



Virtual CICS user group: Newsletter 66

Welcome to the Virtual CICS user group newsletter. The Virtual CICS user group at virtualcics.hostbridge.com is an independently-operated vendor-neutral site run by and for the CICS user community.

```

Display Filter View Print Options Search Help
-----
ISFPCU41 OG 24367.101 s0w2 s0w2 01/10/2022 0w 6,076 COLUMNS 52- 131
COMMAND INPUT ==> SCROLL ==> CSR
0090 +DFHSR0001 T560 An abend (code 0C9/AKEA) has occurred at offset
X'000001AE' in program ESIPCS33.
0090 +DFHME0116 T560 626
0090 (Module:DFHMEME) CICS symptom string for message DFHSR0001 is
0090 PIDS/5655Y0400 LVLS/730 MS/DFHSR0001 RIDS/DFHSRP PTF5/HCI7300
0090 AB/S00C9 AB/UAKEA RIDS/ESIPCS33 ADRS/000001AE
0090 +DFHDU0205 T560 A SYSTEM DUMP FOR DUMPCODE: SR0001 , WAS
SUPPRESSED BY THE DUMP TABLE OPTION FOR THIS DUMPCODE

```

Figure 1: System console message that an abend has occurred

Virtual CICS user group presentation

The latest webinar from the Virtual CICS user group was entitled, "Taking the pain out of CICS application debugging". It was presented by Wendell Lovewell, product manager and systems programmer for MacKinney Systems.

Wendell Lovewell is a product manager and systems programmer for MacKinney Systems. For more than 40 years, he

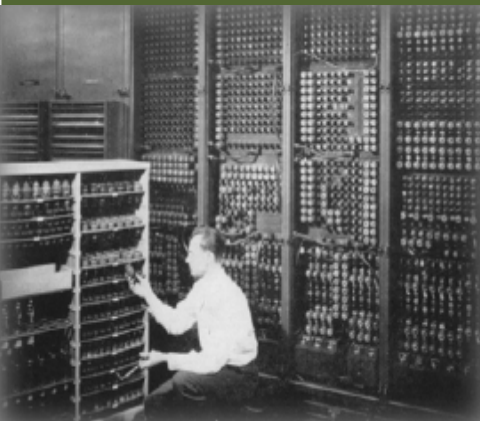
has been providing product support for CICS, ISPF, and batch products, and system programming support for z/OS, z/VSE, z/VM, Windows Server, and Linux. His favourite programming languages include REXX, COBOL, and Assembler.

Wendell Lovewell started his presentation by looking at three qualities of a good programmer (which he got from Larry Wall, original author of the Perl Programming Languages):

- 1 Laziness. You'll go to great lengths to reduce overall energy expenditure.

Contents:

Virtual CICS user group presentation	1
Meeting dates	5
CICS news	5
CICS articles and blogs	6
The Arcati Mainframe Yearbook 2022	6
About the Virtual CICS user group	6



```

L:\STD\DFHDUMPS.ASMLIST      Line=7      Col=1      Size=3064 Alt=0,0;2
====>
T560      --- CICS TRANSACTION DUMP --- CODE=ASRA  TRAN=KASE  ID=1/0002  DATE=22/01/10  TIME=14:36:08  PAGE  1
SYMPTOMS= AB/UASRA PIDS/5655Y0400 FLDS/DFHABAB RIDS/ESIPCS33
CICS LEVEL = 0730
PSW & REGISTERS AT TIME OF INTERRUPT
PSW:      07950000  A137300E  00020009  02A8B400
PSW16:    07950000  80000000  00000000  2137300E
TRANSLATION EXCEPTION ADDRESS: 00000000_02A8B400
BREAKING EVENT ADDRESS:      00000000_21372FFC
ADDRESSING MODE: 31
REGS 0-3  00000000_21100A28  00000000_21100908  00000000_00041800  00000000_A1372E88
REGS 4-7  00000000_00000000  00000000_00000000  00000000_2044BE4C  00000000_20E2F320
REGS 8-B  00000000_1FC94DD8  00000000_211008A0  00000000_21100008  00000000_21100100
REGS C-F  00000000_008B3000  00000000_21100048  00000000_A1372FDA  00000000_00000000
EXECUTION KEY 9
The transaction was in Subspace mode
REGISTERS AT LAST EXEC COMMAND
REGS 0-7  21100A28  21100908  00041800  A1372E88  00000000  20E2F13C  2044BE4C  20E2F320
A137300E - A1372E88 = 390 (hex 186)

```

Figure 2: Working with transaction dumps

- 2 Impatience. The anger you feel when the computer is being lazy.
- 3 Hubris. Excessive pride that makes you write/maintain programs.

