200

CICS update

July 2002

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 £170 ($260) per 1000 words and £100 ($160) per 100 lines of code for the first 200 lines of original material. The remaining code is paid for at the rate of £50 ($80) per 100 lines. In addition, there is a flat fee of £30 ($50) per article. 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.

Refresh user programs in a long-running CICS session

If you are working with a CICS system that is running for more than 24 hours, sometimes it will be necessary to refresh the user programs. In order to ensure that everything runs successfully, you can use the application CSRELOAD.

CSRELOAD

```
*ASM XOPTS(CICS,SP)
CSRELOAD TITLE '*** Reload all user programs in a running CICS ***'
         SPACE
*---------------------------------------------------------------------*
*                C S R E L O A D                                      *
*---------------------------------------------------------------------*
* Note:      Program should be linked with AMODE=31 / RMODE=ANY       *
* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
* Comment:   This program is designed to reload the user programs in  *
*            a running CICS.                                          *
*                                                                     *
*            CSRELOAD can be called via a CICS - API (eg E.C. LINK,   *
*            E.C. XCTL) or via an own transaction.                    *
*                                                                     *
*            The usage of this program is helpful in a CICS which is  *
*            running more  than 24 hours. So that is guaranteed that  *
*            we don't work with obsolete programs.                    *
*---------------------------------------------------------------------*
         SPACE
* ******************************************************************* *
* Author: Claus Reis / April, 2002                                    *
* ******************************************************************* *
* ******************************************************************* *
         EJECT
* ******************************************************************* *
*        D e f i n i t i o n s                                        *
* ******************************************************************* *
         SPACE
*      INCLUDE++ CSCWAA          INCLUDE CICS CWA ASSEMBLER STRUCTURE
         EJECT
*-----------------------------------------------------
*        C I C S       C W A - A R E A                            *
*        INCLUDE-ELEMENT  FOR ASM  PROGRAM  CSCWAA                *
*-----------------------------------------------------
*                                                                  *
```

```
*        ADDRESSNG    :   EXEC CICS ADDRESS CWA(CWAPTR)              *
*                                                                     *
*        ATTENTION    :   NO CHANGES ALLOWED - READ-ONLY             *
*                                                                     *
*                         CWAPTR MUST BE DEFINED                      *
*                                                                     *
*---------------------------------------------------------
               SPACE 3
               USING CWADSECT,CWAPTR
               SPACE 3
CWADSECT       DSECT
               SPACE 3
CWAAREA        DS   ØCL1536         CWA-BEREICH
               SPACE 1
CWAEYECA       DC    CL4' '         EYECATCHER
CWAVSYS        DC    CL4' '         ZUGEORDNETE VSAM-SYSID (CWACICID)
CWATSYS        DC    CL4' '         ZUGEORDNETE TERM-SYSID (CWACICID)
CWASYSID       DC    CL4' '         ORIGINAL SYSTEM-ID
CWAPPLID       DC    CL8' '         ORIGINAL APPLICATION-ID
CWANCVT        DC    AL4(Ø)         POINTER NLV-CVT
CWA_PTR_CUATR  DC    AL4(Ø)         ADRESSE D.CUA-TRANSAKTIONSTABELLE
CWA_CICSLEVEL  DC   ØCL4' '         CICS-LEVEL 'Ø311' OR 'Ø33Ø'
CWA_CICSLEV    DC    CL1' '         CICS-LEVEL
CWA_CICSVER    DC    CL1' '         CICS-VERSION
CWA_CICSREL    DC    CL1' '         CICS-RELEASE
CWA_CICSMOD    DC    CL1' '         CICS-MODIFICATION
CWA_CMFSTOP    DC    PL4'Ø'         STOP-TIME FOR CMF-EVENTS HHMMSSTC
               DS    XL56Ø          .........  FREI  .........
               DS    XL12Ø          .........  FREI  .........
               DS    XL22           .........  FREI  .........
CWACICTX       DC    CL4' '         CICS-ID-BESCHREIBUNG
CWACICID       DC    CL1' '         CICS-ID
CWA$PROD       EQU   C'P'            .. PROD
CWA$TEST       EQU   C'T'            .. TEST
CWA$VPRD       EQU   C'V'            .. VORPROD
CWA$SYST       EQU   C'S'            .. SYSTEM-CICS
CWACICNR       DC    CL1' '         CICS-NR
CWA$TERM       EQU   C'T'            .. TERMINAL
CWA$VSAM       EQU   C'V'            .. DATASET VSAM
CWA$PAIS       EQU   C'P'            .. PAISY
CWA$ODM        EQU   C'O'            .. ODM
CWA$INFO       EQU   C'I'            .. INFO
CWA$APPL       EQU   C'Ø'            .. APPLICATION ØØ-Ø9
*  $APPL       EQU   ????            .. APPLICATION A-C
*  $APPL       EQU   ????            .. APPLICATION E-O
*  $APPL       EQU   ????            .. APPLICATION Q-S
*  $APPL       EQU   ????            .. APPLICATION U-Z
CWADATUM       DC    CL8' '         DATUM FORMAT TT.MM.JJ
CWACTMJ        DC    CL6' '         DATUM TTMMJJ
CWAPTMJ        DC    PL4'Ø'         DATUM ØTTMMJJC
```

```
CWACJMT         DC    CL6' '          DATUM JJMMTT
CWAPJMT         DC    PL4'Ø'          DATUM ØJJMMTTC
CWACTMJ4        DC    CL8' '          DATUM TTMMJJJJ
CWAPTMJ4        DC    PL5'Ø'          DATUM ØTTMMJJJJC
CWACJ4MT        DC    CL8' '          DATUM JJJJMMTT
CWAPJ4MT        DC    PL5'Ø'          DATUM ØJJJJMMTTC
CWACMJ          DC    CL4' '          DATUM MMJJ
CWAPMJ          DC    PL3'Ø'          DATUM ØMMJJC
CWACJM          DC    CL4' '          DATUM JJMM
CWAPJM          DC    PL3'Ø'          DATUM ØJJMMC
CWACMJ4         DC    CL6' '          DATUM MMJJJJ
CWAPMJ4         DC    PL4'Ø'          DATUM ØMMJJJJC
CWACJ4M         DC    CL6' '          DATUM JJJJMM
CWAPJ4M         DC    PL4'Ø'          DATUM ØJJJJMMC
CWACT3J         DC    CL5' '          DATUM TTTJJ
CWAPT3J         DC    PL3'Ø'          DATUM TTTJJC
CWACJT3         DC    CL5' '          DATUM JJTTT
CWAPJT3         DC    PL3'Ø'          DATUM JJTTTC
CWACT3J4        DC    CL7' '          DATUM TTTJJJJ
CWAPT3J4        DC    PL4'Ø'          DATUM TTTJJJJC
CWACJ4T3        DC    CL7' '          DATUM JJJJTTT
CWAPJ4T3        DC    PL4'Ø'          DATUM JJJJTTTC
CWAZEIT         DC    CL5' '          UHRZEIT  SS:MM
*                                     ______________
CWATABLE        DS   ØCL24            ,+Ø123456789-,Ø123456789
*                                     ______________
CWATAB1         DS   ØCL13            TABELLE 1 /,/+/Ø123456789/-/
CWATAB2         DS   ØCL12            TABELLE 2 /,/+/Ø123456789/
CWACHKO1        DC    C','
CWATAB3         DS   ØCL12            TABELLE 3 /+/Ø123456789/-/
CWATAB4         DS   ØCL11            TABELLE 4 /+/Ø123456789/
CWACHARP        DC    C'+'
CWATAB5         DS   ØCL12            TABELLE 5 /Ø123456789/-/,/
CWATAB6         DS   ØCL11            TABELLE 6 /Ø123456789/-/
CWATAB7         DS   ØCL1Ø            TABELLE 7 /Ø123456789/
CWACHØ9         DC    C'Ø123456789'
CWACHARM        DC    C'-'
CWATAB8         DS   ØCL11            TABELLE 8 /,/Ø123456789/
CWACHKO2        DC    C','
CWACHØ92        DC    C'Ø123456789'
CWAZEITP        DC    PL4'Ø'          UHRZEIT  HHMMSSTC
CWADAY          DC    CL1Ø' '         WOCHENTAG
CWAMONTH        DC    CL9' '          MONAT
CWA_PTR_FTT     DC    AL4(Ø)          ADRESSE D. FUNKTIONSTASTENTABELLE
CWA_PTR_ANT     DC    AL4(Ø)          ADRESSE DER AKTIONSNAMENTABELLE
CWA_INFOCICS    DC    C' '            INFO-CICS IDENTIFIER
CWA_INFOCICS_Y  EQU   C'Y'            INFO-CICS IDENTIFIER -JA-
CWA_INFOCICS_N  EQU   C' '            INFO-CICS IDENTIFIER -NEIN-
CWA_DATUM_JJJJ  DC    CL1Ø' '         DATUM FORMAT TT.MM.JJJJ
                SPACE 1
```

```
CWAAREAE        EQU   *              ENDE CWA DEFINITIONEN
                SPACE 5
*---------------------------------------------------------------
*         END  OF  CICS   CWA_AREA                              *
*---------------------------------------------------------------
*
*---------------------------------------------------------------
*         START OF DSECT FOR FUNKTIONSTASTENTABELLE             *
*                                                               *
*         ADRESSING OVER  "CWA_PTR_FTT"                         *
*---------------------------------------------------------------
*
CWAFTTDSECT      DSECT
CWA_FTT_TASTE    DC    XL1'Ø'         TASTENIDENTIFIKATION
CWA_FTT_AKTION   DC    CL16' '        KURZBEZEICHNUNG DER TASTE
*                                     BSP. : HILFE
CWA_FTT_ANZEIGE  DC    CL2Ø' '               TEXT FUER DEN FUNKTIONS-
*                                            TASTENBLOCK EINES BILDES
*                                     BSP. : F1=HILFE
CWA_FTT_PFKEY    DC    CL4' '         PF-TASTE Z.B. "PF1 "
CWA_FTT_KURZTEXT DC    CL8' '         TASTENKUERZEL FUER POP-UP-MENUS
*                                     BSP. : F12=ABBR
CWA_FTT_TEXT     DS    CL2Ø7          BESCHREIBUNG DER AKTION
CWAFTTDSECTE     EQU   *
CWAFTTANZAHL     EQU   3Ø             ANZAHL TABELLENEINTRAEGE FTT
                 SPACE 2
*---------------------------------------------------------------*
*         END OF THE DSECT FOR FUNKTIONSTASTENTABELLE            *
*---------------------------------------------------------------*
*---------------------------------------------------------------*
*         START OF DSECT FOR AKTIONSNAMENTABELLE                 *
*                                                               *
