218

CICS update

January 2004

In this issue

CICS Update

Published by

Xephon
27-35 London Road
Newbury
Berkshire RG14 1JL
England
Telephone: 01635 38342
From USA: 01144 1635 38342
E-mail: trevore@xephon.com

North American office

Xephon
PO Box 350100
Westminster, CO 80035-0100
USA
Telephone: 303 410 9344

Subscriptions and back-issues

A year's subscription to *CICS Update*, comprising twelve monthly issues, costs £175.00 in the UK; $270.00 in the USA and Canada; £181.00 in Europe; £187.00 in Australasia and Japan; and £185.50 elsewhere. In all cases the price includes postage. Individual issues, starting with the December 1999 issue, are available separately to subscribers for £16.00 ($24.00) each including postage.

CICS Update on-line

Code from *CICS Update*, and complete issues in Acrobat PDF format, can be downloaded from our Web site at http://www.xephon.com/cics; you will need to supply a word from the printed issue.

Editor

Trevor Eddolls

Disclaimer

Readers are cautioned that, although the information in this journal is presented in good faith, neither Xephon nor the organizations or individuals that supplied information in this journal give any warranty or make any representations as to the accuracy of the material it contains. Neither Xephon nor the contributing organizations or individuals accept any liability of any kind howsoever arising out of the use of such material. Readers should satisfy themselves as to the correctness and relevance to their circumstances of all advice, information, code, JCL, and other contents of this journal before making any use of it.

Contributions

When Xephon is given copyright, articles published in *CICS Update* are paid for at the rate of £100 ($160) per 1000 words and £50 ($80) per 100 lines of code for the first 200 lines of original material. The remaining code is paid for at the rate of £20 ($32) per 100 lines. To find out more about contributing an article, without any obligation, please download a copy of our *Notes for Contributors* from www.xephon.com/nfc.

Printed in England.

Finding threadsafe programs

One of the major enhancements in CICS Transaction Server Version 2 Release 2 is the CICS-DB2 path length improvement that kicks in when application programs are threadsafe. So the question is, how do I, as a system programmer, know when to mark a program as threadsafe?

CICS TS supplies a load module scanner utility, DFHEISUP, which scans load libraries for CICS commands and identifies which modules contain the specific API or SPI commands that you have chosen. An example might be when a manager asks a simple question like, “How many of our application programs use the basic CICS System Programming Interface (SPI)?” You can then use the load module scanner to identify all the load modules that contain EXEC CICS INQUIRE, SET, and PERFORM commands. The load module scanner can look for specific commands, options, or various combinations.

Having located all the EXEC CICS commands in the load modules in the library you are scanning, it applies a filter that you have specified, and reports the subset of commands that you have listed in the filter tables.

CICS provides two sample filter tables, DFHEIDBR and DFHEIDTH, in the SDFHSAMP library:

- DFHEIDBR contains the set for commands that are not supported by the 3270 bridge.
- DFHEIDTH contains the set for commands that give access to shared storage.

The DFHEIDTH list includes shared storage access commands that could make a program not threadsafe unless the necessary synchronization logic is in place to ensure serialization and prevent concurrent update.

Appendix L of the current *CICS Application Programming*

Reference Manual (APRM) and *Appendix D* of the *Systems Programmer Reference Manual* (SPRM) list the following commands as threadsafe.

The EXEC CICS API threadsafe commands are:

- ABEND
- ADDRESS
- DELETEQ TS
- DEQ
- ENQ
- ENTER TRACENUM
- FREEMAIN
- GETMAIN
- HANDLE ABEND
- HANDLE AID
- HANDLE CONDITION
- IGNORE CONDITION
- LINK
- LOAD
- MONITOR
- POP HANDLE
- PUSH HANDLE
- READQ TS
- RELEASE
- RETURN
- SUSPEND

- WAIT EXTERNAL
- WRITEQ TS
- XCTL.

The EXEC CICS SPI threadsafe commands are:

- DISCARD DB2CONN
- DISCARD DB2ENTRY
- DISCARD DB2TRAN
- INQUIRE DB2CONN
- INQUIRE DB2ENTRY
- INQUIRE DB2TRAN
- INQUIRE EXITPROGRAM
- INQUIRE TASK
- SET DB2CONN
- SET DB2ENTRY
- SET DB2TRAN.

So, you now know how DFHEISUP works – it uses a list of EXEC CICS commands that you supply and it reports which ones it finds. To make things easier, I have supplied a list of all the EXEC CICS API and SPI commands in DFHEISUP format that you can access on www.xephon.com/extras/krasunapi.txt and www.xephon.com/extras/krasunspi.txt. You can pick and choose from this list which commands you want to find; for example, taking the full lists and removing the threadsafe commands listed above will tell you all the non-threadsafe commands used in your CICS programs. Please read the comments on ADDRESS CWA and GETMAIN SHARED!

The technique shown here is expandable to other queries – for example, which programs use EXEC CICS RETURN COMMAREA and do not use the IMMEDIATE option so are possible end points of pseudo-conversations that could be broken into by an EXEC CICS START to the terminal?

Andy Krasun
Senior Enterprise Transaction Consultant
IBM UK Laboratories (UK)

CICS TS V2.3 new features and Java SDK 1.4.1 planning information

Most sites are still running CICS TS 1.3 and wondering why they should move to CICS TS 2.2 or 2.3. This article will provide you with the following: a high-level overview of the new features offered in CICS TS 2.3, Java upgrade planning information, some background to Java evolution under CICS, and an indication of the direction IBM is taking with future CICS development.

IBM announced CICS TS V2.3 on 28/10/03 (announcement letter no 203-296), with the planned GA date of 19/12/03. With this new release IBM further enhanced CICS EJB support and performance by re-architecting JVM and exploiting SDK V1.4.1. CICS TS 2.3 has enhanced Java execution by improving behaviour under stress – reducing storage and start-up requirements for a JVM.

CICS TS 2.3 provides a robust, high-performance environment for enterprise applications written in Java. Java applications now have a high degree of isolation from each other and can achieve execution speeds comparable to that of procedural languages.

WHY USE JAVA UNDER CICS?

The wide use and popularity of the Internet and Java have created a lot of opportunities and challenges for most shops, where the company sees the Internet as a way to reach out directly to its customers. Java, with ‘write once, run anywhere’ capabilities, is seen as the programming language of the future . Most companies develop new applications with GUI front-ends, but still use existing legacy CICS COBOL code. Using Java reduces programming costs and gives developers the freedom to deploy their applications on any platform, and improves time-to-market by exploiting Java technologies and using modern application development tools.

CICS TS Java support helps to evolve from the 3270 procedural COBOL world to a Web-based, object-oriented Java world. New enhancements in CICS TS allow users to exploit existing CICS applications to the full and make them more easily available to e-business applications. CICS exploits a completely new JVM, specially designed to optimize the performance of short-running programs such as CICS transactions.

The persistent reusable JVM supports special techniques that enable system and middleware code to re-initialize the JVM before it is reused for the next Java program. This ensures that there can be no interference between the applications, or between programs running on behalf of different end users, who serially reuse the same JVM.

WHAT ARE CICS CONNECTORS?

CICS TS V2 introduced CICS Connector for CICS TS, providing an option for a Java program or enterprise bean to communicate with an existing (non-Java) CICS application, typically running under another CICS system, executing unchanged existing CICS code. Communication is established by means of a COMMAREA within the local system or to any CICS system that supports DPL.

CICS applications can now use the Common Connector Framework (CCF) client API to allow portability between CICS and non-CICS servers of Java application classes that invoke CICS programs.

Using the CICS Connector enables the use of applications like WebSphere Studio Application Developer or VisualAge for Java

Enterprise Access Builder to create simple command beans and navigator beans. These command and navigator beans are easily generated, requiring limited knowledge of CICS programming. Such command and navigator beans are portable across a number of platforms that support the CICS Connector. For example, they could equally well be used from a servlet running on a WebSphere Advanced Edition on a distributed platform.

Java programmers with little or no knowledge of CICS can easily reuse CICS applications. Java client applications should be portable between Java-enabled platforms with little or no modification, and, in particular, should be portable from a non-CICS environment such as WebSphere Application Server into the CICS environment.

CCI CONNECTOR FOR CICS TS 2.3

CICS TS V2.3 provides a new method for invoking a procedural application from a Java client. This function uses the standard interface defined in the J2EE Connector Architecture (JCA) specification 1.0 - JSR016, called the Common Client Interface Connector for CICS TS (CCI Connector for CICS TS). It replaces the CICS Connector for CICS TS, introduced in CICS TS V2.1.

The CCI Connector for CICS TS enables a Java program or an Enterprise bean running under CICS TS 2.3 to communicate with Java or non-Java COMMAREA application program, wherever that application program resides – whether in the same CICS region as the invoking Java class, or in another CICS region.

The CCI Connector for CICS TS is functionally equivalent to the EXEC CICS LINK PROGRAM() COMMAREA() command, with information being sent to the target CICS application in a COMMAREA.

JAVA EVOLUTION WITH CICS TS V1.3, V2.2, AND V2.3

In order to see how far IBM has advanced with Java support under CICS, let’s take a look at Java evolution under CICS TS V1 and V2.

CICS TS V1.3

CICS TS V1.3 was the first CICS release to support Java applications and provide support for the CORBA IIOP protocol. Support for the following features/products was introduced:

- Java – JDK 1.1.8 for OS/390.
- High Performance Java.
- JCICS.
- JDBC 1.2.
- CORBA IIOP 1.0 (inbound only).
- VisualAge for Java V4 Enterprise Edition (JDK 1.2.2).

CICS TS V2.2

CICS TS V2.2 provides CICS EJB Server support and further enhanced support for the IIOP protocol. Support for the following features/products was introduced:

- SDK 1.3.1 for z/OS – exploiting the persistent JVM.
- HPJ support for migration purposes only.
- JPDA plus Distributed Debugger.
- JDBC 1.2 and 2.0.
- JDBC access to IMS.
- CICS Connector for CICS TS (CCF).
- WebSphere 4.0 distributed container.
- WebSphere Application Assembly Tool.
- EJB 1.1 – session beans only.
- JNDI 1.2 – LDAP plus COS Naming service.

