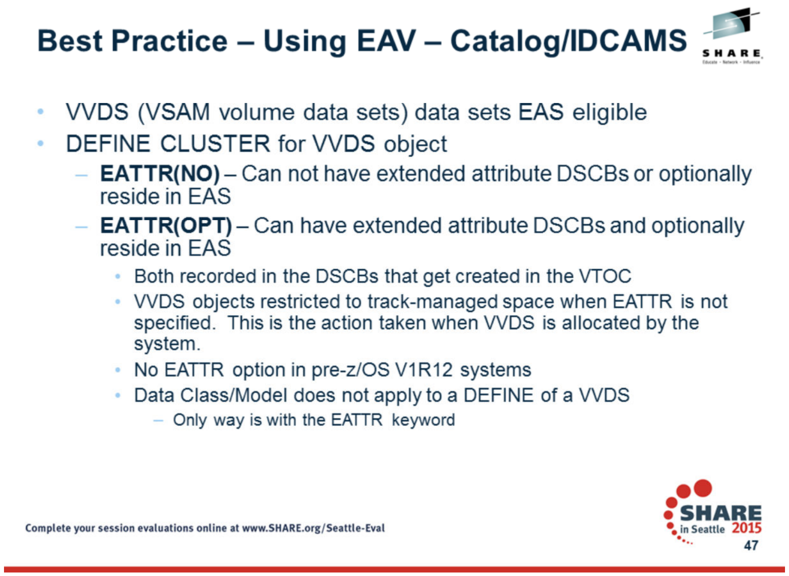
To specify the EATTR value on the IDCAMS DEFINE UCAT or UCAT MODEL, a user would use one of the following:

EATTR(NO)
EATTR(OPT)

The specified value for EATTR of NO or OPT will be recorded in the VVDS for the catalog objects and in the DSCBs that get created in the VTOC.

A DEFINE without the EATTR keyword will result in the catalog object restricted to track-managed space. This is the action taken by the system today for catalog defines.

The releases prior to z/OS V1R12 will not support the EATTR external on the DEFINE UCAT or UCAT MODEL.



## Best Practice – Using EAV – Catalog/IDCAMS

- VVDS (VSAM volume data sets) data sets EAS eligible
- DEFINE CLUSTER for VVDS object
  - **EATTR(NO)** – Can not have extended attribute DSCBs or optionally reside in EAS
  - **EATTR(OPT)** – Can have extended attribute DSCBs and optionally reside in EAS
    - Both recorded in the DSCBs that get created in the VTOC
    - VVDS objects restricted to track-managed space when EATTR is not specified. This is the action taken when VVDS is allocated by the system.
    - No EATTR option in pre-z/OS V1R12 systems
    - Data Class/Model does not apply to a DEFINE of a VVDS
      - Only way is with the EATTR keyword

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

47

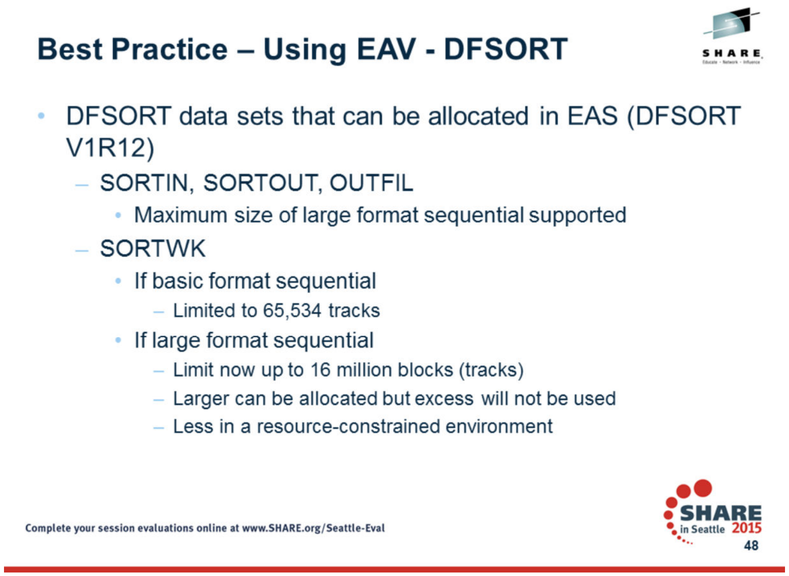
AMS will be changed to support EATTR as a keyword on the DEFINE CLUSTER where the object is a VVDS data set..
To specify the EATTR value on the IDCAMS DEFINE CLUSTER for a VVDS, a user would use one of the following:

EATTR(NO)
EATTR(OPT)

The specified value for EATTR of NO or OPT will be recorded in the VVDS in the DSCBs that get created in the VTOC.

A DEFINE without the EATTR keyword will result in the VVDS object restricted to track-managed space. This is the action taken by the system today for VVDS defines.

The releases prior to z/OS V1R12 will not support the EATTR external on the DEFINE for a VVDS data set.



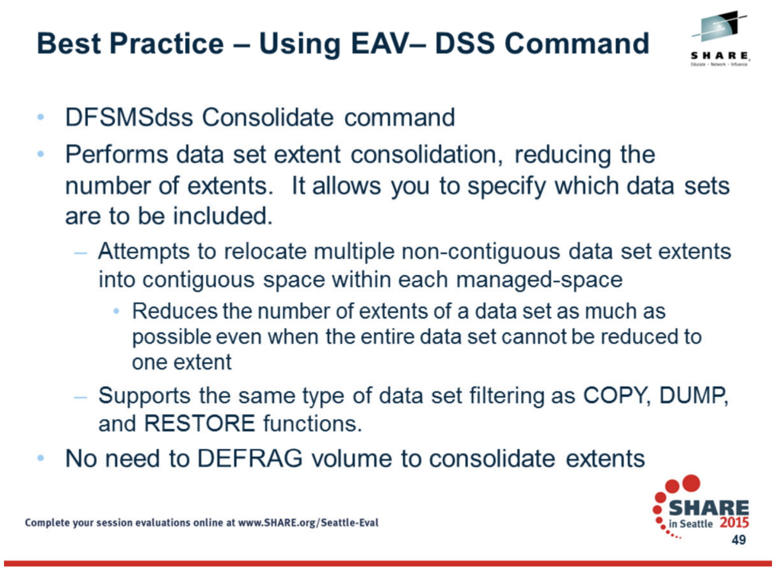
## Best Practice – Using EAV - DFSORT

- DFSORT data sets that can be allocated in EAS (DFSORT V1R12)
  - SORTIN, SORTOUT, OUTFIL
    - Maximum size of large format sequential supported
  - SORTWK
    - If basic format sequential
      - Limited to 65,534 tracks
    - If large format sequential
      - Limit now up to 16 million blocks (tracks)
      - Larger can be allocated but excess will not be used
      - Less in a resource-constrained environment

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015 48

For DFSORT V1R12, DFSORT will support SORTIN, SORTOUT, OUTFIL, and SORTWK data sets as EAS eligible. The largest input and output large format sequential data sets will be supported. However, large format sequential work data sets will support up to 16 million blocks 16,777,216 . A 24-bit internal addressing scheme is now used.



## Best Practice – Using EAV– DSS Command

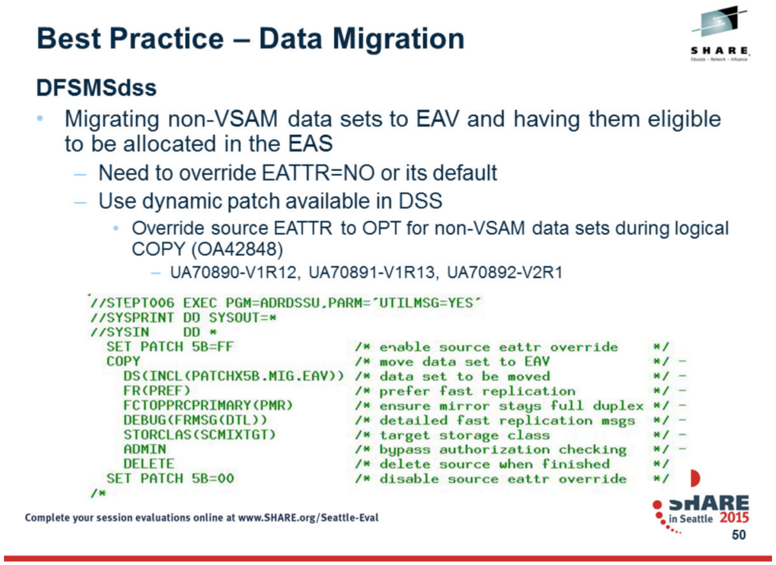
- DFSMSdss Consolidate command
- Performs data set extent consolidation, reducing the number of extents. It allows you to specify which data sets are to be included.
  - Attempts to relocate multiple non-contiguous data set extents into contiguous space within each managed-space
    - Reduces the number of extents of a data set as much as possible even when the entire data set cannot be reduced to one extent
  - Supports the same type of data set filtering as COPY, DUMP, and RESTORE functions.
- No need to DEFRAG volume to consolidate extents

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

A new command, CONSOLIDATE, will perform both extent consolidation and reduction and allow you to specify which data sets are to be processed. This command will try to reduce the number of extents of a data set as much as possible even when the entire data set cannot be reduced to one extent.