*         ADRESSING OVER  "CWA_PTR_ANT"                         *
*---------------------------------------------------------------*
CWAANTDSECT DSECT
*
CWA_ANT_HILFE         DS CL16            HELP TEXT
CWA_ANT_TASTEN        DS CL16            SHOW THE TASTENBELEGUNG
CWA_ANT_AUSGANG       DS CL16            COMPLETE A FUNCTION
CWA_ANT_REFRESH       DS CL16            RESTORE
CWA_ANT_UPDATE        DS CL16            DATE STORE
CWA_ANT_RUECKWAERTS   DS CL16            BACKWARDS BROWSE
CWA_ANT_VORWAERTS     DS CL16            FORWARDS BROWSE
CWA_ANT_AKTION        DS CL16            ACTIVATE ACTION BAR
CWA_ANT_UNTERBRECHEN  DS CL16            EVENTS UNDER VIEW
CWA_ANT_ABBRUCH       DS CL16            ABORT
CWA_ANT_EINSTIEG      DS CL16            BACK TO EINSTIEGSBILD
CWA_ANT_AUSWAHL       DS CL16            BACK TO AUSWAHLBILD
CWA_ANT_SICHERN       DS CL16            FREEZE THE DTA
CWA_ANT_LINKS         DS CL16            LEFT-SIDE PAGES
```

```
CWA_ANT_RECHTS        DS CL16             RIGHT-SIDE PAGE
CWA_ANT_ANFANG        DS CL16             SHOW THE FIRST SIDE
CWA_ANT_SCHLUSS       DS CL16             SHOW THE OTHER SIDE
CWA_ANT_ABMELDEN      DS CL16             ZSS-ABMELDUNG
CWA_ANT_DRUCKEN       DS CL16             PRINT (PA2)
CWA_ANT_LOESCHEN      DS CL16             OUTPUT TO SCREEN
CWA_ANT_DATENFREIGABE DS CL16             DATENFREIGABE
CWA_ANT_HILFE_ANLEGEN DS CL16             BOSHELP HELP START
CWA_ANT_SUCHEN        DS CL16             SEARCH
CWA_ANT_EURODM        DS CL16             CONVERT EURO/DM
CWAANTDSECTE          EQU   *
*-------------------------------------------------------------------*
*         END OF THE DSECT FOR AKTIONSNAMENTABELLE                  *
*-------------------------------------------------------------------*
*      END INCLUDE++
         SPACE
DFHEISTG DSECT
NBR      DS    F                  NBR OF TASKS
RESP     DS    F                  RESPONSE-CODE
LENGTH   DS    H                  OUTPUT LENGTH
PROGRAM  DS    CL8                RELOADED PROGRAMS
FILE     DS    CL8                FILES TO OPEN OR CLOSE
TRAN     DS    CL4                TRANSACTION NAME
TASKNO   DS    F                  TRANSACTIONS TO PURGE
ABCODE   DS    CL4                ABEND CODE
INAREA   DS    CL4                COMMAREA INPUT
OUTAREA  DS    CL5Ø               TERMINAL-OUTPUT-AREA
         EJECT
* ********************************************************************* *
*        M a i n  -  p r o g r a m                                      *
* ********************************************************************* *
         SPACE
CSRELOAD DFHEIENT CODEREG=(R3,R4),DATAREG=R12
*
CSRELOAD AMODE 31
CSRELOAD RMODE ANY
*
         EXEC  CICS ADDRESS    CWA         (CWAPTR)
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BE    CSREØ5ØØ
         MVC   OUTAREA,MSGØØØ          No access to the CWA
         BAS   R7,SENDMSG
         MVC   ABCODE,=C'CWAA'         CWA can't be assigned
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
CSREØ5ØØ EQU   *
         CLI   CWACICID,CWA$PROD       Is it PROD-CICS?
```

```
         BE    CSRE0700               YES: Check whether it's a
*                                          TOR/FOR or an AOR
         CLI   CWACICID,CWA$SYST      Is it SYSTEM-CICS?
         BE    CSRE0700               YES: Check whether it's a
*                                          TOR/FOR or an AOR
         CLI   CWACICID,CWA$TEST      Is it TEST-CICS?
         BE    CSRE0700               YES: Check whether it's a
*                                          TOR/FOR or an AOR
         CLI   CWACICID,CWA$VPRD      Is it VPRD-CICS?
         BE    CSRE0700               YES: Check whether it's a
*                                          TOR/FOR or an AOR
         B     CSRE0550               ERROR!
         SPACE
CSRE0550 EQU   *
         MVC   OUTAREA,MSG001         Invalid MRO - affiliation
         BAS   R7,SENDMSG
         B     ERRORWA
         SPACE
CSRE0700 EQU   *
         CLI   CWACICNR,CWA$TERM      Is it a TOR?
         BE    CSRE0800               YES: Invalid CICS
         CLI   CWACICNR,CWA$VSAM      Is it a FOR?
         BE    CSRE0800               YES: Invalid CICS
         CLI   CWACICNR,CWA$PAIS      Is it PAISY?
         BE    CSRE0800               YES: Invalid CICS
         CLI   CWACICNR,CWA$INFO      Is it a "24 hrs CICS"?
         BNE   CSRE0850               No: Bypass reload
         BAS   R6,DELAY               Delay processing for a second
         BAS   R6,RELOADPG
         BAS   R6,DELAY               Delay processing for a second
         B     RETURN
         SPACE
CSRE0800 EQU   *
         MVC   OUTAREA,MSG002         Function not allowed
         BAS   R7,SENDMSG
         B     RETURN                 Go back
         SPACE
CSRE0850 EQU   *
         MVC   OUTAREA,MSG003         Bypass reload
         BAS   R7,SENDMSG
         B     RETURN                 Go back
         SPACE
RETURN   EQU   *                      Terminate program
*
         EXEC  CICS RETURN
*
         EJECT
* ********************************************************************* *
*        S u b r o u t i n e s                                          *
* ********************************************************************* *
```

```
         SPACE
* ********************************************************************* *
*        U s e r   p r o g r a m s  w i l l  b e  r e l o a d e d     *
* ********************************************************************* *
         SPACE
RELOADPG EQU   *
         MVC   OUTAREA,MSGØ11
         BAS   R7,SENDMSG
*
         EXEC  CICS INQUIRE PROGRAM START AT('CI')                      *
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BE    RELOAØØØ
         MVC   ABCODE,=C'STAR'
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
RELOAØØØ EQU   *
*
         EXEC  CICS INQUIRE PROGRAM(PROGRAM) NEXT                       *
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BE    RELOA5ØØ
         CLC   RESP,DFHRESP(END)
         BE    RELOA6ØØ
         MVC   ABCODE,=C'NEXT'
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
RELOA5ØØ EQU   *
         CLC   PROGRAM(2),=C'CI'
         BNE   RELOA6ØØ
*
         EXEC  CICS SET PROGRAM(PROGRAM) PHASEIN                        *
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BNE   RELOA55Ø
         MVC   OUTAREA,MSGØ16                Move message
         MVC   OUTAREA+2Ø(L'PROGRAM),PROGRAM Move program - name
         BAS   R7,SENDMSG
         B     RELOAØØØ
         SPACE
RELOA55Ø EQU   *
         MVC   OUTAREA,MSGØ17                Move message
         MVC   OUTAREA+2Ø(L'PROGRAM),PROGRAM Move program - name
         BAS   R7,SENDMSG
         B     RELOAØØØ
```

```
         SPACE
RELOA6ØØ EQU   *
*
         EXEC  CICS INQUIRE PROGRAM END                              *
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BE    RELOA9ØØ
         MVC   ABCODE,=C'ENDE'
         BAS   R6,ABEND
         B     ERRORWA
         SPACE
RELOA9ØØ EQU   *
         MVC   OUTAREA,MSGØ12
         BAS   R7,SENDMSG
         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'5Ø'
*
         EXEC  CICS WRITE OPERATOR TEXT(OUTAREA)                      *
               RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BNE   ERRORWA
         BR    R7
         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    DS    ØH
*
         EXEC  CICS ABEND ABCODE(ABCODE)
*
         BR    R6
         SPACE
* ********************************************************************* *
```

```
*        D e l a y  p r o c e s s i n g  f o r  o n e  s e c o n d    *
* ********************************************************************* *
         SPACE
DELAY    DS    ØH
*
         EXEC  CICS DELAY INTERVAL(1)
*
         BR    R6
         EJECT
* ********************************************************************* *
*        R e g i s t e r  e q u a t e s  a n d  c o n s t a n t s    *
* ********************************************************************* *
         SPACE
         EQUREG
CWAPTR   EQU   R8
         SPACE
MSGØØØ   DC    CL5Ø'CSRELOAD-ØØ The CWA cannot be addressed !         '
MSGØØ1   DC    CL5Ø'CSRELOAD-Ø1 Invalid MRO - affiliation !           '
MSGØØ2   DC    CL5Ø'CSRELOAD-Ø2 Function not allowed in this CICS !   '
MSGØØ3   DC    CL5Ø'CSRELOAD-Ø3 Reload bypassed in this CICS !        '
*SGØØ4   DC    CL5Ø'CSRELOAD-Ø4                                       '
*SGØØ5   DC    CL5Ø'CSRELOAD-Ø5                                       '
*SGØØ6   DC    CL5Ø'CSRELOAD-Ø6                                       '
*SGØØ7   DC    CL5Ø'CSRELOAD-Ø7                                       '
*SGØØ8   DC    CL5Ø'CSRELOAD-Ø8                                       '
*SGØØ9   DC    CL5Ø'CSRELOAD-Ø9                                       '
*SGØ1Ø   DC    CL5Ø'CSRELOAD-1Ø                                       '
MSGØ11   DC    CL5Ø'CSRELOAD-11 User - programs will be reloaded      '
MSGØ12   DC    CL5Ø'CSRELOAD-12 User - programs successfully reloaded '
*SGØ13   DC    CL5Ø'CSRELOAD-13                                       '
*SGØ14   DC    CL5Ø'CSRELOAD-14                                       '
*SGØ15   DC    CL5Ø'CSRELOAD-15                                       '
MSGØ16   DC    CL5Ø'CSRELOAD-16 Program xxxxxxxx was reloaded         '
MSGØ17   DC    CL5Ø'CSRELOAD-17 Program xxxxxxxx can not be reloaded  '
*SGØ18   DC    CL5Ø'CSRELOAD-18                                       '
*SGØ19   DC    CL5Ø'CSRELOAD-19                                       '
         EJECT
* ********************************************************************* *
*        L i t e r a l s                                                *
* ********************************************************************* *
         SPACE
         LTORG
         SPACE
         DC     C' '
         END    CSRELOAD
```