- Resource Manager for Enterprise Beans (RMEB).
- IIOP 1.1 inbound and outbound.
- CORBA 2.1 stateless objects.
- WebSphere Studio Application Developer.
- Enterprise Developer V4 or V5.

CICS TS V2.3

In the latest release of CICS, IBM has made enhancements primarily in JVM performance, JCICS functionality, and EJB support. Support for the following features/products was introduced:

- SDK 1.4.1 for z/OS – exploiting the persistent JVM shared class cache.
- HPJ support for migration purposes – to be withdrawn in the future.
- JCICS – new and enhanced classes.
- CICS Web support to the JCICS classes.
- Common Client Interface (CCI) Connector for CICS TS.
- WebSphere V4 and V5 AAT or WebSphere Studio.
- EJB 1.1 session beans only.
- Toleration of EJB 2.0 JARs.
- IIOP 1.2 plus enhanced authentication plus encryption.
- CORBA 2.3 stateless objects.
- WebSphere Studio Application Developer or Enterprise Developer V4 or V5.

CICS TS V2.3 NEW FEATURES

Here are highlights of the CICS TS V2.3 new features:

- E-business:
  - Java enhancements using SDK 1.4.1, exploiting JVM shared class cache.
  - reduced JVM start-up time.
  - storage protection for Java programs.
  - EJB hardening and performance improvements.
  - enhancements to IIOP authentication and encryption.
  - JDBC V2.0 support (requires DB2 Version 7 with APAR PQ56655 (UQ65774) applied).
- Application modernization:
  - new JCICS classes for the CICS APIs WEB, DOCUMENT, and EXTRACT.
  - new CCI Connector for CICS TS.
  - interactive, end-to-end, debugging enhancements.
  - XML capability for COBOL and PL/I.
  - Java remote debug.
  - SOAP for CICS SupportPac.
  - new threadsafe commands.
  - DB2 restart-light and JDBC support.
  - important performance improvements in CICS-DB2 attachment.
  - support of ACCUM option on BMS requests for transactions using the Link3270 bridge.
  - the ability to cancel suspended or looping tasks.
- Enterprise management:
  - CICSPlex SM enhancements:
    - CPSM NT agent.

- CPSM definitional WUI.
- CPSM WLM for LINK3270 Bridge.
- CPSM new function support.
- CPSM architectural limits.

- definitional Web user interface.
- workload management for the LINK3270 Bridge.
- architectural improvements.
- sign-on retention
- CF system-managed rebuild/duplexing.
- enhanced support for CICS data-sharing servers.
- DB2 enhancements:
  - DB2 group attach (requires DB2 UDB for OS/390 Version 7.1, with APARs PQ44614, PQ45691, and PQ45692).
  - CICS-DB2 performance (OTE).
  - RMI purge.
  - CEMT INQ DB2.
- read time-out on MRO connections.
- Java and EJB trace and serviceability.
- statistics enhancements.

ENHANCED JAVA INFRASTRUCTURE

CICS TS 2.3 made major improvements to Java support and JVM infrastructure, building on two previous releases of CICS TS V2. Persistent reusable JVM enables significant performance optimizations for any Java program running in CICS, including Enterprise beans, via the following mechanisms:

1. Serially reusing a JVM – the initialization cost for a Java application is dramatically reduced.
2. Discarding at program termination all the application objects created by the Java program – the performance overhead of the more usual garbage collection mechanism is effectively eliminated.

CICS JVM supports special techniques that enable system and middleware code to re-initialize the JVM before it is reused for the next Java program. This ensures that there can be no interference between the applications, or between programs running on behalf of different end users, that serially reuse the same JVM. CICS supports a set of JVM instances ready for use within each CICS address space and provides management of the pool of JVMs to optimize throughput and to allow Java classes to be replaced without requiring CICS to be restarted.

ENHANCEMENTS IN THE EXPLOITATION OF JVM INFRASTRUCURE

CICS TS 2.3 has the following enhancements in the exploitation of JVM infrastructure, which result in better performance and lower overhead:

- The exploitation of the shared classes configuration of the JVM, with the use of the shared class cache. This improves efficiency by reducing the storage and start-up requirements for a JVM.
- The introduction of a new mode of operating JVMs for applications that do not require the overhead of resetting the JVM’s state between transactions. This mode is referred to as running ‘continuous’ JVMs and it reduces the CPU per transaction.
- The introduction of a dedicated storage monitor for JVMs, which improves behaviour when a CICS region is short on storage used by JVMs.
- A new selection mechanism for creating, managing, and

allocating work into JVMs. JVMs can have different characteristics and are grouped for management purposes. This improves behaviour under stress.

There are also performance improvements for Java classes. The launcher code for these methods has been significantly optimized to reduce the overhead in pathlength imposed on each execution – increasing the Java activity throughput in CICS TS V2.3.

CICS TS AND WEBSPHERE EJB CONTAINER

With CICS TS 2.2 and 2.3 EJB session beans support, Java applications can exploit CICS via open Java Enterprise APIs. A tool like WebSphere Studio may be used to develop these applications and deploy them under CICS.

CICS stateful session beans may be seen as an equivalent of pseudo-conversational programs and are suitable for a wide range of applications. This technology enables transactional peer-to-peer interoperation with WebSphere Application Server and other CORBA-compliant servers using IIOP, which may be used as an advanced connector. By exploiting the WebSphere EJB container, CICS enables construction of reusable business logic components that are binary portable between CICS and WebSphere and may be deployed in either environment using the same toolsets.

JAVA REQUIREMENT

APAR PQ74985 (PTF UQ77468) delivers the Java 2 SDK 1.4.1 upgrade of the IBM SDK for z/OS, Java 2 Technology Edition, V1.4 to the latest service levels and to new levels of functionality.

PQ74985 includes the following:

- Upgrade to the SDK 1.4.1 level, with incorporation of the SUN 1.4.1 level of code.
- With this APAR and the appropriate LE PTFs, IBM SDK for

z/OS Java 2 Technology Edition is now also supported on z/OS V1.2 and z/OS V1.3.

CICS TS 2.3 requires Java SDK 1.4.1 level for run-time, with PTF for APAR PQ79281(PTF UQ81134), which is a service refresh SR1. You need to apply PQ79281 to upgrade from base SDK 1.4.1 code.

SDK 1.4.1 z/OS REQUIRED MAINTENANCE

Please note that Java will not function without the APARs outlined below.

The following APARs (and corresponding PTFs) are required for your z/OS V1.4 (5694-AO1) or z/OS.e V1.4 (5655-G52) system:

- PQ69548 (PTF UQ77316) (Note: this would be the only PTF required if your service is already at 11 April or later.)
- PQ54074 (PTF UQ71321)
- PQ66118 (PTF UQ71530)
- PQ60748 (PTF UQ71263)
- PQ72565 (PTF UQ76197)
- PQ61928 (PTF UQ70204)
- PQ60892 (PTF UQ71361)
- OW54362 (PTF UW95336).

The following APARs (and corresponding PTFs) are recommended for your z/OS V1.4 (5694-AO1) or z/OS.e V1.4 (5655-G52) system:

- PQ72197 (PTF UQ75719)
- PQ72266 (PTF UQ75723)
- PQ71219 (PTF UQ76026).

The following APARs (and corresponding PTFs) are required for your z/OS V1.3 (5694-AO1) or z/OS.e V1.3 (5655-G52) system:

- PQ69546 (PTF UQ77201)
- PQ69548 (PTF UQ77315)
- PQ61928 (PTF UQ70203)
- PQ60748 (PTF UQ71262)
- OW54362 (PTF UW95335).

The following APARs (and corresponding PTFs) are recommended for your z/OS V1.3 (5694-AO1) or z/OS.e V1.3 (5655-G52) system:

- PQ72197 (PTF UQ75718)
- PQ72266 (PTF UQ75722)
- PQ71219 (PTF UQ76025).

The following APARs (and corresponding PTFs) are required for your z/OS V1.2 (5694-AO1) system:

- PQ70450 (PTF UQ77341)
- PQ69546 (PTF UQ77200)
- PQ69548 (PTF UQ77314)
- PQ61928 (PTF UQ70202)
- PQ60748 (PTF UQ71261)
- OW54362 (PTF UW95334).

The following APARs (and corresponding PTFs) are recommended for your z/OS V1.2 (5694-AO1) system:

- PQ72197 (PTF UQ75717)
- PQ72266 (PTF UQ75721)
- PQ71219 (PTF UQ76024) .

SDK 1.4.1 PERFORMANCE ENHANCEMENTS

SDK V1.4 takes advantage of enhanced z/OS linkage capabilities

(XPLINK) for greatly-improved performance. This enhancement is completely transparent to the end users. However, any application code that creates a JVM itself and interacts with the JVM via Java Native Interface (JNI) or any other ‘call’ interface must create the Language Environment (LE) enclave specifying that LE should setup an XPLINK environment. That environment will include an XPLINK-specific run-time library and stack format. This XPLINK LE enclave must be in place prior to creating the JVM. This can be accomplished by including setting a run-time option (XPLINK(ON)) or recompiling the launching application code to be XPLINK. IBM’s Red Book SG245991, *XPLink: OS/390 Extra Performance Linkage* has these options and a description of the XPLINK technology.

CO-EXISTENCE RESTRICTION FOR SMP/E INSTALLED SDK VERSIONS

You will also need to be aware of how to run multiple versions of SDK in your environment, since applications like CICS TS 2.2 and WebSphere for z/OS V4 will still require SDK V1.3.*x* (IBM Developer Kit for OS/390, Java 2 Technology Edition, 5655-D35).

Please be aware that installing the JAVA SDK 1.4 product (HJVA140) in the same SMP/E zones as 1.3 will remove 1.3 (HJVA130). Therefore, you will need to install HJVA140 in separate zones and libraries. This can be accomplished by following these instructions:

Create separate target and dlib zones in your CSI:

```
SET BDY(GLOBAL) .
UCLIN .
  ADD GLOBALZONE ZONEINDEX(
    (#tzone,#csi,TARGET)
    (#dzone,#csi,DLIB)) .
ENDUCL .
SET BDY(#tzone) .
UCLIN .
  ADD TARGETZONE(#tzone) RELATED(#dzone)
  OPTIONS(#options) SREL(ZØ38) .
ENDUCL .
```

```
SET BDY(#dzone) .
UCLIN .
  ADD DLIBZONE(#dzone) RELATED(#tzone)
  OPTIONS(#options) SREL(ZØ38) .
ENDUCL .
```

Change *#csi* to your existing csi, *#tzone* to your new target zone, *#dzone* to your new dlib zone, and *#options* to the existing options name.