Wendell went on to look at how to be a 'better' programmer, reminding us that our computer can do a gazillion things a second, and asking how many we can do? He suggested that we should let the computer do more, so we can do more! He said that we should stop doing time-consuming stuff ourselves.

Wendell went on to ask: when it comes to diagnosing CICS application problems, what is most painful about the way you resolve them? His answer was:

- Reading Dumps
- Analyzing Traces
- Stepping through with Debuggers.

A transaction abending is probably one of the more common indicators that something went wrong. Depending on the system settings, a 'dump' can be created. A dump contains the contents of memory used by the program, and some indication of what instruction the program was trying to execute.

Messages notifying you of an abend can appear:

- On the terminal
- On the system console (see Figure 1)
- In one of the 'MSGUSR' Transient Data destinations.

The first thing you have to know to read a dump is where it is and what kind of dump (if any) was produced for the problem. There are two types of CICS dump:

- Transaction Dump, which contains memory (data

and code) used by a single CICS transaction. Messages will be sent to the CDUL* transient data destination if a transaction dump was produced.

- System Dump, which contains most of the memory allocated to an entire CICS region. Because of their large size, system dumps are often suppressed by CICS.

There are two dump datasets, DFHDMPA and DFHDMPB. Both are defined in the JCL for the CICS region, but only one is active at a time. The CEMT SET DUMP SWITCH command is used to control which is active. You need to know the transaction ID and which dump data set (A or B) was used. You also need JCL to 'print' the dump. The suffix for DFHDUxxx is the 3-digit CICS release level (TS5.5 = 720, TS5.6 = 730, etc). DFHDMPDS must

```

=5.1 TRACK - Stop Display WRL1 T03R 01/06/22 15.01.13
Command-----> -
Offset-----> Password---->

Stop Program--> ESIPCS33 Current Module--> ESIPCS33 At Terminal--> T03R
Phase offset--> 0001AC module offset--> 000184 Statement No.--> 000343
Stop Reason--> FIXED-POINT DIVIDE
Stmnt .....1.....2.....3.....4.....5.....6.....7.....8.....9.....10.....11..
000340 * 2022/01/11: Force an abend for demo purposes
000341 * EX 0,5 D2201
000342 000182 1055 SR 5,5
000343 000184 1D05 DR 0,5
-----Column--> 1-----Type--> Hex-----
GPR 0,1 0 21100A2R 1 21100908
GPR 5 5 00000000
-----Column--> 1-----Type--> Hex-----
NEWCASE(9) 000001C2
-----Column--> 1-----Type--> Hex-----
000344 000186 47F0 318A 0018A B OUTAHERE

```

Figure 3: Using CICS debuggers

specify the correct (A or B) dataset. You can specify inclusion criteria, such as the transaction ID to be printed.

Wendell said that printing a dump is the easy part, and then went on to look at what you have to know to read a dump.

Figure 2 shows an example of a transaction dump. You need to know lots of information to read one, for example that offset x'1AE' goes to instruction x'1D05'.

When it comes to system dumps, if you really need one, you'll probably have to enable CICS for it. There are many types including an SR0001 dump. You need to use CEMT SET SYDUMPCODE to set the MAX count higher than the Cur count. You need to recreate the problem. The console should indicate that a system dump was taken. The second thing to know

about system dumps is that they are BIG! Wendell did not recommend using them unless IBM or a software vendor asks for one.

The advantage of using dumps is that they are free (every z/OS system can produce a dump). And, with training, IPCS can be used to speed the process. No setup is needed prior to the problem, but you do need to ensure that your system has space available for the dump. And, sometimes, they are the only way to resolve an issue.

The disadvantages of using dumps are that reading a dump is usually very labour-intensive and time-consuming. (There are vendor products, and IBM's IPCS, which can be used to simplify using a dump.) A great deal of technical knowledge is required to read dumps. They are 'point in time' only. That means it

can be difficult to determine program flow prior to the abend. However, the trace table entries included in the dump can help.

CICS has an integrated tracing facility that can record every interaction your program has with CICS. Trace entries are recorded in two locations: Internal and External. The internal trace is usually active but tends to wrap quickly. There are two external (AUX) trace files: DFHAUXT and DFHBUXT. The CETR transaction is used to control the Trace Facility. You usually have to start it, and note which dataset is used. You then run your test, and stop the trace as soon as possible. The trace dataset, DFHAUXT, now contains records for your trace. They can be 'printed' using program DFHTUnnn, where nnn = system release (eg TS5.6 = '730').

