233

CICS update

April 2005

In this issue

A tc PUBLICATION

CICS Update

Published by

Xephon Inc
PO Box 550547
Dallas, Texas 75355
USA

Phone: 214-340-5690
Fax: 214-341-7081

Editor

Trevor Eddolls
E-mail: trevore@xephon.com

Publisher

Colin Smith
E-mail: info@xephon.com

Subscriptions and back-issues

A year's subscription to *CICS Update*, comprising twelve monthly issues, costs $270.00 in the USA and Canada; £175.00 in the UK; £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 2000 issue, are available separately to subscribers for $24.00 (£16.00) 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.

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 $160 (£100 outside North America) per 1000 words and $80 (£50) per 100 lines of code for the first 200 lines of original material. The remaining code is paid for at the rate of $32 (£20) 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.

CICS TS 3.1: more new enhancements

This is a follow-up article to ‘CICS TS 3.1 highlights and an overview of new enhancements’, published in *CICS Update*, March 2005, issue 232, which had an overview of Web services and Service-Oriented Architecture (SOA) exploitation by CICS, and discussed a new mechanism that is provided for inter-program data transfer without CICS size limitation. This current article will give you an overview of OTE exploitation enhancements, HTTP enhancements, C/C++ support, CICSPlex SM Web User Interface enhancements, security enhancements, and withdrawn functions.

OPEN TRANSACTION ENVIRONMENT EXPANDED

Open Transaction Environment (OTE) services are not new to CICS TS. OTE introduced a new class of TCB, which can be used by threadsafe applications. This type of TCB is characterized by the fact that it is assigned to a CICS task for the life of the CICS task and multiple OTE TCBs may run concurrently in CICS.

OTE services were initially used to enable Java programs to be executed within a CICS system. In CICS TS V2.2, support was added that enabled CICS/DB2 applications to run in an OTE, which provided the benefit of reduced TCB switching, improving the applications’ performance. CICS TS V3.1 further improved OTE, enabling more CICS commands to function in threadsafe re-entrant environments and to extend multiprocessor usage to non-CICS/DB2 applications. This extends the use of OTE by providing support for PL/I, COBOL, Assembler, and non-XPLink C/C++ OPENAPI application programs. Now your program can run in its own TCB from the start and does not have to go through TCB switching, which engenders additional overhead.

New enhanced OTE can remove a bottleneck in the application throughput running under CICS, and applications can now run as different concurrent OS tasks, exploiting multiprocessors and improving throughput and response time. This enhancement can also greatly help with the integrity of the data by preventing CICS storage violations. CICS storage protection is extended to applications executing under different OS tasks.

In order to benefit from OTE services you need to ensure that their applications are threadsafe. As a result of this requirement, all the EXEC CICS Web API commands have been made threadsafe.

It should be noted that use of any non-threadsafe CICS commands will cause a switch to the QR TCB, then CICS will switch back to the OTE TCB before returning control to the program.

C/C++ SUPPORT ENHANCEMENTS

New support for C and C++ has been introduced in CICS TS 3.1, with the performance of applications written in these languages being comparable to COBOL, PL/I, or Assembler. C/C++ developers can now write CICS code, taking advantage of Language Environment support, with CICS storage management, and can run applications on dedicated TCBs, using OTE services.

Support for the Extra Performance Linkage (XPLINK) feature o f z/OS enables improved performance of applications written in C/C++. The use of XPLINK enables the latest compiler and optimization technologies included with C and C++ to be exploited. You can achieve greater reuse of C and C++ code by using Dynamic Load Libraries (DLLs) that were compiled with the XPLINK option in the CICS environment. With the enhancements delivered by XPLINK, IBM reports that early performance comparisons have indicated that a 40% increase in performance could be attained using C/C++ application.

HTTP ENHANCEMENTS

CICS has had HTTP inbound support since CICS TS V1.3 and now with V3.1 has expanded support to outbound HTTP. This means that CICS can act as both an HTTP server (utilizing CICS Web inbound support) or as an HTTP client (utilizing CICS Web outbound support). CNV= now can serve HTTP requests as an HTTP server, and outbound requests can be created using EXEC CICS commands for a CICS application as an HTTP client. This gives us the capability to code CICS application programs to use the common HTTP protocol for business-to-business communication or to access information in non-browser HTTP applications.

CICS TS 3.1 extended HTTP support to the V1.1 specification, including chunked transfer-coding, pipelining, and persistent connections. Please be aware that CICS Web support is conditionally compliant with the HTTP/1.1 specification, which means that CICS satjies all the '*must*'-level requirements, but not all the '*should*'-level requirements, that are detailed in HTTP/1.1. CICS developers should review the guidance RFC 2616, which is available to download from www.ietf.org/rfc/rfc2616.txt, to make sure that the application code they write is compliant.

The facility for CICS to act as an HTTP client has been fully integrated into CICS Web support. You can use EXEC CICS commands in CICS application programs to open an HTTP connection to a server, make requests, and receive responses for processing by the application program.

Management of the HTTP server facility has been greatly simplified with a new RDO resource, URIMAP, which enables CICS to match the URIs of requests from Web clients or a remote server, and provide information on how to process the requests.

For example, if you have a client invoking a Web service request to CICS, you would define a URIMAP resource that identifies the WEBSERVICE resource that applies to the URI in the inbound message. WEBSERVICE resources can then be installed dynamically in your running CICS system.

The processing for CICS as an HTTP client can be as follows:

1 Your application program does **WEB OPEN** to initiate a connection with the HTTP server through CICS.

2 CICS opens a TCP/IP connection on a socket to establish the connection with the server. When the TCP/IP connection is established, CICS returns a session token to the application program to uniquely identify the connection. This session token is used on all the remaining commands issued by the application program concerning that connection.

3 Your application program issues:

   - **WEB WRITE HTTPHEADER** to write HTTP headers for its request.
   - **WEB SEND** or **WEB CONVERSE** commands to specify components of the request, to produce an entity body for its request, and to initiate transmission of the request.

4 CICS generates some required HTTP headers and adds them to the request, then sends the request to the server.

5 The server receives and processes the request, and provides a response. CICS passes the response back to your program.

6 Your program can examine the response by browsing the HTTP header using the **WEB READ HTTPHEADER** command, and can receive the body of the response by using the **WEB RECEIVE** or **WEB CONVERSE** command.

   At this point you can initiate further requests, if needed. If the server supports persistent connections, the connection identified by the session token remains open for further requests.

7 Your program closes the connection to the server by issuing a **WEB CLOSE** command.

Please note that there is some changed behaviour for HTTP processing, compared with CICS TS V2. Under CICS TS 3.1, connections are now persistent by default and CICS handles a wider range of error situations and unsupported messages.

It also should be noted that some HTTP functions are not supported by CICS Web support:

- CICS does not act as a proxy.
- CICS does not act as a gateway.
- CICS does not provide cacheing facilities, or provide support for user-written cacheing facilities.
- CICS is not designed for use as a Web browser.