Create the new libraries using the AJVALLOC job. Allocate these libraries with a middle-level qualifier to differentiate them from the 1.3 libraries. The recommended MLQ is V1R4M0. Follow the instructions in the job.

Create the new HFS dataset using the AJVISHFS job. The recommended mountpoint is *../usr/lpp/java/J1.4*. Note that J1.4 must be the final directory in this path. Follow the instructions in the job. Edit and run the AJVISMKD job, again following the instructions in the job.

Create the required DDDEFs using the AJVDDDEF job with appropriate modifications as instructed in the job. Be sure to include the MLQ as part of the dataset names.

Continue with the installation as documented in the program directory.

Customization – assuming the recommended mountpoint was used, set the PATH environment variable:

```
    export PATH=.../usr/lpp/java/J1.4/bin:$PATH
```

CONCLUSION

CICS support for Java has been dramatically enhanced. For Web-enabled CICS applications, using Java under CICS and CICS Connectors simplifies development of new e-business applications, reduces the overall cost of computing, and allows users to reuse their existing programming and operational skills without requiring a fundamental re-education. CICS TS 2.3 not only provides functional enhancements in Java functionality under CICS, but also addresses the issue of performance (very high CPU overhead), which has been preventing some shops making use of CICS EJB server.

IBM's strategy is to encourage the development of new e-business applications, facilitating application development for CICS applications in both the traditional and Enterprise Java environments. The benefit that this strategy brings is the ability to evolve the applications without reinventing the existing business logic.

Elena Nanos
IBM Certified Solution Expert in CICS Web Enablement
Zurich NA (USA)

EXEC interface trace tips

CICS application program debugging can be made easier by analysis of the trace entries recorded when CICS is executing commands issued by the program under investigation. In particular, the EXEC CICS trace entries associated with the commands themselves are a good source of information about the program environment being studied.

EXEC CICS events are shown by EIP ENTRY and EIP EXIT trace points. By default, these are standard level 1 trace points. They denote the places where application programs issue EXEC CICS commands to invoke a CICS API or SPI function. The EIP ENTRY trace point is issued on entry to CICS from an application; the EIP EXIT trace point is issued just prior to the flow of control returning to the application from CICS. As such, they delimit the times between CICS and the application being in control.

The EIP ENTRY and EXIT trace points are issued from DFHEIP, the CICS EXEC Interface Program. Their trace point ID is AP 00E1. They contain the basic amount of data needed for analysis of a program’s flow of control when invoking CICS commands. If EIP tracing is extended (by setting the EI component to 1-2 under the CETR trace components panel), an additional pair of DFHEIP trace entries are recorded by CICS. These standard level 2 trace point IDs are AP E160 and AP E161 (on entry to and exit from DFHEIP, respectively). These level 2 EI trace entries give far more detail about the command being executed by CICS, such as formatting the parameters, displaying their storage addresses within the CICS address space, showing the EIB, etc. While not expected to be used during production, they are a great aid in application development and debugging, and can be considered the trace equivalent of stepping through the application logic via CEDF.

This article describes some useful debugging tips and techniques that may be applied when working with EIP trace entries.

THE RETURN ADDRESS FIELD

EIP trace entries contain a return address value in the RET field. This is shown only when the trace entries are formatted using the SHORT or FULL trace format option. (Note: the RET value is not displayed if the ABBREVIATED (ABBREV) formatting option is used.)

For example, the following (edited) trace example shows an EXEC CICS ASKTIME command with a RET address of 0C907770:

```
ØØ337 QR    AP ØØE1 EIP   ENTRY ASKTIME-ABSTIME
     RET-ØC9Ø777Ø 1Ø:16:27.2234375941     =ØØ4281=
```

The example also shows that the trace entry was issued by task number 00337, running under the QR TCB at 10:16:27, and was trace entry number 004281 in this run of CICS tracing.

The RET address is the location within memory in the CICS address space where the flow of execution control will return after CICS has processed the EXEC CICS command. In other words, it is the location of the next instruction to be executed within the application program load module. If the entry point of the executing program can be determined, the offset of the EXEC CICS command within the program may be calculated. The entry point can be found in a number of ways. If a CICS transaction dump is available, the entry point will be formatted by the dump formatter. (Similarly, a CICS system dump will show the entry point via the LD VERBX option.) An example from a transaction dump is shown below:

```
TRANSACTION DUMP MODULE INDEX:
LOAD PT.      NAME      ENTRY PT     LENGTH

ØC9Ø273Ø    TESTPGM1    ØC9Ø275Ø    ØØØØ1F8Ø

ØC9Ø731Ø    ANDYPGM     ØC9Ø733Ø    ØØØØ1D1Ø

ØC9Ø9Ø2Ø    TESTPGM2    ØC9Ø9Ø4Ø    ØØØØ2C98
```

More straightforwardly, preceding trace entries may contain the LD 0002 trace entry showing the successful call to acquire the program in memory, and this will reveal the program's load and entry points. This last technique is viable, provided the loading of the program was recent enough for its CICS trace entries to remain within the internal trace table, and so not have been lost because of new entries wrapping around the table. (Such wrapping is less likely when dealing with the auxiliary trace datasets.)

As an example, the program ANDYPGM, having issued the EXEC CICS ASKTIME command, generated the following CICS Loader exit trace data:

```
LD ØØØ2 LDLD EXIT - FUNCTION(ACQUIRE_PROGRAM) RESPONSE(OK)

                            ENTRY_POINT(ØC9Ø733Ø) LOAD_POINT(ØC9Ø731Ø)
```

Subtraction of the RET address from the program's entry point shows that the EXEC CICS ASKTIME command was located at offset X'440' in the program ANDYPGM.

Note that a program's load point normally addresses the CICS command level stub. This is link-edited to the program and used as a route for the flow of execution from the program into CICS, when executing EXEC CICS commands within the program.

Conversely, the entry point generally points to the start of the application itself. This can be confirmed from the link-edit mapping generated when the program load module was created.

Use of the RET address can be very helpful when tying together the dynamic environment of EXEC CICS commands in a trace and the static contents of a CICS application program listing. For example, suppose an EXEC CICS RETURN command in a COBOL application was being investigated. Using the above technique, assume that in this example an offset of X'51E' was calculated for the command within the program. By locating this offset in the program listing, it can be seen that the offset X'51E' lies immediately after a BALR 14,15 command, and that it is part of the assembly statements generated from a CALL verb (statement number 87):

```
ØØØØ87  CALL
    ØØØ4FA  D21Ø DØAØ AØF3      MVC  16Ø(17,13),243(1Ø)      TS2=8
    ØØØ5ØØ  413Ø DØAØ           LA   3,16Ø(Ø,13)             TS2=8
    ØØØ5Ø4  5Ø3Ø DØ98           ST   3,152(Ø,13)             TS2=Ø
    ØØØ5Ø8  968Ø DØ98           OI   152(13),X'8Ø'           TS2=Ø
    ØØØ5ØC  411Ø DØ98           LA   1,152(Ø,13)             TS2=Ø
    ØØØ51Ø  58FØ AØØØ           L    15,Ø(Ø,1Ø)              V(DFHEI1  )
    ØØØ514  41ØØ 9138           LA   Ø,312(Ø,9)              CLLE@=1
    ØØØ518  58CØ 9Ø8Ø           L    12,128(Ø,9)             TGTFIXD+128
    ØØØ51C  Ø5EF                BALR 14,15

    ØØØ51E  58CØ 9ØE8           L    12,232(Ø,9)             TGTFIXD+232

    ØØØ522  4ØFØ 2ØØØ           STH  15,Ø(Ø,2)               RETURN-CODE
```

The CALL statement is the COBOL instruction generated as part of the CICS translation process that converted the EXEC CICS RETURN command into its COBOL natural language equivalent. The BALR instruction is part of the assembled output generated by the COBOL compilation step; this is how the flow of execution control leaves the application program and passes to CICS, in order to process the EXEC CICS command. The flow is via the CICS command-level stub, link-edited to the application program.

If the COBOL program listing is analysed further, it can be seen that the EXEC CICS RETURN command (statement 86) was commented out as part of the CICS translation process, and replaced by the COBOL CALL verb (statement 87):

```
ØØØØ86     *EXEC CICS RETURN END-EXEC.
ØØØØ87          Call 'DFHEI1' using by content
x'ØeØ8ØØØØØ7ØØØØ1ØØØfØfØfØf1f5  ØØØØ88     -    '4Ø4Ø4Ø' end-call.
FIELD A AND FIELD B DATA VALUES
```

Limited information is provided by EXEC CICS standard level 1 trace data. However, the contents of FIELD A and FIELD B can be of considerable use when debugging an application. In the edited trace entry example shown below, a program running as CICS task number 01310 has issued an EXEC CICS READ command against a VSAM file:

```
Ø131Ø QR    AP ØØE1 EIP   ENTRY READ    ØØ25Ø6B4 , Ø9ØØØ6Ø2
=592452=
```

The content of FIELD A is 002506B4. This is the address of the application’s save area (ie for PL/I it is the program’s DSA, for Assembler the EISTG, for COBOL the working storage, etc). It is useful because it can provide a very quick means of looking through a trace to identify switches in application control and determine the scope of potential loops within application processing. If further analysis of this particular task’s EXEC CICS commands were required, the following trace formatting command could be used to return only the EIP level 1 trace entries:

```
ABBREV,TASKID=(Ø131Ø),TYPETR=(APØØE1)
```

Any switches between programs would be revealed by changing values in the FIELD A data, since different programs would have different save area addresses in memory. Similarly, repeated patterns of EXEC CICS commands in the trace, with matching repeated FIELD A values, may well reveal the scope of a yielding loop within the program or programs. A yielding loop is one where control returns to the CICS dispatcher within the bounds of the loop, so that (as far as CICS is concerned) a looping application is behaving validly. Loop analysis can be further enhanced by studying the return addresses in the RET fields for the EXEC CICS commands contained within the bounds of the loop and applying the technique described earlier for identifying their offsets within the looping programs.

