

**More CICS/TS Problem
Determination and Tips – New
ideas from CICS Level 2 Support**

**Bob Sodan
sodanrob@us.ibm.com**

April 2010

AGENDA

- Introduction
- CICS Support Center update
- CICS Level 2 process
- Problem determination and lookup - site information
- Current known problems and solutions
- Debugging techniques
 - Abends
 - Loops
 - CICS waits
 - Hangs
- Storage tuning hints and tips

Problem determination and lookup - site information

- CICS TS for VSE/ESA MustGathers
- <http://www.ibm.com/support/docview.wss?uid=swg21328819>
- CICS_Transaction_Server. To get to the MustGathers from the Support Portal you click on the "Troubleshooting" in the the left nav and then either "MustGather: Read first" or "Collect Troubleshooting data" in the Featured Troubleshooting links portlet.

MustGather: Read first when collecting troubleshooting data for CICS products

Technote (troubleshooting)

Problem(Abstract)

MustGather documents aid in problem determination and save time resolving problem management records (PMRs). These documents contain a list of the documentation you should gather so CICS® Support can diagnose your specific problem. These documents also include diagnostic tips that will aid in diagnosing and solving problems.

Cause

- Collecting MustGather data early, even before opening a PMR, helps IBM® Support quickly determine if: Symptoms match known problems (rediscovery).
- There is a non-defect problem that can be identified and resolved.
- There is a defect that identifies a workaround to reduce severity.
- Locating root cause can speed development of a code fix.

Resolving the problem

1. Gather the following general information for every problem record: A complete description of the problem, including the following:

1. When did the problem first occur?
2. Is the problem a one time failure or reoccurring?
3. Was software or hardware maintenance applied?
4. Did the failure occur while doing a specific task?
5. Is the failure occurring in more than one address space?

2. CICS product version, release, and maintenance level

3. Operating system version, release, and maintenance level

4. Related products version, and release levels

5. A valid contact phone number and email address



Click on the problem type or component that best describes your CICS problem. This will provide you with a listing of the documentation that the support team requires to diagnose your problem. It might also include diagnostic hints and tips that will aid in diagnosing and solving problems.

[File control wait](#)

[Multiregion operation \(MRO\)](#)

[Program check or abend](#)

[Short on storage](#)

[Storage violation](#)

[Terminal hang](#)

[Wait or loop](#)

Required doc:

CICS message log and the VSE system log.

[CICS Internal Trace](#) that is included in the VSE system dump when tracing is active. The trace should be at least 4096K and when possible level 1 tracing should be on for all CICS components and level 1-2 for the FC component.

A [VSE system dump](#) of the CICS region taken as soon as you notice the wait. Use the following VSE commands to capture the dump (where ***Fa*** is the CICS partition that you want to dump and ***cuu*** is the tape address where the dump will go):

SUSPEND *Fa* - temporarily take the partition out of the dispatch queue to prevent inconsistencies in the dump

[DUMP *Fa,0-7FFFFFFF,cuu*](#) - include the partition, SUP, and SVA in a single dump

RESUME *Fa* - put the partition back on the dispatch queue

(note that SUSPEND *Fa* and RESUME *Fa* are undocumented AR commands)

Optional doc: If feasible, save off the dataset by using [IDCAMS ALTER NEWNAME](#) to rename the dataset and save it in place. If this is not feasible, run [IDCAMS PRINT with SKIP](#) against the dataset and any alternate indexes associated with the dataset before continuing to see if the file is okay.

If you are able to recreate the problem, consider using [CICS Auxiliary Trace](#) in combination with the VSE system dump. The dump is unlikely to tell you anything about system activity in the period leading up to the wait or loop. This is because the trace table will probably wrap before you have had a chance to respond.

See [Exchanging information with IBM Technical Support](#) for FTP and e-mail instructions using the IBM Enhanced Customer Data Repository (ECuRep).

Go to [ServiceLink or IBMLink](#) to open an Electronic Technical Response (ETR). If you need instructions, see [open new PMRs or convert existing PMRs to electronic PMRs](#) using IBMLink ETR.

If you need to speak to an IBM technical support representative call your country representative. If you need to speak to an IBM technical support representative in the US call 1-800-IBM-SERV.

Always update the PMR to indicate that data has been sent.

Diagnostic tips: Review the logs and dumps generated at the point of failure. Use DFHPD410 dump formatter to format domains KE=3, TR=3, DS=1, and LM=3.

Search the [CICS support site](#) for known problems using symptoms like the message number and error codes.

If you find a fixing PTF, see [Ordering CICS products and maintenance](#) for the options that are available to order CICS maintenance.

If the save area addresses are not in application code, gather the documentation and work with the CICS Level2 support team to resolve your problem.

This section provides you with information that can help you learn more about this topic. [File control waits](#)
[CICS TS for VSE/ESA Library](#)

Problem determination and lookup - site information

CICS Support on Twitter

<http://www.ibm.com/support/docview.wss?uid=swg21384915>

Problem determination and lookup - site information

Support Portal - The new URL is

http://www.ibm.com/support/entry/portal/Overview/Software/Other_Software/

CICS Fix Lists – Known Problems

Last year (in February) Fix lists for CICS/VSE V2.3 and CICS TS for VSE/ESA were created.

Here's the link to the Fixes by version document that links to all the CICS Fix lists:

<http://www.ibm.com/support/docview.wss?uid=swg27008833>.



Fix list for CICS Transaction Server for VSE/ESA V1.1.1

Product documentation

Abstract

This document contains a table of APARs for CICS Transaction Server for VSE/ESA (CICS TS) V1.1.1. A comma separated values (CSV) file containing all the fields for each APAR is also included as an attachment. You can use the CSV file to import the list of APARs into a spreadsheet or database for sorting or downloading.

Content

[Fix list table](#)

[CSV file](#)

IBM updates this Fix list every week if there is a change to the APARs for this product. Open APARs are at the top of the Fix list table followed by closed APARs, ordered by close date. There might be a few days after the APAR closes before the table and CSV file are updated with the close date and PTF(s).

Click on an APAR number in the Fix list table to display the APAR. You will need to enter your IBM ID and password when displaying open APARs. There are no links to APARs closed prior to 1999 because they are not available on the Web. However, you can still order the PTFs for these APARs.

[PM06951](#)

DFHSM0002 CODE X'0D0A' DUE TO DOUBLE FREEMAIN OF TIOA FROM DFHZFRE . TIOA WAS ALREADY FREEMAINED BY DFHZRVS .

[PM03817](#)

10_M [UK55](#)
 ar [073](#)
 2010

DFHXS1111 & DFHAC2003 FOR TRANSACTIONS STARTED OVER ISC SESSIONS

[PK95485](#)

8_De [UK52](#)
 c [719](#)
 2009

C PRECOMPILER DFHEDP1\$ DOES NOT INSERT A CONTINUATION CHARACTER IF IN COLUMN 72 IS A SPACE. I

CICS Problem Determination

ABENDS

SCENARIO:

Running normally and all of a sudden the CICS partition crashes.

