

# Optimising CICS TS for VSE/ESA Storage

This document looks at the potential to increase 24-bit storage availability and how to size CICS TS partitions and their DSA sizes.

Mike Poil CICS Level 3 Change Team  
[poilmike@uk.ibm.com](mailto:poilmike@uk.ibm.com)

## Disclaimer:

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "as is" basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the operational environment. There is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

Optimizing the size of the z/VSE 24-bit Shared Area .....	2
Data Required .....	2
Methodology .....	2
Example without using IOBUF31 .....	2
Before.....	2
What do I adjust? .....	3
After .....	3
Benefits .....	3
Side effects.....	4
Using VTAM IOBUF31 .....	4
Other ways to reduce 24-bit usage.....	5
Do you really need a Dynamic Space Getvis Area that Big? .....	5
Do you really need to use LE ALL31(OFF)? .....	5
Resource Definition Parameters for 31-bit Code.....	5
CICS Partition Layout and Storage Usage.....	6
How big should a CICS partition be? .....	7
Example of sizing a Partition and EDSALIM .....	7
Setting DSALIM.....	8

# Optimising CICS TS for VSE/ESA Storage

## Optimizing the size of the z/VSE 24-bit Shared Area

Consider moving to z/VSE 4.3.

### Data Required

1. Supervisor and SVA statements from the z/VSE IPL procedure
2. MAP output
3. LIBR LD SDL output
4. GETVIS SVA output (after z/VSE has been running for some time to get an accurate maximum used figure, the same for (5))
5. Output from VTAM D NET,BFRUSE and D NET,VTAMOPTS

### Methodology

Calculate the amount of free storage that is shown by items (2) to (4) above, and if this adds up to more than 1MB, it should be possible to reduce the IPL command SVA PSIZE and GETVIS parameters to achieve the 1MB reduction. Always verify the results afterwards and monitor System Getvis usage.

Item (5) assesses the possible saving by using VTAM IOBUF31 support to move IOBUF out of the System Getvis-24. This is not as simple, as it will involve testing, and VSE support should be involved to check for any recommended PTFs.

### Example without using IOBUF31

#### Before

```
009,$A$SUPI,VIO=512K,VPOOL=64K,NOPDS,LOG
GETVIS=(2M,6M),PSIZE=(652K,7M)
```

```
MAP
SPACE AREA      V-SIZE   GETVIS   V-ADDR   UNUSED NAME
S  SUP          760K     2188K    0        $A$SUPX
S  SVA-24       1916K     2188K    BE000    384K     ← 384K-64K to be reclaimed
0  BG V        1536K     10752K   600000   249856K ← shared area 6MB
. . .

SDL      TOTAL ENTRIES :    908  (100%)
        USED ENTRIES  :    584  ( 64%)
        FREE ENTRIES  :    324  ( 36%)

SVA(24)  TOTAL SPACE   :   1852K (100%)
        USED SPACE    :   1603K ( 87%)
        - PFXED AREA:    166K (  9%)  START AT: 00273540
        FREE SPACE    :    249K ( 13%)  ← available to be reclaimed
. . .

GETVIS USAGE  SVA-24   SVA-ANY
AREA SIZE:    2,912K   9,704K
USED AREA:    1,516K   5,544K MAX. EVER USED:  1,736K   5,968K
FREE AREA:    1,396K   4,160K LARGEST FREE:    1,320K   2,716K
```

1. VPOOL=64K – this is already optimised, it is also the default.
2. MAP shows SVA-24 UNUSED = 384K.
3. LD SDL Virtual Library SVA(24) FREE SPACE = 249K.

## Optimising CICS TS for VSE/ESA Storage

4. GETVIS SVA AREA SIZE – MAX. EVER USED 2,912K - 1,736K = 1,176K.

The sum of (1) to (3) is 1,809K, which is greater than 1MB, but less than 2MB.

### What do I adjust?

How do you decide what to reclaim? The order would normally be:

1. Unused SVA-24, leaving at least 64K unused for expansion.
2. Unused Virtual Library, leaving at least 64K unused.
3. Unused SVA Getvis-24, leaving perhaps 320K unused.

System Getvis is the most important to have. How close you run the maximum System Getvis usage to the limit is for you to decide. Once loaded, the Virtual Library size remains static unless something is re-loaded without an IPL. I would also remove slightly more than 1MB as IPL rounding sometimes has an unexpected affect. You may need several goes to get it exactly as you want it.

To keep it simple by only changing one value, I decided to use 320K from the unused SVA-24 and reduce the System Getvis by 896K, which is a reduction of 1,216K.