Claus Reis
CICS Systems Programmer
Nuernberger Lebensversicherung AG (Germany)

Managing affinities on specific terminals

In a CICSPLEX environment, IBM said that CEDF cannot support dual screen, ie CEDX must be used instead of CEDF. However, at our site, all branch transactions are called TTOH. So CEDX is not a solution for our environment. We modified EYULWRAM to create EYU9WRAM, which is used for dynamic routing. If the TTOH transaction coming from the branch has an affinity, then EYU9WRAM passes it to an AOR without changing the transaction name. If it doesn't, the transaction name in the AOR would be the function code of TTOH.

As a second step, we wrote a COBOL program, CPSMCREA, to create an affinity on the specific LUname. If the authorized person in the AOR CICS runs this program via AFFC netname <ENTER>, then affinity WLDCEDF*n* (where *n* means the sequence number of the AOR) is automatically created. Now the user should use CEDX TTOH to debug the end-user operation on the specific LUname. After the operation is finished, authorized people use AFFD (program CPSMDELA) to delete the affinity on the LUname automatically.

We wrote another COBOL program, CPSMINQA, to inquire whether there is an active affinity on all the AORs or not.

I believe these programs will be helpful for everybody in a CICSplex environment to manage affinities on specific terminals.

CPSMCREA

```
ENVIRONMENT DIVISION.
DATA DIVISION.
    EJECT
WORKING-STORAGE SECTION.
Ø1 W-CONTEXT      PIC X(8)  VALUE SPACES.
Ø1 W-SCOPE        PIC X(8)  VALUE SPACES.
Ø1 W-THREAD       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-RESULT       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-RESPONSE     PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-REASON       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-BUFFER       PIC X(32767).
Ø1 W-BUFFERLEN    PIC S9(8) COMP.
Ø1 LINE-CNT       PIC 9(4)  COMP.
```

```
Ø1 W-TEXT         PIC X(4Ø).
Ø1 W-MSG-TEXT.
  Ø2 W-TEXT-BDY   PIC X(8Ø) VALUE SPACES.
  Ø2 W-LINECTL    PIC X(1) VALUE X'13'.
Ø1 PICZZZ9A       PIC ZZZ9.
Ø1 PICZZZ9B       PIC ZZZ9.
Ø1 W-INTO-OBJLEN  PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-CRITERIA     PIC X(8Ø) VALUE SPACES.
Ø1 W-CRITERIALEN  PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 DEGISKEN.
   Ø2 DEGIS1      PIC X(12) VALUE 'NAME=WLDCEDF'.
   Ø2 DEGIS2      PIC X(1).
   Ø2 DEGIS3      PIC X(3) VALUE '.'.
Ø1 ISIMDEG.
   Ø2 DEGISI      PIC X(7) VALUE 'WLDCEDF'.
   Ø2 DEGISN      PIC X(1).
Ø1 INPUTPRM.
  Ø2 TRANID       PIC X(4).
  Ø2 ARAB         PIC X(1).
  Ø2 NETIDA       PIC X(8).
Ø1 BOY            PIC 9(4)  COMP.
Ø1 CICS-ADI.
   Ø2 CICSPR      PIC X(4) VALUE 'CICS'.
   Ø2 ORTAM       PIC X(1).
   Ø2 PROJE       PIC X(1).
   Ø2 FONKS       PIC X(1).
   Ø2 SEQNUM      PIC X(1).
Ø1 NETADI.
   Ø2 NOKTA        PIC X(1)  VALUE '.'.
   Ø2 LUISMI       PIC X(8).
Ø1 W-PARM.
   Ø2 WORKLA      PIC X(9) VALUE 'WORKLOAD('.
   Ø2 WORKLOADI   PIC X(8).
   Ø2 WORKLOKA    PIC X(11) VALUE ') OWNER(*).'.
Ø1 W-PARMLEN      PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 TRAN1          PIC X(4) VALUE 'CEDX'.
Ø1 TRAN2          PIC X(4) VALUE 'TTOH'.
COPY WLMDEF.
PROCEDURE DIVISION.
MAIN-PROGRAM.
   EXEC CICS HANDLE CONDITION
                  LENGERR(HATA-YAZ)
   END-EXEC.
   MOVE 13 TO BOY.
   EXEC CICS RECEIVE INTO(INPUTPRM) LENGTH(BOY) END-EXEC.
   IF  ARAB NOT = ' ' OR BOY < 9 THEN
        GO TO  HATA-YAZ.
   MOVE NETIDA TO LUISMI.
   EXEC CICS ASSIGN APPLID(CICS-ADI) END-EXEC.
```

```
IF PROJE = 'S' THEN
 IF ORTAM = 'D' THEN MOVE 'DEVLSUBE' TO W-CONTEXT
 END-IF
 IF ORTAM = 'S' THEN MOVE 'SISPSUBE' TO W-CONTEXT
 END-IF
 IF ORTAM = 'T' THEN MOVE 'TESTSUBE' TO W-CONTEXT
END-IF
MOVE W-CONTEXT TO W-SCOPE.
 EXEC CPSM CONNECT CONTEXT(W-CONTEXT)
      SCOPE(W-SCOPE)
      VERSION('Ø14Ø')
      THREAD(W-THREAD)
      RESPONSE(W-RESPONSE)
      REASON(W-REASON)
 END-EXEC.
 IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-BAGLAN.
 MOVE SEQNUM TO DEGISN.
 MOVE SEQNUM TO DEGIS2.
 MOVE ISIMDEG  TO NAME-R OF WLMDEF.
 MOVE NETADI  TO  LUNAME OF WLMDEF.
 MOVE '*'     TO  USERID OF WLMDEF.
 MOVE 'TRANCEDF' TO TRANGRP-A OF WLMDEF.
 MOVE CICS-ADI  TO AORSCOPE OF WLMDEF.
 MOVE '*'   TO PROCESSTYPE OF WLMDEF.
 MOVE 'CEDF TRANS' TO DESC OF WLMDEF.
 MOVE WLMDEF TO W-BUFFER.
 MOVE W-BUFFER TO W-TEXT.
 MOVE WLMDEF-TBL-LEN TO W-BUFFERLEN.
 EXEC CPSM CREATE
           OBJECT('WLMDEF')
           FROM(W-BUFFER)
           LENGTH(W-BUFFERLEN)
           THREAD(W-THREAD)
           RESPONSE(W-RESPONSE)
           REASON(W-REASON)
 END-EXEC.
 IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CREATE.
 MOVE W-SCOPE  TO WORKLOADI.
 MOVE LENGTH OF W-PARM TO W-PARMLEN.
 MOVE DEGISKEN TO W-CRITERIA.
 MOVE LENGTH OF W-CRITERIA TO W-CRITERIALEN.
 MOVE W-PARM    TO W-TEXT.
 ADD   1         TO LINE-CNT.
 EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(3Ø)
           JUSTIFY(LINE-CNT) WAIT  END-EXEC.
 MOVE W-CRITERIA TO W-TEXT.
 ADD   1         TO LINE-CNT.
 EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(3Ø)
           JUSTIFY(LINE-CNT) WAIT  END-EXEC.
```

```
    EXEC CPSM PERFORM OBJECT('WLMDEF')
              ACTION('INSTALL')
              CRITERIA(W-CRITERIA)
              LENGTH(W-CRITERIALEN)
              PARM(W-PARM)
              PARMLEN(W-PARMLEN)
              RESULT(W-RESULT)
              THREAD(W-THREAD)
              RESPONSE(W-RESPONSE)
              REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-INSTALL.
    GO TO CIKIS-SON.
HATA-BAGLAN.
    MOVE 'HATALI BAGLANTI' TO W-TEXT.
    EXEC CICS SEND FROM(W-TEXT) LENGTH(15) ERASE END-EXEC.
    GO TO CIKIS-DON.
HATA-YAZ.
    MOVE 'HATALI INPUT' TO W-TEXT.
    EXEC CICS SEND FROM(W-TEXT) LENGTH(12) ERASE END-EXEC.
    GO TO CIKIS-DON.
HATA-CREATE.
    MOVE 'BU CICS ORTAMINDA AKTIF CEDX VAR' TO W-TEXT.
    EXEC CICS SEND FROM(W-TEXT) LENGTH(32) ERASE END-EXEC.
    GO TO CIKIS-DON.
HATA-CREATE.
    EVALUATE  W-RESPONSE
       WHEN  EYUVALUE(NOTPERMIT)
             MOVE 'YETK~ HATASI.'
                  TO W-TEXT
             EXEC CICS SEND FROM(W-TEXT) LENGTH(32) ERASE
                  END-EXEC
       WHEN  EYUVALUE(TABLEERROR)
             MOVE 'BU CICS ORTAMINDA AKTIF CEDX VAR.'
                  TO W-TEXT
             EXEC CICS SEND FROM(W-TEXT) LENGTH(32) ERASE
                  END-EXEC
       WHEN  OTHER
             MOVE 'WORKLOAD TANIMI YAPILAMADI.'
                  TO W-TEXT
             EXEC CICS SEND FROM(W-TEXT) LENGTH(32) ERASE
                  END-EXEC
    END-EVALUATE.
    GO TO CIKIS-DON.
HATA-INSTALL.
    MOVE 'TANIMINIZ INSTALL EDILMEDI' TO W-TEXT.
    EXEC CICS SEND FROM(W-TEXT) LENGTH(26) ERASE END-EXEC.
    GO TO CIKIS-DON.
CIKIS-SON.
```

```
    MOVE 'TANIMINIZ BASARILI YAPILDI' TO W-TEXT.
    EXEC CICS SEND FROM(W-TEXT) LENGTH(26) ERASE END-EXEC.
CIKIS-DON.
    EXEC CICS RETURN END-EXEC.
    GOBACK.
```

CPSMDELA

```
IDENTIFICATION DIVISION.
PROGRAM-ID.  CPSMDELA.
ENVIRONMENT DIVISION.
DATA DIVISION.
    EJECT
WORKING-STORAGE SECTION.
Ø1 W-CONTEXT      PIC X(8)  VALUE SPACES.
Ø1 W-SCOPE        PIC X(8)  VALUE SPACES.
Ø1 W-THREAD       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-RESULT       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-RESPONSE     PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-REASON       PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-BUFFER       PIC X(32767).
Ø1 W-BUFFERLE     PIC S9(8) COMP.
Ø1 LINE-CNT       PIC 9(4)  COMP.
Ø1 W-TEXT         PIC X(3Ø).
Ø1 W-MSG-TEXT.
  Ø2 W-TEXT-BDY   PIC X(8Ø) VALUE SPACES.
  Ø2 W-LINECTL    PIC X(1) VALUE X'13'.
Ø1 PICZZZ9A       PIC ZZZ9.
Ø1 PICZZZ9B       PIC ZZZ9.
Ø1 W-INTO-OBJLEN  PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 W-CRITERIA     PIC X(8Ø) VALUE SPACES.
Ø1 W-CRITERIALEN  PIC S9(8) USAGE BINARY VALUE ZERO.
Ø1 DEGISKEN.
   Ø2 DEGIS1      PIC X(12) VALUE 'NAME=WLDCEDF'.
   Ø2 DEGIS2      PIC X(1).
   Ø2 DEGIS3      PIC X(3) VALUE '.'.
Ø1 INPUTPRM.
  Ø2 TRANID       PIC X(4).
  Ø2 ARAB         PIC X(1).
  Ø2 NETIDA       PIC X(8).
Ø1 BOY            PIC 9(4)  COMP.
Ø1 CICS-ADI.
   Ø2 CICSPR      PIC X(4) VALUE 'CICS'.
   Ø2 ORTAM       PIC X(1).
   Ø2 PROJE       PIC X(1).
   Ø2 FONKS       PIC X(1).
   Ø2 SEQNUM      PIC X(1).
Ø1 NETADI.
```

```
   Ø2 NOKTA         PIC X(1)  VALUE '.'.
   Ø2 LUISMI        PIC X(8).
Ø1 W-PARM.
   Ø2 WORKLA      PIC X(9) VALUE 'WORKLOAD('.
   Ø2 WORKLOADI   PIC X(8).
   Ø2 WORKLOKA    PIC X(2) VALUE ').'.
Ø1 W-PARMLEN      PIC S9(8) USAGE BINARY VALUE ZERO.
COPY WLMDEF.
COPY WLMGROUP.
PROCEDURE DIVISION.
MAIN-PROGRAM.
   EXEC CICS ASSIGN APPLID(CICS-ADI) END-EXEC.
   IF PROJE = 'S' THEN
    IF ORTAM = 'D' THEN MOVE 'DEVLSUBE' TO W-CONTEXT
    END-IF
    IF ORTAM = 'S' THEN MOVE 'SISPSUBE' TO W-CONTEXT
    END-IF
    IF ORTAM = 'T' THEN MOVE 'TESTSUBE' TO W-CONTEXT
   END-IF
    MOVE W-CONTEXT TO W-SCOPE.
    EXEC CPSM CONNECT CONTEXT(W-CONTEXT)
         SCOPE(W-SCOPE)
         VERSION('Ø14Ø')
         THREAD(W-THREAD)
         RESPONSE(W-RESPONSE)
         REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CIKIS.
    MOVE SEQNUM    TO DEGIS2.
    MOVE DEGISKEN TO W-CRITERIA.
    MOVE LENGTH OF W-CRITERIA TO W-CRITERIALEN.
    EXEC CPSM GET OBJECT('WLMDEF')
         CRITERIA(W-CRITERIA)
         LENGTH(W-CRITERIALEN)
         RESULT(W-RESULT)
         THREAD(W-THREAD)
         RESPONSE(W-RESPONSE)
         REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CIKIS.
    MOVE LENGTH OF WLMDEF TO W-INTO-OBJLEN.
    EXEC CPSM FETCH INTO(WLMDEF)
         LENGTH(W-INTO-OBJLEN)
         RESULT(W-RESULT)
         THREAD(W-THREAD)
         RESPONSE(W-RESPONSE)
         REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CIKIS.
```

```
    EXEC CPSM REMOVE OBJECT('WLMDEF')
         FROM(WLMDEF)
         LENGTH(W-INTO-OBJLEN)
         THREAD(W-THREAD)
         RESPONSE(W-RESPONSE)
         REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CIKIS.
    MOVE W-SCOPE TO WORKLOADI.
    MOVE LENGTH OF W-PARM TO W-PARMLEN.
    EXEC CPSM PERFORM OBJECT('WLMAWDEF')
              ACTION('DISCARD')
              PARM(W-PARM)
              PARMLEN(W-PARMLEN)
              CRITERIA(W-CRITERIA)
              LENGTH(W-CRITERIALEN)
              RESULT(W-RESULT)
              THREAD(W-THREAD)
              RESPONSE(W-RESPONSE)
              REASON(W-REASON)
    END-EXEC.
    IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO HATA-CIKIS.
CIKIS.
    MOVE 'ISLEMINIZ BASARILI SONLANMISTIR' TO W-MSG-TEXT.
    GO TO SON.
HATA-CIKIS.
    MOVE 'ISLEMINIZ HATALI SONLANMISTIR' TO W-MSG-TEXT.
    GO TO SON.
SON.
    EXEC CICS SEND FROM(W-MSG-TEXT) ERASE LENGTH(81) END-EXEC.
    EXEC CICS RETURN   END-EXEC.
    GOBACK.
```