See *DFSMS Storage Administration Reference* (for DFSMSdfp, DFSMSdss, DFSMShsm) for command syntax.



## Best Practice – Data Migration

### DFSMSdss

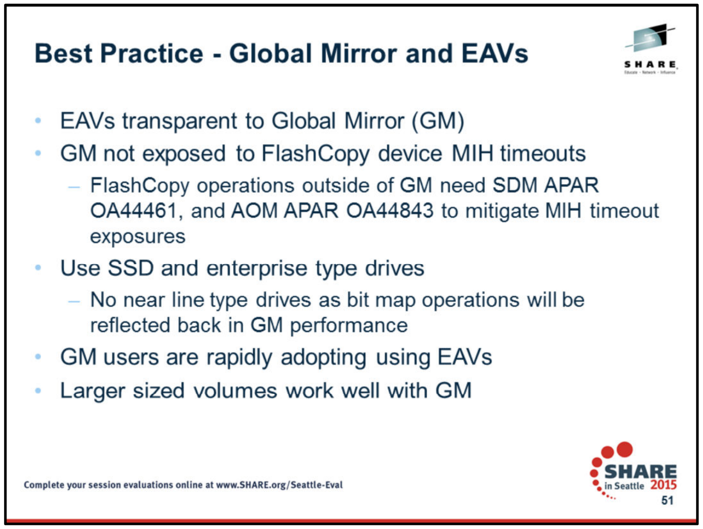
- Migrating non-VSAM data sets to EAV and having them eligible to be allocated in the EAS
  - Need to override EATTR=NO or its default
  - Use dynamic patch available in DSS
    - Override source EATTR to OPT for non-VSAM data sets during logical COPY (OA42848)
      - UA70890-V1R12, UA70891-V1R13, UA70892-V2R1

```
//STEPT006 EXEC PGM=ADRDSSU,PARM='UTILMSG=YES'
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  SET PATCH 5B=FF                   /* enable source eattr override    */
  COPY                              /* move data set to EAV            */ -
    DS(INCL(PATCHX5B.MIG.EAV))      /* data set to be moved            */ -
    FR(PREF)                        /* prefer fast replication         */ -
    FCTOPPRCPRIMARY(PMR)            /* ensure mirror stays full duplex */ -
    DEBUG(FRMSG(DTL))               /* detailed fast replication msgs  */ -
    STORCLAS(SCMIXTGT)              /* target storage class            */ -
    ADMIN                           /* bypass authorization checking   */ -
    DELETE                          /* delete source when finished     */
  SET PATCH 5B=00                   /* disable source eattr override   */
/*
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

50

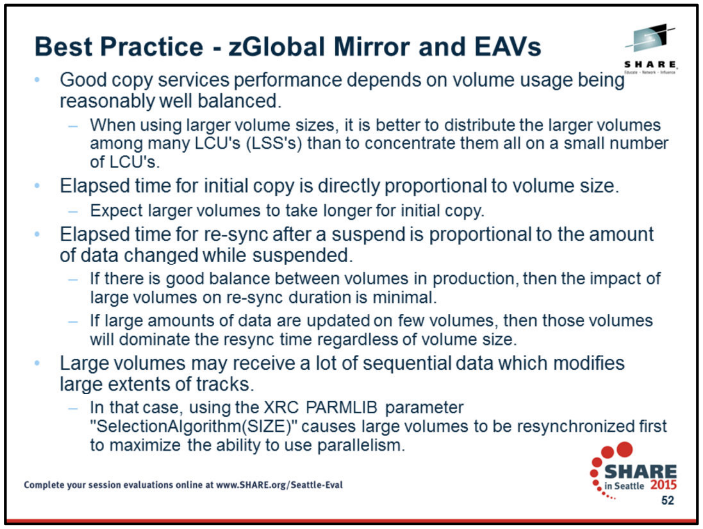


## Best Practice - Global Mirror and EAVs

- EAVs transparent to Global Mirror (GM)
- GM not exposed to FlashCopy device MIH timeouts
  - FlashCopy operations outside of GM need SDM APAR OA44461, and AOM APAR OA44843 to mitigate MIH timeout exposures
- Use SSD and enterprise type drives
  - No near line type drives as bit map operations will be reflected back in GM performance
- GM users are rapidly adopting using EAVs
- Larger sized volumes work well with GM

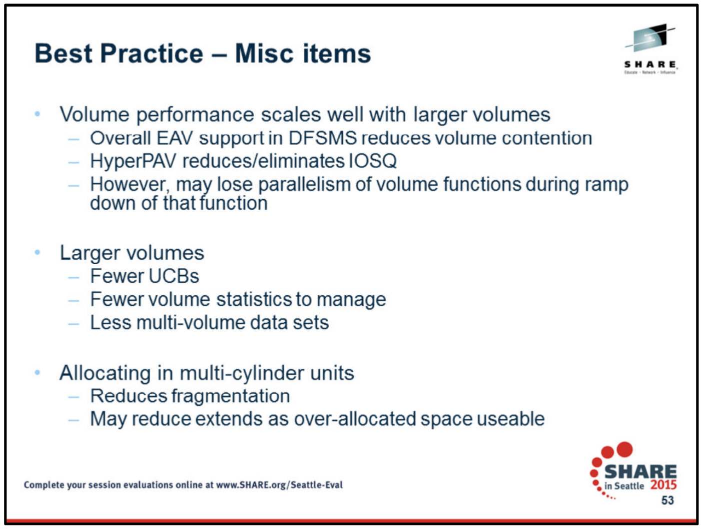
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015



## Best Practice - zGlobal Mirror and EAVs

- Good copy services performance depends on volume usage being reasonably well balanced.
  - When using larger volume sizes, it is better to distribute the larger volumes among many LCU's (LSS's) than to concentrate them all on a small number of LCU's.
- Elapsed time for initial copy is directly proportional to volume size.
  - Expect larger volumes to take longer for initial copy.
- Elapsed time for re-sync after a suspend is proportional to the amount of data changed while suspended.
  - If there is good balance between volumes in production, then the impact of large volumes on re-sync duration is minimal.
  - If large amounts of data are updated on few volumes, then those volumes will dominate the resync time regardless of volume size.
- Large volumes may receive a lot of sequential data which modifies large extents of tracks.
  - In that case, using the XRC PARMLIB parameter "SelectionAlgorithm(SIZE)" causes large volumes to be resynchronized first to maximize the ability to use parallelism.

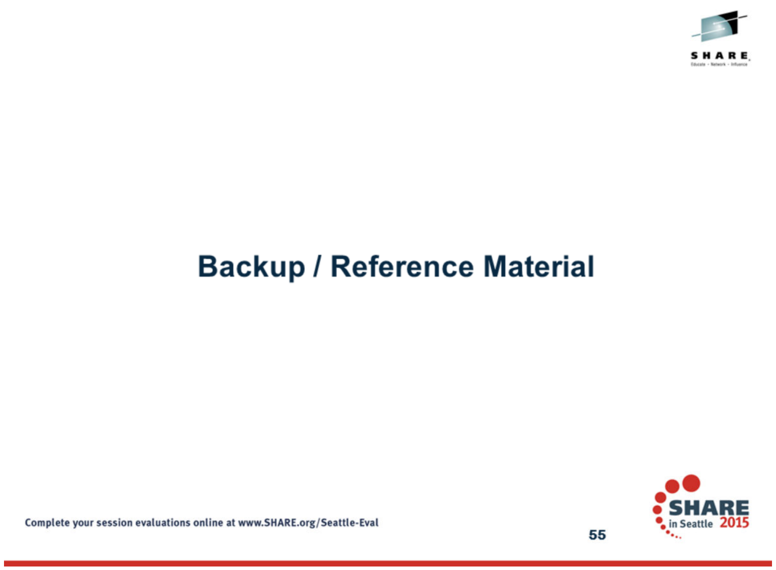


## Best Practice – Misc items

- Volume performance scales well with larger volumes
  - Overall EAV support in DFSMS reduces volume contention
  - HyperPAV reduces/eliminates IOSQ
  - However, may lose parallelism of volume functions during ramp down of that function