CICSPLEX SM WEB USER INTERFACE ENHANCEMENTS

CICS TS 3.1 delivers a totally-enhanced, modern, and more intuitive CICSPlex SM Web user interface. If you have not been using CICSPlex SM in the past you may want to take another look at it. It is now possible to completely configure CICSPlex SM using this interface. The Web user interface has already been providing functions that are not available with the old TSO end-user interface and now has been greatly enhanced to improve usability and reduce the complexity of migration. CICSPlex SM can be configured, set up, and run without involving the TSO or CAS components, saving time and effort for both existing and new users of CPSM.

Some of the new functions added are:

- *Improvements to screen design*. These enhancements maximize the use of screen space in views and menus. The view editor now allows the creation and display of detailed views in two-column format. There is also now the ability to add a view to the Favorites.

- *A single point of control and definition and dramatically reduced time for system configuration* simplify the management of CICS systems, and fewer staff are required to manage a CICS environment.

- *Usability has been improved with the inclusion of filter confirmation*, which allows you to specify a filter before retrieving the data for a particular view.

- *More flexibility in maintaining user profiles*. You can group users, eg operators, system programmers, or application developers, and maintain a common profile for these groups.

- *A new function to submit a Batchrep job using the CPSM WUI*. A Batchrep job can be submitted using the CPSM API PERFORM OBJECT command.

SECURITY ENHANCEMENTS

CICS TS V3.1 introduces a number of improvements to security.

- IBM has been improving CICS security in a 'Web world' and added support for Secure Sockets Layer (SSL) 3.0 in a previous release. It has now added support for the Transport Layer Security (TLS) 1.0 protocol. This includes support for the Advanced Encryption Standard (AES) cipher suites, which offer 128-bit and 256-bit encryption.

- A new transaction, CCRL, is provided for updating the Certificate Revocation Lists (CRL) in an LDAP server. CRLs are now supported when negotiating with clients, allowing any connections using revoked certificates to be closed immediately.

- An enhanced cacheing mechanism enables an SSL handshake to be optimized based on a previous negotiation, thereby improving the performance of connection set-up.

- Additional flexibility is offered in negotiations for a minimum as well as a maximum encryption level. It is now possible to specify whether session IDs are shared across a Sysplex, improving the current use of the cache at the address space level.
- Improved handling of security checks on a START command. The revoked status of a userid or group connection is now tested for by the EXEC CICS START USERID() command when it is issued, in order that the issuer can be notified by the USERIDERR condition if applicable.

INFORMATION CENTER ON ECLIPSE PLATFORMS

CICS Information Center has been a great source for information and for a quick look-up on ‘how to’ questions. It has been enhanced for easier customization and better access to informational resources by using award-winning Eclipse technology, which is being employed by many other IBM products. Eclipse is written in Java and includes extensive plug-in construction toolkits and examples, and can be extended and deployed on a range of desktop operating systems.

The Information Center is provided as a plug-in to the Eclipse platform. This infrastructure allows users to customize their own Information Centers using plug-ins from multiple products, or to write their own plug-ins. The new Information Center enables direct links (eSupport) from CICS TS information to support information. Also, support has been extended beyond z/OS platforms.

Here are highlights of the new functions:

- *A ‘What’s New’ section organized by major functional areas, available through the navigation and welcome pages*. This is similar to the Release Guide but is not a separate document, having integrated links into the rest of the Information Center.

- *Learning Paths* – a sequence of topics that help a user learn about a new area of the product.
- *Information Roadmaps* – a topic that provides a set of comprehensive links, role or function based, to information from a variety of sources for Web services, Java in CICS, and CICSPlex SM.
- *A troubleshooting and support section* – a self-help resource that consists of components for searching external support sites, getting fixes, and contacting IBM support. It will also contain a selection of Technotes.

WITHDRAWN FUNCTIONS

OS/VS COBOL programs

For some time IBM has been making statements of direction to withdraw run-time support for OS/VS COBOL programs and it has now finally been withdrawn in CICS TS 3.1. OS/VS COBOL programs, which had run-time support in CICS Transaction Server for z/OS Version 2, cannot run under CICS TS V3.1. OS/VS COBOL programs must be upgraded to Language Environment conforming COBOL, and recompiled against a level of COBOL compiler supported by CICS.

If you are still using programs compiled under OS/VS COBOL, you will need to reference the *CICS Application Programming Guide for CICS TS V3.1*, which has information on supported compilers for COBOL and guidance on converting OS/VS COBOL programs to Language Environment-conforming COBOL. A program that is compiled under OS/VS COBOL will get a new abend code, ALIK, and CICS will disable the program.

Common Connector Framework (CCF)

The Common Connector Framework (CCF), which was the predecessor interface to the Common Client Interface (CCI), is not supported by CICS TS V3.1. The intention to remove this support was indicated in the announcement of CICS TS V2.3.

Hot-pooling

Run-time support for Java program objects and for hot-pooling (HPJ) has been withdrawn. CICS TS V2.3 still provides run-time support for existing hpj-compiled Java program objects, but no support was provided for developing new Java program objects nor for modifying existing Java program objects. In CICS V3.1, run-time support for hpj-compiled Java program objects has been withdrawn. Now you need to migrate any hpj-compiled Java program objects to run in a Java Virtual Machine. If you attempt to execute a Java program object in CICS TS for z/OS Version 3.1, an ALIG abend is issued. The open TCB mode H8, which was used for hot-pooling Java program objects, no longer exists.

DFHAPH8O and DFHJHPAT

The user-replaceable programs DFHAPH8O and DFHJHPAT have been removed. DFHAPH8O was provided to allow you to alter the default Language Environment run-time options for the Language Environment enclave where a Java program object was to be run.

BTAM and TCAM support

CICS support for the Basic Telecommunications Access Method (BTAM) has been discontinued. Support for the Telecommunications Access Method (TCAM) is limited to indirect support for the DCB interface.

It should be noted that CICS TS V3.1 does not support transaction routing or function shipping from BTAM terminals attached to a pre-CICS TS 3.1 terminal-owning region.

PLANNED CHANGES

If you are running Java Applications, you should run your

JVMs in continuous mode. Support for continuous mode was introduced in CICS TS V2.3 in order to bring CICS use of Java into line with standard practices and to improve performance. IBM intends to remove support for resettable mode in a future release of CICS TS.

SUMMARY

CICS has been around for 35 years, has an excellent reputation for reliability and performance, and has been continually enriched with new functionality; and there is no sign of it slowing down. IBM continues enhancing the product to support new technologies and to improve performance. CICS now provides extensive support for Internet-connected devices, support for Java and Enterprise Java Beans, XML, and Simple Object Access Protocol, and now can exploit Service-Oriented Architecture.