CPSMINQA

```
*****************************************************************
*
* BU PROGRAM, CICSPLEX
*
*****************************************************************
 IDENTIFICATION DIVISION.
 PROGRAM-ID.  CPSMWLD.
 ENVIRONMENT DIVISION.
 DATA DIVISION.
     EJECT
 WORKING-STORAGE SECTION.
*Ø1 W-CONTEXT      PIC X(8)  VALUE SPACES.
 Ø1 W-CONTEXT.
```

```
    Ø5 W-ENV       PIC X(4)  VALUE SPACES.
    Ø5 W-PRJ       PIC X(4)  VALUE SPACES.
 Ø1 W-SCOPE        PIC X(8)  VALUE SPACES.
 Ø1 W-WLDNAME      PIC X(8)  VALUE SPACES.
 Ø1 W-THREAD       PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-RESULT       PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-RESPONSE     PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-REASON       PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-RECCNT       PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-CRITERIA     PIC X(8Ø) VALUE SPACES.
 Ø1 W-CRITERIALEN  PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-INTO-OBJLEN  PIC S9(8) USAGE BINARY VALUE ZERO.
 Ø1 W-CMASNAME     PIC X(8)  VALUE SPACES.
 Ø1 W-TEXT         PIC X(6Ø).
 Ø1 W-TEXTLEN      PIC S9(4) USAGE BINARY VALUE ZERO.
 Ø1 LINE-CNT       PIC 9(4)  COMP.
 Ø1 CICS-ID.
    Ø5 C-PREFIX    PIC X(4)  VALUE SPACES.
    Ø5 C-ENV-ID    PIC X(1)  VALUE SPACES.
    Ø5 C-PRJ-ID    PIC X(1)  VALUE SPACES.
    Ø5 C-SUBJ-ID   PIC X(1)  VALUE SPACES.
    Ø5 C-SEQ-NO    PIC X(1)  VALUE SPACES.
 Ø1 W-WLD-OUTLEN   PIC S9(4) USAGE BINARY VALUE ZERO.
 Ø1 W-WLD-OUT.
    Ø5 REC-HEAD    PIC X(6)  VALUE '****> '.
    Ø5 W-WLD-NAME  PIC X(8)  VALUE SPACES.
    Ø5 FILLER      PIC X(4)  VALUE SPACES.
    Ø5 W-CICSID    PIC X(8)  VALUE SPACES.
    Ø5 FILLER      PIC X(3)  VALUE SPACES.
    Ø5 W-LUNAME.
       1Ø W-LUNAME1   PIC X(1) VALUE SPACES.
       1Ø W-LUNAME2   PIC X(16) VALUE SPACES.
    Ø5 REC-TRAIL   PIC X(6)  VALUE ' <****'.
 Ø1 III            PIC S9(8) VALUE ZERO.
 Ø1 CPSM-ERROR.
    Ø5 FILLER       PIC X(12) VALUE 'CPSM HATA : '.
    Ø5 CPSM-ERROR-RESPONSE   PIC Z(Ø4)9.
 Ø1 ACQSTATUS      PIC S9(ØØØ8) USAGE BINARY.
 Ø1 SERVSTATUS     PIC S9(ØØØ8) USAGE BINARY.
 COPY WLMAWDEF.
 PROCEDURE DIVISION.
 MAIN-PROGRAM.
*    MOVE 'CMASTC9Ø' TO W-CMASNAME.
*    MOVE 'DEVLSUBE' TO W-CONTEXT.
*    MOVE 'DEVLSUBE' TO W-SCOPE.
     MOVE 'Ø'        TO LINE-CNT.
     MOVE LENGTH OF W-TEXT    TO W-TEXTLEN.
     MOVE LENGTH OF W-WLD-OUT TO W-WLD-OUTLEN.
     EXEC CICS INQUIRE SYSTEM
```

```
               JOBNAME(CICS-ID) END-EXEC.
*??????????????
*    MOVE 'CICSDSA1' TO CICS-ID.
     EVALUATE  C-ENV-ID
        WHEN  'D' MOVE 'DEVL' TO W-ENV
        WHEN  'T' MOVE 'TEST' TO W-ENV
        WHEN  'P' MOVE 'PROD' TO W-ENV
        WHEN  OTHER GO TO CIKIS
     END-EVALUATE.
     EVALUATE  C-PRJ-ID
        WHEN  'S' MOVE 'SUBE' TO W-PRJ
        WHEN  OTHER GO TO CIKIS
     END-EVALUATE.
     MOVE W-CONTEXT  TO W-SCOPE.
     MOVE W-CONTEXT  TO W-WLDNAME.
*********************************************
*    MOVE 'CPSM BALANTISI KURULUYOR ... ' TO W-TEXT.
*    ADD   1         TO LINE-CNT.
*    EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*              JUSTIFY(LINE-CNT) ERASE END-EXEC.
     EXEC CPSM CONNECT CONTEXT(W-CONTEXT)
          SCOPE(W-SCOPE)
          VERSION('Ø14Ø')
          THREAD(W-THREAD)
          RESPONSE(W-RESPONSE)
          REASON(W-REASON)
     END-EXEC.
     IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO ERROR-CONNECT.
*    MOVE 'CPSM BALANTISI KURULDU   ... ' TO W-TEXT.
*    ADD   1         TO LINE-CNT.
*    EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*              JUSTIFY(LINE-CNT) WAIT  END-EXEC.
*********************************************
*    MOVE 'WORKLOAD TABLOSUNA ERIIM ... ' TO W-TEXT.
*    ADD   1         TO LINE-CNT.
*    EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*              JUSTIFY(LINE-CNT) WAIT  END-EXEC.
*    MOVE 'WORKLOAD=DEVLSUBE AND NAME=WLDCEDF+.'
*         TO W-CRITERIA.
     STRING 'WORKLOAD=' DELIMITED BY SIZE
             W-WLDNAME  DELIMITED BY SIZE
             ' AND NAME=WLDCEDF+.' DELIMITED BY SIZE
             INTO W-CRITERIA.
     MOVE LENGTH OF W-CRITERIA TO W-CRITERIALEN.
     EXEC CPSM GET OBJECT('WLMAWDEF')
          CRITERIA(W-CRITERIA)
          LENGTH(W-CRITERIALEN)
          COUNT(W-RECCNT)
          RESULT(W-RESULT)
```

```
           THREAD(W-THREAD)
           RESPONSE(W-RESPONSE)
           REASON(W-REASON)
      END-EXEC.
      IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO ERROR-GET.
*     MOVE 'WORKLOAD TABLOSUNA ERIILDI.. ' TO W-TEXT.
*     ADD   1         TO LINE-CNT.
*     EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
*********************************************
      MOVE '         AKTF WORKLOAD TANIMLARI'    TO W-TEXT.
      ADD   1         TO LINE-CNT.
      EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
      MOVE '      WLD-NAME    CICS-NAME   LU-NAME' TO W-TEXT.
      ADD   2         TO LINE-CNT.
      EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
      MOVE '      --------    --------   ------' TO W-TEXT.
      ADD   1         TO LINE-CNT.
      EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
*********************************************
      MOVE 1 TO III.
 LOOP-TOP.
      IF III > W-RECCNT GO TO CIKIS.
*     MOVE 'WORKLOAD DATA FETCH       ... ' TO W-TEXT.
*     ADD   1         TO LINE-CNT.
*     EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
      MOVE LENGTH OF WLMAWDEF TO W-INTO-OBJLEN.
      MOVE 12 TO W-CRITERIALEN
      EXEC CPSM FETCH INTO(WLMAWDEF)
           LENGTH(W-INTO-OBJLEN)
           RESULT(W-RESULT)
           THREAD(W-THREAD)
           RESPONSE(W-RESPONSE)
           REASON(W-REASON)
      END-EXEC.
      IF (W-RESPONSE NOT = EYUVALUE(OK)) GO TO ERROR-FETCH.
*     MOVE 'WORKLOAD DATA FETCH EDILDI .. ' TO W-TEXT.
*     ADD   1         TO LINE-CNT.
*     EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
*********************************************
*     MOVE 'DATA DISPLAY              ... ' TO W-TEXT.
*     ADD   1         TO LINE-CNT.
*     EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
*               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
```

```
     MOVE NAME-R       TO W-WLD-NAME.
     MOVE AORSCOPE     TO W-CICSID.
     MOVE LUNAME       TO W-LUNAME.
     MOVE SPACES       TO W-LUNAME1.
     ADD   1           TO LINE-CNT.
     EXEC CICS SEND TEXT FROM(W-WLD-OUT) LENGTH(W-WLD-OUTLEN)
               JUSTIFY(LINE-CNT) WAIT  END-EXEC.
     ADD 1 TO III.
     GO TO LOOP-TOP.
 LOOP-BOT.
*********************************************
 ERROR-CONNECT.
        STRING 'CONTEXT=' DELIMITED BY SIZE
                W-CONTEXT DELIMITED BY SIZE
                '   '  DELIMITED BY SIZE
                'SCOPE='  DELIMITED BY SIZE
                W-SCOPE   DELIMITED BY SIZE
                INTO W-TEXT
        ADD   1         TO LINE-CNT.
        EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                  JUSTIFY(LINE-CNT) WAIT  END-EXEC.
        MOVE 'CPSM BALANTISI KURULAMADI.' TO W-TEXT.
        ADD   1         TO LINE-CNT.
        EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                  JUSTIFY(LINE-CNT) WAIT  END-EXEC.
     GO TO CIKIS.
*********************************************
 ERROR-GET.
     EVALUATE  W-RESPONSE
        WHEN  EYUVALUE(NODATA)
            STRING 'WORKLOAD=' DELIMITED BY SIZE
                    W-WLDNAME  DELIMITED BY SIZE
                    ' ''DE AKTF WORKLOAD BULUNAMADI.'
                                       DELIMITED BY SIZE
                    INTO W-TEXT
*           MOVE 'AKTF WORKLOAD BULUNAMADI.'      TO W-TEXT
            ADD   1         TO LINE-CNT
            EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                      JUSTIFY(LINE-CNT) WAIT  END-EXEC
        WHEN  EYUVALUE(TABLEERROR)
            STRING 'WORKLOAD=' DELIMITED BY SIZE
                    W-WLDNAME  DELIMITED BY SIZE
                    ' ''DE AKTF WORKLOAD BULUNAMADI.'
                                       DELIMITED BY SIZE
                    INTO W-TEXT
*           MOVE 'AKT~F WORKLOAD BULUNAMADI.'      TO W-TEXT
            ADD   1         TO LINE-CNT
            EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                      JUSTIFY(LINE-CNT) WAIT  END-EXEC
```

```
        WHEN  OTHER
            STRING 'WORKLOAD=' DELIMITED BY SIZE
                    W-WLDNAME  DELIMITED BY SIZE
                    ' TABLOSUNA ERIILEMEDI.'
                                       DELIMITED BY SIZE
                    INTO W-TEXT
*           MOVE 'WORKLOAD TABLOSUNA ERIILEMEDI.' TO W-TEXT
            ADD   1         TO LINE-CNT
            EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                      JUSTIFY(LINE-CNT) WAIT  END-EXEC
     END-EVALUATE.
     GO TO CIKIS.
*********************************************
 ERROR-FETCH.
        MOVE 'CPSM FETCH LEMNDE HATA OLUTU.' TO W-TEXT.
        ADD   1         TO LINE-CNT.
        EXEC CICS SEND TEXT FROM(W-TEXT) LENGTH(W-TEXTLEN)
                  JUSTIFY(LINE-CNT) WAIT  END-EXEC.
     GO TO CIKIS.
*********************************************
 CIKIS.
     EXEC CICS
         SEND PAGE
     END-EXEC.
     EXEC CICS RETURN   END-EXEC.
     GOBACK.
```

EYU9WRAM

The following is an IBM program that has been modified. Only the changes and the surrounding code are published here:

```
000100 IDENTIFICATION DIVISION.
000200 PROGRAM-ID. EYULWRAM.
000300*************************************************************
000400*                                                           *
000500* MODULE NAME = EYULWRAM                                    *
000600*                                                           *
000700* DESCRIPTIVE NAME = WORKLOAD ROUTING ACTION MODULE         *
000800*                    (DEFAULT/SAMPLE COBOL VERSION)         *
.
.
.
010400    COPY EYULWSVD.
010500
010600 EJECT
010700
010800* ---------------------------------------------------------- *
```

```
010900* DEFINE Local Variables                                        *
011000* ------------------------------------------------------------ *
AKNET  01  AKNT-TASK-SW-WRK.
AKNET      02 AKNT-TASKNO       PIC S9(7) COMP-3.
AKNET      02 AKNT-TASKNO-TST REDEFINES AKNT-TASKNO.
AKNET         03 FILLER         PIC X(3).
AKNET         03 FILLER         PIC S9(1) COMP-3.
AKNET         88 AKNT-ODD-TASK  VALUE 1 3 5 7 9.
AKNET         88 AKNT-EVEN-TASK VALUE 0 2 4 6 8.
AKNET      02 AKNT-CNT          PIC S9(4) COMP.
AKNET      02 AKNT-SELECTED-AOR.
AKNET         03 FILLER         PIC X(4) VALUE 'CICS'.
AKNET         03 AKNT-SEL-AOR   PIC X(4).
AKNET      02 AKNT-SYS-SYS1     PIC X(4) VALUE 'TSA1'.
AKNET      02 AKNT-SYS-AOR1     PIC X(8) VALUE '        '.
AKNET      02 AKNT-SYS-SYS2     PIC X(4) VALUE 'TSA2'.
AKNET      02 AKNT-SYS-AOR2     PIC X(8) VALUE '        '.
011100 01  TERMINAL-MESSAGE PIC X(160).
011200 01  MESSAGE-LENGTH   PIC 9(2) BINARY.
011300 01  TERM-LINE-POS    PIC 9(3) BINARY.
.
.
.
032500    COPY EYULWSVE REPLACING ==WSVE-SCOPE-VECTOR== BY
032600    ==WSVE-SCOPE-VECTOR OCCURS 1000 TIMES DEPENDING ON
032625      WCOM-SCOP-CNT==.
032650
032700
032800 EJECT
AKNET  01  AKNT-TASK-INP-AREA.
AKNET      02 AKNT-TASK-TTOH    PIC X(4).
AKNET      02 AKNT-TASK-ENCODE  PIC X(1).
AKNET      02 FILLER            PIC X(2).
AKNET      02 AKNT-TASK-FUNC    PIC X(4).
AKNET      02 AKNT-USER-ID      PIC X(8).
032900
033000 PROCEDURE DIVISION.
033100
033200* ------------------------------------------------------------ *
033300* CHECK THAT THE COMMAREA HAS ACTUALLY BEEN PASSED             *
.
.
.
036800* ------------------------------------------------------------ *
036900* Return to the Caller.                                        *
037000* ------------------------------------------------------------ *
AKNET? AKNT-RETURN.
037100     EXEC CICS RETURN
037200     END-EXEC
```

```
037300     GOBACK.
037400
037500 EJECT
047500* ------------------------------------------------------------ *
04990000
047600*                      DO-RTSEL-AOR                            *
047700*                                                              *
047800*  Select an AOR for Route Select                              *
047900*                                                              *
048000* ------------------------------------------------------------ *
05040000
048100 DO-RTSEL-AOR.
048200     MOVE '0' TO LOOP-CONTROL
AKNET      SET ADDRESS OF AKNT-TASK-INP-AREA TO WCOM-INP-BUFF
048300     MOVE WRAM-SM-SCOPE TO CUR-FUNC
048400     CALL 'WAPIENPT' USING WCOM-DA-TOKEN, WCOM-SM-SCOPE
048600     MOVE RETURN-CODE TO API-RETCODE
048700     MOVE WCOM-API-RESP TO API-RESP
048800     MOVE WCOM-API-REASON TO API-REASON
048900     IF RETURN-CODE = 0
049000        PERFORM RESP-PROC
049100        IF WCOM-RET-RESP = WCOM-RET-NORM
049200          IF WCOM-SCOP-CNT > 0
AKNET              IF  WCOM-AFF-STAT NOT =  WCOM-AFF-ACTIVE
AKNET              AND WCOM-SCOP-CNT > 1
AKNET              AND EIBTRNID        = 'TTOH'
AKNET              AND AKNT-TASK-TTOH  = 'TTOH'
AKNET?*                 MOVE AKNT-USER-ID   TO WCOM-USERID
AKNET?*                 MOVE AKNT-TASK-FUNC TO WCOM-TRANSID
AKNET                   MOVE AKNT-TASK-FUNC TO WCOM-REM-TRANID
AKNET              END-IF
049300            PERFORM CHECK-AFF
049400            EVALUATE CHECK-AFF-RC
049500              WHEN CHECK-AFF-RC-BAL
AKNETP*************************************************************
AKNETP* for production environment pls comment out following logic
AKNETP* to allow proper work load balancing (disable switching)
AKNETP*************************************************************
AKNETP                IF WCOM-AFF-STAT NOT = WCOM-AFF-ACTIVE
AKNETP                AND WCOM-SCOP-CNT > 1
AKNETP                   PERFORM AKNT-SWITCHING-ROUTE
AKNETP                END-IF
AKNETP*************************************************************
049600                MOVE WRAM-SM-BALANCE TO CUR-FUNC
049700            CALL 'WAPIENPT' USING WCOM-DA-TOKEN, WCOM-SM-BALANCE
049900                MOVE RETURN-CODE TO API-RETCODE
050000                MOVE WCOM-API-RESP TO API-RESP
050100                MOVE WCOM-API-REASON TO API-REASON
050200                IF RETURN-CODE = 0
```

```
050300                  PERFORM RESP-PROC
050400                ELSE
050500                  MOVE ERR-API TO ERR-INDICATOR
050600                  PERFORM ERR-PROC
050700                END-IF
.
.
.
120500*  Send a Message                                               *
120600* ------------------------------------------------------------ *
120700
120800 WRITE-TRM.
120900     SUBTRACT 1 FROM TERM-LINE-POS GIVING MESSAGE-LENGTH
121000     EXEC CICS SEND
121100               FROM(TERMINAL-MESSAGE)
121200               LENGTH(MESSAGE-LENGTH)
121300               ERASE
121400     END-EXEC.
AKNET * ------------------------------------------------------------ *
AKNET *                    SWITCHED ROUTING                          *
AKNET *                      BY EIBTASKN (even/odd)                  *
AKNET *  To ensure every time selecting different AOR                *
AKNET *  for testing any affinity remained after investigation       *
AKNET * ------------------------------------------------------------ *
AKNET  AKNT-SWITCHING-ROUTE.
AKNET      MOVE EIBTASKN TO AKNT-TASKNO.
AKNET      IF   AKNT-ODD-TASK MOVE AKNT-SYS-SYS1 TO AKNT-SEL-AOR
AKNET      ELSE               MOVE AKNT-SYS-SYS2 TO AKNT-SEL-AOR
AKNET      END-IF.
AKNET      SET WSVE-PTR TO WCOM-SCOP-VECT.
AKNET      SET ADDRESS OF EYURWSVE TO WSVE-PTR.
AKNET        PERFORM WITH TEST BEFORE VARYING AKNT-CNT FROM 1 BY 1
AKNET              UNTIL AKNT-CNT > WCOM-SCOP-CNT
AKNET          IF  WSVE-APPLID (AKNT-CNT) NOT = AKNT-SELECTED-AOR
AKNET              MOVE WSVE-IGNORE-YES TO WSVE-IGNORE(AKNT-CNT)
AKNET          END-IF
AKNET        END-PERFORM.
AKNET  AKNT-SWITCHING-ROUTE-END.
AKNET      EXIT.
```