- Larger volumes
  - Fewer UCBs
  - Fewer volume statistics to manage
  - Less multi-volume data sets
- Allocating in multi-cylinder units
  - Reduces fragmentation
  - May reduce extends as over-allocated space useable

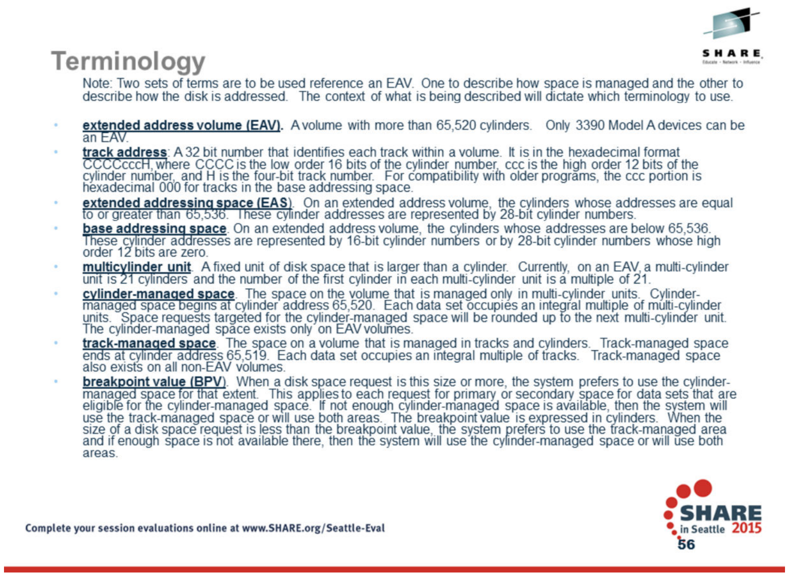
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

5
4



## Backup / Reference Material

The usage and invocation of the EAV support is discussed next.



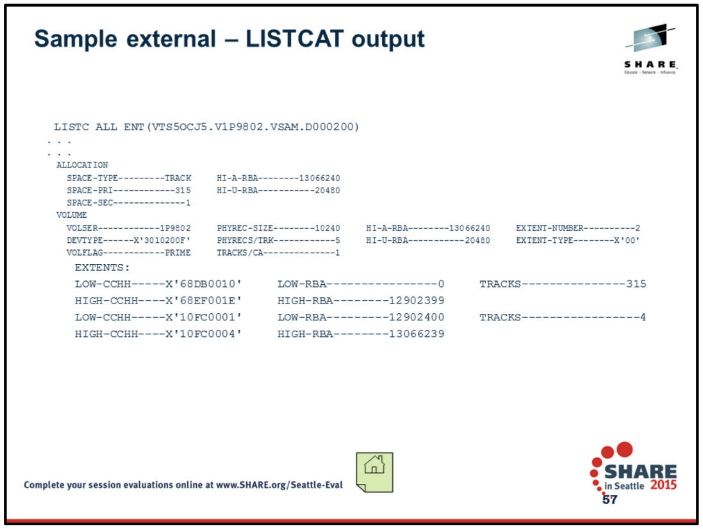
## Terminology

Note: Two sets of terms are to be used reference an EAV. One to describe how space is managed and the other to describe how the disk is addressed. The context of what is being described will dictate which terminology to use.

- **extended address volume (EAV).** A volume with more than 65,520 cylinders. Only 3390 Model A devices can be an EAV.
- **track address**: A 32 bit number that identifies each track within a volume. It is in the hexadecimal format CCCCcccH, where CCCC is the low order 16 bits of the cylinder number, ccc is the high order 12 bits of the cylinder number, and H is the four-bit track number. For compatibility with older programs, the ccc portion is hexadecimal 000 for tracks in the base addressing space.
- **extended addressing space (EAS)**. On an extended address volume, the cylinders whose addresses are equal to or greater than 65,536. These cylinder addresses are represented by 28-bit cylinder numbers.
- **base addressing space**. On an extended address volume, the cylinders whose addresses are below 65,536. These cylinder addresses are represented by 16-bit cylinder numbers or by 28-bit cylinder numbers whose high order 12 bits are zero.
- **multicylinder unit**. A fixed unit of disk space that is larger than a cylinder. Currently, on an EAV, a multi-cylinder unit is 21 cylinders and the number of the first cylinder in each multi-cylinder unit is a multiple of 21.
- **cylinder-managed space**. The space on the volume that is managed only in multi-cylinder units. Cylinder-managed space begins at cylinder address 65,520. Each data set occupies an integral multiple of multi-cylinder units. Space requests targeted for the cylinder-managed space will be rounded up to the next multi-cylinder unit. The cylinder-managed space exists only on EAV volumes.
- **track-managed space**. The space on a volume that is managed in tracks and cylinders. Track-managed space ends at cylinder address 65,519. Each data set occupies an integral multiple of tracks. Track-managed space also exists on all non-EAV volumes.
- **breakpoint value (BPV)**. When a disk space request is this size or more, the system prefers to use the cylinder-managed space for that extent. This applies to each request for primary or secondary space for data sets that are eligible for the cylinder-managed space. If not enough cylinder-managed space is available, then the system will use the track-managed space or will use both areas. The breakpoint value is expressed in cylinders. When the size of a disk space request is less than the breakpoint value, the system prefers to use the track-managed area and if enough space is not available there, then the system will use the cylinder-managed space or will use both areas.

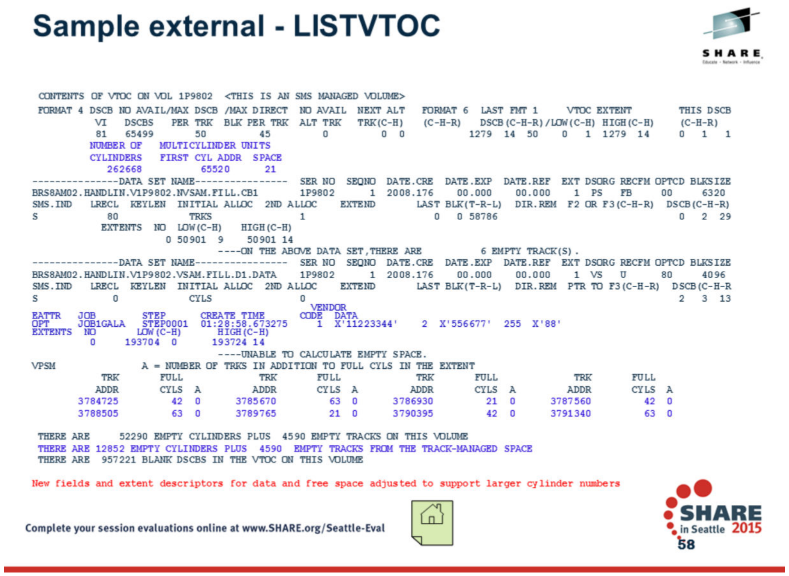
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

56



## Sample external – LISTCAT output