MESSAGES:

DFHSR0615 Program interrupt has occurred in recovery task

DFHDU0201 ABOUT TO TAKE SDUMP. DUMPCODE: SR0615 ,
DUMPID: 1/0002

0S24I AN SDUMP OR SDUMPX MACRO WAS ISSUED 28/04-15:43:47

0S29I DUMP STARTED 28/04-15:43:47

0S30I DUMP STARTED. MEMBER=DF600034.DUMP IN SUBLIB=SYSDU 28/04-
15:43:47

PTF Availability not on RSU

Fix information

Fixed component name

CICSTS FOR VSE

Fixed component ID

564805400

Applicable component levels

RB0P PSY [UK55073](#)

UP10/03/11 I 1000 <<<<< not on a RSU but orderable

Fix is available

PTF Availability on a RSU

Fix information

Fixed component name

CICSTS FOR VSE

Fixed component ID

564805400

Applicable component levels

RB0P PSY [UK22082](#)

UP07/02/14 P E411 <<<< available on the RSU 411 level

Fix is available

CICS Problem Determination

ABENDS Example 1

DFHSR0615 applid Program interrupt has occurred in recovery task

Explanation: An operating system abnormal termination occurred. CICS started to abend the task with an abend code of ASRB when a program check occurred. DFHSRP terminates CICS.

System Action: CICS abnormally terminates with system dump SR0615 and exception trace entries giving the kernel error data for the operating system abend and the program check.

User Response: The most likely cause of the program check is an error in a global user exit program running at the XSRAB exit. This is the global user exit that can be invoked when an abend code is found in the SRT. If such a program was running, determine the cause of the program check and correct it.

CICS Problem Determination

ABENDS Example 1

We issue: CALL DFHPD410 DATA KE=3 in the Infoana job to see what the running task is doing:

```
00A2 046CB080 ***Running** 007E6680 00055 XXXX 09A13380 09B51020
*YES*
```

We then display this task's stack entry to see what the module flow is:

```
00A2 046CC020 0120 Bot 89B8ABE8 89B8AEB8 02D0 DFHKETA
00A2 046CC330 03F0 Dom 89C2A898 89C2B396 0AFE DFHXMTA
00A2 046CC720 03E0 Dom 841400C8 8414218C 20C4 DFHPGPG
      Int +00D4 84140156 008E INITIAL_LINK
      Int +1A06 841404A6 03DE ACQUIRE_FOR_LINK
00A2 046CCB00 0510 Dom 841AF480 841AFB9E 071E DFHAPLI1
00A2 046CD010 0450 Sub 841B4418 840612F6 0000 *YES* DFHAPLI3
00A2 046CD460 03A0 Sub 84061500 84061E14 0914 *YES* DFHSRP
00A2 046CD800 0F80 Dom 89BDEDD8 89BE23A2 35CA DFHMEME
```

CICS Problem Determination

ABENDS Example 1

We then examine KE Domain Error Summary to see the error flow:

==KE: KE Domain Error Table Summary

ERR_NUM OFFSET	ERR_TIME	KE_NUM	ERROR TYPE	ERR_CODE	MODULE
=====	=====	=====	=====	=====	=====
00000003 00000020	15:41:28	0000	TRAN_ABEND_PERCOLATE	---/APSI	DFHLIRET
00000004 00003022	15:43:47	00A2	ABEND	2C5/AKEB	DFHYC220
00000005 0000071A	15:43:47	00A2	PROGRAM_CHECK	0C1/AKEA	-noheda-

CICS Problem Determination

ABENDS Example 1

We see the initial abend2C5 happened so we find this KERRD information:

Error Code: 2C5/AKEB Error Type: ABEND Timestamp: C083E66DA3200980
Reason Code: 47020033

KE_NUM: 00A2 KE_TASK: 046CB080 TCA_ADDR: 007E6680
DS_TASK: 09A13380 Error happened in program DFHYC220 at offset 00003022

PSW: 07DD0000 0086E372 Instruction Length: 2 Interrupt Code: 25
Exception Address: 00000000 Execution key at Program Check/Abend: D

REGISTERS 0-15

0086E66E	0086EA40	0086F740	00826A10
0086B380	009C02AC	0086F740	00000D28
0086BEF0	6086E336	0086B380	0086B380
0086C1A8	0086E9A8	6086DBB2	0086E330

CICS Problem Determination

ABENDS Example 1

We now print storage go to the PSW address area x'0086E372' and go backwards to find the beginning of the module:

The abend happened in:

```
*0.j.....f.y.f...fAy.f.0*  
*.b...f..COBF3000XXXXXXXXX .fA.15/0*  
*1/9710.06.09.....0000.....*  
* .....*
```

We then go to the PSW address area in storage and look for something which would give us an indication of the cause of the abend:

```
*.....&.o..0..ILBDDDBG03010/05/90* 0086E330
```

CICS Problem Determination

ABENDS Example 1

Analysis:

DFHSR0615 ABEND2C5 occurs after migrating to CICS TS for VSE/ESA 1.1.1

CICS Kernel error data (KERRD) offset x'2C' contains the reason code x'47020033'. The description of this reason code in the VSE Message and Codes manual indicates that a program has issued a STXIT AB macro although an ESTAEX exit is already defined.

The Program Status Word (PSW) at time of abend points into COBOL module **ILBDDBG0**.

CICS Problem Determination

ABENDS Example 1

Cause

A COBOL program has been compiled with a **restricted COBOL option or verb**. This causes ILBDDBG0 to attempt to set up its own abend recovery routines using the STXIT AB. However, VSE rejects the STXIT AB request because CICS has already set up its abend recovery routine using an ESTAE.

Solution

Change the COBOL application to remove the restricted verbs or options. To determine which COBOL programs contain the CICS restricted verbs and options, you can use LIBR LIST and look for eyecatcher ILBDDBG0. The presence of ILBDDBG0 indicates that the COBOL application was compiled with one of the restricted options.

The CICS Application Programming Guide contains a list of the restricted verbs and options under the topic COBOL Considerations .