Dibbe Edwards, director of CICS Products at IBM, made the following statement about CICS TS 3.1: “The latest version of CICS Transaction server continues the path of IBM’s investment in the mainframe infrastructure and joins with the latest versions of WebSphere, DB2, and IMS, enabling companies to deploy Web Services and participate in SOA.”

CICS now offers you technology that your company can take advantage of to meet its business demands and stay competitive by capturing new opportunities derived from the latest technologies, while protecting investments in existing CICS applications.

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

DFHEISUP load module scanner

The utility DFHEISUP is provided by CICS to allow you to search load modules for specific CICS API and SPI commands. For example, if you know that a particular API or SPI command should be changed to take advantage of a new feature in CICS, you can use the load module scanner to identify all the load modules containing the command. The load module scanner can look for specific commands, options on commands, or combinations of options. It can also scan for commands where certain options are not specified.

You can create two types of report by running the program. The summary report will give you a list of load modules containing the commands specified by the filter you have set. It also shows how many of these commands exist in the load module. A detailed report will show the specific commands each module contains, and the offset of the command. The reports also inform you what the language type of the load module was when originally written.

Normally I would run the summary report and then use the information it provides to run detailed reports for the load modules containing the commands I am interested in.

The program is very useful for finding out what applications require changes to make them threadsafe.

IBM supplies a sample job that resides in the SDFHSAMP dataset named DFHEILMS. The example below contains a modified version of this JCL to execute DFHEISUP:

```
//JXB7884C   JOB    (7884),CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//STEP1      EXEC  PGM=DFHEISUP,
//                 PARM=('SUMMARY,DETAILMODS'),REGION=512M
//STEPLIB    DD     DSN=SCICS.SDFHLOAD,DISP=SHR
//SYSPRINT   DD     SYSOUT=*
//SYSERR     DD     SYSOUT=*
//DFHFLTR    DD     DSN=SCICS.FILTER,DISP=SHR
//DFHDTL     DD     DSN=SCICS.MODLIST,DISP=(NEW,CATLG,DELETE),
//               DCB=(RECFM=FB,LRECL=8Ø,BLKSIZE=8ØØØ),SPACE=(CYL,(1,1))
```

```
//DFHLIST     DD      DSN=SCICS.MODLIST2,DISP=SHR
//DFHIN       DD      DSN=SDEVCICS.SDFHLOAD,DISP=SHR
```

The load module scanner can scan all the load modules in a load library, individual load modules, or even load modules containing specific commands that you specify in filter lists.

You specify the commands to be reported on as a filter list input file. In these lists you can name the commands in which you are interested. You can also include or exclude commands that have certain parameters. The 'any' operator can be used to search for all parameters or for all commands. IBM supplies several sample command filter tables. Two such tables are DFHEIDTH and DFHEIDNT, again stored in SDFHSAMP. The first of these tables is shown below:

```
# The extract command gives addressibilty to a global work area of a
# GLUE or TRUE.
EXTRACT EXIT GASET *
#
# Getmain shared storage can be shared between CICS transactions.
GETMAIN SHARED *
#
# The CWA is shared between all CICS transactions.
ADDRESS CWA *
```

They can be used to search for programs that are not threadsafe compatible.

On the PARM statement, specify SUMMARY. If you want to produce a list of modules to be used when running a detailed scan on this load library, specify DETAILMODS as well.

SUMMARY specifies that a summary scan and report is required for the entire library, except for certain CICS modules (those beginning with XDFH and the DL/I modules ASMTDLI, CBLTDLI, and PLITDLI), CICS tables, and those modules that cannot be loaded because of an error.

DETAILMODS specifies the names of those modules that may contain at least one of the EXEC CICS commands identified in your filter and that are to be written to the sequential file defined by the DFHDTL DD statement.

The DFHIN DD statement will contain the name of the load library to be scanned. These must be PDS-type datasets. PDSE cannot be used.

SYSPRINT DD is where the summary report will be produced.

DFHFLTR DD is the input file for the filter, containing details of the commands in which you are interested.

DFHDTL DD is an optional statement and is used to specify the name of a sequential dataset where the list of modules with potential command matches is to be sent. To send the list of modules to the dataset, specify DETAILMODS on the PARM statement. You can edit the dataset to alter the list of modules to be scanned, but note that all the modules listed must be in the same load library.

The DFHEIDTH filter list contains a list of the three commands, which are threadsafe inhibitors. This term refers to commands that could make the program not threadsafe because they allow accessibility to shared storage.

The three commands that are listed in DFHEIDTH as being threadsafe inhibitors are:

- EXTRACT EXIT GASET
- GETMAIN SHARED
- ADDRESS CWA.

All these commands return addresses of data areas that can be shared across an application. If any of these commands are identified as being used in any application program, a more detailed analysis of the whole application should be undertaken. It is possible the address returned by one of these commands can be passed to another program that may not execute any of the above commands itself, but could still modify the data at the address specified.

DFHEIDNT contains a list of all those commands that cause a TCB switch back to the QR TCB. Note that this table will appear only if you have APAR PQ82603 applied to your system. Use of these commands will not prevent you from defining the program as threadsafe; however, it is possible that they might prevent your application from achieving the performance benefits of allowing programs to stay on the L8 TCB following a DB2 call.

It is possible that the scanner may return some false hits. This could happen because there is static data in the module. You may have to do some manual post processing to filter out the false hits. Also be aware that the scanner is designed to scan up to about 100 load modules on a single run. If you wish to scan more, you may need to increase the region size of the job, but IBM does warn against coding 0M as the region size. This could cause performance hits. If you have a large number of load modules in a load library that you want to scan it is better to copy the modules to separate libraries and run several scan jobs.

Full details on the filter lists and how to code them can be found in the *CICS Operations and Utilities Guide* manual.

Two types of information will be created when a summary mode report is run. All of this information will be reported to the SYSPRINT DD statement by the program.

The information will contain a summary of the whole load library, detailing how many modules were scanned, how many modules are in the library, how many were not scanned, and how many of the requested commands were found in the whole library. This is shown below:

```
Total modules in library = 98
Total modules Scanned = 98
Total CICS modules/tables not scanned = Ø
Total modules possibly containing requested commands = 15
```

You will also obtain a list of members from the library that have the commands you are interested in, as detailed in your filter list. This information will show how many commands are in each load module and what language the module was written in. The example below shows a sample of this information:

```
Module Name                         Commands Found        Language

'CICS.LOAD (BDMANY)'                   2                  Assembler
'CICS.LOAD(BDTWICE)'                   3                  Assembler
'CICS.LOAD(BDPGMA)'                    1                  Assembler
'CICS.LOAD(BDPGMB)'                    1                  Assembler
'CICS.LOAD(BDPGMØ)'                    1                  Assembler
'CICS.LOAD(SS2PRGA)'                   2                  Assembler
'CICS.LOAD(SS2PRG1)'                   2                  Assembler
'CICS.LOAD(SS2PRG2)'                   2                  Assembler
'CICS.LOAD(SS2PRG3)'                   2                  Assembler
'CICS.LOAD(SS2PRG4)'                   2                  Assembler
'CICS.LOAD(SS3PRG5)'                   2                  Assembler
'CICS.LOAD(SS3PRG6)'                   2                  Cobol
'CICS.LOAD(SS3PRG7)'                   2                  Cobol
'CICS.LOAD(SS3PRG8)'                   2                  Assembler
'CICS.LOAD(SS3PRG9)'                   2                  Assembler
```

This list of modules can also be written to a file, which is allocated to the DFHDTL DD statement by specifying the DETAILMODS parameter along with the SUMMARY parameter on the EXEC statement of the job step. Then it can be modified and used as input to a detailed report.