```
 LISTC ALL ENT(VTS5OCJ5.V1P9802.VSAM.D000200)
. . .
. . .
  ALLOCATION
    SPACE-TYPE---------TRACK     HI-A-RBA--------13066240
    SPACE-PRI------------315     HI-U-RBA-----------20480
    SPACE-SEC--------------1
  VOLUME
    VOLSER------------1P9802     PHYREC-SIZE--------10240     HI-A-RBA--------13066240     EXTENT-NUMBER----------2
    DEVTYPE------X'3010200F'     PHYRECS/TRK------------5     HI-U-RBA-----------20480     EXTENT-TYPE--------X'00'
    VOLFLAG------------PRIME     TRACKS/CA--------------1
     EXTENTS:
     LOW-CCHH-----X'68DB0010'     LOW-RBA----------------0     TRACKS---------------315
     HIGH-CCHH----X'68EF001E'     HIGH-RBA--------12902399
     LOW-CCHH-----X'10FC0001'     LOW-RBA---------12902400     TRACKS-----------------4
     HIGH-CCHH----X'10FC0004'     HIGH-RBA--------13066239
```



## Sample external - LISTVTOC

```
 CONTENTS OF VTOC ON VOL 1P9802  <THIS IS AN SMS MANAGED VOLUME>
 FORMAT 4 DSCB NO AVAIL/MAX DSCB /MAX DIRECT  NO AVAIL  NEXT ALT   FORMAT 6  LAST FMT 1     VTOC EXTENT        THIS DSCB
          VI    DSCBS    PER TRK  BLK PER TRK  ALT TRK   TRK(C-H)   (C-H-R)   DSCB(C-H-R)/LOW(C-H) HIGH(C-H)    (C-H-R)
          81    65499      50          45          0         0  0              1279  14  50     0   1  1279  14     0   1   1
         NUMBER OF   MULTICYLINDER UNITS
         CYLINDERS   FIRST CYL ADDR  SPACE
           262668        65520       21
---------------DATA SET NAME----------------  SER NO  SEQNO  DATE.CRE  DATE.EXP  DATE.REF  EXT DSORG RECFM OPTCD BLKSIZE
BRS8AM02.HANDLIN.V1P9802.NVSAM.FILL.CB1       1P9802      1  2008.176   00.000    00.000    1  PS    FB     00     6320
SMS.IND   LRECL  KEYLEN  INITIAL ALLOC  2ND ALLOC    EXTEND       LAST BLK(T-R-L)  DIR.REM  F2 OR F3(C-H-R)  DSCB(C-H-R)
S            80            TRKS             1                        0   0 58786                               0   2  29
            EXTENTS  NO  LOW(C-H)    HIGH(C-H)
                      0 50901  9     50901 14
                               ----ON THE ABOVE DATA SET,THERE ARE          6 EMPTY TRACK(S).
---------------DATA SET NAME----------------  SER NO  SEQNO  DATE.CRE  DATE.EXP  DATE.REF  EXT DSORG RECFM OPTCD BLKSIZE
BRS8AM02.HANDLIN.V1P9802.VSAM.FILL.D1.DATA    1P9802      1  2008.176   00.000    00.000    1  VS    U      80     4096
SMS.IND   LRECL  KEYLEN  INITIAL ALLOC  2ND ALLOC    EXTEND       LAST BLK(T-R-L)  DIR.REM  PTR TO F3(C-H-R)  DSCB(C-H-R
S            0             CYLS             0                                                                      2   3  13
                                             VENDOR
EATTR   JOB        STEP      CREATE TIME     CODE  DATA
OPT     JOB1GALA   STEP0001  01:28:58.673275   1   X'11223344'    2  X'556677'  255  X'88'
EXTENTS  NO       LOW(C-H)      HIGH(C-H)
         0     193704  0       193724 14
                               ----UNABLE TO CALCULATE EMPTY SPACE.
VPSM                A = NUMBER OF TRKS IN ADDITION TO FULL CYLS IN THE EXTENT
            TRK        FULL            TRK        FULL            TRK        FULL            TRK        FULL
           ADDR        CYLS  A        ADDR        CYLS  A        ADDR        CYLS  A        ADDR        CYLS  A
        3784725          42  0     3785670          63  0     3786930          21  0     3787560          42  0
        3788505          63  0     3789765          21  0     3790395          42  0     3791340          63  0

 THERE ARE     52290 EMPTY CYLINDERS PLUS  4590 EMPTY TRACKS ON THIS VOLUME
 THERE ARE 12852 EMPTY CYLINDERS PLUS  4590  EMPTY TRACKS FROM THE TRACK-MANAGED SPACE
 THERE ARE  957221 BLANK DSCBS IN THE VTOC ON THIS VOLUME
```

New fields and extent descriptors for data and free space adjusted to support larger cylinder numbers

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

IEHLIST LISTVTOC changes . . . Update programs or procedures that depend on the LISTVTOC output.

With the FORMAT option for volumes that support cylinder-managed space the format 4 DSCB display identifies the size of the volume, the location of the address of the cylinder where cylinder-managed space begins, and the allocation size of multiple cylinder unit that the system uses to allocate space in cylinder-managed space.

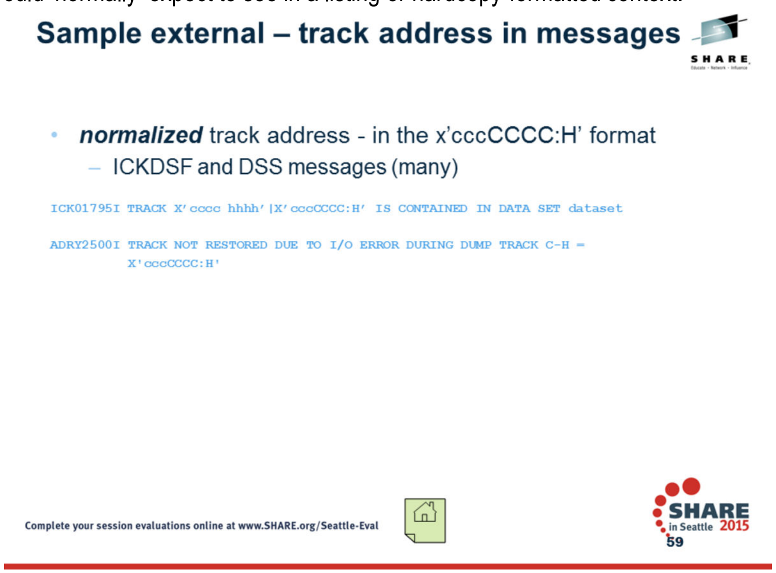
With the FORMAT option for data sets that support extended attribute DSCBs, the extent descriptions are adjusted to support larger cylinder numbers. With the DUMP option Format 9 DSCBs will be displayed.

With the FORMAT option in the VPSM report of free extents, the TRK ADDR and FULL CYLS columns are adjusted to support larger numbers for the track address and full cylinders when the volume is an EAV.

At the end of the DUMP and FORMAT report there is a listing of the number of empty cylinders and additional empty tracks for the track-managed space of volumes that support cylinder-managed space.

See Appendix D. IEHLIST VTOC Listing, in z/OS V1R10.0 DFSMSdfp Utilities for complete examples.

A normalized track address is what one would ‘normally’ expect to see in a listing or hardcopy-formatted context.



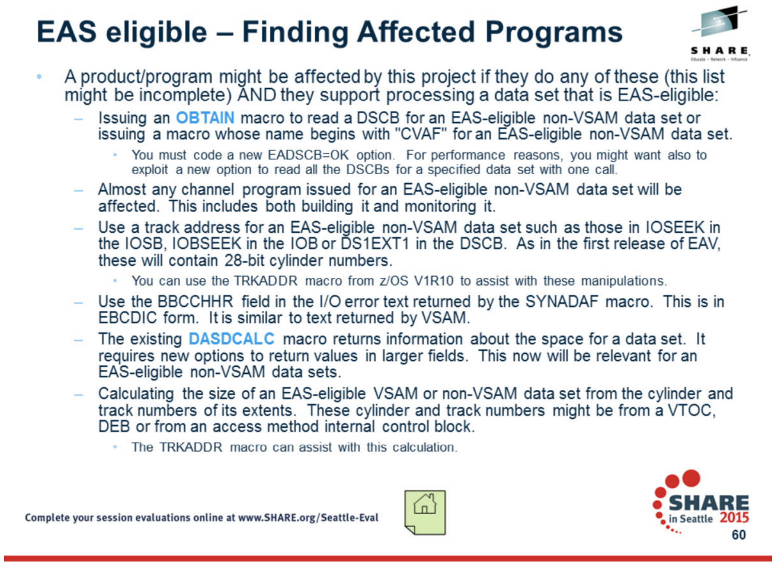
## Sample external – track address in messages

- ***normalized*** track address - in the x’cccCCCC:H’ format
  - ICKDSF and DSS messages (many)

```
ICK01795I TRACK X’cccc hhhh’|X’cccCCCC:H’ IS CONTAINED IN DATA SET dataset

ADRY2500I TRACK NOT RESTORED DUE TO I/O ERROR DURING DUMP TRACK C-H =
          X'cccCCCC:H'
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015
59