CICS Problem Determination

ABENDS Example 2

Scenario:

Our customer upgraded to TCP/IP 15E Beta version for VSE. When they shutdown TCP/IP they get the following errors:

Messages:

```
DFHSO0106 06/28/06 09:45:21 A Language Environment Callable  
      Service error (code X'0226') has occurred in module DFH SOCK.  
SYMPTOMS: PIDS/564805400 LVLS/411 MS/DFHSO0106 RIDS/DFH SOCK  
      PTFS/UK08369 PRCS/00000226
```

```
DFHSO0002 A severe error (code X'0211') has occurred In module DFH SOCK.
```

CICS Problem Determination

ABENDS Example 2

We issue **CALL DFHPD410 DATA KE=3** in the Infoana job to find the running task:

KE_#	KE_TASK	STATUS	TCA_ADDR	TRAN_#	TRANSID	DS_TASK	KE_KTCB
0027	0531DB00	*Running	04D8B080	00024	CSOL	06680480	06790020

We then examine the **task's stack output** to see what the module flow shows:

```
.
KE_# @STACK  LEN  TYPE ADDRESS  LINK REG OFFS          ERROR NAME
0027  053D5020 0120 Bot  867C9BE8 867C9EB8  02D0          DFHKETA

                               Int  +00D4 84E40166 008E          INITIAL_LINK
0027  053D5B00 0510 Dom  84FEA3E0 851D7148 0000          DFHAPLI1
                               Int  +166C 84FEA986 05A6          CICS_INTERFACE
0027  053D6010 0AF0 Dom  84E57AB0 84E5DABC 600C          DFH SOCK
0027  053D6B00 0F80 Dom  8681DDD8 868214C8 36F0          DFHMEME
                               Int  +2CD6 8681DF4E 0176          SEND
```

CICS Problem Determination

ABENDS Example 2

We then issue **CALL DFHPD410 DATA TR=3** in the Infoana job and find the *EXC exception trace entries:

```
TR 0101 TRPT *EXC* - OVERLENGTH-ENTRY : CALLING DOMAIN WAS .. :  
PARAMETER LIST WAS FUNCTION(TRACE_PUT) POINT_ID(226) DATA1(035DF580  
    , 00000040) DATA2(02590000 , 00000398) DATA3(00000000 , 00000248)  
DATA4(00000000 , 02590308)
```

```
TASK-00054 KE_NUM-0007 TCB-00364000 RET-82017AC8 TIME-  
11:54:28.5079103310 INTERVAL-00.0000673281 =000113=  
1-0000 0021 * .. *
```

```
TASK-00054 KE_NUM-0007 TCB-00364000 RET-8365FD4C TIME-  
11:54:28.5083155029 INTERVAL-00.0000474296 =000124=
```

```
1-0000 00268050 C4C6C8E2 D6F0F1F0 F640D7D9 D6C4C3C9 C3E24040  
DFHSO001
```

CICS Problem Determination

ABENDS ---- NOTES

CICS always performs exception tracing when it detects an exception condition (for example, bad parameters on a domain call, or an abnormal response from a called routine). The aim is "first failure data capture", to record data that might be relevant to the exception as soon as possible after it has been detected.

CICS uses a similar mechanism for both exception tracing and "normal" tracing. Exception trace entries are made from specific points in CICS code, and data is taken from areas that might provide information about the cause of the exception. The first data field in the trace entry is usually the parameter list from the last domain call, because this can indicate the reason for the exception.

Exception trace entries are always written to the internal trace table, even if no trace destinations are currently STARTED. That is why there is always an internal trace table in every CICS partition, to make sure there is always somewhere to write exception trace entries. If the other trace destinations are STARTED, the exception trace entries are written there, as well.

CICS Problem Determination

ABENDS ---- NOTES

You can select tracing options so that exception traces only are made to an auxiliary trace data set. This is likely to be useful for production regions, because it enables you to preserve exception traces in auxiliary storage without incurring any general tracing overhead. You need to disable all standard and special task tracing, and enable auxiliary trace:

1. Ensure that special tracing has not been specified for any task.
2. Set the master system trace flag off.
3. Set the auxiliary trace status to `STARTED`, and the auxiliary trace data set and the auxiliary switch status to whatever values you want.

Exception traces are now made to an auxiliary trace data set, but there is no other tracing overhead.

The format of an exception trace entry is almost identical to that of a normal trace entry. However, you can identify it by the eye-catcher `*EXC*` in the header.

CICS Problem Determination

ABENDS Example 2

SO 0211 **SOCK *EXC*** - ASYNCIO_ERROR RETURN_VALUE(1)
RETURN_CODE(1124) REASON_CODE(000000FF)

TASK-00054 KE_NUM-0007 TCB-00364000 RET-82017AC8 TIME-
11:54:42.9993327062 INTERVAL-00.0016391423 =000233=
1-0000 FFFFFFFF 00000464 000000FF 00000000 00000000 02590000
00000080 02590308 **
0020 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 **
2-0000 03986EC4 C6C8E2D6 D3E3C540 40404040 01C22084 01C22084
043D0000 00008000 *.q>DFHSOLTE .B.d.B.d.....*
0020 00000000 00000000 001C6EC4 C6C8E2D6 E2E3C5C8 C5C1C440
02590028 02590028 *>DFHSOSTEHEAD*
0040 00000000 00000000 00000001 00008000 00000005 00000000 00000000
78000000 **

CICS Problem Determination

ABENDS Example 2

ANALYSIS:

We can see the errors are happening during SHUTDOWN processing. If the CICS Internal Trace is not started, this failure does not happen when closing the TCPIP SERVICE connection initially. Once the connection is closed and restarted, the error does not happen.

The trace domain shows us two errors:

1. An error response is returned on a `sock_call_async_accept_service`. The return codes are: `AioRv=-1` and `AioRc=ENOTCONN (1124)`. This error will give us the DFHSO0002 abend.
2. This in turn results in `call listener_error(tid_sock_accept_failure)`. `Listener_error` issues `trace_bpx_call_failure(trace_point_id)` this calls `trace_bpx` which selects which data areas to trace depending on the trace id.

In this case (`tid_sock_accept_failure`) we trace data1,2,3,4. However, we have no data3 or data4 (STE and `SOCKADDR_ADDR` respectively) so both addresses and the lengths are unpredictable.

In this case the length of `SOCADDR_ADDR` is overlength.

CICS Problem Determination

ABENDS Example 2

SOLUTION:

This failure resulted in creating two APARS for each error we found:

APAR Identifier PK29184 Last Changed 06/12/08

TR 0101 TRPT *EXC* - OVERLENGTH-ENTRY : CALLING DOMAIN WAS .. :
CLOSING A TCPIP SERVICE CONNECTION

The second APAR is:

APAR Identifier PK39175 Last Changed 07/03/09

SHUTTING DOWN TCP/IP LEADS TO A CICS DUMP DFHSO0106 X'0226' IN
DFH SOCK FOLLOWED BY DFHSO0002 WHEN INTERNAL TRACE IS ACTIVE.

CICS Problem Determination

ABENDS --- Summary

- Suggested steps to debug an ABEND
 - Ensure the DUMP=YES SIT parameter is set (this to get a dump of the abend)
 - Get all the CICS and VSE message logs
 - Issue “CALL DFHPD410 DATA KE=3,TR=3,LD=3” in the Infoana job
 - Call the IBM Support Center if you have any questions/problems
-

CICS Problem Determination

HANGS

SCENARIO

CICS shutdown hangs following the DFHTM1782 message that says that all non-system tasks have quiesced. The DFHSTP shutdown task is waiting on AP QUIES CSASSI2 (which means it is waiting on terminal control quiesce.)

A CEMT SET VTAM FORCE has caused CICS to close the VTAM ACB, but the TCP task has not finished quiescing because the CSNE Task is still active.