When you run in detail mode, a report is output to the SYSPRINT DD statement showing exactly which commands are in each of the load modules scanned. You run a detailed report by coding PARM=('DETAIL') on the EXEC statement in the JCL.

The detail run will scan only those modules listed in the input file pointed to by DD DFHLIST unless you add ALL to the parameter statement.

The example below shows a sample report produced by running with DETAIL coded:

```
SCAN PERFORMED ON Wed June 15 19:23:Ø5 2ØØ4 USING TABLE RSTABLE2.3

DETAILED LISTING OF DD:DFHLIST

Module Name 'CICS.LOAD(BDMANY)'
Module Language Assembler
Offset/EDF Command
---------------- -----------------------------------------
ØØØØ181Ø/no-edf START TRANSID FROM LENGTH INTERVAL
ØØØØ1993/no-edf SEND TEXT FROM LENGTH FREEKB TERMINAL
```

```
Module Name 'CICS.LOAD(BDTWICE)'
Module Language Assembler
Offset/EDF Command
---------------- -----------------------------------------
ØØØØ1861/no-edf INQUIRE CLASSCACHE PROFILE
ØØØØ2174/no-edf INQUIRE JVM PROFILE
ØØØØ22Ø3/no-edf ASKTIME ABSTIME

Total possible commands located = 5

LOAD LIBRARY STATISTICS
=======================
Total modules in library = 2
Total modules Scanned = 2
Total CICS modules/tables not scanned = Ø
Total modules possibly containing requested commands = 2
```

One important point to mention is that you must ensure that APAR PQ78531 is applied if you intend to scan very large libraries of load modules in a single run. This will prevent possible storage problems when running against load libraries with 80 or more load modules. Be aware, even with this fix applied, that DFHEISUP is a resource-intensive program, particularly where CPU is concerned. So discretion should be used as to when it is run and against how many modules.

John Bradley
Systems Programmer
Meerkat Computer Services (UK)

CICS transaction segregation and region creation – part 2

This month we conclude the article looking at CICS security.

CICS PARAMETERS

While taking all this time to focus on the RACF side of security, let us not forget that all we’ve discussed won’t be worth anything if you don’t configure CICS to actually hand over security control.

Parameters must be set within the CICS System Initialization Table (SIT) to activate resource security management for CICS regions. The following list details those SIT parameters that are coded to enable transaction security:

- CMDSEC = ALWAYS – CICS uses RACF to secure systems programming commands.
- SEC = YES – CICS uses full RACF support.
- SECPRX = YES – CICS prefixes the resource names with its RACF userID when passing authorization requests to RACF.
- DFLTUSER = ? – identifies to CICS the default RACF userID to be used when users have not signed on to CICS. See Figure 1.
- XTRAN = YES – CICS uses TCICSTRN and GCICSTRN for resource class profiles for transaction security checking.
- CMDSEC=ALWAYS is the preferred setting for security, and should be implemented for all regions unless a

CICS Region Type	*Prefix*	*UserID*
ACCEPTANCE TESTING	A	ACIXDFLT
DEVELOPMENT TESTING	D	DCIXDFLT
ENGINEERING PERFORMANCE TESTING	E	ECIXDFLT
IMPLEMENTATION TESTING	I	ICIXDFLT
INFRASTRUCTURE INTEGRATION TEST	F	FCIXDFLT
PRODUCTION	P	PCIXDFLT
REGRESSION TESTING	R	RCIXDFLT
SECURITY TESTING	Z	ZCIXDFLT
SYSTEM TESTING	S	SCIXDFLT
UNIT TESTING	U	UCIXDFLT

Figure 1: Naming convention

specific business case can be made to use CMDSEC=YES instead. ALWAYS does have some potential for impact on operational overhead, but provides more comprehensive protection of CICS command functions and transactions. IBM's documentation recommends CMDSEC=YES because of the potential overhead issue, but the overhead actually comes from the CEMT transaction and EXEC CICS commands, not from the garden-variety CICS transactions. This issue has been raised to IBM's Hursley Laboratories, and I believe they're going to switch to recommending CMDSEC=ALWAYS in their future documentation.

- CMDSEC=ALWAYS should remain in effect even in cases where the overhead increase is>5% of processing capacity and/or adds 1.5 seconds per standard CICS transaction, unless the region owner can prove that this is measurably detrimental to income.

The standard for the DFLTUSER SIT parameter should be in a standard naming convention, with a different one for each general type of CICS region. See Figure 1.

Remember that you have actually to define those user IDs within RACF. Don't, however, give those IDs access to anything. They should be protected IDs, with no connection to the CICS regions or transactions except as a default definition in the SIT.

STARTED TASK AND REGION OWNERSHIP

When it comes to CICS regions, I'm a firm believer in defining them as started tasks – always! I've yet to come across a situation where a CICS region should not be defined that way. If you have, please let me know. You can follow whatever naming conventions you wish for the primary user ID, which is then linked to the STARTED General Resource class, although a structured name that can easily match back to the four-character prefix used in the examples here should be considered. It's important to ensure that the user ID is marked as a PROTECTED user in RACF and that no terminal access is available to the ID proper.

Region ownership is a slightly different issue, and while not technical from a RACF standpoint it should still be discussed. The technical ownership within RACF is simple enough – a high-level group profile. But that's not the same as ownership of the region. Who is the person or department responsible for the region? Who provides the permission to the business side of the transactions (as opposed to the technical)? It is important to identify these people (primary and back-up) and include them on the installation data. Why? Because you want to make sure that access to the region is explicitly permitted by the owner of the region. If something goes wrong, you want to know as quickly as possible who to contact, and this is the most efficient way.

By the way, for those of you concerned about changing the names and phone numbers of the authorizers, all you need to do is keep a copy of the region creation JCL. Use a subset of the JCL to change RDEFINE to RALTER, strip off the transaction ADDMEM instructions and PERMIT commands, and just update the installation data with the new name(s). This can be done once again with a simple find/replace command, provided you keep the spacing correct (see my comments on those annoying installation data formats below).