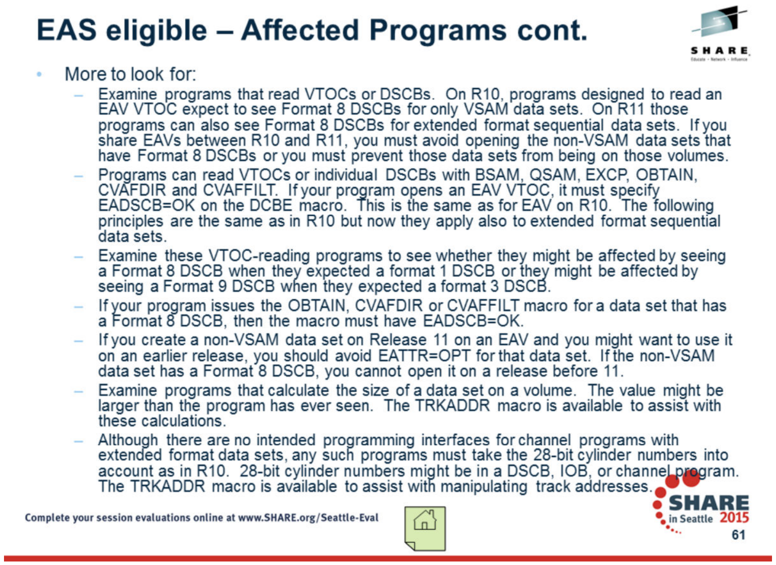
## EAS eligible – Finding Affected Programs

- A product/program might be affected by this project if they do any of these (this list might be incomplete) AND they support processing a data set that is EAS-eligible:
  - Issuing an **OBTAIN** macro to read a DSCB for an EAS-eligible non-VSAM data set or issuing a macro whose name begins with "CVAF" for an EAS-eligible non-VSAM data set.
    - You must code a new EADSCB=OK option. For performance reasons, you might want also to exploit a new option to read all the DSCBs for a specified data set with one call.
  - Almost any channel program issued for an EAS-eligible non-VSAM data set will be affected. This includes both building it and monitoring it.
  - Use a track address for an EAS-eligible non-VSAM data set such as those in IOSEEK in the IOSB, IOBSEEK in the IOB or DS1EXT1 in the DSCB. As in the first release of EAV, these will contain 28-bit cylinder numbers.
    - You can use the TRKADDR macro from z/OS V1R10 to assist with these manipulations.
  - Use the BBCCHHR field in the I/O error text returned by the SYNADAF macro. This is in EBCDIC form. It is similar to text returned by VSAM.
  - The existing **DASDCALC** macro returns information about the space for a data set. It requires new options to return values in larger fields. This now will be relevant for an EAS-eligible non-VSAM data sets.
  - Calculating the size of an EAS-eligible VSAM or non-VSAM data set from the cylinder and track numbers of its extents. These cylinder and track numbers might be from a VTOC, DEB or from an access method internal control block.
    - The TRKADDR macro can assist with this calculation.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

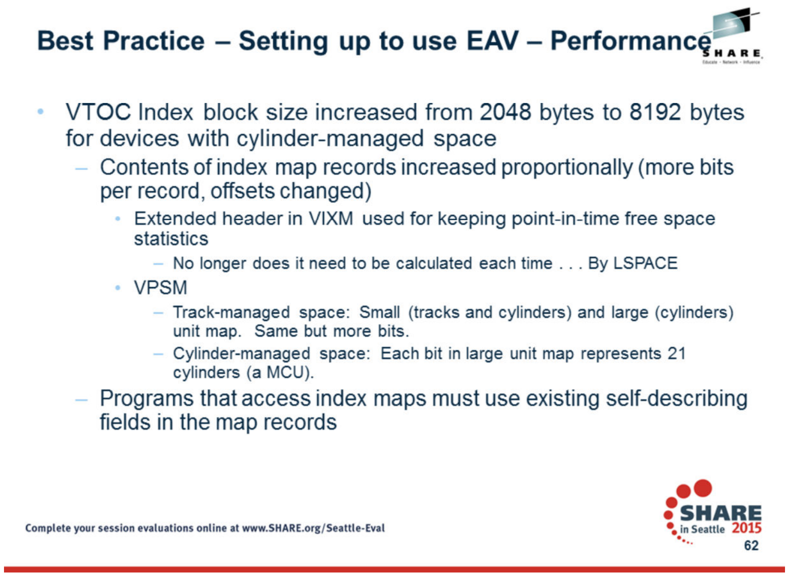
60



## EAS eligible – Affected Programs cont.

- More to look for:
  - Examine programs that read VTOCs or DSCBs. On R10, programs designed to read an EAV VTOC expect to see Format 8 DSCBs for only VSAM data sets. On R11 those programs can also see Format 8 DSCBs for extended format sequential data sets. If you share EAVs between R10 and R11, you must avoid opening the non-VSAM data sets that have Format 8 DSCBs or you must prevent those data sets from being on those volumes.
  - Programs can read VTOCs or individual DSCBs with BSAM, QSAM, EXCP, OBTAIN, CVAFDIR and CVAFFILT. If your program opens an EAV VTOC, it must specify EADSCB=OK on the DCBE macro. This is the same as for EAV on R10. The following principles are the same as in R10 but now they apply also to extended format sequential data sets.
  - Examine these VTOC-reading programs to see whether they might be affected by seeing a Format 8 DSCB when they expected a format 1 DSCB or they might be affected by seeing a Format 9 DSCB when they expected a format 3 DSCB.
  - If your program issues the OBTAIN, CVAFDIR or CVAFFILT macro for a data set that has a Format 8 DSCB, then the macro must have EADSCB=OK.
  - If you create a non-VSAM data set on Release 11 on an EAV and you might want to use it on an earlier release, you should avoid EATTR=OPT for that data set. If the non-VSAM data set has a Format 8 DSCB, you cannot open it on a release before 11.
  - Examine programs that calculate the size of a data set on a volume. The value might be larger than the program has ever seen. The TRKADDR macro is available to assist with these calculations.
  - Although there are no intended programming interfaces for channel programs with extended format data sets, any such programs must take the 28-bit cylinder numbers into account as in R10. 28-bit cylinder numbers might be in a DSCB, IOB, or channel program. The TRKADDR macro is available to assist with manipulating track addresses.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



## Best Practice – Setting up to use EAV – Performance

- VTOC Index block size increased from 2048 bytes to 8192 bytes for devices with cylinder-managed space
  - Contents of index map records increased proportionally (more bits per record, offsets changed)
    - Extended header in VIXM used for keeping point-in-time free space statistics
      - No longer does it need to be calculated each time . . . By LSPACE
    - VPSM
      - Track-managed space: Small (tracks and cylinders) and large (cylinders) unit map. Same but more bits.
      - Cylinder-managed space: Each bit in large unit map represents 21 cylinders (a MCU).
  - Programs that access index maps must use existing self-describing fields in the map records

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

62

This new block size will be recorded in the format 1 DSCB for the index and is necessary to allow for scaling to largest sized volumes

The VTOC index space map (VIXM) will have a new bit, VIMXHDRV, to indicate that new fields exist in the new VIXM extension.

The VIXM will contain a new field for the RBA of the new large unit map and new space statistics

'LSPACE' like free space statistics

The VIXM will contain a new field for the 'minimum allocation unit' in cylinders for the cylinder-managed space. Each extent in the cylinder-managed space must be a multiple of this on an EAV.

NOTE: If you omit the VTOC index size when formatting a volume with ICKDSF and do not preallocate the index, the default before this release has been 15 tracks. In EAV Release 1, the default size for EAV and non-EAV volumes will be calculated and may be different from earlier releases.

The small unit map (for tracks) will describe only the tracks in the first 65,520 cylinders (the track-managed space).

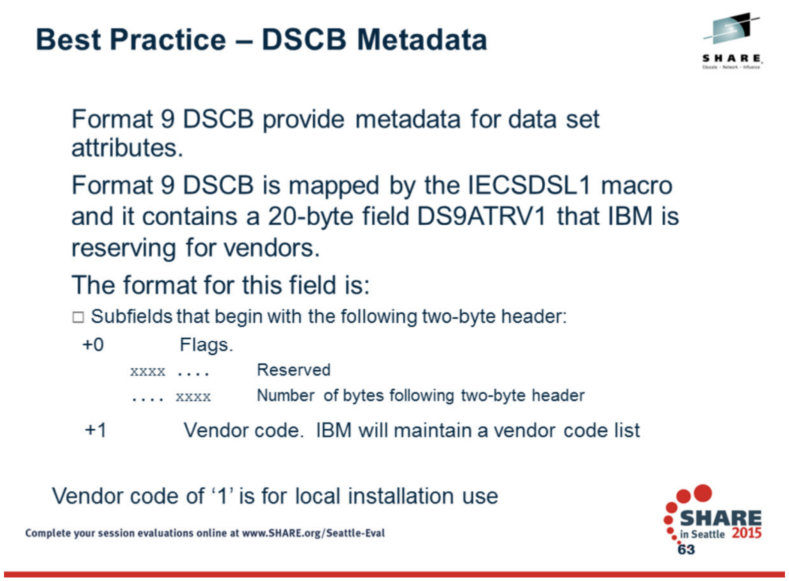
There will be two large unit maps (for cylinders)

The existing one will be for the first 65,520 cylinders.