The CSNE task has been stuck in an ICDELAY wait for several minutes because of an EXEC CICS DELAY INTERVAL(000001) done by a NEP. The CSNE task should have only suspended for 1 second with that command, but because of a problem, the CSNE task is now hung indefinitely.

CICS Problem Determination

HANGS

MESSAGES

DFHTM1782I applid All non-system tasks have been successfully terminated.

Explanation: This message is issued during shutdown of the CICS session after successful termination by the user of any active tasks which had previously prevented termination.

System Action: CICS shutdown continues normally.

User Response: None

Destination: Console

Modules: DFHSTP

XMEOUT Parameter: applid

CICS Problem Determination

Hangs

ANALYSIS:

The problem which causes the CSNE task to hang in the ICWAIT longer than it should is; the fact that DFHSTP breaks the ICE chain (near label STTSBY) after non-system tasks have quiesced and before waiting on the terminal control quiesce.

A break in the ICE chain while a task is waiting in an ICDELAY will cause that task to wait forever.

A clue that this problem has happened is the fact that the Time Of Suspend of the ICDELAY task (the CSNE task) is in the same second as time of the DFHTM1782 message.

CICS Problem Determination

TASK Hang --- SOLUTION

APAR Identifier PK42130 Last Changed 07/09/14

CICS SHUTDOWN HANGS AFTER DFHTM1782 MESSAGE. DFHSTP SHUTDOWN IS WAITING ON TERMINAL CONTROL QUIESCE (CSASSI2) .

Symptom IN INCORROUT Status CLOSED DOC

Severity 3 Date Closed 07/09/14

Component 564805400 Duplicate of

Reported Release B0P Fixed Release

Component Name CICSTS FOR VSE Special Notice

Current Target Date ..07/10/30 Flags

SCP

Platform

LOCAL FIX:

Don't do an EXEC CICS DELAY from a NEP. (Doing an EXEC CICS DELAY from a NEP isn't a good idea anyway.)

CICS Problem Determination

TASK Hang -- SOLUTION

The following **documentation change** will be made to the CICS Transaction Server for VSE/ESA Customization Guide (SC33-1652-00). Chapter 9 (Writing a node error program), the section titled 'Restrictions on the use of EXEC CICS commands' will be amended by the addition of the following:

- o Terminal control. For example, issuing EXEC CICS DELAY may cause the CSNE task to suspend and never resume, which can cause shutdown of the region to hang. ("CEMT-type" commands, such as EXEC CICS INQUIRE TERMINAL are permissible).
-

CICS Problem Determination

HANGS

SCENARIO

Because of a HANG Condition, a CICS partition had to be CANCELLED. This because when a shutdown had been started, it would not complete. CICS had been started, and even though TCPIP=NO was specified in the SIT, the CWBG transaction was in this hang state.

MESSAGES

MESSAGE: DFHXM0306 XXXXXXXX A severe error (code X'130C') has occurred while initializing task number 02395 with transaction identifier CWBG. The task is suspended indefinitely.

SYMPTOMS: PIDS/564805400 LVLS/411 MS/DFHXM0306 RIDS/DFHXMTA
PTFS/UQ58418
PRCS/0000130C

CICS Problem Determination

HANGS

MESSAGE: DFHXM0306 applid A severe error (code X'code') has occurred while initializing task number tasknum with transaction identifier tranid. The task is suspended indefinitely.

Explanation: An internal error has prevented the initialization of task number tasknum with identifier tranid.

The task cannot run and cannot be abended. The task has no principal facility bound to it. Rather than terminate CICS, the transaction manager keeps CICS running and preserves its integrity by suspending the task. The suspended task will hold its MXT slot until CICS is terminated.

Note that the task may hold resources (for example, locks and enqueues) so you should cancel CICS at your earliest convenience. Otherwise you may risk other tasks being prevented from running because they also need access to the same resources. You may wish to add a dump table entry to always terminate CICS on this message.

CICS Problem Determination

HANGS

MESSAGE: DFHXM0306 CONTD...

System Action: The task is suspended indefinitely. First failure diagnostics should be produced by the component which first detects the error. The transaction manager also takes a dump. Message DFHME0116 is normally produced containing the symptom string for this problem.

The task is suspended with a resource type of FOREVER and a resource name of DFHXMTA.

User Response: You must cancel CICS if you need to destroy the task. You cannot quiesce CICS since this task will not terminate. You cannot purge or force purge the task.

Note the error code X'code'. You need further assistance from IBM to resolve this problem. See Part 4 of the CICS Problem Determination Guide for guidance on how to proceed.

Destination: Console

CICS Problem Determination

Hangs

ANALYSIS:

In this scenario, the KERNEL and TRACE output did not show much information of much use for our customers. However, the stack and trace information was quite valuable to CICS support.

The problem is due a CICS module, DFHXMTA, called DFHUSXM for INIT_TRANSACTION_USER call and passing an invalid SESSION_USER_TOKEN. CWBG is a non terminal task and should never have a SESSION_USER_TOKEN.

DFHXMTA's stack storage was not being initialized to zeros. Thus causing CICS to make the bad choice of passing a SESSION_USER_TOKEN based on residual data within the area of the stack address.

CICS Problem Determination

Hangs

We first go into the KE=3 Infoana output to see what the running task shows:

```
0064 02D73780 ***Running** 020AA080 02395 CWBG 01DF8780 053B4020
```

We then look at the STACK entry for KE_NUM 0064 to see what that flow shows:

```
KE_NUM @STACK  LEN  TYPE ADDRESS LINK REG OFFS ERROR NAME
0064 02D7A020 0120 Bot 853E2BE8 853E2EB8 02D0 DFHKETA
0064 02D7A140 01F0 Dom 85407530 8540761E 00EE DFHDSKE
0064 02D7A330 03F0 Dom 854826A8 854854FC 2E54 DFHXMTA
      Int  +2240 85482BBA 0512 PROCESS_ATTACH_ERROR
      Int  +2A16 85484C12 256A ATTACH_WAIT_FOREVER
      Int  +2DF2 854852AA 2C02 WAIT_FOREVER
0064 02D7A720 0F80 Dom 85436BE8 8543A2D8 36F0 DFHMEME
Int  +3618 85438108 1520 TAKE_A_DUMP_FOR_CALLER
0064 02D7B6A0 0480 Dom 8540DA88 8540E6A8 0C20 DFHDUDU
      Int  +0A3C 8540DB70 00E8 SYSTEM_DUMP
      Int  +178E 8540E95C 0ED4 TAKE_SYSTEM_DUMP
```