In the example, the EXEC CICS READ command had a FIELD B value of 09000602. The first byte of the data shows that the application was executing in storage key 9 at the time it issued the EXEC CICS command. CICS runs in storage key 8, and DFHEIP will switch the storage key environment between keys 8 and 9 as appropriate when executing EXEC CICS commands. Storage protection, by means of key switching, is controlled by the STGPROT CICS system initialization parameter.

The last halfword of FIELD B is the EIBFN value for the command being executed. In the case of an EXEC CICS READ command, the EIBFN is 0602. The 06 denotes that it is a CICS file control command, the 02 that it is a READ. In the earlier EXEC CICS RETURN example, the EIBFN is 0E08, and can be seen as the start of the parameter argument data constructed by the CICS translator, in statement 87 of the COBOL program. EIBFN values are documented in the CICS *Application Programming Reference* manual.

CONCLUSION

I hope this article has helped shed some light on the use of EIP trace entries within CICS, and the role they can play in assisting with the debugging of CICS application programs.

Andy Wright (andy_wright@uk.ibm.com)
CICS Change Team
IBM (UK)

CICS USER EXIT LIST with the transaction EXIT – part 2

This month we conclude the code for the transaction to list all exits (GLUEs and TRUEs) in a CICS address space.

```
CWAANTDSECT DSECT
CWA_ANT_HILFE        DS CL16              HILFETEXT
```

```
CWA_ANT_TASTEN        DS CL16                ANZEIGE DER TASTENBELEGUNG
CWA_ANT_AUSGANG       DS CL16                BEENDEN EINER FUNKTION
CWA_ANT_REFRESH       DS CL16                WIEDERHERSTELLEN
CWA_ANT_UPDATE        DS CL16                DATEN SPEICHERN
CWA_ANT_RUECKWAERTS   DS CL16                RÜCKWÄRTS BLÄTTERN
CWA_ANT_VORWAERTS     DS CL16                VORWÄRTS BLÄTTERN
CWA_ANT_AKTION        DS CL16                AKTIVIERUNG ACTIONBAR
CWA_ANT_UNTERBRECHEN  DS CL16                VORGANGSUNTERBRECHUNG
CWA_ANT_ABBRUCH       DS CL16                ABBRUCH
CWA_ANT_EINSTIEG      DS CL16                ZURÜCK ZUM EINSTIEGSBILD
CWA_ANT_AUSWAHL       DS CL16                ZURÜCK ZUM AUSWAHLBILD
CWA_ANT_SICHERN       DS CL16                EINFRIEREN VON DATEN
CWA_ANT_LINKS         DS CL16                BLÄTTERN NACH LINKS
CWA_ANT_RECHTS        DS CL16                BLÄTTERN NACH RECHTS
CWA_ANT_ANFANG        DS CL16                ANZEIGE DER ERSTEN SEITE
CWA_ANT_SCHLUSS       DS CL16                ANZEIGE DER LETZTEN SEITE
CWA_ANT_ABMELDEN      DS CL16                ZSS-ABMELDUNG
CWA_ANT_DRUCKEN       DS CL16                DRUCKEN (PA2)
CWA_ANT_LOESCHEN      DS CL16                LÖSCHEN BILDSCHIRM
CWA_ANT_DATENFREIGABE DS CL16                DATENFREIGABE
CWA_ANT_HILFE_ANLEGEN DS CL16                BOSHELP HILFE ANLEGEN
CWA_ANT_SUCHEN        DS CL16                SUCHEN
CWA_ANT_EURODM        DS CL16                UMRECHNEN EURO/DM
CWAANTDSECTE          EQU   *
*---------------------------------------------------------------------
*         ENDE DER DSECT FUER AKTIONSNAMENTABELLE                    *
*---------------------------------------------------------------------
*      END INCLUDE++
         SPACE
*      INCLUDE++ CSGWA             INCLUDE GWA ASSEMBLER STRUCTURE
*          DATA SET CSGWA      AT LEVEL Ø18 AS OF 25/Ø8/ØØ
***********************************************************************
* CHANGE ACTIVITY:                                                    *
***********************************************************************
*                                  ------------------------------------
GWADSECT             DSECT         GLOBAL WORK AREA FOR CICS/ESA-EXIT'S
*                                  ------------------------------------
GWA_CWA              DC   AL4(Ø)   ADDR OF CWA
GWA_UNUSED           DC   CL96' '  UNUSED SPACE
*                                  ------------------------------------
GWA_EXIT             EQU  *        LAYOUT FOR EXIT-DEPENDENCIES ONLY
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XDLIPRE          ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XDLIPRE_OLDPSB   DC   CL8' '   ORIGINAL PSB NAME
GWA_XDLIPRE_NEWPSB   DC   CL8' '   SUBSTITUTED PSB NAME
GWA_XDLIPRE_MSGAREA  DC   CL1ØØ' ' MESSAGE-AREA FOR MVS-OPERATOR
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XLGSTRM          ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S     Ø1-Ø2
```

```
GWA_XLGSTRM_LSNAME   DC   CL26' '  LOG STREAM NAME                Ø1-Ø2
GWA_XLGSTRM_WORK     DC   CL8' '   WORK AREA                      Ø1-Ø2
GWA_XLGSTRM_MSGAREA  DC   CL1ØØ' ' MESSAGE-AREA FOR MVS-OPERATOR  Ø1-Ø2
                     ORG  GWADSECT+248                            Ø1-Ø2
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XMEOUT           ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XMEOUT_EXT_AREA  DS   F
GWA_XMEOUT_EXT_SIZE  DS   F
GWA_XMEOUT_NEXT      DS   F
GWA_XMEOUT_LAST      DS   F
GWA_XMEOUT_LENGTH    DS   F
GWA_XMEOUT_DOWO      DS   D
GWA_XMEOUT_MSG       DS  ØCL6
GWA_XMEOUT_MSG_DOM   DS   CL2
GWA_XMEOUT_MSG_NUM   DS   CL4
GWA_XMEOUT_MSGAREA   DS  ØCL1ØØ
GWA_XMEOUT_TIME      DS   CL8,CL1
GWA_XMEOUT_DFHMSG    DS   CL9,CL1
GWA_XMEOUT_APPLID    DS   CL8,CL1
GWA_XMEOUT_TRANID    DS   CL4,CL1
GWA_XMEOUT_PROGRAM   DS   CL8,CL1
GWA_XMEOUT_TERMID    DS   CL4,CL1
GWA_XMEOUT_USERID    DS   CL8,CL1
GWA_XMEOUT_INSERT1   DS   CL14,CL1
GWA_XMEOUT_INSERT2   DS   CL14,CL1
GWA_XMEOUT_INSERT3   DS   CL14
                     ORG  GWADSECT+248
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XPCABND          ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XPCABND_UNUSED   DC   CL1Ø' '  UNUSED AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XPCFTCH          ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XPCFTCH_TIME     DC   CL1Ø' '  ' HH:MM:SST'
GWA_XPCFTCH_TIMEP    DC   PL4'Ø'   HHMMSSTC
GWA_XPCFTCH_UNUSED   DC   CL1Ø' '  UNUSED AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XTSERE           ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XTSERE_DOWO      DC   D'Ø'     ONLY FOR CVD
GWA_XTSERE_MASK1FLD  DC   CL4' '   ONLY FOR EDIT
GWA_XTSERE_MSGAREA   DC   CL1ØØ' ' MESSAGE-AREA FOR MVS-OPERATOR
GWA_XTSERE_QUEUE     DC   CL16' '  DATAID FOR TS-QUEUES           Ø1-Ø1
GWA_XTSERE_SYSID     DC   CL4' '   SYSID FOR FUNCTION SHIPPING
GWA_XTSERE_WORK      DC   CL1Ø' '  WORK AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XZCATT           ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
```

```
GWA_XZCATT_UNUSED    DC   CL1Ø' '  WORK   AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XICTENF          ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XICTENF_UNUSED   DC   CL1Ø' '  WORK   AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XMNOUT           ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XMNOUT_WORK      DC   CL16' '  WORK   AREA
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XTDOUT           ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XTDOUT_UNUSED    DC   CL1Ø' '  WORK   AREA   (UNUSED)
*                                  ------------------------------------
*                                  ------------------------------------
GWA_XEIIN            ORG  GWA_EXIT START-VALUE FOR ALL EXIT'S
GWA_XEIIN_MSGAREA    DC   CL1ØØ' ' MESSAGE-AREA FOR MVS-OPERATOR
GWA_XEIIN_EIPFN      DC   CL2' '   EIP-FUNCTION IN HEX
GWA_XEIIN_EIPFN_TR   DC   CL5' '   EIP-FUNCTION IN CHARACTER
*                                  ------------------------------------
*                                  ------------------------------------
                     ORG  GWADSECT+248
GWADSECT_E           EQU  *        END OF GLOBAL WORK AREA (LENGTH)
*                                  ------------------------------------
*      END INCLUDE++
         SPACE
DFHEISTG       DSECT
ABCODE         DS   CL4            ABEND CODE
BYTE           DS   CL2
COUNTER        DS   PL2            LINE COUNTER
DOUBLEW        DS   D
ENTRYNAME      DS   CL8
EXIT           DS   CL8            EXIT-POINT
EXITADDR       DS   CL8            CONVERTED EXIT ADDRESS
EXITPROGRAM    DS   CL8
GAENTRY        DS   F              EXIT ENTRY ADDRESS
GAENTRYNAME    DS   CL8
GAEXITPOINT    DS   CL8
GALENGTH       DS   H
LENGTH         DS   H              OUTPUT LENGTH
NOTAGAIN       DS   C
NUMEXITS       DS   H
OUTAREA        DS   CL7Ø           TERMINAL-OUTPUT-AREA
OUTLINE        DS   ØCL79          PREPARE SINGLE LINE
               ORG  OUTLINE+79
QUEUE          DS   CL8
RESP           DS   F              RESPONSE-CODE
SAVENTRNA      DS   CL8
SAVEXIPGM      DS   CL8
STARTSTATUS    DS   F              CVDA
```

```
STOR8          DS   F              SAVE REGISTER 8
USRID          DS   CL8            USERID
         SPACE
         COPY  CMEXIT              INCLUDE MAP STRUCTURE
         EJECT
* ******************************************************************* *
*        M a i n  p r o g r a m                                       *
* ******************************************************************* *
         SPACE
CSEXIT   DFHEIENT CODEREG=(R2,R3),DATAREG=(R11),EIBREG=(R12)
         SPACE
CSEXIT   AMODE ANY
CSEXIT   RMODE ANY
         EXEC  CICS ADDRESS    CWA         (CWAPTR)                    *
               RESP(RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BE    CSPRØ1ØØ
         MVC   ABCODE,=C'CWAA'         SYSID can't be assigned
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
CSPRØ1ØØ EQU   *
         ST    CWAPTR,STOR8            Save CWA address
         EXEC  CICS ASSIGN     USERID      (USRID)                     *
               RESP(RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BE    CSPRØ2ØØ
         MVC   ABCODE,=C'USID'         USERID can't be assigned
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
CSPRØ2ØØ EQU   *
         BAS   R6,RECEMAP              Receive the map
         CLC   RESP,DFHRESP(MAPFAIL)   First invoking or empty screen?
         BE    CSPR1ØØØ
         MVC   NERRO32O,BLANKS         Clear error message
         CLI   EIBAID,DFHCLEAR         Complete TA?
         BE    ENDCLEAU
         CLI   EIBAID,DFHENTER         Enter?
         BE    CSPRØ4ØØ
         CLI   EIBAID,DFHPF7           Backward?
         BE    CSPR1ØØØ
         CLI   EIBAID,DFHPF8           Forward?
         BE    CSPR1ØØØ
         SPACE
CSPRØ4ØØ EQU   *
         MVC   NERRO32O(L'MSGINVKE),MSGINVKE Invalid function key