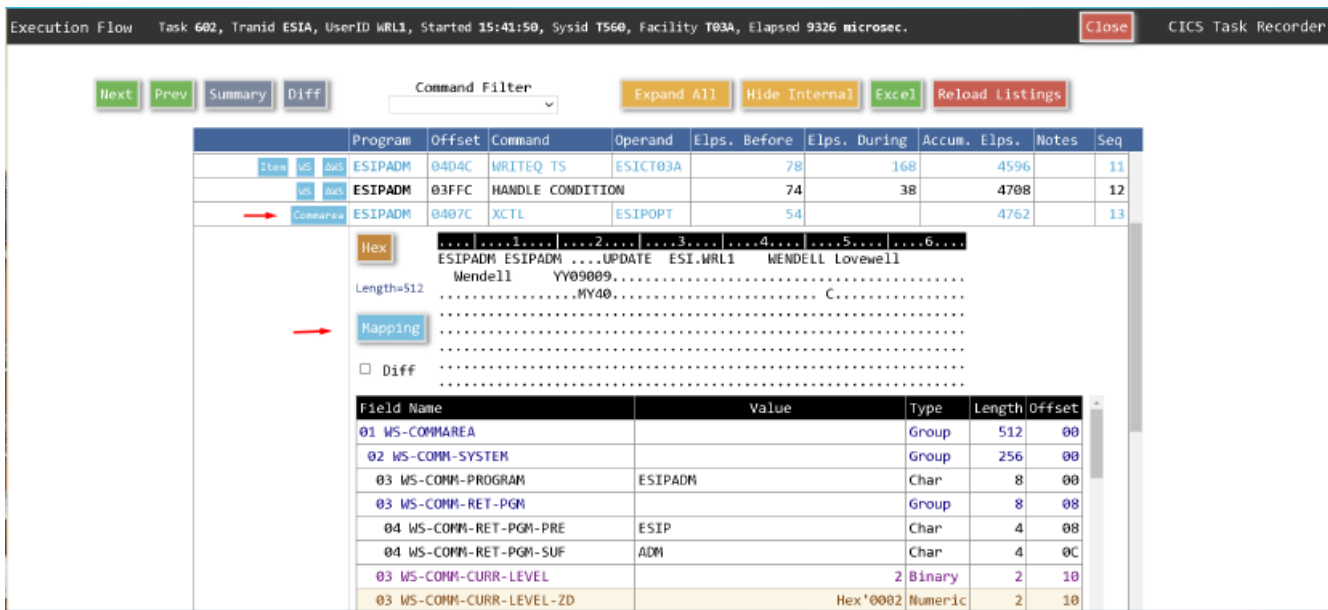


Figure 4: Example Task Recorder screen

Using an AUXTRACE listing is like looking for a needle in a really big haystack! Almost everything CICS does under the covers is exposed. Each domain call is shown with the matching ENTRY and EXIT displays. With an editor (or IPCS), you can find *EXC* entries for the exception, but it is hard.

There are products, including IPCS, that help you decipher the trace data, but without a lot of knowledge (ie experience), you may find trace data, like system dumps, something you only send to IBM or a vendor. One thing also to note, DFHDUnnn (the dump 'print' program) includes an abbreviated trace table following the dump of memory.

The advantages of using AUXTRACE are that it's free (built into every version of CICS). And, with training, IPCS can be used to speed the process. No setup is needed prior to the problem, although, depending on the application, you may need large trace datasets. And, sometimes, using CICS trace is the only way to resolve an issue

The disadvantages to using AUXTRACE are that a great deal of technical knowledge is required to use them, although there are products that make it easier. Lots of data generated, which means that it's likely to take a long time to identify a problem.

A third application diagnosis tool is the Debugger. This is an add-on product that allows you to step through your program, generally one statement at a time. You can display and change the program's data, and alter the sequence of statements that get executed.

Debuggers work by matching a program's executable code with the matching compile or assembly listing. When the machine code for a particular statement is about to be executed, the debugger will position your display to that statement in the compile/assembly listing. This requires that the debugger has access to either the ADATA generated by the compiler/assembler, or the

listing. There is often a step added to the compiler JCL to make it ready for debugging.