A new one will be for cylinders after the first 65,520 cylinders (the cylinder-managed space). In the new map, each bit will represent 21 cylinders instead of each bit representing one cylinder.

A new field will define the number of cylinders per large unit map.

The system rounds the size of each extent that happens to be satisfied in the cylinder-managed space up to a multiple of 21-cylinders. Programs must allow for the possibility that any extent might be larger than requested.



## Best Practice – DSCB Metadata

Format 9 DSCB provide metadata for data set attributes.

Format 9 DSCB is mapped by the IECSDSL1 macro and it contains a 20-byte field DS9ATRV1 that IBM is reserving for vendors.

The format for this field is:

□ Subfields that begin with the following two-byte header:

```
+0        Flags.
      xxxx ....     Reserved
      .... xxxx     Number  of bytes following two-byte header
+1        Vendor code.  IBM will maintain a vendor code list
```

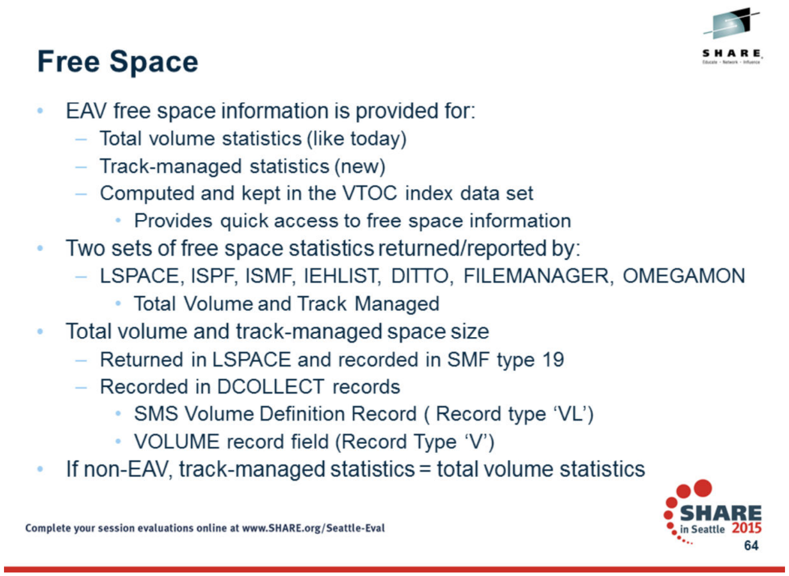
Vendor code of ‘1’ is for local installation use

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

63

Before z/OS 1.10, some vendor products have used fields in the format 1 DSCB that IBM reserved for IBM use. IBM makes rules on DSCB updates but does not want to enforce content rules here. IBM will recommend this format.



## Free Space

- EAV free space information is provided for:
  - Total volume statistics (like today)
  - Track-managed statistics (new)
  - Computed and kept in the VTOC index data set
    - Provides quick access to free space information
- Two sets of free space statistics returned/reported by:
  - LSPACE, ISPF, ISMF, IEHLIST, DITTO, FILEMANAGER, OMEGAMON
    - Total Volume and Track Managed
- Total volume and track-managed space size
  - Returned in LSPACE and recorded in SMF type 19
  - Recorded in DCOLLECT records
    - SMS Volume Definition Record ( Record type 'VL')
    - VOLUME record field (Record Type 'V')
- If non-EAV, track-managed statistics = total volume statistics

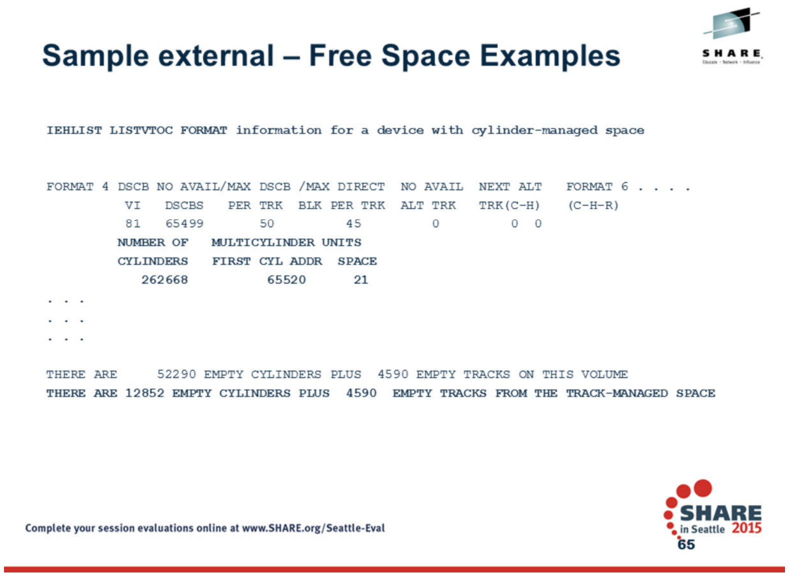
Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

64

Free space on the volume today is kept, reported and processed by the system on a total volume basis. For a device that supports cylinder-managed space, free space information will be kept, reported and processed by the system using the total volume free space statistics and also free space statistics pertaining to the track-managed space. These two sets of free space statistics will be maintained in the index data set for quick access from functions like LSPACE for volumes that have cylinder-managed space. Today's old slow methods to obtain free space information will be used for all others.

Since all volumes have track-managed space, free space statistics from track-managed space will be reported on all volumes. Track-managed space statistics will be the same as full volume statistics for volumes that do not support cylinder-managed space.



## Sample external – Free Space Examples

```
IEHLIST LISTVTOC FORMAT information for a device with cylinder-managed space


FORMAT 4 DSCB NO AVAIL/MAX DSCB /MAX DIRECT  NO AVAIL  NEXT ALT   FORMAT 6 . . . .
          VI   DSCBS   PER TRK  BLK PER TRK  ALT TRK   TRK(C-H)   (C-H-R)
          81   65499       50         45         0         0  0
         NUMBER OF   MULTICYLINDER UNITS
         CYLINDERS   FIRST CYL ADDR  SPACE
            262668          65520     21
. . .
. . .
. . .

THERE ARE     52290 EMPTY CYLINDERS PLUS  4590 EMPTY TRACKS ON THIS VOLUME
THERE ARE 12852 EMPTY CYLINDERS PLUS  4590  EMPTY TRACKS FROM THE TRACK-MANAGED SPACE
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

65

IEHLIST LISTVTOC changes . . .

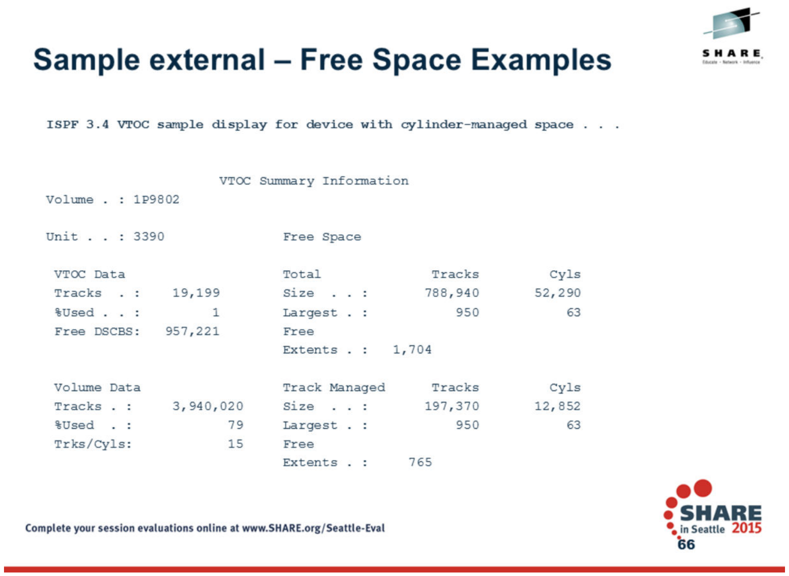
With the FORMAT option for volumes that support cylinder-managed space the format 4 DSCB display identifies the size of the volume, the location of the address of the cylinder where cylinder-managed space begins, and the allocation size of multiple cylinder unit that the system uses to allocate space in cylinder-managed space.

With the FORMAT option for data sets that support extended attribute DSCBs, the extent descriptions are adjusted to support larger cylinder numbers. With the DUMP option Format 9 DSCBs will be displayed.

With the FORMAT option in the VPSM report of free extents, the TRK ADDR and FULL CYLS columns are adjusted to support larger numbers for the track address and full cylinders when the volume is an EAV.

At the end of the DUMP and FORMAT report there is a listing of the number of empty cylinders and additional empty tracks for the track-managed space of volumes that support cylinder-managed space.

See Appendix D. IEHLIST VTOC Listing, in z/OS V1R10.0 DFSMSdfp Utilities for complete examples.



## Sample external – Free Space Examples

```
ISPF 3.4 VTOC sample display for device with cylinder-managed space . . .


