March 2001

In this issue

3  An unwanted effect of the new MROFSE SIT parameter
5  Displaying the autoinstall-model-name and the associated terminal-name
10 Yet another cold start next time – revisited
11 CICS dynamic hardcopy printer assignment – part 2
36 Simple tool to manage the data extracted from CICS CSD in a DB2 environment
48 CICS news

© Xephon plc 2001
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 January 1994 issue, are available separately to subscribers for £16.00 ($23.50) each including postage.

CICS Update on-line
Code from CICS Update can be downloaded from our Web site at http://www.xephon.com/cicsupdate.html; you will need the user-id shown on your address label.

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 contact us at any of the addresses above and we will send you a copy of our Notes for Contributors, or you can download a copy from www.xephon.com/contnote.html.

© Xephon plc 2001. All rights reserved. None of the text in this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior permission of the copyright owner. Subscribers are free to copy any code reproduced in this publication for use in their own installations, but may not sell such code or incorporate it in any commercial product. No part of this publication may be used for any form of advertising, sales promotion, or publicity without the written permission of the publisher. Copying permits are available from Xephon in the form of pressure-sensitive labels, for application to individual copies. A pack of 240 labels costs $36 (£24), giving a cost per copy of 15 cents (10 pence). To order, contact Xephon at any of the addresses above.

Printed in England.
An unwanted effect of the new MROFSE SIT parameter

CICS 5.3.0 brings with its set of changes a new System Initialization Table (SIT) parameter – MROFSE. When activated, this parameter can elongate response times for File Owning Regions (FOR).

New parameters are brought into existence for good reasons. You only get the benefit if you turn them on. Here’s what we found when we activated MROFSE.

The parameter is designed to bring the Application Owning Region (AOR) in step with the long-running mirror transaction in the FOR. If the AOR transaction issues more than a couple of file requests and user syncpoints, why have CICS go through the overhead of setting up its connection to the mirror task for each syncpoint? Wouldn’t it be better to have this stick around for the life of the task? Sounded good to me!

When CICS 5.3.0 was brand new, the documentation was a bit scarce for this SIT parameter. The CICS Intercommunication Guide now says, “MROFSE=YES, specified on the front-end region, extends the retention of the mirror task and the session from the next syncpoint to the end of the task”. It continues, “Conceptually, MROLRM is specified on the back-end region and MROFSE is specified on the front-end region. However, if the distinction between ‘back end’ and ‘front end’ is not clear, it is safe to code both parameters on each region if necessary.”

The System Definition Guide has been modified to say that MROFSE specifies “whether you want to extend the lifetime of the long-running mirror to keep it allocated until the end of the task rather than after a user syncpoint for function shipping applications”. This is subtly different, and it would seem at first glance to be a misleading definition. The long-running mirror is only present in FORs and MROLRM must be specified in order to get it. MROFSE is for AORs, right? Well, it turns out that if you have it coded in your FOR, it really will do just that! The mirror tasks last even longer if you set both these
parameters to YES than if you have just MROLRM turned on. And this can play havoc with your response time reports. We monitor FOR response times fairly closely and were quite surprised to see a 60% increase in the numbers! The averages went from .125 to .2 secs! Average CPU times did not change. Luckily, the AOR response times did not change. The same work was being done as before, but now the long-running mirror stayed connected to the AOR for a longer time – for the life of the task and not just the life of one syncpoint.

Another implication of this parameter is that the session between the AOR and the FOR will stay active longer. In a production environment, with many sessions defined, this should not be a problem. Six active sessions can handle a lot of traffic. Defining 50 would be more likely than just having six, I would think. Test environments might be different – ours was. We have a low number of sessions defined for function shipping in our test regions. We ran into a situation where an application that featured some ‘normal’ long-running tasks, acquired all the sessions defined to the FOR. No other application in the region could get a file request satisfied! More sessions were added and the problem was eliminated. This also shed light on the FOR numbers. Super long-running AOR transactions used to give up the FOR mirror at every syncpoint. Now they didn’t. So if a transaction runs continually, say scanning a queue, now the corresponding FOR mirror transaction will too. That could be bad if you have Max Task (MXT) set low and have a lot of long-running transactions. You might run out of slots and have tasks queue up. It definitely means that the system programmer must scan to see how close your regions run to Max Task.

The System Definition Guide concludes its section on MROFSE by saying, “It should be used with caution. For additional information, see the Performance Guide.” While I (now) agree with the statement, I could not find MROFSE referenced in our on-line copy of the Performance Guide. In fairness, and in retrospect, the entries in the manuals do talk about the kinds of effects we saw. However, the entries seem a lot clearer now that I know what can happen first hand!

CONCLUSION
Before you choose to activate MROFSE:
• Make sure you have enough sessions defined in your AORs and FORs to handle your long-running transactions.

• Tell your boss the FOR response time numbers will go up a bit, but the AOR response time numbers will not.

• Make sure the AORs and FORs aren’t running close to your MXT value. If they are, you must raise it.