THOSE ANNOYING INSTALLATION DATA FORMATS...

Wow, that didn't take long, did it?

You'll note that the JCL in the appendices looks unusual when it comes to the installation data, especially for the General Resource profiles. This is because of IBM's rather arcane formatting of the data for different types of RACF profile. While all RACF profiles have a maximum length of 255 characters for the installation data field, they are displayed differently. Below is an example of each type of installation data for the various profiles, so that you can get an idea of the actual display structure.

General Resource profiles:

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
FULL NAME AND DESCRIPTION OF CICS REGION HERE - SHOULD BE SAME FOR ALL PROFILES
IBM SUPPLIED TRANS - CAT 1 - NON-TERMINAL TRANS - CICS INTERNAL ONLY - NO USERS
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818
RRRRRRRR     &&&&&
```

Dataset profiles:

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
CICS PRODUCTION REGION -  RRRRRRRR  - DATASET  PROFILE
DESCRIBE DATASET HERE  DESCRIBE DATASET HERE  DESCRIBE
DESCRIBE DATASET HERE  DESCRIBE DATASET HERE  DESCRIBE
DESCRIBE DATASET HERE  DESCRIBE DATASET HERE  DESCRIBE
SEE HIGH LEVEL QUALIFIER FOR OWNER$$$$$
```

User ID profiles:

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
CICS PRODUCTION REGION - STARTED TASK USER PROFILE
DESCRIBE REGION HERE DESCRIBE REGION HERE DESCRIBE
DESCRIBE REGION HERE DESCRIBE REGION HERE DESCRIBE
OWNER=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA 777-777-1717
PRGMR=BBBBBBBBBBBBBBBBBBBBBBBBBBBBBB 666-666-1616
#####
```

Group profiles:

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
CICS PRODUCTION REGION - RRRRRRRR - APPL G001
DESCRIBE REGION HERE DESCRIBE REGION HERE DES
DESCRIBE TRANSACTIONS OR FUNCTIONS USED  HERE
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414
APPLICATN GROUP RRRRRRRR %%%%%
```

I don’t know about you, but I find this exceedingly aggravating. Why IBM didn’t just pick one of them and stick to it is beyond me. Personally, I prefer the user ID layout (five lines of 50 characters each, plus a five character tag line), so I wish that IBM would consider updating RACF in the future to cover *all* record types with the same installation data layout.

Since trying to count characters when entering the installation data freeform on a TSO screen is nigh on impossible, the only way to adequately control the formatting of the output is through the use of structured JCL. And, since JCL allows only 72 characters of effective input space, it is necessary to split the installation data input for the General Resource profiles. It is important to keep the continuation character (-) in the correct position when copying and updating the sample JCL in the appendices. Otherwise, the displayed installation data will look decidedly weird.

One way to keep the data properly formatted is to use Microsoft Word's replace function to edit a standard set of JCL. For example, when updating the base JCL for a new system, take a single formatted line of JCL and put that in the 'From' slot, and then put the changed data in the 'To' slot, making sure the number of spaces is exactly correct. Keeping the font as Courier New should help with the alignment process.

One tip: when using MS Word to edit your JCL, be sure to do a final find/replace to change the long dash (the one that automatically goes between typed words with spaces) with the short dash (which IBM recognizes as the continuation character). The (–) versus (-) issue can cause you some potential problems unless you remember to address it before uploading your JCL.

REFERENCES

- *CICS RACF Security Guide* – http://publibfp.boulder.ibm.com/epubs/pdf/dfht5p09.pdf.
- *CICS Supplied Transactions* – http://publibfp.boulder.ibm.com/epubs/pdf/dfha7p08.pdf.
- *CICS Resource Definition Guide* – http://publibfp.boulder.ibm.com/epubs/pdf/dfha4p11.pdf.
- *CICS Front-End Programming Interface User's Guide* – http://publibfp.boulder.ibm.com/epubs/pdf/dfhp7p00.pdf.

Appendix A contains a detailed list of all the IBM-supplied CICS transactions and their individual General Resource profile groupings. It can be downloaded from our Web site at www.xephon.com/extras/DFAppA.doc.

Appendix B contains the IBM-supplied JCL template to actually create these profiles. It can be downloaded from our Web site at www.xephon.com/extras/DFAppB.doc.

Appendix C contains a full description of the CICS commands and how they are segregated. It can be downloaded from our Web site at www.xephon.com/extras/DFAppC.doc.