                       VTOC Summary Information
Volume . : 1P9802

Unit . . : 3390               Free Space

 VTOC Data                    Total             Tracks        Cyls
 Tracks  . :     19,199       Size  . . :       788,940       52,290
 %Used . . :          1       Largest . :           950           63
 Free DSCBS:    957,221       Free
                              Extents . :    1,704

 Volume Data                  Track Managed     Tracks        Cyls
 Tracks . :     3,940,020     Size  . . :       197,370       12,852
 %Used  . :            79     Largest . :           950           63
 Trks/Cyls:            15     Free
                              Extents . :      765
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

A volume with cylinder-managed space will have this expanded ISPF 3.4 V display where track-managed free space is reported along with larger fields for volume data and free space for the entire volume. For a device that does not have cylinder-managed space, then the normal display will be presented.

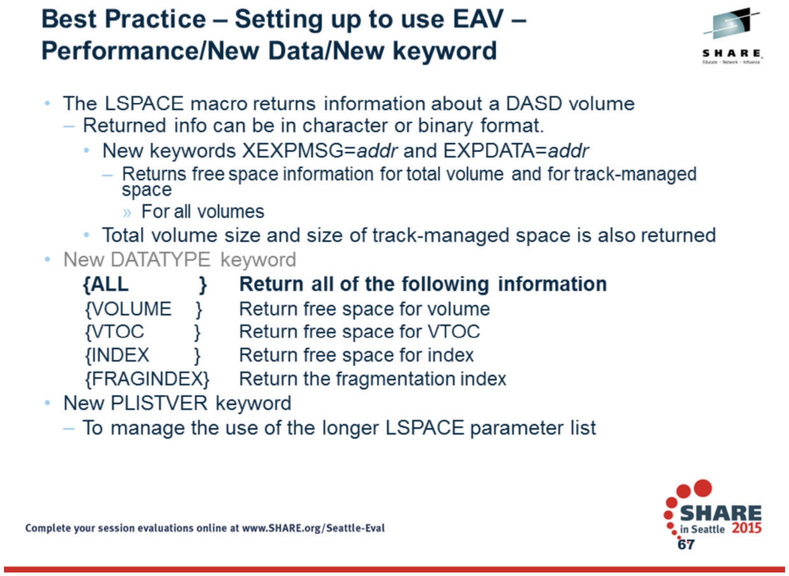
The display shows that the largest free extent is in track-managed space, even though cylinder-managed space has more total free space.

When processing a volume that does not have cylinder-managed space, the old format of the display will be shown.

Volume . : 1P9801

Unit . . : 3390

```
Volume Data        VTOC Data          Free Space   Tracks    Cyls
Tracks . :  982,800   Tracks  . :    4,800  Size  . . : 196,560  13,104
%Used  . :      80   %Used . . :        1  Largest . : 196,560  13,104
Trks/Cyls:      15   Free DSCBS:   239,980  Free
                              Extents . :      1
```



## Best Practice – Setting up to use EAV – Performance/New Data/New keyword

- The LSPACE macro returns information about a DASD volume
  - Returned info can be in character or binary format.
    - New keywords XEXPMSG=*addr* and EXPDATA=*addr*
      - Returns free space information for total volume and for track-managed space
        - » For all volumes
    - Total volume size and size of track-managed space is also returned
- New DATATYPE keyword

| | |
|---|---|
| **{ALL }** | **Return all of the following information** |
| {VOLUME } | Return free space for volume |
| {VTOC } | Return free space for VTOC |
| {INDEX } | Return free space for index |
| {FRAGINDEX} | Return the fragmentation index |

- New PLISTVER keyword
  - To manage the use of the longer LSPACE parameter list

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

67

You can use the LSPACE macro to get free space, volume fragmentation, and volume table of contents (VTOC) status information for a direct access storage device (DASD) volume. The LSPACE macro returns status information (such as LSPACE sub-function, return code, and reason code) in the parameter list. The LSPACE macro also returns the return code in register 15. For volumes that are configured with more than 9999 cylinders, you can use the EXPMSG option to create an expanded message return area that the LSPACE macro needs. For volumes that are configured with more than 65,520 cylinders, you can use the XEXPMSG option to create an extended expanded message return area that the LSPACE macro needs. The expanded data return area (EXPDATA) will return binary data of free space information for volumes with more than 65,520 cylinders. You can have LSPACE return additional information such as the format 4 DSCB, the total number of free extents on the volume or the fragmentation index.

Reference: *Hot Topics Newsletter*, March 2009, Issue 20, pp. 36-38, by Keith Cowden, Jan Redding, and William Smith. Article: *“LSPACE: Above and Beyond - New EAV Free Space Information and Options to Improve Performance”*

A new DATATYPE keyword is available to limit the amount of data and thus processing time to return LSPACE information. Any combination of the new keywords can be specified.

Details of the macro changes is in the *DFSMSdfp Advanced Services* documentation.

LSPACE reference material:
. The new keywords are described below.

XEXPMSG=addr or (reg) or 0

Specifies the address of a caller-provided 95-byte extended expanded message return area into which LSPACE returns either a free space message or, for unsuccessful requests, status information. Specify this keyword if you wish to obtain free space information in the message return area for volumes that are configured with more than 65,520 cylinders. The returned free space will include space for the total volume and space from the track-managed space on a volume. For volumes with 65,520 cylinders or less both sets of free space information will be returned but they will be the same.

EXPDATA=addr or (reg) or 0

Specifies the address of a caller-provided expanded data return area into which LSPACE returns expanded free space and volume information. Specify this keyword if you wish to obtain free space information in the LSPACE data return area for volumes that are configured with more than 65,520 cylinders. The returned free space will include space for the total volume and space from the track-managed space on a volume. For volumes with 65,520 cylinders or less both sets of free space information will be returned but they will be the same.

DATATYPE= (VOLUME,VTOC,INDEX,FRAGINDEX,ALL)

This keyword is only allowed when the DATA or EXPDATA keyword is specified. Only the information specified will be returned to the caller. DATATYPE is valid for both non-EAV and EAV. This keyword will eliminate unnecessary I/O required to retrieve free space information that is not be required by the caller. DATATYPE=ALL is the default.

VOLUME – Provide free space information for the VOLUME
VTOC – Provide free space information related to the VTOC
INDEX – Provide free space information related to the VTOC INDEX
FRAGINDEX – Provide the fragmentation index
ALL – Provide all available LSPACE statistics. This is the default

PLISTVER=plistver | IMPLIED_VERSION | MAX

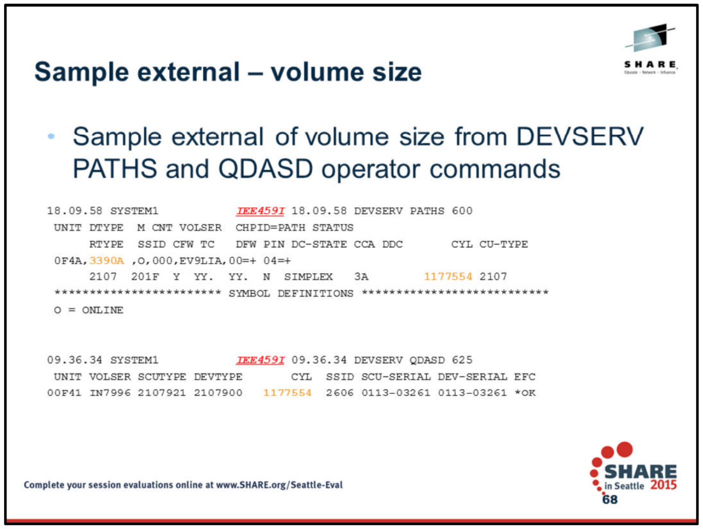
This keyword defines the version of the LSPACE parameter list that should be generated for the MF=L form of the LSPACE macro. The value for plistver is a byte input decimal value in the "1-2" range that specifies the version of the LSPACE parameter list that should be generated. The macro keys associated with each supported version of the macro are listed below. This PLISTVER= keyword is required for any macro keys associated with version 2 or larger to be specified:

| VERSION | KEY |
|---|---|
| 1 | MSG |
| | EXPMSG |
| | DATA |
| | SMF |
| | F4DSCB |
| 2 | XEXPMSG |
| | EXPDATA |
| | DATATYPE |

When MAX is specified, the generated parameter list is the largest size currently supported. This size may grow from release to release thus possibly affecting the amount of storage needed by your program. If your program can tolerate this, IBM recommends that you always specify MAX when creating the list form parameter list as that will ensure that the list form parameter list is always long enough to hold whatever parameters might be specified on the execute form.