         SPACE
CSPR1ØØØ EQU   *
         CLC   RESP,DFHRESP(MAPFAIL)
         BNE   CSPR2ØØØ
         BAS   R6,SENDMAPO             First invoking or empty screen!
         SPACE
CSPR2ØØØ EQU   *
         BAS   R6,INQUIRE              Browse the exit's
         BAS   R6,FILHDR               Fill map header
         BAS   R6,SENDMAP              Send map
         SPACE
RETURN   EQU   *
         EXEC  CICS RETURN
         SPACE
ENDCLEAU EQU   *                       Complete with clean-up program
         EXEC  CICS XCTL PROGRAM('CSTCTUC')
         EJECT
* ******************************************************************* *
*        S u b r o u t i n e s                                        *
* ******************************************************************* *
         SPACE
* ******************************************************************* *
*        F i l l  T h e  H e a d e r                                  *
* ******************************************************************* *
         SPACE
FILHDR   EQU   *
         L     CWAPTR,STOR8                           LOAD SAVED ADDR
         MVC   NTACODEO,EIBTRNID                      TRANSACTION CODE
         MVC   NUSRIDO,USRID                          RACF USERID
         MVC   NRZDATO(L'CWADATUM),CWADATUM           DATE FROM THE CWA
         MVC   NRZDATO+L'CWADATUM+1(L'CWAZEIT),CWAZEIT     CWA TIME
         MVC   NTERMNRO,EIBTRMID                      TERMINALID
         BR    R6
         SPACE
* ******************************************************************* *
*        S e n d  T h e  M a p                                        *
* ******************************************************************* *
         SPACE
SENDMAP  EQU   *
*
         EXEC  CICS SEND MAP('MEXIT')                                   *
                         MAPSET('CMEXIT')                               *
                         ERASE                                          *
                         FREEKB
*
         BR    R6
         SPACE
* ******************************************************************* *
*        S e n d  T h e  M a p  (M a p  o n l y!)                     *
* ******************************************************************* *
         SPACE
SENDMAPO EQU   *
         EXEC  CICS SEND MAP('MEXIT')                                   *
```

```
                         MAPSET('CMEXIT')                              *
                         MAPONLY
         BR    R6
         SPACE
* ******************************************************************* *
*        R e c e i v e  T h e  M a p                                  *
* ******************************************************************* *
         SPACE
RECEMAP  EQU   *
         EXEC  CICS RECEIVE MAP('MEXIT')                               *
                            MAPSET('CMEXIT')                           *
                            RESP(RESP)
         MVI   NOTAGAIN,C'Ø'               Set flag to zero
         MVC   SAVEXIPGM,NEX23O+25         Save for
         MVC   SAVENTRNA,NEX23O+37         browsing
         BR    R6
         SPACE
* ******************************************************************* *
*        I n q u i r e  E x i t p g m s                               *
* ******************************************************************* *
         SPACE
INQUIRE  EQU   *
         LA    LINEREG,NEXØØO              FIRST LINE IN THE MAP
         ZAP   COUNTER,=P'Ø'               FORMAT COUNTER
         EXEC  CICS INQUIRE EXITPROGRAM START                          *
                            RESP        (RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BNE   INQEND
         SPACE
INQNEXT  EQU   *
         EXEC  CICS INQUIRE EXITPROGRAM (EXITPROGRAM) NEXT             *
                            GAENTRYNAME (GAENTRYNAME)                  *
                            ENTRY       (WORKREG)                      *
                            GALENGTH    (GALENGTH)                     *
                            ENTRYNAME   (ENTRYNAME)                    *
                            NUMEXITS    (NUMEXITS)                     *
                            STARTSTATUS (STARTSTATUS)                  *
                            RESP        (RESP)
         CLC   RESP,DFHRESP(END)
         BE    INQEND
         CP    COUNTER,=P'24'                          Page full?
         BL    INQ1ØØØ
         MVC   NADVI31O(L'MSGADDEX),MSGADDEX           Add.exits av.
         B     INQEND
         SPACE
INQ1ØØØ  EQU   *
         ST    WORKREG,GAENTRY
         LA    R8,GAENTRY
         LA    R14,EXITADDR
         LA    WORKREG,4
         SPACE
INQ15ØØ  EQU   *
         SR    R4,R4
         SR    R5,R5
         IC    R4,Ø(R8)
         SRDL  R4,4
         SRL   R5,28
         STC   R4,BYTE
         STC   R5,BYTE+1
         TR    BYTE(2),TRTB
         MVC   Ø(2,R14),BYTE
         LA    R14,2(,R14)
         LA    R8,1(,R8)
         BCT   WORKREG,INQ15ØØ
         MVC   Ø(L'NEXØØO,LINEREG),BLANKS
*        MVC   13(L'GAENTRYNAME,LINEREG),GAENTRYNAME
         CLC   NUMEXITS,=H'Ø'                            TRUE?
         BNE   INQ2ØØØ
         MVC   13(4,LINEREG),=C'TRUE'
         B     INQ25ØØ
         SPACE
INQ2ØØØ  EQU   *
         MVC   13(4,LINEREG),=C'GLUE'
         SPACE
INQ25ØØ  EQU   *
         MVC   25(L'EXITPROGRAM,LINEREG),EXITPROGRAM    OK
         EXEC  CICS EXTRACT EXIT PROGRAM(EXITPROGRAM)                  *
                            ENTRYNAME   (GAENTRYNAME)                  *
                            GALENGTH    (GALENGTH)                     *
                            GASET       (DSECTREG)                     *
                            RESP        (RESP)
         CLI   NOTAGAIN,C'1'
         BE    INQ3ØØØ
         CLI   EIBAID,DFHPF8                             Forward?
         BNE   INQ3ØØØ
         CLC   MSGADDEX,NADVI31O                         Add. exits?
         BNE   INQ3ØØØ
         CLC   EXITPROGRAM,SAVEXIPGM                     Last
         BNE   INQNEXT
         CLC   ENTRYNAME,SAVENTRNA                       Entry?
         BNE   INQNEXT
         LA    LINEREG,NEXØØO              FIRST LINE IN THE MAP
         ZAP   COUNTER,=P'Ø'               FORMAT COUNTER
         MVI   NOTAGAIN,C'1'               SET FLAG TO "ON"
         MVC   NADVI31O,BLANKS             CLEAR LINE 31
         LH    WORKREG,DLENGTH             AREA LENGTH
         LA    R5,NEXØØL                   FIRST BYTE
         SPACE
INQ27ØØ  EQU   *
         MVI   Ø(R5),C' '                  FORMAT/CLEAR
         LA    R5,1(,R5)                   NEXT BYTE
         BCT   WORKREG,INQ27ØØ
         B     INQNEXT
         SPACE
INQ3ØØØ  EQU   *
*        MVC   25(L'EXITPROGRAM,LINEREG),EXITPROGRAM
*        MVC   13(L'GAEXITPOINT,LINEREG),GAEXITPOINT
*        MVC   37(L'GAENTRYNAME,LINEREG),GAENTRYNAME    NOT OK
         MVC   37(L'GAENTRYNAME,LINEREG),ENTRYNAME
         MVC   OUTLINE,BLANKS
         MVC   OUTLINE(L'EXITPROGRAM),EXITPROGRAM
         MVC   OUTLINE+1Ø(L'ENTRYNAME),ENTRYNAME
         MVC   49(L'EXITADDR,LINEREG),EXITADDR
         LH    WORKREG,NUMEXITS
         CVD   WORKREG,DOUBLEW
         OI    DOUBLEW+L'DOUBLEW-1,X'ØF'
         UNPK  OUTLINE+2Ø(2),DOUBLEW
         CLC   STARTSTATUS,DFHVALUE(STARTED)
         MVC   OUTLINE+3Ø(1Ø),=CL1Ø'STARTED'
         MVC   6Ø(1Ø,LINEREG),=CL1Ø'STARTED'
         BE    INQ4ØØØ
         MVC   OUTLINE+3Ø(1Ø),=CL1Ø'STOPPED'
         MVC   6Ø(1Ø,LINEREG),=CL1Ø'STOPPED'
         SPACE
INQ4ØØØ  EQU   *
         MVC   QUEUE(L'EIBTRMID),EIBTRMID
         MVC   QUEUE+L'EIBTRMID(L'EIBTRNID),EIBTRNID
         EXEC  CICS WRITEQ TS QUEUE(QUEUE)                             *
                              FROM(OUTLINE)                            *
                              RESP(RESP)
         LA    LINEREG,L'NEXØØL+L'NEXØØA+L'NEXØØO(LINEREG)    NEXT
         AP    COUNTER,=P'1'                                  LINE
         B     INQNEXT
         SPACE
INQEND   EQU   *
         EXEC  CICS INQUIRE EXITPROGRAM END                            *
                            RESP        (RESP)
         BR    R6
         SPACE
* ******************************************************************* *
*        S e n d  M e s s a g e  T o  C o n s o l e                   *
* ******************************************************************* *
         SPACE
SENDMSG  EQU   *
         MVC   LENGTH,=H'7Ø'
         MVC   OUTAREA+52(L'USRID),USRID
         MVC   OUTAREA+61(L'CWAPPLID),CWAPPLID
         EXEC  CICS WRITE OPERATOR TEXT(OUTAREA)                       *
               RESP(RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BNE   ERRORWA
         BR    R6
         SPACE
* ******************************************************************** *
*        E r r o r  w i t h  a b e n d  ØC1 (operation exception)     *
* ******************************************************************** *
         SPACE
ERRORWA  EQU   *                     Error with abend ØC1
         DC    D'Ø'                  Never come back statement
         BR    R6
         SPACE
* ******************************************************************** *
*        A b e n d  w i t h  a b e n d c o d e  "ABCODE"               *
* ******************************************************************** *
         SPACE
ABEND    EQU   *
         EXEC  CICS ABEND ABCODE(ABCODE)
         BR    R6
         SPACE
* ******************************************************************** *
*        C o n s t a n t s                                             *
*        F u n c t i o n  k e y s                                      *
*        R e g i s t e r  e q u a t e s                                *
* ******************************************************************** *
         SPACE
BLANKS   DC    CL8Ø' '
CWAPTR   EQU   R8
DSECTREG EQU   R1Ø
LINEREG  EQU   R7
DLENGTH  DC    Y(NFUNC3ØL-NEXØØL)
MSGADDEX DC    CL5Ø'Additional Exits available!'
MSGINVKE DC    CL5Ø'Function key not supported!'
TRTB     DC    C'Ø123456789ABCDEF'
WORKREG  EQU   R9
         SPACE
         COPY  DFHAID
         SPACE
         EQUREG
         EJECT
* ******************************************************************** *
*        L i t e r a l s                                               *
* ******************************************************************** *
         SPACE
         LTORG
         SPACE
         DC     C' '
         END
```