### After

GETVIS=(1152K,6M),PSIZE=(652K,7M)

```
MAP
SPACE AREA      V-SIZE  GETVIS  V-ADDR  UNUSED NAME
S   SUP         760K      0        0        $$A$SUPX
S   SVA-24      1916K    2188K   BE000    256K ← still too large
0   BG V        1536K   10752K  500000   249856K ← shared area now 5MB

GETVIS USAGE    SVA-24      SVA-ANY      SVA-24      SVA-ANY
AREA SIZE:      2,016K     98,808K
USED AREA:      1,508K     5,368K MAX. EVER USED:  1,648K     5,578K
FREE AREA:       508K     3,440K LARGEST FREE:    492K     2,928K
```

This gave me 368K of System Getvis, which will be OK if that is the real high-water mark. I could increase the System Getvis by another 192K by factoring in the unused SVA-24. I could even swap some of the Virtual Library 249K for System Getvis.

### Benefits

You now have an improved CICS 24-bit Partition Getvis size. You can now choose whether or not to increase the DSALIM size.

# Optimising CICS TS for VSE/ESA Storage

## Side effects

Partitions that have an ALLOC value that now results in only having a few MB of 31-bit storage will find that they now have 1MB less, and that could cause side effects, e.g. TCP/IP warning message about having control blocks in 24-bit storage. In which case increase their ALLOC by at least 1MB.

## Using VTAM IOBUF31

D NET,BFRUSE showed:

```
IST350I DISPLAY TYPE = BUFFER POOL DATA
...
IST790I MAXIMUM SGA USED = 2592K
IST449I SGA24 LIMIT = NO LIMIT, CURRENT = 301K, MAXIMUM = 314K
IST790I MAXIMUM SGA24 USED = 314K
```

D NET,VTAMOPTS showed:

```
IST1189I INITDB = ***NA*** IOBUF31 = NO
```

We have something less than 314K that could be moved out of System Getvis. Adding that to previous 1,809K gives approximately 2,100K, which too close to 2MB to be able to safely reduce the shared area size down to 4MB.

## Optimising CICS TS for VSE/ESA Storage

### Other ways to reduce 24-bit usage

#### Do you really need a Dynamic Space Getvis Area that Big?

```
getvis g1
AR 0015 GETVIS USAGE      G1-24      G1-ANY      G1-24      G1-ANY
AR 0015 AREA SIZE:       10,236K    39,932K
AR 0015 USED AREA:       5,596K    28,800K MAX. EVER USED:  5,628K    28,976K
AR 0015 FREE AREA:       4,640K    11,132K LARGEST FREE:     4,608K    10,956K
AR 0015 DYNAMIC-SPACE GETVIS USAGE
AR 0015 AREA SIZE:       1,024K
AR 0015 USED AREA:       84K          MAX. EVER USED:     84K
AR 0015 FREE AREA:       940K          LARGEST FREE:       940K
AR 0015 1I40I  READY
```

#### Do you really need to use LE ALL31(OFF)?

You may be able to run with ALL31(ON) and let LE figure out what is 24-bit and what is 31-bit and handle it appropriately. EXEC CICS LINK is not an issue as it starts a new LE enclave.

If you need ALL31(OFF) in some partitions, compile a private CEECOPT for the other partitions, and link-edit it to an appropriate sublibrary that is only accessible to them. Set ALL31(ON), and ensure that STACK(4096,4080,ANYWHERE,KEEP) is set. STACK has more of an effect on C and PL/I than on COBOL, as COBOL uses HEAP storage for its DSAs. IBM and OEM software may be written in C, for example, WebSphere MQ uses C.

### Resource Definition Parameters for 31-bit Code

Transaction TASKDATALOC(ANY), program DATALOCATION(ANY).

# Optimising CICS TS for VSE/ESA Storage

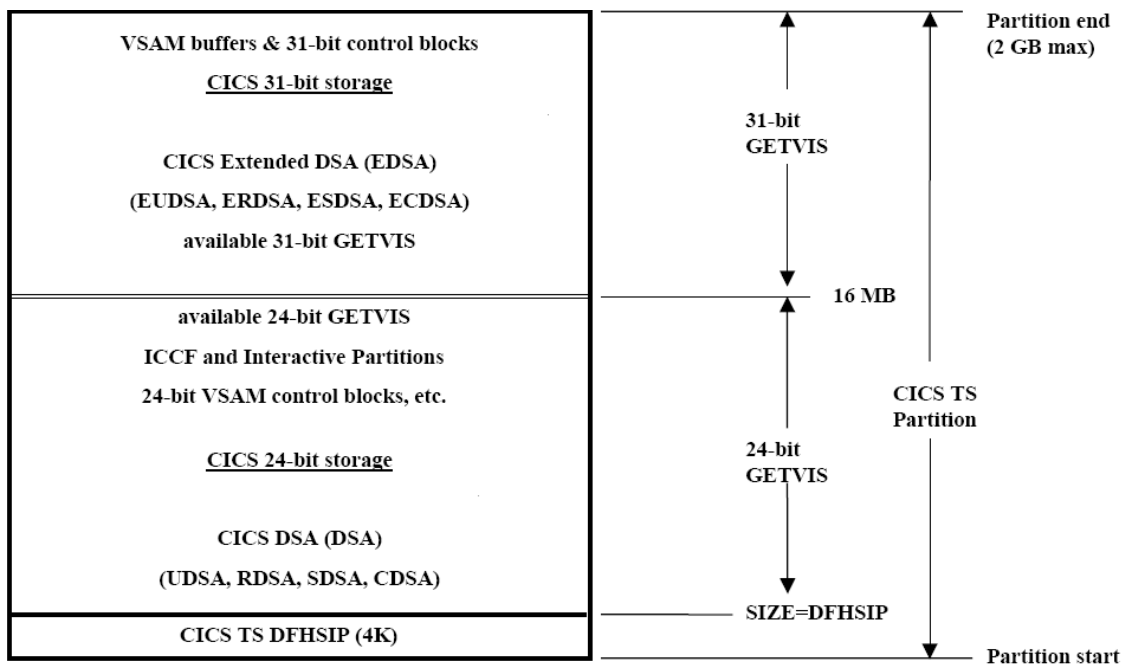
## CICS Partition Layout and Storage Usage