When IMPLIED_VERSION is specified the generated parameter list is the lowest version which allows all of the parameters on the invocation to be processed.

When PLISTVER is omitted, the default is the lowest version of the parameter list, which is version 1.



## Sample external – volume size

- Sample external of volume size from DEVSERV PATHS and QDASD operator commands

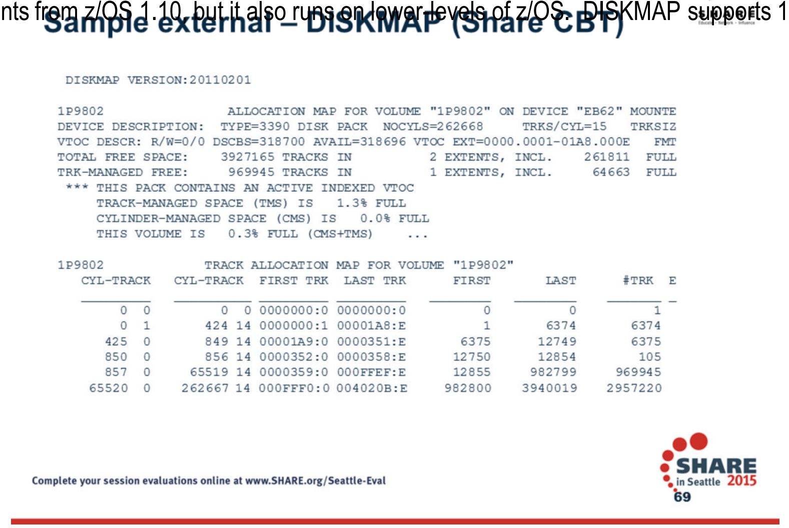
```
18.09.58 SYSTEM1           IEE459I 18.09.58 DEVSERV PATHS 600
 UNIT DTYPE  M CNT VOLSER  CHPID=PATH STATUS
      RTYPE  SSID CFW TC   DFW PIN DC-STATE CCA DDC       CYL CU-TYPE
 0F4A,3390A ,O,000,EV9LIA,00=+ 04=+
      2107  201F  Y  YY.  YY.  N  SIMPLEX   3A        1177554 2107
 ************************ SYMBOL DEFINITIONS ***************************
 O = ONLINE


09.36.34 SYSTEM1           IEE459I 09.36.34 DEVSERV QDASD 625
 UNIT VOLSER SCUTYPE DEVTYPE       CYL  SSID SCU-SERIAL DEV-SERIAL EFC
00F41 IN7996 2107921 2107900   1177554  2606 0113-03261 0113-03261 *OK
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

SHARE in Seattle 2015

68

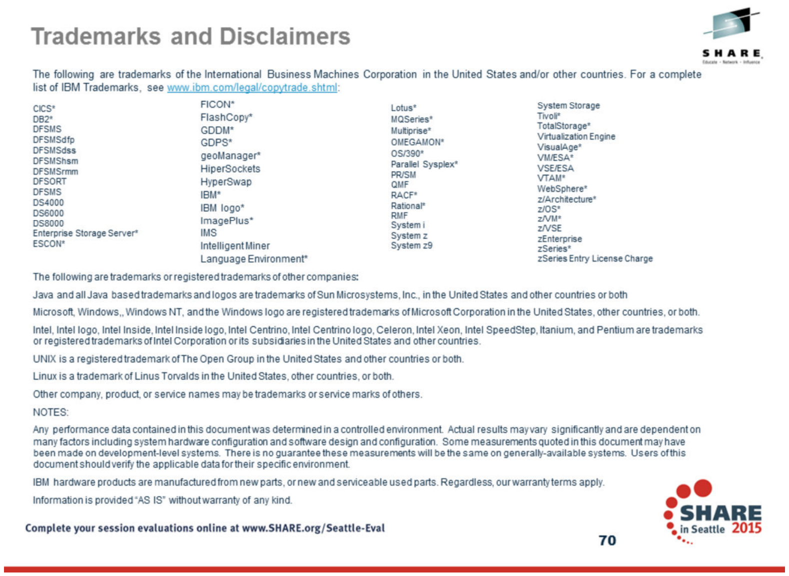
Contributed in the public domain by IBM DFSMS Device Support Development, the DISKMAP APF-authorized assembler source is available without support and at no-charge from the SHARE CBT web site http://www.cbttape.org. File 792 contains an updated copy of the DISKMAP program from File 260. This version of DISKMAP contains the EAV enhancements from z/OS 1.10, but it also runs on lower levels of z/OS. DISKMAP supports 1TB EAVs.



## Sample external – DISKMAP (Share CBT)

```
 DISKMAP VERSION:20110201

1P9802                 ALLOCATION MAP FOR VOLUME "1P9802" ON DEVICE "EB62" MOUNTE
DEVICE DESCRIPTION:  TYPE=3390 DISK PACK  NOCYLS=262668     TRKS/CYL=15   TRKSIZ
VTOC DESCR: R/W=0/0 DSCBS=318700 AVAIL=318696 VTOC EXT=0000.0001-01A8.000E   FMT
TOTAL FREE SPACE:    3927165 TRACKS IN          2 EXTENTS, INCL.    261811  FULL
TRK-MANAGED FREE:     969945 TRACKS IN          1 EXTENTS, INCL.     64663  FULL
 *** THIS PACK CONTAINS AN ACTIVE INDEXED VTOC
     TRACK-MANAGED SPACE (TMS) IS   1.3% FULL
     CYLINDER-MANAGED SPACE (CMS) IS   0.0% FULL
     THIS VOLUME IS   0.3% FULL (CMS+TMS)     ...

1P9802              TRACK ALLOCATION MAP FOR VOLUME "1P9802"
   CYL-TRACK   CYL-TRACK  FIRST TRK  LAST TRK      FIRST       LAST      #TRK  E

   _________   __________ _________ _________    ________   ________    _______ _
        0  0         0  0 0000000:0 0000000:0           0          0          1
        0  1       424 14 0000000:1 00001A8:E           1       6374       6374
      425  0       849 14 00001A9:0 0000351:E        6375      12749       6375
      850  0       856 14 0000352:0 0000358:E       12750      12854        105
      857  0     65519 14 0000359:0 000FFEF:E       12855     982799     969945
    65520  0    262667 14 000FFF0:0 004020B:E      982800    3940019    2957220
```

Complete your session evaluations online at www.SHARE.org/Seattle-Eval



## Trademarks and Disclaimers

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

CICS*
DB2*
DFSMS
DFSMSdfp
DFSMSdss
DFSMShsm
DFSMSrmm
DFSORT
DFSMS
DS4000
DS6000
DS8000
Enterprise Storage Server*
ESCON*

FICON*
FlashCopy*
GDDM*
GDPS*
geoManager*
HiperSockets
HyperSwap
IBM*
IBM logo*
ImagePlus*
IMS
Intelligent Miner
Language Environment*

Lotus*
MQSeries*
Multiprise*
OMEGAMON*
OS/390*
Parallel Sysplex*
PR/SM
QMF
RACF*
Rational*
RMF
System i
System z
System z9

System Storage
Tivoli*
TotalStorage*
Virtualization Engine
VisualAge*
VM/ESA*
VSE/ESA
VTAM*
WebSphere*
z/Architecture*
z/OS*
z/VM*
z/VSE
zEnterprise
zSeries*
zSeries Entry License Charge

The following are trademarks or registered trademarks of other companies:

Java and all Java based trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries or both

Microsoft, Windows,, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

NOTES:

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Users of this document should verify the applicable data for their specific environment.

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

Information is provided "AS IS" without warranty of any kind.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

70

## Trademarks and Disclaimers (continued)

NOTES:

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 are suggested US list prices and are subject to change without notice. Starting price may not include a hard drive, operating system or other features. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use.

The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM makes no representation or warranty regarding third-party products or services including those designated as ServerProven, ClusterProven ,or BladeCenter Interoperability Program products. Support for these third-party (non-IBM) products is provided by non-IBM Manufacturers.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

Complete your session evaluations online at www.SHARE.org/Seattle-Eval