CICS Problem Determination

Hangs

We now go into the TR=2 Infoana output to see if there is good trace information:

```
US 0402 USXM EXIT - FUNCTION(INIT_TRANSACTION_USER)
RESPONSE(EXCEPTION) REASON(INVALID_USER_TOKEN) USDOM_TRANSACTION_TOKEN(00000000
, 00000000) PRIORITY(0)
```

```
TASK-02395 KE_NUM-0064 TCB-003E6000 RET-85482BA4 TIME-23:36:53.2703683759
INTERVAL-00.0000011718 =002282=
```

```
XM 130C XMTA *EXC* - Logic_error - FUNCTION(TASK_REPLY) USER_TOKEN(02AA6500)
TASK_TOKEN(040E0027) SUSPEND_TOKEN(040E0027)
```

```
TASK-02395 KE_NUM-0064 TCB-003E6000 RET-8540761E TIME-23:36:53.2703691416
INTERVAL-00.0000007656 =002283=
```

```
1-0000 00580000 00000003 00000001 00000000 A6000400 00000000 02F001F0 F0F04040
*.....w.....0.000
0020 40404040 02AA6500 040E0027 00000048 40404040 40404040 40404040 40404040 * .....
*
0040 40404040 40404040 040E0027 40404040 40404040 40404040 * .... *
```

CICS Problem Determination

TASK Hang -- SOLUTION

APAR Identifier PK37353 Last Changed 07/05/01

TRANSACTION CWBG WAIT FOREVER STATE CAUSES SHUTDOWN TO
HANG

Symptom MS MSGDFHXM0306

PTF List:

Release B0P : UK24146 available 07/05/01 (1000)

CICS Problem Determination

CICS Hangs

- Steps to debug a HANG in a CICS partition
 - Inspect the MSG log for signs of abends
 - If there are no abends dump the CICS partition
 - Run Infoana against the CICS partition dump
 - Issue “CALL DFHPD410 DATA
KE=3,TR=3,LD=3”
 - If you have questions/problems, call the IBM Support Center
-

CICS Problem Determination

VSE SUB-TASK Hangs

- Steps to debug a HANG
 - Issue the STATUS command (more than once if you suspect a Loop condition)
 - Issue MSG xx,dump 0-7FFFFFFF,uuu or
 - Create a Standalone dump or
 - Dump the Partition(s) in question, SVA and Supervisor
 - If you have questions/problems, call the IBM Support Center
-

CICS Problem Determination LOOPS

A loop is the repeated execution of some code. If you have not planned the loop, or if you have designed it into your application but for some reason it fails to terminate, you get a set of symptoms that vary depending on what the code is doing.

In some cases, a loop may at first be diagnosed as a wait or a performance problem, because the looping task competes for system resources with other tasks that are not involved in the loop.

CICS Problem Determination LOOPS

- The following are some characteristic symptoms of loops:
 - The 'system busy' symbol is permanently displayed in the operator information area of a display unit, or stays displayed for long periods.
 - The transaction abends with abend code AICA.
 - CPU usage is very high, perhaps approaching 100%, yet some tasks stay suspended or ready, but not running, for a long time. CPU usage is very high, perhaps approaching 100%, yet some tasks stay suspended or ready, but not running, for a long time. You can check what the CPU usage is for any VSE job by using the DISPLAY SYSTEM ACTIVITY screen of the VSE/ESA Interactive Interface. See the VSE/ESA Operation manual for more information.
 - There is reduced activity at terminals, or possibly no activity at all.
 - One or more CICS partitions appear to be stalled, or to be continuing only slowly.
-

CICS Problem Determination

LOOPS

➤ SCENARIO

- Our customer at 13:28 03/24, received msg DFHSM0133 CICS is under stress (short on storage above 16MB).
 - This was accompanied by multiple messages from CA-Explore/CICS reporting an Short on Storage (SOS) condition.
 - At the time the problem was occurring, a VSE GETVIS command showed 3.5M 24-Bit and 11M 32-bit GETVIS available in the partition.
 - From that point on, the CICS/TS System was unresponsive. Our customer was unable to sign on at a tn3270 terminal. An active CICS/TS session would not respond.
 - CICS/TS would respond to a MSG xx command from the console, and would accept commands entered from the console, but it does not appear that any of the commands that were entered were ever executed.
 - The CICS/TS partition had to be cancelled and restarted to recover.
-

CICS Problem Determination

LOOPS

The first symptom was the **SOS** condition above the line. The **SM (Storage Manager)** output showed:

==SM: ECDSA Summary

Size: 31744K
Cushion size: 128K
Current free space: 80K (0%)
* Lwm free space: 0K (0%)
* Hwm free space: 1024K (3%)
Largest free area: 36K
* Times cushion released: 1570
Currently SOS: YES <<<<<<<<<< SOS CONDITION
* Times went SOS: 3
* Time at SOS: 00:00:00.067
* Storage violations: 0
* Extents added: 26
* Extents released: 0
Number of extents: 31
Extent list: Start End Size Free
07600000 076FFFFF 1024K 0K

CICS Problem Determination

LOOPS

Further on down in the SM (Storage Manager) domain output is the subpool summary. This presents information on different subpools to see who might be using more Pagestg than would be used for normal operation:

==SM: Domain subpool summary (ECDSA)

Name	Id	Initf	Bndry	Fxlen	Q-c	Gets	Frees	Elms	Elemstg	Pagestg
JCDYNLOG	C1			16		13142	6413	6729	13135056	13468K

The unequal number of Gets/Frees ratio and Pagestg amount shows us we are writing a lot to this JCDYNLOG and not freeing up the storage.

The Infoana command to get this output is:

CALL DFHPD410 DATA SM

CICS Problem Determination

LOOPS

In the TRace output there appears to be a looping condition as I see this pattern over and over:

```
31734 1 AP 00F9 JCP  ENTRY GETB 31734 1 AP F101 SMSCP ENTRY
DFHSC_FREEMAIN      DFHSC TYPE=FREEMAIN
31734 1 SM 0D01 SMMF  ENTRY FREEMAIN          007D7008,00000000,MACR
31734 1 SM 0D02 SMMF  EXIT  FREEMAIN/OK      CICS24 storage at 007D
31734 1 AP F102 SMSCP EXIT  DFHSC_GETMAIN    DFHSC TYPE=GETMAIN
31734 1 AP 00F9 JCP  EXIT  NORMAL
31734 1 AP EA00 TMP  ENTRY LOCATE            FCT,XXXXXXXX
31734 1 AP EA01 TMP  EXIT  LOCATE            FCT,XXXXXXXX,07A6C300,N
31734 1 AP 04E0 FCFR  ENTRY READ_SET_UPDATE  00000000,07A6C300,007D
31734 1 SM 0301 SMGF  ENTRY GETMAIN          7D0,YES,FCSS SET,CICS
31734 1 SM 0302 SMGF  EXIT  GETMAIN/OK      07D471B8
31734 1 AP F00A XCPC  ENTRY CONDITIONAL_ENQ
31734 1 AP F00B XCPC  EXIT  CONDITIONAL_ENQ
```