APPENDIX D: CICS COMMAND SEGREGATION

The JCL to create the General Resource profiles is listed below:

```
RDEFINE VCICSCMD xxxxMØØ1  -
OWNER(zzzzGMØ1)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 1 - INQUIRY-
 ONLY COMMANDS - TECHNICAL SUPPORT USER-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ))    -
UACC(NONE)         -
ADDMEM(RRRRRRRR.BEAN     -
   RRRRRRRR.CFDTPOOL   -
   RRRRRRRR.EXCI     -
   RRRRRRRR.MVSTCB    -
   RRRRRRRR.RRMS     -
   RRRRRRRR.STORAGE    -
   RRRRRRRR.STREAMNAME  -
   RRRRRRRR.SUBPOOL    -
   RRRRRRRR.UOWDSNFAIL  -
   RRRRRRRR.UOWENQ)
PERMIT xxxxMØØ1       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ1       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ1)
RDEFINE VCICSCMD xxxxMØØ2  -
OWNER(zzzzGMØ2)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 2 - DB2 COM-
MANDS - DATABASE TECHNICAL SUPPORT USER-
AUTH1:MMMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(UPDATE)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.DB2CONN   -
   RRRRRRRR.DB2ENTRY   -
   RRRRRRRR.DB2TRAN)
PERMIT xxxxMØØ2       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ2       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ2)
RDEFINE VCICSCMD xxxxMØØ3  -
OWNER(zzzzGMØ3)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 3 - TERMINA-
L-MONITOR-TCPIP COMMANDS - TECH SUPPORT-
AUTH1:MMMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(UPDATE)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.CONNECTION -
   RRRRRRRR.IRC      -
   RRRRRRRR.MONITOR    -
   RRRRRRRR.TCPIP     -
   RRRRRRRR.TCPIPSERVICE -
   RRRRRRRR.TERMINAL   -
   RRRRRRRR.TSMODEL    -
   RRRRRRRR.TSPOOL    -
   RRRRRRRR.TSQNAME    -
   RRRRRRRR.VTAM     -
   RRRRRRRR.WEB)
PERMIT xxxxMØØ3       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ3       -
```

```
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ3)
RDEFINE VCICSCMD xxxxMØØ5  -
OWNER(zzzzGMØ5)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 5 - GENERAL-
 USAGE COMMANDS -  TECHNICAL SUPPORT-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(UPDATE)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.AUTINSTMODEL -
   RRRRRRRR.AUTOINSTALL  -
   RRRRRRRR.BRFACILITY  -
   RRRRRRRR.CORBASERVER  -
   RRRRRRRR.DELETSHIPPED -
   RRRRRRRR.DISPATCHER  -
   RRRRRRRR.DJAR     -
   RRRRRRRR.DOCTEMPLATE  -
   RRRRRRRR.DSNAME    -
   RRRRRRRR.DUMPDS    -
   RRRRRRRR.ENQMODEL   -
   RRRRRRRR.FILE     -
   RRRRRRRR.JOURNAL    -
   RRRRRRRR.JVMPOOL    -
   RRRRRRRR.MODENAME   -
   RRRRRRRR.PARTNER    -
   RRRRRRRR.PROFILE    -
   RRRRRRRR.PROGRAM    -
   RRRRRRRR.REQUESTMODEL -
   RRRRRRRR.SYSDUMPCODE  -
   RRRRRRRR.SYSTEM    -
   RRRRRRRR.TASK     -
   RRRRRRRR.TCLASS    -
   RRRRRRRR.TDQUEUE    -
   RRRRRRRR.TRANDUMPCODE -
   RRRRRRRR.TRANSACTION  -
   RRRRRRRR.UOW)
PERMIT xxxxMØØ5       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ5       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ5)
RDEFINE VCICSCMD xxxxMØØ6  -
```

```
OWNER(zzzzGM06)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 6 - EXEC CI-
CS-LEVEL COMMANDS - TECHNICAL SUPPORT-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(READ)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.EXITPROGRAM -
   RRRRRRRR.REQID     -
   RRRRRRRR.STATISTICS  -
   RRRRRRRR.TRACEDEST   -
   RRRRRRRR.TRACEFLAG   -
   RRRRRRRR.TRACETYPE   -
   RRRRRRRR.TSQUEUE)
PERMIT xxxxM006       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxM006       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGM06)
RDEFINE VCICSCMD xxxxM007  -
OWNER(zzzzGM07)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 7 - CEMT CI-
CS-LEVELCOMMANDS - TECHNICAL SUPPORT-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(READ)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.DUMP    -
   RRRRRRRR.JOURNALMODEL -
   RRRRRRRR.LINE     -
   RRRRRRRR.PROCESSTYPE  -
   RRRRRRRR.UOWLINK)
PERMIT xxxxM007       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxM007       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGM07)
RDEFINE VCICSCMD xxxxM008  -
```

```
OWNER(zzzzGMØ8)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 8 - HIGH-EN-
D TECH SUPPORT ONLY COMMANDS - TECH SUP-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(READ)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.FEPIRESOURCE -
   RRRRRRRR.LSRPOOL    -
   RRRRRRRR.MAPSET    -
   RRRRRRRR.PARTITIONSET -
   RRRRRRRR.RESETTIME   -
   RRRRRRRR.SESSIONS   -
   RRRRRRRR.SHUTDOWN   -
   RRRRRRRR.TYPETERM)
PERMIT xxxxMØØ8       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ8       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ8)
RDEFINE VCICSCMD xxxxMØØ9  -
OWNER(zzzzGMØ9)       -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
CICS SUPPLIED COMMANDS - LVL 9 - SECURIT-
Y ONLY COMMANDS - SECURITY DEPT ONLY-
AUTH1:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 AUTH2:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
AUDIT(FAILURES(READ)SUCCESS(READ)) -
UACC(NONE)         -
ADDMEM(RRRRRRRR.SECURITY)
PERMIT xxxxMØØ9       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(RRRRRRRR)
PERMIT xxxxMØØ9       -
   CLASS(VCICSCMD)    -
   ACCESS(READ)      -
   ID(xxxxGMØ9)
```

APPENDIX E: CICS REGION STARTED TASK DEFINITION

```
ADDUSER RRRRRRRR         -
```

```
OWNER(CIX#PRD)          -
DFLTGRP(CIX#PRD)         -
NAME(XXXXXXXXXXXXXXXXXXXX)    -
NOPASSWORD            -
DATA('CICS PRODUCTION REGION - STARTED TASK USER PROFILE-
DESCRIBE REGION HERE DESCRIBE REGION HERE DESCRIBE-
DESCRIBE REGION HERE DESCRIBE REGION HERE DESCRIBE-
OWNER=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA 777-777-1717-
PRGMR=BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB 666-666-1616-
#####')
RDEFINE STCGROUP (RRRRRRRR.*)  -
STDATA(USER(RRRRRRRR)      -
OWNER(RRRRRRRR)         -
GROUP(CIX#PRD))         -
DATA('FULL NAME AND DESCRIPTION OF CICS REGION-
 HERE - SHOULD BE SAME FOR ALL PROFILES-
DESCRIBE REGION HERE DESCRIBE REGION HER-
E DESCRIBE REGION HERE DESCRIB REGION H-
OWNER:MMMMMMMMMMMMMMMMMMMM 999-999-1919-
 PRGMR:NNNNNNNNNNNNNNNNNNNN 888-888-1818-
RRRRRRRR  &&&&&')          -
UACC(NONE)
```