Nilufer Kaya
Tamer Tezgel
Aknet AS (Turkey)

A CICS template utility – part 2

This month we conclude the code for a set of programs and templates that allow users to view and refresh (reinstall) document templates through a browser.

CICSDOCL

(Note: change HOST to an appropriate value.)

```
         TITLE ' CICSDOCL - LIST DOCUMENTS IN REGION'
DFHEISTG DSECT
*    BROWSES AND LISTS DOCUMENTS
         DS    ØF
TOKEN    DS    CL16
R6       EQU   6
R7       EQU   7
R1Ø      EQU   1Ø
RESP     DS    F
CVRTAREA DS    D
DOCTYPE  DS    F
JOBNAME  DS    CL8
CICSDOCL CSECT
CICSDOCL AMODE 31
CICSDOCL RMODE ANY
START    EQU   *
         EXEC CICS INQUIRE SYSTEM JOBNAME (JOBNAME)
         MVC    JOBNM(8),JOBNAME
*      FIND THE PORT NUMBER FOR THIS REGION
CKTCPS   EQU   *
         EXEC CICS INQUIRE TCPIPSERVICE (TCPNAME)                    -
              PORT (PORTNO) RESP (RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BE    CVRTPORT
CVRTPORT EQU   *
         L     R7,PORTNO
         CVD   R7,CVRTAREA
         OI    CVRTAREA+7,X'ØF'
         UNPK  PORTOUT(5),CVRTAREA+5(3)
         MVC   PORTOUTD(5),PORTOUT
         MVC   PORTOUTI(5),PORTOUT
         MVC   PORTOUTJ(5),PORTOUT
PRESDOC  EQU    *
         EXEC CICS DOCUMENT CREATE DOCTOKEN(TOKEN)
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                   -
```

```
               TEXT (TOP) LENGTH(TOPLEN)
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (TBL) LENGTH(TBLLEN)
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (TITLELN) LENGTH(TITLLEN)
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (DUMMYOUT) LENGTH(DUMMYLEN)
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (HEADOUT) LENGTH(HEADLEN)
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (DUMMYOUT) LENGTH(DUMMYLEN)
*       BROWSE DOCUMENTS
          EXEC CICS INQUIRE DOCTEMPLATE START
BROWSE    EQU   *
          EXEC CICS INQUIRE DOCTEMPLATE(DOCT) NEXT RESP(RESP)            -
               TYPE (DOCTYPE)
          CLC   RESP(4),DFHRESP(END)
          BE    ENDIT
*****************************************************************
          CLC   DOCTYPE,DFHVALUE(EBCDIC)
          BE    DOCHTML
          B     IMAGEO
DOCHTML   EQU   *
          MVC   DISPNAME(8),BLANKS
          MVC   RFSHNAME(8),BLANKS
          MVC   DOCNAME(8),BLANKS
          MVC   DOCTYPEO(5),BLANKS
          MVC   DOCTYPEO(5),=C'HTML '
          MVC   DISPNAME(8),DOCT
          MVC   RFSHNAME(8),DOCT
          MVC   DOCNAME(8),DOCT
          EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (LINEOUT) LENGTH(TLEN)
          B     BROWSE
IMAGEO    EQU   *
          MVC   IMAGTEMP(12),BLANKS
          MVC   IMAGDISP(12),BLANKS
          MVC   IMAGRFSH(8),BLANKS
          MVC   IMAGNAME(8),BLANKS
          MVC   IMAGNAMO(9),BLANKS
          MVC   IMAGTYPE(5),BLANKS
          MVC   IMAGTYPE(5),=C'Image'
          MVC   IMAGNAME(8),DOCT
          MVC   IMAGRFSH(8),DOCT
          MVC   IMAGNAMO(8),DOCT
CKINPUT   EQU    *
          SR     R6,R6
          SR     R7,R7
          SR     R1Ø,R1Ø
```

```
         L      R6,8               LENGTH OF TEMPLATE NAME
         LA     R7,IMAGNAMO
         LA     R1Ø,IMAGTEMP
CLCBLNK  EQU    *
         CLC    Ø(1,R7),=C' '       SEARCH FOR END OF INPUT
         BE     MVCGIF
         MVC    Ø(1,R1Ø),Ø(R7)
         LA     R7,1(R7)             BUMP
         LA     R1Ø,1(R1Ø)
         BCT    R6,CLCBLNK
MVCGIF   EQU   *
         MVC   Ø(4,R1Ø),=C'.GIF'          USE 'GIF' FOR ALL
         MVC   IMAGDISP(12),IMAGTEMP        IMAGES
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (IMAGOUT) LENGTH(ILEN)
         B     BROWSE
*********************************************************************
ENDIT    EQU   *
         EXEC  CICS INQUIRE DOCTEMPLATE END
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (TBLEND) LENGTH(TBLELEN)
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (HROUT) LENGTH(HRLEN)
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (BOTTOM) LENGTH(BOTMLEN)
         EXEC CICS WEB SEND DOCTOKEN(TOKEN)                             -
               CLNTCODEPAGE('ISO-8859-1')
EXIT     EQU   *
         EXEC  CICS RETURN
******
         DS    ØF
TMPL     DC    CL12' '
BLANKS   DC    CL48' '
DOCT     DC    CL8' '
TCPTRANS DC    CL8'CICSTCP'
PORTNO   DC    F'Ø'
TCPNAME  DC    CL8'HTTPNSSL'
IMAGNAMO DC    CL9' '
IMAGTEMP DC    CL12' '
*      TOP OF PAGE DEFINITION
TOP      DC    CL23'<!DOCTYPE HTML PUBLIC "'
         DC    CL39'-//W3C//DTD HTML 4.Ø Transitional//EN">'
         DC    CL6'<HTML>'
         DC    CL6'<HEAD>'
         DC    CL31'<TITLE>CICS region list</TITLE>'
         DC    CL7'</HEAD>'
         DC    CL6'<BODY '
         DC    CL31'BGCOLOR="#FFFF99" TEXT="BLACK">'
TOPLEN   DC    F'149'
```

```
*      TABLE DEFINITION LAYOUT
TBL      DC    CL8'<CENTER>'
         DC    CL30'<TABLE WIDTH="600" BORDER="0">'
         DC    CL4'<TR>'
         DC    CL28'<TD WIDTH="150">& </TD>'       document name
         DC    CL27'<TD WIDTH="50">& </TD>'         space
         DC    CL28'<TD WIDTH="100">& </TD>'       document type
         DC    CL27'<TD WIDTH="50">& </TD>'         space
         DC    CL28'<TD WIDTH="100">& </TD>'       display?
         DC    CL27'<TD WIDTH="50">& </TD>'         space
         DC    CL28'<TD WIDTH="100">& </TD>'       refresh?
         DC    CL5'</TR>'
TBLLEN   DC    F'240'
*
*      TITLE LINE
TITLELN  DC    CL4'<TR>'
         DC    CL31'<TD COLSPAN="7" align="center">'
         DC    CL29'<FONT SIZE="+1" COLOR="BLUE">'
         DC    CL19'Documents found in '
         DC    CL3'<B>'
JOBNM    DC    CL8' '
         DC    CL4'</B>'
         DC    CL7'</FONT>'
         DC    CL5'</TD>'
         DC    CL5'</TR>'
TITLLEN  DC    F'115'
*
*      HEADING LINE
HEADOUT  DC    CL4'<TR>'
         DC    CL4'<TD>'
         DC    CL29'<FONT SIZE="+1" COLOR="BLUE">'
         DC    CL13'Document Name'
         DC    CL7'</FONT>'
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'        space
         DC    CL4'<TD>'
         DC    CL29'<FONT SIZE="+1" COLOR="BLUE">'
         DC    CL13'Document Type'
         DC    CL7'</FONT>'
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'        space
         DC    CL4'<TD>'
         DC    CL29'<FONT SIZE="+1" COLOR="BLUE">'
         DC    CL7'Display'
         DC    CL7'</FONT>'
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'        space
         DC    CL4'<TD>'
         DC    CL29'<FONT SIZE="+1" COLOR="BLUE">'
```

```
         DC    CL7'Refresh'
         DC    CL7'</FONT>'
         DC    CL5'</TD>'
HEADEND  DC    CL5'</TR>'
HEADLEN  DC    F'277'
*
*      HORIZONTAL RULE
HROUT    DC    CL16'<HR WIDTH="15Ø">'
HRLEN    DC    F'16'
*
*      DUMMY LINE FOR SPACING
DUMMYOUT DC    CL4'<TR>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
         DC    CL16'<TD>& </TD>'
DUMMYEND DC    CL5'</TR>'
DUMMYLEN DC    F'121'
*
*      HTML LINE
LINEOUT  DC    CL4'<TR>'
         DC    CL7'<TD><B>'
DOCNAME  DC    CL8' '
         DC    CL9'</B></TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'
DOCTYPEO DC    CL5' '
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'                               display...
         DC    CL21'<A HREF="http://HOST:' (Change HOST as appropriate)
PORTOUTD DC    CL5' '
         DC    CL23'/cics/cwba/cicsdocd?pg='
DISPNAME DC    CL8' '
         DC    CL2'">'
         DC    CL7'Display'
         DC    CL4'</A>'
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'
         DC    CL21'<A HREF="http://HOST:' (Change HOST as appropriate)
PORTOUT  DC    CL5' '
         DC    CL23'/cics/cwba/cicsdocr?pg='
RFSHNAME DC    CL8' '
         DC    CL2'">'
         DC    CL7'Refresh'
```

```
         DC    CL4'</A>'
         DC    CL5'</TD>'
LINEEND  DC    CL5'</TR>'
TLEN     DC    F'253'
*
*
*      Image LINE
IMAGOUT  DC    CL4'<TR>'
         DC    CL7'<TD><B>'
IMAGNAME DC    CL8' '
         DC    CL9'</B></TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'
IMAGTYPE DC    CL5' '
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'                               display...
         DC    CL21'<A HREF="http://HOST:' (Change HOST as appropriate)
PORTOUTI DC    CL5' '
         DC    CL9'/graphix/'
IMAGDISP DC    CL12' '
         DC    CL2'">'
         DC    CL7'Display'
         DC    CL4'</A>'
         DC    CL5'</TD>'
         DC    CL16'<TD>& </TD>'                    space
         DC    CL4'<TD>'
         DC    CL21'<A HREF="http://HOST:'  (Change HOST as appropriate)
PORTOUTJ DC    CL5' '
         DC    CL23'/cics/cwba/cicsdocr?pg='
IMAGRFSH DC    CL8' '
         DC    CL2'">'
         DC    CL7'Refresh'
         DC    CL4'</A>'
         DC    CL5'</TD>'
IMAGEND  DC    CL5'</TR>'
ILEN     DC    F'243'
*
*      TABLE END
TBLEND   DC    CL8'</TABLE>'
         DC    CL9'</CENTER>'
TBLELEN  DC    F'17'
*
*      BOTTOM OF PAGE
BOTTOM   DC    CL7'</BODY>'
         DC    CL7'</HTML>'
BOTMLEN  DC    F'14'
         LTORG
         END
```