CICS Problem Determination

LOOPS

31734 1 AP 0460 FCJL ENTRY JOURNAL_REQUEST

07A6C300,007CEA3C,07D471B8,07D471B8

31734 1 AP 00F9 JCP ENTRY WRITE

31734 1 AP 00F9 JCP EXIT NORMAL

31734 1 AP 0461 FCJL EXIT JOURNAL_REQUEST/OK

31734 1 AP 04E1 FCFR EXIT READ_SET_UPDATE/OK

7D0,07D471B8,197,00000000,

31734 1 AP 04E0 FCFR ENTRY REWRITE_DELETE

00000000,07A6C300,YES,NO

31734 1 AP 0460 FCJL ENTRY JOURNAL_REQUEST

07A6C300,007CEA3C,SYNCHRONISE_READ_

31734 1 AP 00F9 JCP ENTRY WAIT

31734 1 DS 0004 DSSR ENTRY RESUME

008E0005

CICS Problem Determination

LOOPS

31734 1 DS 0005 DSSR EXIT RESUME/OK
31734 1 DS 0004 DSSR ENTRY WAIT_OLDC
J0191335.....ql.,JCIOBLOK,0068E5DC,
J01 1 DS 0005 DSSR EXIT SUSPEND/OK
J01 1 DS 0004 DSSR ENTRY WAIT_OLDW
JCTICA,JCIOCOMP,0068E639,NO,IO
J01 1 DS 0005 DSSR EXIT WAIT_OLDW/OK
J01 1 DS 0004 DSSR ENTRY SUSPEND 008E0005,DFHJ01A
.W^.....,JCJOURDS,
31734 1 DS 0005 DSSR EXIT WAIT_OLDC/OK
31734 1 AP 00F9 JCP EXIT NORMAL
31734 1 AP 0461 FCJL EXIT JOURNAL_REQUEST/OK
31734 1 AP EA00 TMP ENTRY LOCATE AFCT,XXXXXXXX
31734 1 AP EA01 TMP EXIT LOCATE
AFCT,XXXXXXXX,07A70060,NORMAL
31734 1 DS 0004 DSSR ENTRY WAIT_OLDW

CICS Problem Determination LOOPS

```
XXXXXXXX,FCIOWAIT,00C54841,NO,IO
31734 1 DS 0005 DSSR EXIT WAIT_OLDW/OK
31734 1 SM 0301 SMGF ENTRY FREEMAIN          07D471B8,7D0,FCSS
SET,CICS
31734 1 SM 0302 SMGF EXIT FREEMAIN/OK
31734 1 AP 04E1 FCFR EXIT REWRITE_DELETE/OK  00000000
```

The Infoana command to see the Trace output is:

CALL DFHPD410 DATA TR

CICS Problem Determination

LOOPS

Here is the KERNAL output showing the running TASK-31734 and its stack entry and error information:

005B 0799A080 ***Running** 0073E680 31734 XXXX 0991E080 09A51020

KE_NUM @STACK LEN TYPE ADDRESS LINK REG OFFS ERROR NAME

005B 0799B020 0120 Bot 89A8ABE8 89A8AEB8 02D0 DFHKETA

ABNORMAL_TRANSACTION_END

005B 007140A0 0478 Lifo 07704000 87704790 0790 DFHDBP1\$

005B 0799B9A0 0240 Sub 90DD7F18 90DD8B62 0C4A DFHF CFR

Int +0BEA 90DD8326 040E ACCMTEST

005B 0799BBE0 03B0 Sub 90DF3BE8 90DF78C2 3CDA DFHF CVS

Int +3C7A 90DF400E 0426 FCJRNSRU

005B 0799BF90 0480 Sub 90DF28B0 90DF361C 0D6C DFHF CJL

Int +07EA 90DF296A 00BA JOURNAL_REQUEST

Int +0DAE 90DF31C8 0918

SYNCHRONISE_READ_UPDATE

Int +0D4E 90DF367C 0DCC

SYNCHRONISE_JOURNAL

005B 007D1020 0338 Lifo 00293E10 80295B94 1D84 DFHJCP

005B 0799C410 0310 Dom 89AA8A80 89AAA2F8 1878 DFHDSSR

CICS Problem Determination

LOOPS

==KE: KE Domain Error Table Summary

ERR_NUM OFFSET	ERR_TIME	KE_NUM	ERROR TYPE	ERR_CODE	MODULE
=====	=====	=====	=====	=====	=====
=====					
00000009	13:49:57	005B	TRAN_ABEND_PERCOLATE	---/AFCY	DFHPCP
000004EC					
0000000A	13:49:57	005B	TRAN_ABEND_PERCOLATE	---/AFCY	DFHFCEI
00000F70					
0000000C	13:50:04	005B	TRAN_ABEND_PERCOLATE	---/AFCY	DFHEPC
00000148					
0000000D	13:51:58	0001	ABEND	122/AKEB	DFHKETCB
00000A48					

CICS Problem Determination LOOPS

It looks like the task was purged and this resulted in the AFCY abend:

AFCY

Explanation: The transaction issued a file request resulting in a call to the main file control program (DFHFCFR). During the processing of the request the transaction was purged (that is, was the subject of an explicit PURGE or FORCEPURGE request, was timed out, or was selected by CICS for termination in an attempt to alleviate an SOS condition). A purged response was returned from DFHFCFR to its caller.

System Action: The task is abnormally terminated with a CICS transaction dump.

CICS Problem Determination

LOOPS

We then looked at the dynamic log to see what its contents show:

```
* ..>DFHJCDBL_BUFF.J...'.....*  
* .....@.....a.....C.XXXXXXX*  
*X .....XX XXX .....XX XXX*  
* 40500025254 ...0 ....XX XXX .*  
* .....XX XXX 40500025254 ...0 .*  
*8L0...I01 *  
* 0001SHD4070 2523 *  
TACTICAL CO*
```

The Infoana command to get this output is:

CALL DFHPD410 DATA AP

LOCATE ON THE STRING DBL

CICS Problem Determination

LOOPS -- SOLUTION

The dynamic log shows that the task was in a loop issuing repeated commands against the same file control record in file XXXXXXXX.

Each dynamic log record is the same which is the tell-tale symptom of a looping task that changes the same RIDFLD over and over.

It would seem that the RIDFLD is the data beginning 'XX XXX...'. The FCT shows that the key length is x'22'.

Eventually, this task was purged and had to backout the changes so DFHDBP1\$ validly had to read back along the dynamic log issuing repeated READ UPDATE / REWRITE DELETE commands against this same record key.

This eventually lead to the SOS condition/CICS hang that was reported. To resolve this problem the Customer needs to determine the cause of the original application loop.