The CICS // EXEC DFHSIP,SIZE=DFHSIP means that the Program Area at the start of the partition is, and should always be 4K in size, which means that the remainder of the partition ALLOC is Getvis storage.

CICS allocates some Getvis storage and then allocates the remainder of the 24-bit storage, maps the SIT DSALIM value and frees the excess. Whatever is left is then available for things such as VSAM 24-bit control blocks e.g. the ACB for an open FCT entry. As the whole of the 24-bit Getvis area is used, even if temporarily, the GETVIS operator command shows the GETVIS-24 high-water-mark as being equal to the AREA SIZE.

Always issue the GETVIS xx,RESET command after CICS has started to be able to accurately calculate how much Getvis-24 has actually been used.

The SIT EDSALIM value is allocated out the Getvis-31 area, as is the CICS trace table, as are VSAM buffers and 31-bit control blocks.



The ICCF areas are only allocated when ICCF is started under CICS.

# Optimising CICS TS for VSE/ESA Storage

## How big should a CICS partition be?

The minimum allocation size comprises the amount of storage below the line (fixed), plus the amount that is required above the line, plus a safety factor. The amount of storage required above the line is partly a function of EDSALIM. As always, ensure that CICS has been running at a known peak time before producing data, and look at several executions.

1. Print CICS statistics output and look at the Summary report "Global Statistics", set EDSALIM to a minimum of the largest "Peak EDSA total" + safety factor; alternatively use the STAT transaction.
2. Run CICS in partition "xx" with the new DSALIMs, and use the GETVIS xx command to display the Getvis usage.
3. The partition size should be something like 8MB greater than the MAX. EVER USED xx-ANY.

## Example of sizing a Partition and EDSALIM

Based on STAT, which includes the equivalent of the GETVIS xx command:

```
Partition size established from ALLOC parameter . . : 393,215K (actually 384MB)
Storage BELOW 16MB
-----
Partition GETVIS area size under 16 Mb . . . . . : 11,260K (11MB below 16MB)
Partition GETVIS used area below 16 Mb . . . . . : 9,556K
Partition GETVIS free area below 16 Mb . . . . . : 1,704K
Partition GETVIS maximum used below 16 Mb . . . . : 11,260K (no GETVIS ,RESET)
Partition GETVIS largest free area below 16 Mb . : 1,704K
-----
Current DSA Limit . . . . . : 8,192K
Current Allocation for DSAs . . : 6,912K
Peak Allocation for DSAs . . . : 6,912K
. . .
Storage ABOVE 16MB
-----
Partition GETVIS area size above 16 Mb . . . . . : 393,204K
Partition GETVIS used area above 16 Mb . . . . . : 380,104K
Partition GETVIS free area above 16Mb . . . . . : 13,100K
Partition GETVIS maximum used above 16 Mb . . . . : 382,668K
Partition GETVIS largest free area above 16 Mb . : 12,968K
-----
Current EDSA Limit . . . . . : 357,376K (349MB)
CICS Trace table size . . . . . : 256K (use at least 4MB)
Current Allocation for EDSAs . : 49,152K (48MB)
Peak Allocation for EDSAs . . : 49,152K
```

We have used almost all of the 384MB. Can we reduce it? Yes.

1. EDSALIM could be reduced from 349MB to 52MB.
2. Increase the trace table to at least 4MB (CICS level 2 or 3 will always ask for at least this amount to be used for debugging), adding 4 MB.
3. The total saving is 293MB.
4. Conversely, it could show that you are running too close to the limit.

You could reduce the partition allocation to about 96MB, or you could leave it alone - if it is not broken . . .

## **Optimising CICS TS for VSE/ESA Storage**

### **Setting DSALIM**

Based on the same data as collected above. Set DSALIM to the reported maximum plus at least 512K, which allows for two additional DSA component area expansion increments.

The above example shows that a maximum of 6.75MB out of 8MB was used. Adjust this if you need to obtain more Getvis-24. As GETVIS xx,RESET was not used, it is impossible to see what the maximum usage really was.

EDSALIM grows in 1MB increments not 256K.