CMEXIT

```
         SPACE 3
         PRINT NOGEN
         SPACE 3
***********************************************************************
*              MAP FOR THE CICS USER EXIT LIST                        *
***********************************************************************
         SPACE 3
CMEXIT   DFHMSD TYPE=MAP,TERM=327Ø,MODE=INOUT,LANG=ASM,                *
               CTRL=(FREEKB,FRSET),DATA=FIELD,STORAGE=AUTO
         SPACE 3
MEXIT    DFHMDI LINE=1,COLUMN=1,SIZE=(32,8Ø),TIOAPFX=YES
NTACODE  DFHMDF POS=(Ø1,Ø1),ATTRB=(ASKIP,FSET),LENGTH=4
NUSRID   DFHMDF POS=(Ø1,1Ø),ATTRB=(ASKIP,FSET),LENGTH=8
NTITEL1  DFHMDF POS=(Ø1,32),ATTRB=(ASKIP,BRT),LENGTH=26,               *
               INITIAL='CICS USER EXIT LIST'
         SPACE 3
NRZDAT   DFHMDF POS=(Ø1,61),ATTRB=(ASKIP),LENGTH=14
NTERMNR  DFHMDF POS=(Ø1,76),ATTRB=(ASKIP),LENGTH=4
         SPACE 3
NTITEL2  DFHMDF POS=(Ø3,14),ATTRB=(ASKIP,BRT),LENGTH=54,               *
               INITIAL='EXIT TYPE   PROGRAM     ENTRY NAME  ADDRESS    *
               STATUS'
         SPACE 3
NEXØØ    DFHMDF POS=(Ø5,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ1    DFHMDF POS=(Ø6,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ2    DFHMDF POS=(Ø7,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ3    DFHMDF POS=(Ø8,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ4    DFHMDF POS=(Ø9,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ5    DFHMDF POS=(1Ø,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ6    DFHMDF POS=(11,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ7    DFHMDF POS=(12,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ8    DFHMDF POS=(13,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEXØ9    DFHMDF POS=(14,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX1Ø    DFHMDF POS=(15,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX11    DFHMDF POS=(16,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX12    DFHMDF POS=(17,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX13    DFHMDF POS=(18,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX14    DFHMDF POS=(19,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX15    DFHMDF POS=(2Ø,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX16    DFHMDF POS=(21,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX17    DFHMDF POS=(22,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX18    DFHMDF POS=(23,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX19    DFHMDF POS=(24,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX2Ø    DFHMDF POS=(25,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX21    DFHMDF POS=(26,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX22    DFHMDF POS=(27,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
NEX23    DFHMDF POS=(28,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
         SPACE 3
```

```
NFUNC3Ø  DFHMDF POS=(3Ø,Ø1),ATTRB=(ASKIP),LENGTH=45,                   *
               INITIAL='END = CLEAR; PF7 = BACKWARD; PF8 = FORWARD'
NADVI31  DFHMDF POS=(31,Ø1),ATTRB=(ASKIP,FSET),LENGTH=79
         SPACE 3
NERRO32  DFHMDF POS=(32,Ø1),ATTRB=(ASKIP),LENGTH=79
         SPACE 3
         DFHMSD TYPE=FINAL
         END
```

Claus Reis
CICS Systems Programmer
Nuernberger Lebensversicherung AG (Germany)

Automatic management of CICSplex offsite recovery program for VSAM record-level sharing

You can use RLS access mode to share VSAM datasets between CICS application-owning regions throughout a sysplex.

Record-level sharing is an access mode for VSAM datasets that enables VSAM data to be shared, with full update capability, between many applications running in many CICS regions.

With RLS, CICS regions that share VSAM datasets can reside in one or more MVS images within an MVS parallel sysplex.

When you use the RLS support, multiple CICS regions can open the same dataset concurrently.

In order to use RLS we must consider that:

- You need a level of DFSMS (1.3 and later releases) that supports RLS, and RLS=YES specified as a CICS system initialization parameter.
- The CICS regions must all run in the same parallel sysplex.
- There must be one SMSVSAM server started in each MVS image.
- You must specify RLSACCESS(YES) in the CICS file resource definition to provide full update capability for datasets accessed by multiple CICS regions.

It is possible to specify RLS access for all files supported by CICS file control, except for the following:

- Key range datasets are not supported.
- VSAM clusters defined with the IMBED attribute are not supported. (IMBED is a performance option that is generally unnecessary with modern cacheing disk controllers; therefore, you can remove the IMBED attribute from the cluster definition without loss of function.)

- Opening individual components of a VSAM cluster (which is not supported by CICS for any mode of access).
- Temporary datasets are not supported.
- Key-sequence datasets (KSDS) in Relative Byte Address (RBA) mode are not supported. Application programs that specify the RBA keyword on file control API commands for a KSDS opened RLS mode receive an INVREQ with RESP2 51 exception condition.
- Direct opening of alternative index (AIX) data is not supported in RLS access mode, but path access to data is supported.
- VSAM catalogs and VVDS datasets are not supported.

If the RLS support is not used (RLS=NO, system initialization parameter), it is not possible for multiple CICS regions to open the same VSAM dataset concurrently using a non-RLS mode (such as LSR or NSR). These access modes mean that to share VSAM data between CICS regions you must either use shared data tables or allocate the VSAM datasets to one CICS region (a File-Owning Region – FOR), and function-ship file requests from the applications to the CICS FOR using either MRO or APPC connections.

Generally, you choose which datasets need to be shared and updated in RLS mode by multiple CICS regions. When you have made this choice, you are recommended always to update these datasets in RLS mode.

When it is decided to use RLS support, in addition to specifying the parameter RLS=YES in the CICS initialization parameter, it is possible to specify when CICS starts the OFFSITE parameter to indicate that RLS offsite recovery processing is to be carried out during this CICS run.

Therefore, specifying OFFSITE=YES means CICS will perform an offsite restart at a remote site following a disaster at the primary site.

CICS performs this special processing for an offsite restart because some information (for example, a VSAM lock structure) is not available at the remote site.

CICS performs an emergency restart, even if the global catalog indicates that it should do a warm start.

The OFFSITE=YES parameter is valid with START=AUTO only, and CICS initialization is terminated if you specify START=COLD or INITIAL.

The recovery condition is managed from the CICS subsystem through two messages:

- DFHFC0574 applid – RLS offsite recovery will be performed. Normal RLS access is not allowed.
- DFHFC0575D applid – reply 'GO' only after all CICS regions have completed offsite recovery and issued this message.

These messages are issued when a CICS system is participating in the offsite recovery of a CICSplex.

The DFHFC0574 message is issued during file control initialization to indicate that RLS offsite recovery processing is to be carried out during this CICS run. In this condition, RLS access is not allowed until after this CICS has performed its RLS recovery work. Only tasks performing the recovery work are allowed RLS access.

The end of recovery process is indicated by a DFHFC0575D message. If this does not happen shortly after CICS restart has completed, there are probably some backout failed or indoubt failed units of work that had updated RLS datasets, and which are now delaying the completion of RLS recovery.

The WTOR type DFHFC0575D message is issued when all RLS recovery has been completed by CICS. CICS has backed out or committed all units of work that had made updates to datasets open in RLS mode, and that were either inflight or shunted at the time of the disaster at the primary site (or, more exactly, that were in that state at the common point in time to which the CICS system logs have been pruned).

RLS access for normal work (user applications) is not allowed until CICS has issued message DFHFC0575D and received the reply GO.

Since we are in a CICSplex environment, we must watch for the completion of the recovery of the other CICS systems.

In fact, this message indicates that RLS recovery work has been completed by a particular CICS region only.

It is important to specify that we must not reply until all CICS regions in the CICSplex have issued the DFHFC0575D message.

Replying to it indicates that all RLS recovery for the offsite CICSplex has been completed.

All the offsite recovery process should be automated in order to simplify its management and to prevent any operational errors.

TOOL DESCRIPTION

This tool, CICS Offsite Recovery Utility, automatically manages the record level sharing recovery process in a CICSplex environment.

This tool is an aid only for the users who manage the CICS environment and for application customers.

The essential scope of the tool is to simplify the RLS recovery process.

The CORUTOOL program is a NetView EXEC, written in REXX, that assists in the disaster recovery of a CICSplex when datasets have been used in RLS mode and OFFSITE=YES has been specified as a system initialization parameter.