CICS Problem Determination LOOPS

- Steps to debug LOOPS
 - Issue the STATUS command several times (this to see if the PSW is constantly changing)
 - Issue "CEMT P SNAP" to get a dump of the partition
 - Dump the SVA and SUPERVISOR
 - Using the DFHPD410 dump formatter, format KE=3, TR=3, AP=3
 - Gather all the CICS and message logs
 - Contact the IBM Support Center for assistance
-

CICS Problem Determination

WAITS

Table 20. Resources that a suspended task might be waiting on

Resource type	Resource name	Suspending module	DSSR call	Task
(none)	DMWTQUEU	DFHDMWQ	SUSPEND	System
(none)	LMQUEUE	DFHMLM	SUSPEND	User
AP_QUIES	CSASSI2	DFHSTP	WAIT_OLDC	System
AP_TERM	STP_DONE	DFHAPDM	WAIT_	System
			EXTERNAL	only
EKCWAIT	Value of NAME argument	DFHEKC	WAIT_OLDW	User
FCIOWAIT	file ID	DFHFCBD or DFHFCVR	WAIT_OLDW	User
FCPSWAIT	file ID	DFHFCVR	WAIT_OLDC	User
FOREVER	DFHXMTA	DFHXMTA	WAIT_	User
			EXTERNAL	
TCP_NORM	DFHZDSP	DFHZDSP	WAIT_OLDW	System
ZCIOWAIT	DFHZARQ1	DFHZARQ	SUSPEND	User

CICS Problem Determination

WAITS

DMWTQUEU

A system wait typically when domains are shutting down.

LMQUEUE

It means that the suspended task cannot acquire the lock on a resource it has requested, probably because another task has not released it.

CSASSI2

It means the AP domain is trying to shutdown and it has not finished. Typically it waits on the Terminals to finish their shutdown process.

STP_DONE

This wait is done to wait for DFHSTP to complete its processing before returning to Domain Manager, as DM will assume we have completed QUIESCE when we return and set the phase point etc, allowing other domains to complete their QUIESCE processing.

EKCWAIT

EKCWAIT indicates that a task has issued an EXEC CICS WAIT EVENT command. USERWAIT indicates that a task has issued an EXEC CICS WAITCICS or EXEC CICS WAIT EXTERNAL command. If the wait is prolonged, you should identify the event being waited on.

CICS Problem Determination

WAITS

FCIOWAIT

A wait on resource type FCIOWAIT occurs when the exclusive control conflict is deferred internally by VSAM and not returned as an error condition to CICS. An example of this is when a request against an LSR file is made for exclusive control of a control interval (for example, by EXEC CICS WRITE or READ UPDATE) and either this task or another task already holds shared control of this control interval (for example, by STARTBR).

FCPSWAIT

If your task is waiting on either of resource types FCPSWAIT or FCSRSUSP, it means that it cannot get a VSAM string. FCPSWAIT shows that the wait is for a private string, and FCSRSUSP shows that the wait is for a shared resource string.

FOREVER

If you have found that a user task is waiting on a resource type of FOREVER, and resource name DFHXMTA, transaction manager has detected a severe error during task initialization or task termination. Transaction manager has suspended the task.

ZCIOWAIT

Suspends on resource type ZCIOWAIT occur when the task is waiting for some terminal I/O. Once the expected I/O event occurs, the task is resumed.

DFHZARQ1 - resource type and name for all application requested waits involving NON-LU 6.2 devices.

DFHZARR1 - waiting for a receive issued to a LU6.2 ISC connection.

DFHZARL1 - waiting for a send issued to a LU6.2 device.

CICS Problem Determination

WAITS

SCENARIO

Our customer found CICS was in a “hang”, customer has to cancel CICS partition and dump was generated at 11:09:08. Last trace entry time is 10:50:59 which means CICS was “hung” after this time.

CICS Problem Determination

WAITS

Here is the CICS KERNAL entries showing the running task and what it is doing at the time of the "hang":

10 078EEB00 ***Running** 07897680 58065 XXXX 1785AD80 1832D020

KE_NUM @STACK LEN TYPE ADDRESS LINK REG OFFS ERROR NAME

0010 078F8020 0120 Bot 98366BE8 98366EB8 02D0 DFHKETA

0010 078F8720 03E0 Dom 879400C8 87940A5C 0994 DFHPGPG

Int +00D4 87940156 008E INITIAL_LINK

0010 078F8B00 0510 Dom 87AEA410 89982D3E 0000 DFHAPLI1

Int +223E 87AEA9AE 059E LE370_INTERFACE

Int +2022 87AECC5C 284C

INVOKE_FOR_RECURSION

10 078F9010 06A0 Lifo 006EB358 879CD894 0000 DFHERM

CICS Problem Determination

WAITS

We now know that there was a call to an external resource manager to do some work. We need to see what the last trace entry shows as this will give us the request information:

```
AP 00E7 ERM ENTRY APPLICATION-EXEC-DLI          REQ(0004) FIELD-  
A(C22D71F4 B..4) FIELD-B(D5075904 N...) RESOURCE(DLI  
          TASK-58065 KE_NUM-0010 TCB-0041F000 RET-89982D3E  
TIME-10:50:59.0021395947 INTERVAL-00.0000005312  =116992=
```

CICS Problem Determination

WAITS

We then go to the RETurn address listed in the previous trace entry to see who is making the call to the external resource manager.

This call is made at offset x'1BFC' in the application:

```
| .....00..CEE..... | *  
| .....00.ϕ.....q.. | *  
| ..... | *  
| ..0.q.0<.....q. | *  
| .q...q...q...q. | *  
| .q...q...q..... | *  
| ....XXXXXXXX02008 | *  
| 0327161010010100 | *  
| .....-...h... | *  
| ..... | *  
*****
```

CICS Problem Determination

WAITS -- SOLUTION

- We know CICS was last dispatched at the call to the External Resource Manager DL/I.
 - We know that control was never returned from this DL/I call.
 - There was CPU activity going on in the DL/I partition.
 - Our customer needed to find out what is going on with the DL/I activity and see why it never returned to CICS.
-

CICS Problem Determination

WAITS

- Steps to DEBUG WAITS
 - DUMP the CICS Partition --- CEMT P SNAP
 - Run the "Analyze CICS Dumps" format the following domains: KE=3, TR=3, DS=1
 - Contact the IBM CICS Support Center
-

CICS dump formatter

```
* $$ JOB JNM=DMPACD1,DISP=D,PRI=8,  
* $$ NTFY=YES,  
* $$ CLASS=0  
* $$ LST DISP=H,RBS=1000  
// JOB DMPACD1 ANALYZE CICS/TS DUMP  
// EXEC PROC=DTRINFOA  
// EXEC INFOANA,SIZE=INFOANA,OS390  
    SELECT DUMP MANAGEMENT  
    DUMP NAME SYSDUMP.BG.DBG00002  
    RETURN  
  
    SELECT DUMP VIEWING  
    CALL DFHPD410 DATA AP=0,KE=3,DS=1,TR=2,LD=3  
    RETURN  
        DUMP NAME SYSDUMP.BG.DBG00002  
    RETURN  
    SELECT END  
  
/*  
/&  
* $$ EOJ
```

```
C /====/  
C *====*  
  *====*  
  *====*  
  *====*  
  *====*  
    *====*  
    *====*  
    *====*  
      *====*  
      *====*  
      *====*  
        *====*  
        *====*  
        *====*  
          *====*  
          *====*  
          *====*  
            *====*  
            *====*  
            *====*
```

CICS/TS for z/VSE Storage Tuning Issues

We will now look at the potential to increase 24-bit storage availability and how to size CICS TS partitions and DSA sizes

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

Data Required

- 1.SVA statement 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:

SVA SDL=700, GETVIS=(2M,6M),PSIZE=(652K,7M)

MAP

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

. . .