APPENDIX F: CICS REGION GROUP DEFINITION

```
ADDGROUP zzzzGRPS        -
OWNER(RRRRRRRR)         -
SUPGROUP(CIX#PRD)        -
NAME(RRRRRRRR - PRIME GRP)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - PRIME GRP-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
TOP GROUP FOR CICS REGION - TREE STARTS HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
PRIMARY GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGRPS)      -
   AUTHORITY(USE)
ADDGROUP zzzzGØØØ        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGRPS)        -
NAME(RRRRRRRR - APPL TRAN)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - APPL TRAN-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
TOP GROUP FOR CICS APPLICATION TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
APPLICATN GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
```

```
   GROUP(zzzzGØØØ)      -
   AUTHORITY(USE)
ADDGROUP zzzzGCØØ        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGRPS)        -
NAME(RRRRRRRR - IBM TRANS)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
TOP GROUP FOR IBM SUPPLIED CICS TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGCØØ)      -
   AUTHORITY(USE)
ADDGROUP zzzzGSØØ        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGRPS)        -
NAME(RRRRRRRR - UTIL GRPS)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - UTIL GRPS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
TOP GROUP FOR CICS UTILITY  TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
UTILITY GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGSØØ)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØØ        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGRPS)        -
NAME(RRRRRRRR - IBM CMDS)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØØ)      -
   AUTHORITY(USE)
ADDGROUP zzzzGØØ1        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGØØØ)        -
NAME(RRRRRRRR - APPL GØØ1)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - APPL GØØ1-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
```

```
APPLICATN GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGØØ1)      -
   AUTHORITY(USE)
ADDGROUP zzzzGØØ2        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGØØØ)        -
NAME(RRRRRRRR - APPL GØØ2)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - APPL GØØ2-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
APPLICATN GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGØØ2)      -
   AUTHORITY(USE)
ADDGROUP zzzzG999        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGØØØ)        -
NAME(RRRRRRRR - APPL G999)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - APPL G999-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
APPLICATN GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzG999)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC11        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC11)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS - NON-TERM - REGION ONLY-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC11)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC21        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC21)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS  SYSTEM ADMINISTRATION-
```

```
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC21)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC22        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC22)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS       DEVELOPMENT-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC22)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC23        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC23)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS       INQUIRY ONLY-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC23)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC24        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC24)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS  OPERATOR TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC24)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC25        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC25)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS  INTERCOM TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC25)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC26        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC26)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS   RPCUSER TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC26)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC27        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC27)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS AFFINITIES TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC27)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC28        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC28)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS   WEBUSER TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC28)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC29        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
```

```
NAME(RRRRRRRR - IBM GC29)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS ALLUSER GM TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC29)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2A        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGC00)        -
NAME(RRRRRRRR - IBM GC2A)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS ALLUSER CRTX TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2A)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2B        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGC00)        -
NAME(RRRRRRRR - IBM GC2B)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS  IIOPUSER TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2B)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2C        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGC00)        -
NAME(RRRRRRRR - IBM GC2C)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS    DBCTL TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2C)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2D        -
```

```
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC2D)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS **C E C I** TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2D)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2E        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC2E)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS **C I N D** TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2E)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC2F        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC2F)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS **C E M T** TRANSACTIONS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC2F)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC31        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC31)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM-SUPPLIED TRANS ALL CICS TERMINAL USERS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC31)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC91        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC91)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
CICS RESOURCE DEF GUIDE & CICS SUPPLIED TRANS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC91)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC92        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC92)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS RESOURCE DEFINITION TRANSACTION ONLY-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC92)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC93        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC93)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS SUPPLIED TRANSACTION MANUAL ONLY-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGC93)      -
   AUTHORITY(USE)
ADDGROUP zzzzGC94        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGCØØ)        -
NAME(RRRRRRRR - IBM GC94)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM TRANS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS FRONT-END PROGRAMMING INTERFACE ONLY-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
IBMTRANS GROUP RRRRRRRR %%%%%')
```

```
CONNECT RRRRRRRR         -
   GROUP(zzzzGC94)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ1        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ1)     -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS   INQUIRY ONLY COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ1)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ2        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ2)     -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS       DB2 COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ2)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ3        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ3)     -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS TERM-MONITOR-TCPIP COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ3)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ5        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ5)     -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS   GENERAL USE COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
```

```
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ5)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ6        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ6)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS EXEC CICS-LEVEL COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ6)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ7        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ7)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS    CEMT-LEVEL COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ7)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ8        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ8)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
IBM CICS COMMANDS HIGH-END TECH SUPP COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ8)      -
   AUTHORITY(USE)
ADDGROUP zzzzGMØ9        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGMØØ)        -
NAME(RRRRRRRR - IBM GMØ9)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - IBM CMDS-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
```

```
IBM CICS COMMANDS HIGH-END SECURITY COMMANDS-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
COMMAND GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGMØ9)      -
   AUTHORITY(USE)
ADDGROUP zzzzGSØ1        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGSØØ)        -
NAME(RRRRRRRR - APPL GSØ1)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - UTIL GSØ1-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
UTILITY GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGSØ1)      -
   AUTHORITY(USE)
ADDGROUP zzzzGSØ2        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGSØØ)        -
NAME(RRRRRRRR - APPL GSØ2)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - UTIL GSØ2-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
UTILITY GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGSØ2)      -
   AUTHORITY(USE)
ADDGROUP zzzzGS99        -
OWNER(RRRRRRRR)         -
SUPGROUP(zzzzGØØØ)        -
NAME(RRRRRRRR - UTIL GS99)    -
DATA('CICS PRODUCTION REGION - RRRRRRRR - UTIL GS99-
DESCRIBE REGION HERE DESCRIBE REGION HERE DES-
DESCRIBE TRANSACTIONS OR FUNCTIONS USED HERE-
OWNER=CCCCCCCCCCCCCCCCCCCCCCCCCC 555-555-1515-
PRGMR=DDDDDDDDDDDDDDDDDDDDDDDDDD 444-444-1414-
UTILITY GROUP RRRRRRRR %%%%%')
CONNECT RRRRRRRR         -
   GROUP(zzzzGS99)      -
   AUTHORITY(USE)
```

Doc Farmer
Independent Security Consultant (USA)

CICS Interdependency Analyzer

This article discusses the CICS Interdependency Analyzer. First I’ll look at what the product aims at doing, then go on to describe its main components. Finally, information is given on what is required to run the product and where to obtain detailed information.

There are numerous business situations that the CICS Interdependency Analyzer can be used to address. These obviously vary in nature and by industry.

Here are some of the main reasons why this product is needed:

- *Mergers and acquisitions*. Many companies have merged or been acquired, resulting in a merger of CICS workloads. This merging can be for performance reasons, to isolate work, or even for cost savings to be realized. Because these applications may not be well understood by the acquiring company, or documentation may be inadequate, there is a need to understand all of the resources that are associated with a given application.
- *Outsourcing*. Outsourcing companies are continually facing the problem of running CICS applications with which they are unfamiliar. Naming standards are lacking or conflict with other applications running on the same LPAR. Documentation may be non-existent or incomplete – a common theme at most installations, sadly.
- *Maintenance*. During the normal application life-cycle, CICS applications require maintenance and enhancement. When a programmer who is unfamiliar with the application that he is required to modify starts the modification process, much time can be spent in trying to understand the application and the flow of transactions. The use of CICS Interdependency Analyzer can reduce this learning curve.

CICS Interdependency Analyzer for z/OS and OS/390 is a batch and run-time system that is used with CICS Transaction server for z/OS. It can be used to identify CICS application resources and their associated interdependencies. This means you can then understand the functionality of a specific application – such as which transactions use which programs, which resources are no longer utilized, which programs use which files, maps, queues, etc.

CICS Interdependency Analyzer captures interdependency information while CICS is running and stores it in VSAM files. You can then produce detailed reports from the data in these files. You can also use the data from the VSAM files to populate DB2 tables. CICS Interdependency Analyzer loads the affinity data collected by The Affinities Utility (TAU), a component of CICS TS, into its DB2 databases. The data can be queried with CICS Interdependency Analyzer on-line queries, with batch SQL queries, or with other SQL query software tools. You can also use batch SQL to produce detailed reports.

Because many CICS applications have a long history, their growth has not been as structured as it might have been, with the result that many applications and services share common resources, and changes in one area typically affect many others. In some installations this often results in a fear of changing applications because of the potential knock-on effect.

By using the CICS Interdependency Analyzer to help you achieve a level of understanding of an application, changes can be made more easily. If you need to change the content or structure of a file, knowing which programs use this file is a must because they will need to be changed too. CICS Interdependency Analyzer can tell you this and also inform you which transactions drive the programs. CICS Interdependency Analyzer records the interdependencies between resources such as files, programs, and transactions by monitoring programming commands that operate on resources.