Each participating MVS image in the sysplex should have NetView configured so that, when any CICS region issues messages DFHFC0574 or DFHFC0575D, CORUTOOL is called.

It is a mechanism that protects the RLS datasets from being accessed by new work until all the recovery work in the CICSplex has been completed.

As well as managing the recovery process, the CORUTOOL program automatically executes a notification function for CICS administrators. Such notification is sent at the end of the recovery process.

The notification of the event is carried out using the SMTP protocol to send mail.

The functions of this tool are:

- Survey the conditions at the beginning and end of the offsite recovery process.
- Execute the CORUTOOL program.
- Reply to DFHFC0575D message for all CICS systems that participate in the offsite recovery.
- Notify the completion of the recovery process.

CORUTOOL extracts the relevant input parameters from the message; these parameters are the message ID, CICS ID (APPLID), and the message reply number (only for DFHFC0575D).

It records the state of every CICS system using a control file.

CORUTOOL's control file should contain a record for each participating CICS region and should have dataset organization PS, record format of FB, and record length 80.

In the initialization phase of this tool, the control file records will look like:

```
Name_cics1
Name_cics2
    …
    …
Name_cicsn
```

Particularly with the DFHFC0574 message, the CORUTOOL program executes automatically.

It updates its control file with the CICS state, which should be accessible from any participating MVS image within the sysplex.

All entries that are not in ‘message issued’ state are updated to ‘message waiting’. Otherwise the existing state is preserved.

In this phase, therefore, the control file records will look like:

```
Name_cics1 MSGWAITING
Name_cics2 MSGWAITING
        …
        …
Name_cicsn MSGWAITING
```

If the message ID is DFHFC0575D, this program updates the record for the CICS entry, denoted by the input CICS ID, to ‘message issued’.

In this phase, therefore, the control file records will look like:

```
Name_cics1 MSGISSUED 312 (312 represents the message reply number)
Name_cics2 MSGISSUED 591 (591 represents the message reply number)
        …
        …
Name_cicsn MSGISSUED 878 (878 represents the message reply number)
```

If this CICS is not in the control file, it is ignored. All other entries that are not in a ‘message issued’ state are set to ‘message waiting’. Otherwise the existing state is preserved.

When all entries in the control file are in the ‘message issued’ state, this program generates an automatic reply to each DFHFC0575D message issued and then it prepares and sends a notification of recovery completed to CICS system administrators.

After all CICS regions have issued the message DFHFC0575D the control file records will look like:

```
Name_cics1 MSGREPLIED
Name_cics2 MSGREPLIED
        …
        …
Name_cicsn MSGREPLIED
```

This utility was tested and used in OS/390 2.10, Tivoli NetView for OS/390 1.4.0, and CICS Transaction Server 1.3.0.

NETVIEW CONFIGURATION

Add the following entry to DSIPARM, the NetView message table:

```
      IF MSGID = 'DFHFCØ57'. & TEXT = MSG
           THEN EXEC(CMD('CORUTOOL' MSG) ROUTE(ONE *))
               DISPLAY(Y) NETLOG(Y);
```

This causes NetView to invoke the CORUTOOL program whenever a message is issued that is prefixed with DFHFC057, passing the message text as input parameters.

To copy the CORUTOOL program to a NetView library use the DSICLD DD statement.

CORUTOOL EXEC

```
/* Corutool */
/*
   This program is called by DFHFCØ574 and DFHFCØ575D messages.
   It carries out, for every CICS present in its control file, the
   update of the state of the offsite recovery process.
   When all the recovery is ended, it answers to message DFHFCØ575D
   for every CICS.
   The answer to message DFHRFØ575D allow CICS to resume normal
   activity.
                                                                   */
Trace ?o
CORUTOOL:
Arg var1 var2 var3 varx
cntrf  = 'CICSTS.CORUTOOL.CNTRFILE'
messw = 'MSGWAITING'
messi = 'MSGISSUED'
messr = 'MSGREPLIED'
Call CoruTool_Var_Message
Call CoruTool_Alloc_CntrFile(cntrf)
Select
    When cxmess = 'DFHFCØ574' then
      Call CoruTool_Update_CntrF_Wait(messw messi messr)
    When cxmess = 'DFHFCØ575' then
      Call CoruTool_Update_CntrF(cxappl mesrplno messw messi messr)
    Otherwise Nop
End
Call CoruTool_Dealloc_CntrF
Exit
/********************************************************************/
/*                                                                  */
/* Select variables from CICS message and control message type.    */
/*                                                                  */
/********************************************************************/
CoruTool_Var_Message:
If datatype(var1) = 'NUM' Then Do
                                cxmess = substr(var2,1,9)
                                cxappl = var3
                                mesrplno = var1
                         say time() 'CoruTool: DFHFCØ575 condition ...'
                               End
                          Else Do
                                cxmess = substr(var1,1,9)
                                cxappl = ''
                                mesrplno = ''
                         say time() 'CoruTool: DFHFCØ574 condition ...'
                               End
Return
/********************************************************************/
/*                                                                  */
/* Allocation CICS Offsite Recovery control file.                   */
/*                                                                  */
/********************************************************************/
CoruTool_Alloc_CntrFile: procedure expose corutool.
Arg cntrf
nvolte = Ø
say time() 'CoruTool: Allocation control file ...'
Do until rc = Ø
  'ALLOCATE DSN('''cntrf''') DDNAME('corutool') OLD KEEP'
  retc = rc
  if retc = 4Ø then do
         say time() '************************************************'
         say time() '***                                          ***'
         say time() '*** CoruTool : Allocation Control File error ***'
         say time() '***            Dataset not found             ***'
         say time() '***            Notify CICS Admin group       ***'
         say time() '***                                          ***'
         say time() '************************************************'
                   exit
                    end
  If retc ¬= Ø & retc ¬= 4Ø Then Do
                   nvolte = nvolte + 1
                   if nvolte < 1ØØ then do
                              say time() 'CoruTool: Control file ' ,
                                         'in use. Retrying ...'
                                         'WAIT 2'
                                         rc = 99
                                        end
             Else Do
        say time() '************************************************'
        say time() '***                                          ***'
        say time() '*** CoruTool : Allocation Control File error ***'
        say time() '***            Notify CICS Admin group       ***'
        say time() '***                                          ***'
        say time() '************************************************'
                   exit
                  End
  End
End
Return
/********************************************************************/
/*                                                                  */
/* Dealloc CICS Offsite Recovery control file.                      */
/*                                                                  */
/********************************************************************/
CoruTool_Dealloc_CntrF: procedure expose corutool.
'FREE DDNAME('corutool')'
Return
/********************************************************************/
/*                                                                  */
/* Update CICS status in Offsite Recovery control file.             */
/*                                                                  */
/********************************************************************/
CoruTool_Update_CntrF: procedure expose corutool rec.
Arg cxappl mesrplno messw messi messr
yes = 'Y'
no = 'N'
Call CoruTool_Read_CntrF
reply = yes
Do i = 1 to rec.Ø
  parse upper var rec.i  cxname messtat mesrplnu varx.
  If cxname = cxappl Then Do
                           rec.i = cxname||' '||messi||' '||mesrplno
                          End
                     Else Do
                           if messtat ¬= messi then do
                                          rec.i = cxname||' '||messw
                                          if reply = yes then reply = no
                                                    end
                          End
End
Call CoruTool_Write_CntrF
If reply = yes then Call CoruTool_Rpl_MessAll(messw messi messr)
Return
/********************************************************************/
/*                                                                  */
/* Recovery Offsite process ended.                                  */
/* Reply to each CICS region and update CICS status in control file.*/
/*                                                                  */
/********************************************************************/
CoruTool_Rpl_MessAll: procedure expose corutool
```

```
Arg messw messi messr
Call CoruTool_Read_CntrF
Do i = 1 to rec.Ø
  Parse Upper Var rec.i cxname messtat mesrplno varx.
  Call CoruTool_Rpl_Mess(mesrplno)
  rec.i = cxname||' '||messr
End
Call CoruTool_Write_CntrF
/*******************************************************************/
/*                                                                 */
/* Offsite Recovery process ended.                                 */
/* Send e-mail to CICS Administrators.                             */
/*                                                                 */
/*******************************************************************/
Call CoruTool_Send_Notify_Recovery_End
Return
/*******************************************************************/
/*                                                                 */
/* Reply to CICS message.                                          */
/*                                                                 */
/*******************************************************************/
CoruTool_Rpl_Mess: procedure
Arg mesrplnu
'MVS R '||mesrplnu||',GO'
If rc ¬= Ø then Do
        say time() '**********************************************'
        say time() '***                                        ***'
        say time() '*** CoruTool : Reply Recovery error        ***'
        say time() '***            Notify CICS Admin group     ***'
        say time() '***                                        ***'
        say time() '**********************************************'
                 exit
                End
Return
/*******************************************************************/
/*                                                                 */
/* Write CICS status in control file.                              */
/*                                                                 */
/*******************************************************************/
CoruTool_Update_CntrF_Wait: procedure expose corutool
Arg messw messi messr
Call CoruTool_Read_CntrF
Do i = 1 to rec.Ø
  Parse Upper Var rec.i cxname messtat mesrplno varx.
  If messtat ¬= messi Then Do
                           rec.i = cxname||' '||messw
                          End
End
Call CoruTool_Write_CntrF
Return
```

```
/********************************************************************/
/*                                                                  */
/* Read CICS status from control file.                              */
/*                                                                  */
/********************************************************************/
CoruTool_Read_CntrF: procedure expose corutool rec.
Address MVS 'EXECIO * DISKR  corutool  (FINIS STEM REC.'
If rc ¬= Ø Then Do
        say time() '************************************************'
        say time() '***                                          ***'
        say time() '*** CoruTool : Read Control File error       ***'
        say time() '***            Notify CICS Admin group       ***'
        say time() '***                                          ***'
        say time() '************************************************'
                 exit
                End
Return
/********************************************************************/
/*                                                                  */
/* Write CICS status in control file.                               */
/*                                                                  */
/********************************************************************/
CoruTool_Write_CntrF: procedure expose corutool rec.
Address MVS 'EXECIO ' rec.Ø ' DISKW corutool (FINIS STEM REC.'
If rc ¬= Ø Then Do
        say time() '************************************************'
        say time() '***                                          ***'
        say time() '*** CoruTool : Write Control File error      ***'
        say time() '***            Notify CICS Admin group       ***'
        say time() '***                                          ***'
        say time() '************************************************'
                 exit
                End
Return
CoruTool_Send_Notify_Recovery_End:
/*
 **********************************************************************
 *
 * When recovery process is ended, this procedure executes:
 *
 *     - Read Mail-list.
 *     - Write and send e-mail to CICS Administrators users.
 *
 **********************************************************************
                                                                      */
emaill = 'TCPSMPT.SA53.MAILLIST(CICSADMØ)' /* mailing list */
utentea = Ø
gruppo = substr(emaill,23,8)
tempo = date(e)
gg = substr(tempo,1,2)
```

```
aa = substr(tempo,7,2)
daw = date(w)
dam = date(m)
/********************************************************************/
/*                                                                  */
/*                                                                  */
/* Alloc/Read Mail-list file.                                       */
/*                                                                  */
/********************************************************************/
ADDRESS NETVIEW
"ALLOC DATASET('"emaill"') F(TCPML) SHR FREE"
say time() ' Read mail-list... 'emaill
ADDRESS MVS
  "NEWSTACK"
  "EXECIO * DISKR TCPML (STEM RECML. FINIS"
  "DELSTACK"
/********************************************************************/
/*                                                                  */
/* Prepare e-mail to CICS administrators.                           */
/*                                                                  */
/********************************************************************/
DO I = 1 TO RECML.Ø
  recno = substr(recml.i,1,1)
  if recno = '*' | recno = ' ' then ITERATE
  aemail = substr(recml.i,2,5Ø)
  utentea = utentea + 1
  tt = time()
  tt = translate(tt,'',':')
  tt = space(tt,Ø)
  mm = substr(time(l),1Ø,6)
  ADDRESS NETVIEW
  "ALLOC DATASET('MAILS.SVIL.D"||DATE(J)||".T"||TT||".M"||mm||"')",
  "FILE(FMAIL"||utentea||") UNIT(339Ø) VOLUME(DSKHØØ) SPACE(1 1)",
  "DSORG(PS) LRECL(132) BLKSIZE(136) RECFM(VBA) NEW CATALOG"
  ADDRESS MVS
  "NEWSTACK"
  QUEUE 'HELO JESB'
  QUEUE 'MAIL FROM:<NETOPER@NZ5326.Legacy.Svil.it>'
  QUEUE 'RCPT TO:<'aemail'>'
  QUEUE 'DATA'
  QUEUE 'TO:<'aemail'>'
  QUEUE 'DATE: 'daw' , 'gg' 'dam' 'aa' 'time()
  QUEUE 'Subject: CICS Offsite Recovery'
  QUEUE ' '
  QUEUE ' --------------------------------------------'
  QUEUE ' E-mail: NETOPER@NZ5326.LEGACY.SVIL.IT'
  QUEUE ' Data: 'tempo
  QUEUE ' Ora:  'time()
  QUEUE ' --------------------------------------------'
  QUEUE ' '
  QUEUE '            CICS OFFSITE RECOVERY UTILITY '
  QUEUE ' '
  QUEUE ' Automatic management of CICSPLEX OFFSITE RECOVERY program'
  QUEUE ' for Vsam Record-Level Sharing (RLS): '
  QUEUE ' '
  QUEUE 'RECOVERY PROCESS ENDED, TO VERIFY.'
  QUEUE ' '
  QUEUE
  "EXECIO * DISKW FMAIL"||utentea||"(FINIS)"
  "DELSTACK"
  ADDRESS NETVIEW
  "FREE FILE(FMAIL"||utentea||")"
End
/********************************************************************/
/*                                                                  */
/* Prepare and submit job to send e-mail.                           */
/*                                                                  */
/********************************************************************/
filesend = 'TEMP.MAILS.D'DATE(J)'.T'TT'.M'mm
  ADDRESS NETVIEW
  "ALLOC DATASET('"filesend"')",
  "FILE(FSEND) UNIT(SYSDA) SPACE(1 1)",
  "DSORG(PS) LRECL(8Ø) BLKSIZE(8Ø) RECFM(FB) NEW CATALOG"
filemail = 'MAILS.SVIL.D'DATE(J)'.T'TT'.M'mm
  ADDRESS MVS
  "NEWSTACK"
  QUEUE '//NETOPER3 JOB (AMKJØ21Ø),'
  QUEUE '//         CLASS=S,REGION=4M,'
  QUEUE '//         MSGCLASS=X,'
  QUEUE '//         MSGLEVEL=(1,1)'
  QUEUE '//STEP1    EXEC PGM=IKJEFTØ1'
  QUEUE '//SYSTSPRT     DD SYSOUT=*'
  QUEUE '//SYSTSIN      DD DDNAME=SYSIN'
  QUEUE '//SYSIN        DD *'
  QUEUE "  XMIT JESB.SMTP53 DA('"filemail"')"
  QUEUE '/*'
  QUEUE '/*'
  QUEUE
  "EXECIO * DISKW FSEND (FINIS)"
  ADDRESS NETVIEW
  "SUBMIT '"filesend"'"
  ADDRESS MVS
  "DELSTACK"
ADDRESS NETVIEW
"FREE FILE(TCPML)"
"FREE FILE(FSEND) DATASET('"filesend"') DELETE"
say time() ' E-Mail creation for group 'gruppo' number users...'utentea
Return
```