Debuggers are real-time: you can't use a debugger to diagnose a problem that occurred previously. The debugger must be told to start debugging a program. Each debugger has its own way to do this. The user interface can be a 3270 session or web browser. An example of using a debugger is shown in Figure 3. You can step through the code one line at a time, or continue to selected "halt points" to avoid stepping through each line. Variables can be displayed and changed in three ways, as they change, on request, or all the time. If the program is abending, you can just continue until theabend occurs. You can also change the order of the program statements.

The advantages of using debuggers are that you can watch your program executing. You can also change data on-the-fly to test exceptions. You can verify the results of every statement (or you can skip to the interesting parts). And it's fairly easy to use. If you can code, you will understand the debugger output.

The disadvantages of using debuggers are that it can take a very long time to

observe every line of code! However, functions like 'continue' and 'halt' can reduce this, but debugging sessions tend to take a long time. Sometimes, observing the program changes the results, particularly timing-sensitive issues. And, generally, you have to be 'live and in person'. If something fails when you're not there, a debugger won't show you what happened, and you may have to read a dump anyway.

Wendell went on to suggest that there could be a better way of doing things. Something that could combine the ability to capture everything that happened, like a trace, with the ability to display the contents of any or all variables, like a dump (where you could see not only all the variables in every 'step', but let the tool compare and point out the differences each step made!). Plus, have the ability to display every step, like a debugger, but not have to wait for it. He then went on to describe a new product called CICS Task Recorder (see Figure 4).

A copy of Wendell Lovewell's presentation is available for download from the Virtual CICS user group website at virtualcics.hostbridge.com/presentations/CICSDebugJan22.pdf.

You can see and hear the whole user group meeting at <https://youtu.be/bOK1fgHtu3w>.

Meeting dates

The following meeting dates have been arranged for the Virtual CICS user group:

- On 18 March we have Joe Winchester, IBM Senior Technical Staff Member from Hursley, who will be discussing, "CICS and open source tooling, with a focus on Zowe".
- The following meeting is on 10 May, when Robert Barnes, CEO Jazz Software, will be discussing "CICS Web services – bridging two worlds".

We are using Zoom for the user group meetings.

CICS news

IntelliMagic has announced IntelliMagic Vision for z/OS release 10.9.0, which includes major enhancements to reporting on CICS SMF data that lets users monitor the performance and resource usage of applications and transactions. New functionality includes support for 19 fields in the Statistics data, and more than 70

automated Health Insights for the CICS 110 SMF data.

More information can be found at: <https://www.intellimagic.com/resources/zos/news/intellimagic-vision-adds-cics-statistics-automated-analysis/>.

CICS articles and blogs

CICS VR-Executing batch backout when the RCDSs are unavailable by Dionne A Nerissa in the CICS part of the IBM Z and LinuxONE Community (24 November 2021). You can find the article at: <https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/dionne-a-nerissa/2021/11/24/cics-vr-executing-batch-backout-when-the-rcdss-are>.

Seamless VSAM data transfer between Host & PC by Dionne A Nerissa in the CICS part of the IBM Z and LinuxONE Community (24 November 2021). You can find the article at: <https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/dionne-a-nerissa/2021/11/24/seamless-vsam-data-transfer-between-host-pc>.

CICS VR - CHANGE ACCUMULATION by Chitra Devi S in the CICS part of the IBM Z and LinuxONE Community (24 November 2021). You can find the article at: <https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/chitra-devi-s/2021/11/24/cics-vr-change-accumulation>.

The Arcati Mainframe Yearbook 2022

Many members of the Virtual CICS user group contributed to this year's user survey in the Arcati Mainframe Yearbook. The 2022 edition of the Yearbook is now available. It includes an annual user survey, an up-to-date directory of vendors and consultants, a resources guide, a strategy section with papers on mainframe trends and directions, a glossary of terminology, and a mainframe evolution section. The Yearbook is available in PDF format and is completely FREE to download from itech-ed.com/arcati/

About the Virtual CICS user group

The Virtual CICS user group was established as a way for individuals using IBM's CICS TS systems to exchange information, learn new techniques, and advance their skills with the product.

The Web site at virtualcics.hostbridge.com provides a central point for coordinating periodic meetings (which contain technically-oriented topics presented in a webinar format), and provides articles, discussions, links, and other resources of interest to IBM CICS practitioners. Anyone with an interest in CICS is welcome to join the Virtual CICS user group and share in the knowledge exchange.

To share ideas, and for further information, contact trevor@itech-ed.com.

The Virtual CICS user group is free to its members.