Example without using IOBUF31

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

SVA(24) TOTAL SPACE : 1852K (100%)
 USED SPACE : 1603K (87%)
 - PFIXED AREA: 166K (9%) START AT: 00273540
 FREE SPACE : 249K (13%)

. . .

GETVIS USAGE	SVA-24	SVA-ANY	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

Example without using IOBUF31

1. MAP shows SVA-24 UNUSED = 384K.
 2. GETVIS SVA AREA SIZE – MAX. EVER USED 2,912K - 1,736K = 1176K.
 3. LD SDL Virtual Library SVA(24) FREE SPACE = 249K.
- 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.
 - 2.Unused Virtual Library, leaving at least 64K unused.
 - 3.Unused SVA Getvis-24, leaving perhaps 320K unused.
-

What do I adjust?

As 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 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 1216K.

After.....

```
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
```

After.....

SVA SDL=700 ,GETVIS=(1152K,6M),PSIZE=(652K,7M)

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. I could probably 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.

RESULTS...

Benefits

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

Side effects

Partitions that have an ALLOC value that 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. 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 another 314K that could be moved out of System Getvis. Adding that to previous 1,809K gives 2,123K, which is perhaps a bit too close to 2MB to be able to safely reduce the shared area size down to 4MB.

Other ways to reduce 24-bit usage

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

Perhaps you only need ALL31(OFF) in the AOR where the 24-bit code resides. If you do not need ALL31(OFF) in other partitions, compile a private CEECOPT for these 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. IBM and OEM software may be written in C, for example, WebSphere MQ uses C.

CICS Partition 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-watermark as being equal to the AREA SIZE.

CICS Partition Storage Usage

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.

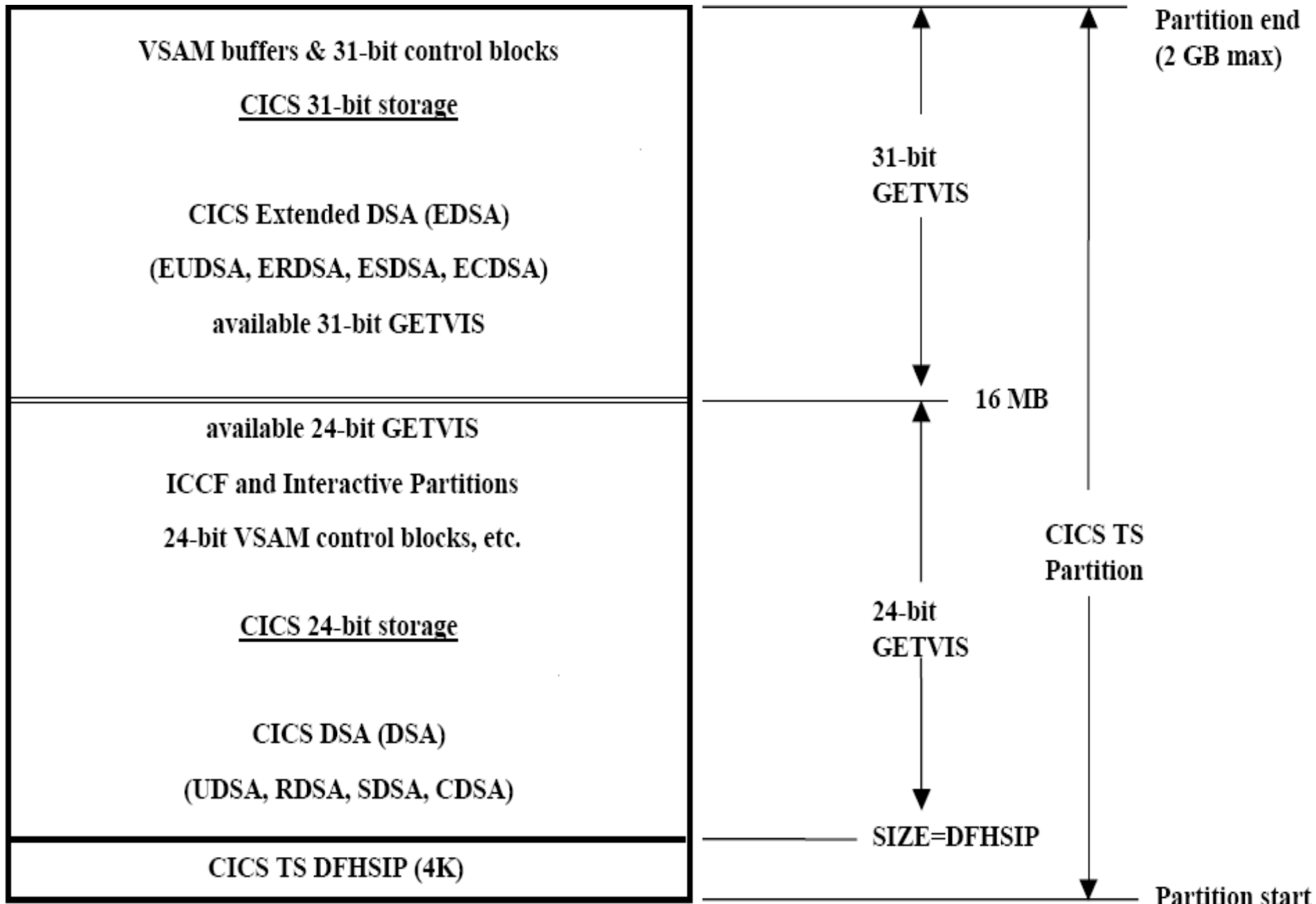
The SIT EDSALIM value is allocated out the Getvis-31 area, as is the CICS trace table, as are VSAM buffers and 31-bit control blocks.

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

How big should a CICS partition be?

The required 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 for a suitable amount of time to enable it to reach peak usage before producing data, and look at several executions of CICS.

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.
-



VSAM buffers & 31-bit control blocks

CICS 31-bit storage

CICS Extended DSA (EDSA)
(EUDSA, ERDSA, ESDSA, ECDSA)
available 31-bit GETVIS

available 24-bit GETVIS

ICCF and Interactive Partitions
24-bit VSAM control blocks, etc.

CICS 24-bit storage

CICS DSA (DSA)
(UDSA, RDSA, SDSA, CDSA)

CICS TS DFHSIP (4K)

Partition end
(2 GB max)

31-bit
GETVIS

16 MB

CICS TS
Partition

24-bit
GETVIS

SIZE=DFHSIP

Partition start

Example.....

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
  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
  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
. . .
```

Example.....

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
-------------------------------------	---	---------

Example.....

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

1.Reduce EDSALIM 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.

You should be able to reduce the partition allocation to about 96MB.

Remember that unused Partition storage costs nothing but page dataset space.

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.

SUMMARY