SAMPLE CORUTOOL CONTROL FILE

```
EDIT       CICSTS.CORUTOOL.CNTRFILE                  Columns ØØØØ1 ØØØ72
 Command ===>                                            Scroll ===> CSR
 ****** ************************* Top of Data **************************
 ØØØ4ØØ CICSSVI1 MSGISSUED 798
 ØØØ5ØØ CICSSVI8 MSGISSUED 812
 ØØØ51Ø CICSTSIA MSGISSUED 815
 ØØØ6ØØ CICSINT3 MSGWAITING
 ØØØ61Ø CICSTSIB MSGISSUED 818
 ØØØ7ØØ CICSCOLD MSGWAITING
 ØØØ8ØØ CICSMNT5 MSGISSUED 793
 ****** ************************ Bottom of Data ************************
```

SAMPLE E-MAIL

```
  --------------------------------------------
  E-MAIL: NETOPER@NZ5326.LEGACY.SVIL.IT
  DATA: Ø1/Ø9/Ø3
  ORA:  11:27:34
  --------------------------------------------

             CICS OFFSITE RECOVERY UTILITY

  Automatic management of CICSPLEX OFFSITE RECOVERY program
  for Vsam Record-Level Sharing (RLS):

  RECOVERY PROCESS ENDED, TO VERIFY.
```

SAMPLE MAILING LIST

```
BROWSE    TCPICS.SA53.MAILLIST(CICSADMØ) - Ø1.ØØ   Line ØØØØØØØØ Col ØØ1
 Command ===>                                                Scroll ===>
****************************** Top of Data *****************************
                Mailing List CICS Offsite Recovery Utility

*----------------------------------------------------------------------*
* E-Mail Address                            | Notes                    *
*----------------------------------------------------------------------*

 The records with the character "*" o " " to column 1 are considers
comments

_cicsadmØ@nn3273.napoli.legacy.svil.it
_cicsadm7@nn3273.napoli.legacy.svil.it
_cicsadmC@rx5ØØ2.roma.legacy.svil.it
 cicsadmP@rx5ØØ2.roma.legacy.svil.it
```

```
_cicsadm3@fh27Ø2.firenze.legacy.svil.it
_cicsadm5@fh27Ø2.firenze.legacy.svil.it
```

Espedito Morvillo
Systems Programmer (Italy)

CICS news

Cyanea Systems has announced Cyanea/One for CICS, a new release designed to provide application management support for CICS platform.

Cyanea/One for CICS helps customers gain visibility into transactions that span across J2EE and CICS platforms, and enables composite transactions between J2EE application servers and CICS to be traced and correlated. Users can pinpoint and resolve complex application performance problems, minimizing downtime, enhancing availability, and improving capacity planning.

For further information contact:
Cyanea Systems, 2001 Broadway, Third Floor, Oakland, CA 94612, USA.
Tel: (510) 587 7000.
URL: http://www.cyanea.com/solution_subnav_01_how_cyanea_works.html.

* * *

Wily Technology's Introscope Version 4.2 can now monitor the health and availability of Java applications running on CICS.

Introscope helps identify performance problems whether inside the application itself, the application server, or in back-end transactional and supporting systems. Introscope's Blame technology isolates bottlenecks all the way down to individual servlets, JSPs, EJBs, Classes, Methods, and more. Introscope's ability to monitor Java applications on CICS TS makes it easier to eliminate performance issues before they affect the end-user.

For further information contact:
Wily Technology, 8000 Marina Boulevard, Suite 700, Brisbane, California 94005, USA.
Tel: (415) 562 2000.
URL: http://www.wilytech.com/solutions/products/PP_IBM_CICS.html.

Cue-Metamon has announced a new release of Metamon/CICS, its software trace and analysis tool.

The product automates the capture of CICS calls, eliminating the need to manually step through code. The recorder can be stopped, started, and paused by the programmer, while CICS calls can be selectively captured based on parameters that are defined by the user. A full event log of all calls is captured and can be viewed on-line, or printed in expanded and compressed formats. Summaries for programs executed, files accessed, and calls made are readily accessible.

For further information contact:
Cue-Metamon, 45 Accord Park Drive, Norwell, MA 02061, USA.
Tel: (781) 740 8866.
URL: http://www.cue-metamon.com/solution5.htm.

* * *

Data 21 has announced Version 3.0 of ZIP/CICS, its cross-platform compatible compression utility. The product combines compression and encryption technologies with built-in TCP/IP communications (ie e-mail and FTP).

Version 3.0 offers improved performance and ease of use. A refinement in the products UNZIP algorithm reduces UNZIP time on the mainframe by an average of 30%. New features such as dynamic file allocation and generic file name support greatly simplify the JCL.

For further information contact:
Data 21, 3510 Torrance Blvd, Suite 300, Torrance, CA 90503, USA.
Tel: (310) 792 1771.
URL: http://www.data21.com/products/zip/default.asp.