It is also possible to instruct CICS Interdependency Analyzer to monitor some types of non-CICS commands that operate on non-CICS resources. Examples of non-CICS commands that can be monitored include:

- EXEC DLI calls to IMS database resources.
- Dynamic COBOL calls to other programs.
- DB2 calls.
- MQ calls to WebSphere MQ resources.

The Collector component of CICS Interdependency Analyzer collects the dependencies that apply to a single CICS region. It can be run against production CICS regions or in test environments.

From the interactive interface of CICS Interdependency Analyzer you can control Collectors running on multiple regions.

Some of the product's features are:

- A batch job is provided to load data captured by the Transaction Affinities Utility. Extensions are provided to the CICS query interface to assist in the analysis of data in the affinities data tables.
- A set of sample SQL queries is provided to allow resource comparisons on the data in the DB2 tables.
- Comparison of applications across regions.
- Comparison of all interdependencies across regions.
- Improved data capture facilities are provided in the Collector for the DB2 subsystem.
- The Collector function has been extended so that it now supports long resource names of up to 200 bytes. Previously this was limited to 50 bytes.
- New installation verification procedures, including sample data, are now provided to assist the installation process.

- Task Control Block information is now added to the data gathered by the Collector.
- A new capability allows differentiation between types of Temporary Storage Queues (TSQs) – main, auxiliary, and Coupling Facility.
- A set of sample SQL queries and jobs is provided to enable housekeeping functions to be performed on the data in the DB2 tables.
- Provides an option to load Scanner data in a DB2 table.

Below are examples of some of the questions you might ask the CICS Interdependency Analyzer:

- Which regions does a CICS application execute in?
- What are all of the CICS resources used by an application?
- What are all of the CICS resources used by an transaction?
- What transactions belong to a given application?
- What programs does a transaction invoke?
- What transactions access a particular file and how?
- What resources are used by a specific program?
- How is a file accessed by a particular program?

This gives a small insight into how powerful the tool can be if utilized correctly.

The whole design of the CICS Interdependency Analyzer is based on the concept of examining the EXEC CICS commands used by applications. Each command and its parameters indicate the resources that will be used by the program. An analysis of these calls provides a view of resource interdependencies in an application.

The Analyzer’s Scanner component examines the program load modules and reports on the EXEC CICS commands and their parameters. It produces a report that tells you the commands issued by each program, the programming language used, and the resources that are involved. The data collected can then be loaded into DB2 tables.

The Scanner does not show the execution-time path through the code and which commands are executed.

The second component of the product is the Collector, which provides a capture function and stores the data in an MVS data space. The collector function can be activated across multiple CICS regions from a single point of control, and the data can be collected across these regions and written to a VSAM file shared between these regions using a File-Owning Region (FOR) or using RLS.

After data collection is complete, the CICS Interdependency Analyzer provides a set of utilities to enable the data you have collected to be loaded into DB2 tables. This provides you with an opportunity to perform detailed analysis using standard SQL queries or the on-line interface provided by CICS.

The data being collected is initially stored in a data space. This data is written to VSAM files at a scheduled time or when certain operator commands are issued. The files can be examined using the Reporter component of the product. This component uses a group of batch programs to produce summary or detailed reports.

You can monitor and control the Collector component using the CINT transaction. This transaction enables you to start, pause, continue, and stop the collection of dependency data in the tables in the data space. Using the CINT transaction, you can also specify for which dependency commands, and for which transactions, data is to be collected.

To use CICS Interdependency Analyzer you need the following:

- OS/390 Version 2.10 or later.
- CICS Transaction Server for OS/390 or CICS Transaction Server for z/OS.

- Each CICS region on which the CICS Interdependency Analyzer Collector is to run must have either Language Environment installed and active, or a COBOL run-time environment.
- A CICS region on which the CICS Interdependency Analyzer Query interface is to be run must have DB2 installed and active.
- To control CICS Interdependency Analyzer Collectors on multiple regions from a single CICS terminal, the VSAM files to which CICS saves dependency data and control information must be shared across all the regions. To share these files, you can use VSAM Record-Level Sharing (RLS). If you use VSAM RLS, all regions must be in the same MVS parallel Sysplex.
- Function shipping to a File-Owning Region (FOR).

Note that CICS Interdependency Analyzer is a one-time-charge product and is not included as a part of CICS Transaction Server. The program product number for CICS Interdependency Analyzer is 5655-G76.

Below is a list of IBM publications from which more information on the CICS Interdependency Analyzer can be obtained:

- *CICS IA User Guide and Reference* – SC34-6365.
- *DB2 UDB for OS/390 and z/OS Administration Guide V7* – SC26-9931
- *DB2 UDB for OS/390 and z/OS Administration Guide V8* – SC18-7413
- *CICS Transaction Affinity Utility Guide* – SC34-6013
- *CICSPlex SM Administration* – SC34-6256.

John Bradley
Systems Programmer
Meerkat Computer Services (UK)

CICS news

GT Software has announced Release 2.0 of Ivory Web services. The Ivory Process for Web services offers methods for bottom-up and top-down processing for CICS Transaction Server composite services. Ivory Studio and Ivory Server combine to provide composite outbound/inbound Web services in addition to the composite application process provided in Release 1. The new WSDL discovery and import function allows Ivory to reuse any Web service meta data and map that data onto existing CICS applications.

Ivory provides CICS programmers with access to several services using a single API call to Ivory Server. In addition to composite Web services, Ivory has an IDE with drag-and-drop capabilities to build Web services.

CICS programmers do not need to learn complex Web service processing; they use a standard CICS call to access Ivory Server, and Ivory acts as a SOAP client using HTTP or HTTPS processing for the programmer.

The Ivory wizard provides a method for building CICS Web services by stepping through a set of Wizard dialogs. This process provides a very fast path to building and exposing CICS applications as Web services. The process will automatically build the WSDL for the application, removing the COBOL look and feel from the exposed WSDL meta data.

For further information contact:
GT Software, 1314 Spring Street NW, Atlanta, GA 30309-2810, USA.
Tel: (404) 253 1300.
URL: www.gtsoftware.com/products/ivory.php.

***

IBM has announced a new release of its CICS Interdependency Analyzer for z/OS (CICS IA), which supports run-time analysis of CICS applications. The new release automates the detection of run-time resource relationships within a CICS system, and records this data within a DB2 database. CICS IA also provides flexible reports to help users analyse the collected information, build a relationship roadmap, and use this data in daily operations to improve the ability to maintain, enhance, and migrate their business applications.

For further information contact your local IBM representative.

***

IBM plans to release a Linux-based CICS offering this year to provide a migration path for TXSeries CICS customers who want to consolidate transactional application workloads onto a single Linux operating environment.

For further information contact your local IBM representative.

***