CICSDOCR

```
*********************************************************************
         TITLE ' CICSDOCR - INSTALL DOCUMENT TEMPLATES'
*********************************************************************
* THIS PROGRAM WILL INSTALL A DOCUMENT TEMPLATE.....
*   1 - CHECK THAT DFHCSD FILE IS NOT BEING USED IN ANOTHER REGION
*   2 - DISCARD DOCTEMPLATE
*   3 - EXEC CEDA DISPLAY COMMAND TO DETERMINE WHICH GROUP THE
*       TEMPLATE IS IN
*   4 - USE GROUP NAME FROM PREVIOUS STEP TO INSTALL DOCTEMPLATE.
*********************************************************************
DFHEISTG DSECT
*
SYSID    DS    CL4
TMPLNAME DS    CL48
RESP     DS    F
REGNSAVE DS    CL8
OSTAT    DS    F
TOKEN    DS    CL16
*
CICSDOCR CSECT
*
*                   GET THE SYSID FOR THIS REGION
         EXEC CICS ASSIGN SYSID (SYSID)
*
*   TRY TO OPEN AND CLOSE THE CSD DATASET. IF UNABLE TO OPEN IT, IT
*  MAY BE IN USE IN ANOTHER REGION. IF UNABLE TO ACCESS CSD FILE AND
*  DO NOT WANT TO DISCARD THE TEMPLATE ENTRY, SEND MESSAGE AND
*  GET OUT.
*
*                   TRY TO OPEN CSD FILE
         MVC   OSTAT,DFHVALUE(OPEN)
*
         EXEC CICS SET FILE (CSDFILE) OPENSTATUS (OSTAT) RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BNE   ERROR1              IF UNABLE TO OPEN, MIGHT BE IN
*                                  USE IN ANOTHER REGION. GET OUT.
*                                  TRY TO CLOSE CSD FILE
         MVC   OSTAT,DFHVALUE(CLOSED)
*
         EXEC CICS SET FILE (CSDFILE) OPENSTATUS (OSTAT) RESP(RESP)
*
         CLC   RESP,DFHRESP(NORMAL)
         BNE   ERROR1              IF UNABLE TO CLOSE, GET OUT
**
** GET TEMPLATE NAME
**
```

```
         MVC    STRLEN,=F'12'
*
         MVC    TMPL(12),BLANKS
         MVC    TMPLNAME(48),BLANKS
*     GET TEMPLATE NAME
CONTINUE EQU   *
         EXEC CICS WEB EXTRACT QUERYSTRING (TMPL)                      -
               QUERYSTRLEN (STRLEN)
*
CKINPUT  EQU    *
         L      R6,STRLEN           R6 HAS LEN OF STRING RETURNED
         S      R6,=F'3'            (MAX 11) SUBTRACT 3 FOR 'PG='
         LA     R7,TMPL+3           POINT PAST 'PG='
         LA     R8,TMPLNAME         POINT TO START OF TMPLNAME
CLCBLNK  EQU    *
         CLC    Ø(1,R7),=C' '       SEARCH FOR END OF INPUT
         BE     TNAME
         MVC    Ø(1,R8),Ø(R7)  BUILD TMPLNAME FOR DOCUMENT TO DISPLAY
         LA     R7,1(R7)            BUMP
         LA     R8,1(R8)            REGISTERS
         BCT    R6,CLCBLNK
*
TNAME    EQU   *
         MVC   INSTDOC(8),TMPLNAME
*
*  DISCARD THE CURRENT TEMPLATE DEFINITION...
         EXEC CICS DISCARD DOCTEMPLATE (INSTDOC) RESP (RESP)
         CLC   RESP,DFHRESP(NORMAL)
         BE    DOCDISP
         B     OTHERROR            IF NOT NORMAL RETURN,
*                                   GO TO OTHER ERROR.
**   LINK TO DFHEDAP WITH PARMS TO PERFORM THE DISPLAY GROUP
*
DOCDISP  EQU   *
**   MOVE FIELDS TO BUILD PARMS TO PASS TO CEDA TO FIND
***   GROUP NAME
*
         MVC   DISPDOC(8),TMPLNAME
         MVC   DISPGRP(3),SYSID+1   END OF SYSID NAME - S(TX.)
         MVC   DISPGRP+3(1),=C'*'
         EXEC CICS LINK PROGRAM('DFHEDAP') COMMAREA(DISPPARM)          *
               LENGTH(162)
**   EXAMINE RETURN FOR ERRORS
         CLC   DREMSGS,=H'5'        IF RETURN CODE GREATER THAN 4
         BL    DOCINST              MOVE RESPONSE MESSAGE.
         B     OTHERROR
**
*
**   LINK TO DFHEDAP WITH PARMS TO PERFORM THE INSTALL
```

```
*
DOCINST  EQU   *
         MVC   INSTGRP(8),DISPRESP+96     USE GROUP NAME
*                                  EXTRACTED FROM DISPLAY RESPONSE.
         EXEC CICS LINK PROGRAM('DFHEDAP') COMMAREA(INSTPARM)          *
               LENGTH(1Ø9)
**   EXAMINE RETURN FOR ERRORS
         CLC   IREMSGS,=H'5'        IF RETURN CODE GREATER THAN 4
         BL    NORMRET              MOVE RESPONSE MESSAGE.
         B     OTHERROR
**
**           NORMAL RETURN DISPLAY
NORMRET  EQU    *
*
         MVC    NORTMPL(8),TMPLNAME
*
         EXEC CICS DOCUMENT CREATE DOCTOKEN(TOKEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (TOP) LENGTH(TOPLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (NORLINE) LENGTH(NORLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (BOTTOM) LENGTH(BOTMLEN)
*
         EXEC CICS WEB SEND DOCTOKEN(TOKEN)                            -
               CLNTCODEPAGE('ISO-8859-1')
*
         B    RETURN
**
**           OTHER ERROR RETURN DISPLAY
OTHERROR EQU    *
*
         MVC   OTHTMPL(8),TMPLNAME
         MVC   OTHERROR(62),IREDIAG+3
         EXEC CICS DOCUMENT CREATE DOCTOKEN(TOKEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (TOP) LENGTH(TOPLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (OERLINE) LENGTH(OERLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                     -
               TEXT (BOTTOM) LENGTH(BOTMLEN)
*
         EXEC CICS WEB SEND DOCTOKEN(TOKEN)                            -
```

```
               CLNTCODEPAGE('ISO-8859-1')
*
         B    RETURN
**
**           ERROR RETURN DISPLAY
ERROR1   EQU    *
*
         EXEC CICS DOCUMENT CREATE DOCTOKEN(TOKEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (TOP) LENGTH(TOPLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (ERRLINE) LENGTH(ERRLEN)
*
         EXEC CICS DOCUMENT INSERT DOCTOKEN(TOKEN)                      -
               TEXT (BOTTOM) LENGTH(BOTMLEN)
*
         EXEC CICS WEB SEND DOCTOKEN(TOKEN)                             -
               CLNTCODEPAGE('ISO-8859-1')
*
         B    RETURN
**
RETURN   EQU   *
         EXEC  CICS RETURN
**
*
INSTPARM DS    ØF             CEDA PARAMETERS FOR INSTALL
INSTCOMM DC    A(INSTD1)
INSTLEN  DC    A(INSTILEN)
INDFIELD DC    A(INSTINDR)
IOUTPUT  DC    A(INSTRESP)
IOUTLEN  DC    A(IRESPLEN)
*
         DS    ØF
INSTD1   DC    CL21'INSTALL DOCTEMPLATE ('
INSTDOC  DC    CL8' '
INSTD2   DC    CL9') GROUP ('
INSTGRP  DC    CL8' '
INSTDEND DC    CL1')'
*
INSTILEN DC    H'47'               LENGTH OF COMMAND
INSTINDR DC    X'ØØ'               DO NOT DISPLAY OUTPUT AT TERMINAL
         DS    ØH
INSTRESP DS    ØCL62               RESPONSE TO COMMAND
*                          TRANSLATION STAGE
IRTLEN   DS    H                   LENGTH OF RESPONSE FIELD
IRTMSGN  DS    H                   NUMBER OF MESSAGES PRODUCED
IRTMSGS  DS    H                   HIGHEST SEVERITY MESSAGE
```

```
*                         EXECUTION STAGE
IRELEN   DS    H                    LENGTH OF RESPONSE FIELD
IREMSGN  DS    H                    NUMBER OF MESSAGES PRODUCED
IREMSGS  DS    H                    HIGHEST SEVERITY MESSAGE
IREDIAG  DS    CL5Ø                 DIAGNOSTIC MESSAGES
IRESPLEN DC    H'62'                LENGTH OF RESPONSE FIELD
*
**
*
DISPPARM DS    ØF                   CEDA PARAMETERS FOR DISPLAY
DISPCOMM DC    A(DISPD1)
DISPLEN  DC    A(DISPILEN)
DNDFIELD DC    A(DISPINDR)
DOUTPUT  DC    A(DISPRESP)
DOUTLEN  DC    A(DRESPLEN)
*
         DS    ØF
DISPD1   DC    CL21'DISPLAY DOCTEMPLATE ('
DISPDOC  DC    CL8' '
DISPD2   DC    CL9') GROUP ('
DISPGRP  DC    CL4' '
DISPDEND DC    CL1')'
*
DISPILEN DC    H'43'                LENGTH OF COMMAND
DISPINDR DC    X'ØØ'                DO NOT DISPLAY OUTPUT AT TERMINAL
         DS    ØH
DISPRESP DS    ØCL112               RESPONSE TO COMMAND
*                         TRANSLATION STAGE
DRTLEN   DS    H                    LENGTH OF RESPONSE FIELD
DRTMSGN  DS    H                    NUMBER OF MESSAGES PRODUCED
DRTMSGS  DS    H                    HIGHEST SEVERITY MESSAGE
*                         EXECUTION STAGE
DRELEN   DS    H                    LENGTH OF RESPONSE FIELD
DREMSGN  DS    H                    NUMBER OF MESSAGES PRODUCED
DREMSGS  DS    H                    HIGHEST SEVERITY MESSAGE
DREDIAG  DS    CL1ØØ                DIAGNOSTIC MESSAGES
DRESPLEN DC    H'112'               LENGTH OF RESPONSE FIELD
*
**
TMPL     DC    CL12' '              DOCUMENT(TEMPLATE) NAME
BLANKS   DC    CL5Ø' '
STRLEN   DC    F'12'                QUERY STRING LENGTH
FOPEN    DC    C'OPEN'
FCLOSE   DC    C'CLOSED'
CSDFILE  DC    CL8'DFHCSD'
*
*      TOP OF PAGE DEFINITION
TOP      DC    C'<!DOCTYPE HTML PUBLIC "'
         DC    C'-//W3C//DTD HTML 4.Ø TRANSITIONAL//EN">'
```

```
         DC    C'<HTML>'
         DC    C'<HEAD>'
         DC    C'<TITLE>REFRESH ERROR</TITLE>'
         DC    C'</HEAD>'
         DC    C'<BODY BGCOLOR="WHITE"'
         DC    C' TEXT="BLACK">'
TOPLEN   DC    F'146'
*
*     ERROR LINE
ERRLINE  DC    C'<FONT COLOR="RED">'
         DC    C'Error refreshing template.'
         DC    C'<br> The CSD file may be open in another region.'
         DC    C'<br><br> Click the "BACK" button to return to'
         DC    C' the previous screen.'
         DC    C'</FONT>'
ERRLEN   DC    F'168'
*
*
*     NORMAL RESPONSE LINE
NORLINE  DC    C'<FONT COLOR="GREEN">'
         DC    C'Refresh complete for '
NORTMPL  DC    CL8' '
         DC    C'<br><br> Click the "BACK" button to return to'
         DC    C' the previous screen.'
         DC    C'</FONT>'
NORLEN   DC    F'124'
*
*
*     OTHER ERROR LINE
OERLINE  DC    C'<FONT COLOR="RED">'
         DC    C'Refresh failed for '
OTHTMPL  DC    CL8' '
         DC    C'<BR>'
         DC    C'Refresh manually.'
         DC    C'<BR>'
OTHERR   DC    CL62' '
         DC    C'<br><br> Click the "BACK" button to return to'
         DC    C' the previous screen.'
         DC    C'</FONT>'
OERLEN   DC    F'2Ø8'
*
*      BOTTOM OF PAGE
BOTTOM   DC    CL7'</BODY>'
         DC    C'</HTML>'
BOTMLEN  DC    F'16'
*
         LTORG
*********************************************************************
         COPY  DFHAID              ATTENTION ID DEFINITIONS
```

```
         COPY  DFHBMSCA            BMS ATTRIBUTE DEFINITIONS
         DFHREGS
         END
```

INDXDOC

(Note: change HOST:PORT to appropriate values.)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
     <title>CICS Documents</title>
</head>
<frameset border="0" frameborder="no" cols="150,*">
<frame frameborder="0" scrolling="auto" noresize name="cicsframe"
src="http://sysa:8001/cics/cwba/cicsdoct">
<frame frameborder="0" noresize name="displayframe"
src="http://HOST:PORT/cics/cwba/cicsdocd?pg=DOCINFO">
</frameset>
</html>
```

Jim Smith
System Programmer
Onondaga County Dept of Information Technology (USA)

Why not share your expertise and earn money at the same time? *CICS Update* is looking for macros, program code, REXX, etc, that experienced CICS users have written to make their life, or the lives of their users, easier. We will publish it (after vetting by our expert panel) and send you a cheque when the article is published. Articles can be of any length and can be sent or e-mailed to Trevor Eddolls at any of the addresses shown on page 2. A free copy of our *Notes for contributors* is available from our Web site at www.xephon.com/nfc.

CICS session reuse and the DFHSHUNT logstream

INTRODUCTION

The 'implicit forget flow' optimization of two-phase commit processing can lead to an excessive growth of the DFHSHUNT logstream. This article discusses the background to this situation, and also explains how it has been addressed by CICS PTF.

THE DFHSHUNT LOG STREAM

The Log Manager component of CICS Transaction Server writes information about changes made to recoverable system activity into the CICS system log. This is a single item conceptually; physically it is represented by two MVS System Logger logstreams, known as the primary and secondary CICS system logs. They are more commonly referred to as DFHLOG and DFHSHUNT. DFHLOG is used to store log records for those tasks with reasonably short-lived Units Of Work (UOWs). DFHSHUNT is used to store log records for tasks that are regarded as 'long-running' under CICS. A task is recognized as long-running by CICS if it does not cause any log records to be written within the time between two adjacent activity keypoint operations (CSKP system tasks). CICS uses activity keypoint processing as the time to review the logging activity for the various tasks on the system, and to move their log data between the system logstreams if appropriate. By moving log records for such long-running tasks from the DFHLOG to the DFHSHUNT logstream, CICS is then able to trim this data from the DFHLOG logstream and so better manage the MVS System Logger primary storage usage for the logstream – that is, within the Coupling Facility structure or on the Staging Dataset.

Tasks can fail to generate log records within the interval between successive activity keypoint operations for a number of reasons. One is that they are busy performing non-recoverable work of sufficiently long duration to cause them to remain within the CICS system and span two activity keypoints. Tasks executing conversational programs under

CICS are one example of this, where much of their time is spent waiting for further end user input from a terminal. Another example is a 'batch-style' or 'background' task, typically a non-terminal one, running within CICS and performing some long-running non-recoverable operation such as numerical calculations or browsing user files. Such tasks generally have low dispatching priorities, and hence are more likely to remain within the system for sufficient duration to span successive activity keypoints.

Other examples of long-running work within CICS relate to syncpoint activity. When a task is executing a syncpoint operation, a failure may occur within a crucial period of syncpoint activity while the task is in an 'indoubt' state with respect to the syncpoint's outcome. If so, CICS can 'shunt' the UOW for the task. This preserves aspects of the UOW, such as resource locks, until such time as the situation can be resolved and the UOW 'unshunted' and allowed to complete the syncpoint. The duration between a shunt and its corresponding unshunt operation may be considerable; during this window, the UOW will not be generating log records and so CICS will deem it long-running and eligible for movement of its log data from DFHLOG to DFHSHUNT at a subsequent activity keypoint.

SYNCPOINT PROCESSING

Perhaps the most common reason for log data to be moved from DFHLOG to DFHSHUNT relates to UOWs that have completed their syncpoint operations on a local CICS system, but which are awaiting an 'implicit forget' flow from a connected CICS system. Once a UOW has begun, it is in an 'inflight' state until a syncpoint occurs. During the syncpoint operation, CICS will optimize the series of events needed to commit the UOW, as appropriate. If recoverable changes by the UOW are distributed across interconnected CICS systems, a two-phase commit (2PC) will be performed. CICS will first prepare, then later commit, the UOW's changes. Between these stages, a UOW can enter the indoubt state whilst the CICS system is awaiting confirmation of whether to commit the UOW forwards or backwards. This occurs if the CICS system is participating in a distributed syncpoint involving a number of interconnected regions. Finally, a successful completion of a syncpoint will place the UOW in a 'committed' state. At this point,

CICS can release various resources, terminate the transaction if appropriate, and complete the UOW. (Note: a successful syncpoint completion may well have committed backwards rather than forwards, if, say, the task was abending and backing out, or honouring an EXEC CICS SYNCPOINT ROLLBACK command. A backout is a backwards commit.)

SESSION ALLOCATION

A connection between two CICS systems will define a number of sessions that can be used for distributed activity, such as function-shipping. These sessions are used as required. During periods of peak activity, a high-water-mark number of sessions will be in use. As distributed activity drops away, so parallel session usage will fall too.

Because of the use of an optimization known as 'implicit forget', a UOW may need to be retained after the syncpoint completes, until some further information is received from an interconnected CICS system. Implicit forget avoids the need for excessive network traffic between systems. The next flow across the session between the two CICS systems is used as evidence that the previous commit flow was received and honoured by the remote side. UOWs being retained by CICS until a session is reused – using implicit forget to denote that the UOW (which had previously syncpointed across that session) may now be forgotten – may need to be retained for a considerable time. For example, peak CICS system activity may occur daily, and so the high-water-mark session will not see further activity flow across it for a number of hours after the peak. This means that a UOW which had previously used this session will be retained for this length of time, after it has syncpointed and so locally committed its resources. This has little bearing on the CICS system itself; maintaining the UOW is trivial in terms of system management and storage use. However, it will mean that the UOW will be deemed long-running because it will not write to the CICS system log until caused to do so by session traffic. As such, CICS will move the UOW's log data from DFHLOG to DFHSHUNT once a complete activity keypoint interval has elapsed. This is good from the point of view of space management on the logstreams. However, it will mark a point on DFHSHUNT after which log data

cannot be deleted (via CICS 'log tail trimming') until the UOW is subsequently discarded by CICS. This deletion will not take place until an implicit forget flow is received across the session.

Such a use of DFHSHUNT can lead to the growth of this secondary logstream, as other UOWs have their data moved to it for the same reason. These may in turn receive implicit forget flows across their own sessions, and be discarded by CICS. However, since this will happen after the delimiting point of the log records for the UOW awaiting reuse of the high-water-mark session, their log data will not be removable from the DFHSHUNT logstream. Eventually, this may result in the MVS System Logger initiating offload processing for the DFHSHUNT logstream when its primary storage usage reaches the Highoffload percentage threshold. This manifests itself in additional MVS System Logger I/O activity and Offload Dataset allocations ('DASD shifts'). The situation will persist until the session is reused, when CICS can discard the UOW, and an activity keypoint can then invoke the MVS System Logger to trim DFHSHUNT of this unrequired data.

Using the CEMT INQUIRE UOW command, such UOWs appear with a UOWSTATE of COMMIT (abbreviated to 'Com' on the principal display). They will typically have much longer lifetimes than other inflight UOWs. The age of a UOW is shown in the AGE field; the value is in number of seconds since the UOW was created.

CICS MODIFICATIONS TO ADDRESS THE PROBLEM

CICS Transaction Server 1.3 and 2.2 have been changed to address this situation. PTF UQ63466 has been provided for CICS Transaction Server Version 1.3, and PTF UQ63918 for Version 2.2. The modification to CICS causes a summary of the UOW's pertinent log data, relating to its obligations with interconnected systems, to be relogged at activity keypoint time. The UOW log data can be summarized in this manner since only a subset of the information has to be maintained once the UOW has syncpointed and locally committed its changes. Data pertaining to changes to local resources can now be discarded; only that data relating to obligations with other systems needs to be retained. CICS needs to re-log the subset of data that is still required. The old log data can then be deleted when CICS trims the system log at the end of

activity keypoint processing. Such a summarizing approach avoids the need to move the log data for such a long-running UOW (ie one awaiting a forget flow to the DFHSHUNT logstream); it also optimizes storage use on DFHLOG for such UOWs. Any secondary effects, such as MVS System Logger offload activity, are also avoided for such UOWs' log data.

By making this change, the potential for any considerable build up of log data held on DFHSHUNT is reduced. It is anticipated that this design enhancement will result in DFHSHUNT being used to store log data only for UOWs that are shunted or for those that are deemed long-running for reasons other than awaiting an implicit forget flow (such as conversational and 'background' tasks).

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

A generic CICS compiler

The IBM way to compile programs is by means of PROCs, a parametrized JCL skeleton that can be invoked by jobs. They may be useful as examples of what JCL is needed to perform a certain task, but apart from that, they are not very user-friendly. Of course, I never use them. Instead, I create my own JCL, usually through a REXX program. Over the years I've created a number of EXECs to generate and submit JCL to compile all types of programs.

The program presented here is a CICS pre-compiler, compiler, and link editor for Assembler or COBOL code. It covers most common options, and is fully parameterized in terms of libraries, program names, etc. It can compile to an arbitrary number of CICSs, each designated by a suffix letter or number. Each CICS can have its specific copybook or macro input libraries as well as its output LOADLIB.

The input for this EXEC is the source code file and the CICS suffix. The input is done by an ISPF panel. The EXEC automatically detects the programming language.

Looking at the code, you can see at the beginning a number of variables that represent the standard libraries for CICS, COBOL, Assembler, and LE, as well as the names of the pre-compiler, compiler, and link editor programs. Modify these values according to your installation configuration.

Immediately below, there is a table with the CICS suffixes known by this program. The suffix is simply a code that represents a specific CICS, and that will be used to differentiate the output LOADLIB and eventually some input libraries (typically, copybooks or Assembler macros that might have different versions for different CICS).

Once this table is set, you must enter the library-specific names for each suffix entry. This is done in the 'Select' statement, where each 'when' corresponds to an entry in the suffix table. You can leave the 'copy' entries blank, as I did in the second entry. In this case, the corresponding JCL line is not generated. If you need to concatenate more than one library, you can easily do so – create a variable with a similar name, and double the relevant JCL line in the 'queue' statements.

The JCL generated is fairly simple and should cover most needs. The options used for compile and link edit can be easily modified or parameterized to achieve greater flexibility. For example, you can add the 'SP' option for specific users; or you can implement CICS access restrictions, depending on the user.

CICSCOMP REXX SOURCE CODE

```
/*  REXX ISPF *=========================================================*/
/*                                                                     */
/*  CICSCOMP - Generates and submits JCL to pre-compile, compile,      */
/*             and link edit COBOL or Assembler CICS programs.         */
/*                                                                     */
/*=====================================================================*/
arg inpfile .                    /* optional argument: input file */
/*=====================================================================*/
/*  CICS-independent libraries and programs                            */
/*=====================================================================*/
precomp_asm    = "DFHEAP1$"            /* asm precompiler program */
precomp_cob    = "DFHECP1$"            /* cob precompiler program */
comp_asm       = "ASMA90"              /* asm compiler program    */
comp_cob       = "IGYCRCTL"            /* cob compiler program    */
linkeditor     = "HEWLH096"            /* linkeditor program      */
cics_steplib   = "CICS.SDFHLOAD"       /* cics loadlib            */
```

```
cob_steplib    = "IGY.SIGYCOMP"        /* cobol loadlib           */
asm_syslib     = "SYS1.MACLIB"         /* asm maclib              */
cicsasm_syslib = "CICS.SDFHMAC"        /* cics asm maclib         */
cicscob_syslib = "CICS.SDFHCOB"        /* cics cob maclib         */
cee_steplib1   = "CEE.SCEECICS"        /* LE cics lib             */
cee_steplib2   = "CEE.SCEERUN"         /* LE lib                  */
cee_steplked   = "CEE.SCEELKED"        /* LE linkedit lib         */
/*================================================================*/
/*  List of CICS suffixes known by this program                   */
/*================================================================*/
cicsuf.Ø =  3                         /* total number of suffixes */
cicsuf.1 = 'A'                        /* cics suffix list         */
cicsuf.2 = 'B'
cicsuf.3 = 'C'
/*================================================================*/
/* CICS-dependent libraries for job creation are specified below. */
/* Outlib (the program's LOADLIB destination) is mandatory.       */
/* Copy entries (COBOL copybooks or ASM macros) are optional.     */
/* Each suffix entry in the above "cicsuf" table should have a    */
/* correspondind "when" in the select statement below.            */
/*================================================================*/
call display_panel                /* Call ISPF panel to get input */
                                  /* file and sf (cics suffix)    */
select
  when sf = cicsuf.1 then do
     asm_copy = "ASM.SIS.MACLIB.A"
     cob_copy = "CICS.COPY.CICSA"
     outlib   = "CICS.SDFHLOAD.CICSA"
  end
  when sf = cicsuf.2 then do
     asm_copy = ""
     cob_copy = ""
     outlib   = "CICS.SDFHLOAD.CICSB"
  end
  when sf = cicsuf.3 then do
     asm_copy = "ASM.SIS.MACLIB.C"
     cob_copy = "CICS.COPY.CICSC"
     outlib   = "CICS.SDFHLOAD.CICSC"
  end
end
/*================================================================*/
/*  create and submit job                                         */
/*================================================================*/
call alloc_jobfile

select
 when type = "COB" then call queue_job_cobol
 when type = "ASM" then call queue_job_assembler
end

"execio * diskw jobe (finis "
```

```
"submit '"jobname"'"
say "Job submitted for cics " sf
say "Program type assumed " type

exit:
 xx = msg(off)
 "free dd(input)"
 "free dd(jobe)"
exit
/*====================================================================*/
/*                          Subroutines                               */
/*====================================================================*/
/*  Display ISPF input panel and validate entries                     */
/*====================================================================*/
display_panel:
 inpfile = strip(inpfile,,"'")
 curpos = 'inpfile'
 do panel_loop = 0
    address ispexec
    'addpop row(1) column(1)'
    'display panel(cicscomp) cursor('curpos')'
    if rc=8 then do
       address tso
       signal exit
    end
    'rempop'
    address tso
    inpfile = strip(inpfile,,"'")
    call alloc_file inpfile input
    if result <> 0 then do
       msg0 = "Error allocating Dataset"
       iterate panel_loop
    end
    call find_input_type
    if result <> 0 then do
       msg0 = "Could not determine input file language"
       iterate panel_loop
    end
    parse var inpfile with pds1 "(" name ")"
    sf = space(sf,0)
    do k = 1 to cicsuf.0
       if sf = cicsuf.k then leave panel_loop
    end
    msg0 = "Invalid CICS"
    curpos = 'sf'
 end
return
/*====================================================================*/
/*  Create job for Assembler program                                  */
/*====================================================================*/
```

```
queue_job_assembler:

dropbuf
queue "//"userid()"A JOB "userid()","
queue "//         MSGCLASS=X,"
queue "//         MSGLEVEL=(1,1),"
queue "//         CLASS=A,"
queue "//         REGION=2048K,"
queue "//         NOTIFY="USERID()
queue "//*"
queue "//PRECOMP  EXEC PGM="precomp_asm
queue "//SYSIN    DD DISP=SHR,DSN="inpfile
queue "//STEPLIB  DD DISP=SHR,DSN="cics_steplib
queue "//SYSPUNCH DD DSN=&&TEMP1,"
queue "//         DCB=(BLKSIZE=800),"
queue "//         DISP=(,PASS),"
queue "//         UNIT=SYSDA,"
queue "//         SPACE=(CYL,(1,1))"
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"

queue "//ASMCOMP  EXEC PGM="comp_asm","
queue "//         COND=(3,LT,PRECOMP),"
queue "//         PARM='NODECK,OBJECT,NOXREF'"
queue "//SYSLIB   DD DISP=SHR,DSN="asm_syslib
queue "//         DD DISP=SHR,DSN="cicsasm_syslib
if asm_copy <> "" then,
   queue "//         DD DISP=SHR,DSN="asm_copy
queue "//SYSIN    DD DSN=&&TEMP1,DISP=(OLD,DELETE)"
queue "//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT2   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT3   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSLIN   DD DSN=&&TEMP2,"
queue "//         DCB=(BLKSIZE=800),"
queue "//         DISP=(,PASS),"
queue "//         UNIT=SYSDA,"
queue "//         SPACE=(CYL,(1,1))"
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"

queue "//LINKED   EXEC PGM="linkeditor","
queue "//         COND=(3,LT,ASMCOMP),"
queue "//         PARM='NOXREF,RENT,AMODE=31,RMODE=ANY'"
queue "//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1)),"
queue "//         DCB=(BLKSIZE=1024)"
queue "//SYSLIN   DD DISP=SHR,DSN="cicsasm_syslib"(DFHEILIA)"
queue "//         DD DISP=(OLD,DELETE),DSN=&&TEMP2"
queue "//SYSLIB   DD DISP=SHR,DSN="cics_steplib
queue "//         DD DISP=SHR,DSN="outlib
queue "//SYSLMOD  DD DISP=SHR,DSN="outlib"("name")"
```

```
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"
queue ""

return
/*=================================================================*/
/*  Create job for COBOL program                                   */
/*=================================================================*/
queue_job_cobol:

dropbuf
queue "//"userid()"C JOB "userid()","
queue "//         MSGCLASS=X,"
queue "//         MSGLEVEL=(1,1),"
queue "//         CLASS=A,"
queue "//         REGION=3Ø72K,"
queue "//         NOTIFY="USERID()
queue "//*"
queue "//PRECOMP  EXEC PGM="precomp_cob","
queue "//         PARM='COBOL2'"
queue "//SYSIN    DD DISP=SHR,DSN="inpfile
queue "//STEPLIB  DD DISP=SHR,DSN="cics_steplib
queue "//SYSPUNCH DD DSN=&&TEMP1,"
queue "//         DCB=(BLKSIZE=8ØØ),"
queue "//         DISP=(,PASS),"
queue "//         UNIT=SYSDA,"
queue "//         SPACE=(CYL,(1,1))"
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"

queue "//COBCOMP  EXEC PGM="comp_cob","
queue "//         COND=(3,LT,PRECOMP),"
queue "//         PARM='NODYNAM,LIB,APOST,OBJECT,DATA(31)'"
queue "//STEPLIB  DD DISP=SHR,DSN="cics_steplib
queue "//         DD DISP=SHR,DSN="cee_steplib1
queue "//         DD DISP=SHR,DSN="cee_steplib2
queue "//         DD DISP=SHR,DSN="cob_steplib
queue "//SYSLIB   DD DISP=SHR,DSN="cicscob_syslib
if cob_copy <> "" then,
   queue "//         DD DISP=SHR,DSN="cob_copy
queue "//SYSIN    DD DSN=&&TEMP1,DISP=(OLD,DELETE)"
queue "//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT2   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT3   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT4   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT5   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT6   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSUT7   DD UNIT=SYSDA,SPACE=(CYL,(1,1))"
queue "//SYSLIN   DD DSN=&&TEMP2,"
queue "//         DCB=(BLKSIZE=8ØØ),"
```

```
queue "//         DISP=(,PASS),"
queue "//         UNIT=SYSDA,"
queue "//         SPACE=(CYL,(1,1))"
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"

queue "//LINKED   EXEC PGM="linkeditor","
queue "//         COND=(7,LT,COBCOMP),"
queue "//         PARM='NOXREF,RENT,AMODE=31,RMODE=ANY'"
queue "//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(1,1)),"
queue "//         DCB=(BLKSIZE=1Ø24)"
queue "//SYSLIN   DD DISP=SHR,DSN="cicscob_syslib"(DFHEILIC)"
queue "//         DD DISP=(OLD,DELETE),DSN=&&TEMP2"
queue "//SYSLIB   DD DISP=SHR,DSN="cics_steplib
queue "//         DD DISP=SHR,DSN="cee_steplked
queue "//         DD DISP=SHR,DSN="cee_steplib1
queue "//         DD DISP=SHR,DSN="outlib
queue "//SYSLMOD  DD DISP=SHR,DSN="outlib"("name")"
queue "//SYSPRINT DD SYSOUT=*"
queue "//*"
queue ""

return
/*====================================================================*/
/*  Find if the input file type is COBOL or Assembler                 */
/*====================================================================*/
find_input_type:
 type = ""
 do alpha = Ø
    execio 1 diskr input
    if rc <> Ø then leave
    pull line
    kmax = words(line)
    do k = 1 to kmax -1
       if left(word(line,k),2) = "ID" & ,
          left(word(line,k+1),8) = "DIVISION" then do
          type = "COB"
          leave alpha
       end
    end
    do k = 1 to kmax
       if word(line,k) = "CSECT" | ,
          word(line,k) = "DSECT" then do
          type = "ASM"
          leave alpha
       end
    end
 end
 execio Ø diskr input "(finis"
 "free dd(input)"
 if type = "" then retcod = -1
```

```
 else retcod = 0
return retcod
/*====================================================================*/
/* Input file allocation to check for its existence and type      */
/*====================================================================*/
alloc_file:
 xx = msg(off)
 arg dsname ddname
 "free dd("ddname")"
 "alloc da('"dsname"') dd("ddname") shr"
return rc
/*====================================================================*/
/*  Allocate job temporary file                                       */
/*====================================================================*/
alloc_jobfile:
 zz = msg(off)
 jobname = userid()".CICSJOB"
 "free dd(jobe)"
 "alloc dd(jobe) da('"jobname"') new delete blksize(8000)
        lrecl(80) recfm(f,b) dsorg(ps) space(1 1) tracks"
 if rc <> 0 then do
    say "Error "rc" allocating" jobname
    signal exit
 end
return
```

CICSCOMP PANEL SOURCE CODE

```
)ATTR
  _ TYPE(INPUT) CAPS(ON) JUST(LEFT)  COLOR(YELLOW)
  % TYPE(INPUT) CAPS(ON) JUST(LEFT)  COLOR(RED)
  $ TYPE(TEXT)   INTENS(HIGH) SKIP(ON) COLOR(YELLOW)
  * TYPE(OUTPUT) INTENS(HIGH) SKIP(ON) COLOR(WHITE) CAPS(OFF)
)BODY WINDOW(66,8)
$
$ Input program.:_inpfile                                    $
$
$ Cics suffix...:_sf$
$
$   *msg0
$
)INIT
&ZWINTTL = 'Cics preprocessor and compiler'
)PROC
VER(&inpfile,NONBLANK,dsname)
VER(&sf,NONBLANK)
)END
```

CICS news

Mackinney Systems has released Version 2.5 of Easy Help for CICS. The product gives users the ability to request help on individual fields on the screens they use. It also allows users to maintain the help text themselves without involving data processing personnel. Easy Help will display the help text for the requested field in a pop-up window so that the field in question and most of the rest of the screen is still visible.

The latest version has a 'Find' function. This has been added to the help display to allow users to find a string within the help text for the field being displayed.

The 'Sticky Cursor' function has been enhanced by allowing users to specify which text can be used for a field by surrounding it with a '~' (tilde) or other specified character. The user can tab directly to text indicated as 'sticky'. Optionally, users can specify that the user can only select 'sticky' text to be copied to a screen input field.

Help text for a particular field or for the screen overview is no longer limited to a maximum of 25 17-line screens. (425 lines). Up to 5,000 lines of help text can be created and updated if using the Mackinney QEDITOR editor.

The HELPUTIL program has been enhanced to allow loading, unloading, and deleting of help text for a specified field. This allows for editing of help text using editors other than the native and CICS/QEDITOR editors.

For further information contact:
Mackinney Systems, 2740 South Glenstone, Suite 103 , Springfield, MI 65804, USA.
Tel: (417) 882-8012.
URL: http://www.mackinney.com/news.htm.

* * *

Mackinney Systems has released Version 5.4 of CICS/Spooler. The product gives users the ability to direct reports to a destination name associated with a printer profile (which may or may not have the same name as the actual printer). Printer profiles allow various options to be set for the physical printer such as maximum number of print characters per line, command codes to be sent before and after the report is printed, and printer translate table for automatic translation of unwanted characters. A physical printer can have more than one printer profile to allow for printing with different options.

The latest version provides support for 31-bit CICS programs.

For VSE installations, it now provides a feature to display and view the VSE/Power In-Creation queue.

The new version provides a feature to specify the amount of time to wait before scanning for reports to be printed to be less than one minute (as determined by the installation option 'AUTO TIME').

For further information contact:
Mackinney Systems, 2740 South Glenstone, Suite 103 , Springfield, MI 65804, USA.
Tel: (417) 882-8012.
URL: http://www.mackinney.com/news.htm.