Look for a decrease in AOR CPU times for transactions that do a lot of function shipping and issue many syncpoints.

Paul C Gordon  
Assistant Vice President  
Bank of America (USA)  
© Xephon 2001

Displaying the autoinstall-model-name and the associated terminal-name

This is a simple program to display the autoinstall-model-name and the associated terminal-name, which can be used with CEDC or CEDA to look at its characteristics. The transaction is ‘MODE’ and the calling program is ‘CSMODEL’. This gives you the list shown in Figure 1, which will differ depending on the installation on which the transaction is running.

So you need to define the transaction ‘MODE’ in your PCT and the program ‘CSMODEL’ in your PPT. When you have done this, compile the program as a command-level program and run ‘MODE’.

CSMODEL

*ASM XOPTS(CICS,SP)  
***********************************************************************
*                                                                       *
*  PROGRAM NAME:   CSMODEL                                            *
*  FUNCTION:       THIS IS A SIMPLE PROGRAM TO DISPLAY THE AUTOINST.- *
*       MODEL-NAME AND THE ASSOCIATED TERMINAL-NAME.                  *
*                                                                    *
Figure 1: Example output
**THE TERMINAL PRESENTATION OUTPUT AREA BEGINS HERE**

---

**OUT** DS OCL1800

**HEADER LINE**

**HDRMSG** DS XL79
**NL** DS XL1
**FILL0** DS XL62
**NL0** DS XL1

**ENVIRONMENT LINE**

**CICSVER** DS CL8
**CICS** DS CL7
**SYSIDENT** DS CL8
**SYSID** DS CL4
**TRANID** DS CL10
**TRANSID** DS CL4
**PROGNA** DS CL10
**PROGRAM** DS CL8

**STARTER STARS**

**FILL1** DS XL62
**NL2** DS XL1

**1ST LINE OF DATA OUTPUT**

**CSMODEL1** DS CL8
**FILL2** DS CL2
**TERMINA1** DS CL8
**FILL3** DS CL4
**CSMODEL2** DS CL8
**FILL4** DS CL2
**TERMINA2** DS CL8
**FILL5** DS CL4
**CSMODEL3** DS CL8
**FILL6** DS CL2
**TERMINA3** DS CL8

**LIST** DS CL1500

---

**THE OUTPUT AREA ENDS HERE**

---

*HERE BEGINS THE CSECT ITSELF*

---

**CSMODEL CSECT**
**CSMODEL AMODE 31**
**CSMODEL RMODE ANY**

B START

**PROGNAME** DC CL8'CSMODEL' SET
DC CL8'&SYSDATE' EYECATCHER
DC CL8'&SYSTIME' INFORMATION
BEGIN DS ØH

*******************************************************************
** CICS BASIC CODING **
*******************************************************************
L 12,X'21C' ADDRESS CURRENT TCB
L 12,X'D0'(,12) ADDRESS TCB EXTN
L 12,X'14'(,12) ADDRESS AFCX
L 12,X'0B'(,12) ADDRESS THE CSA
L 5,X'C8'(,12) ADDRESS THE CSAOPFLA
L 5,X'1C0'(,5) ADDRESS THE STATIC STOR AREA (SSA)
L 5,X'14'(,5) ADDRESS THE TMP SSA (SSATMP/TMS)

*******************************************************************
CICS41Ø DS ØH
CLI X'9F'(12),X'41' CICS 4.1.Ø ?
BNE CICS51Ø NO CHECK NEXT
MVC CICS,=C'CICS41Ø' MOVE VER/REL INTO OUTPUT FIELD
L 5,X'1AC'(,5) ADDRESS THE TMSCATTER TABLE 4.1.Ø
B CONTINUE GO AHEAD

CICS51Ø DS ØH
CLI X'9F'(12),X'51' CICS 5.1.Ø ?
BNE CICS52Ø NO TRY NEXT
MVC CICS,=C'CICS51Ø' MOVE VER/REL INTO OUTPUT FIELD
L 5,X'1E8'(,5) ADDRESS THE TMSCATTER TABLE 5.1.Ø
B CONTINUE GO AHEAD

CICS52Ø DS ØH
CLI X'9F'(12),X'52' CICS 5.2.Ø ?
BNE CICS53Ø CICS VERSION/RELEASE NOT SUPPORTED
MVC CICS,=C'CICS52Ø' MOVE VER/REL INTO OUTPUT FIELD
L 5,X'1E8'(,5) ADDRESS THE TMSCATTER TABLE 5.2.Ø
B CONTINUE GO AHEAD

CICS53Ø DS ØH
CLI X'9F'(12),X'53' CICS 5.3.Ø ?
BNE ERRMSG1 CICS VERSION/RELEASE NOT SUPPORTED
MVC CICS,=C'CICS53Ø' MOVE VER/REL INTO OUTPUT FIELD
L 5,X'1E8'(,5) ADDRESS THE TMSCATTER TABLE 5.3.Ø

*******************************************************************
CONTINUE DS ØH
L 7,X'24'(,5) ADDRESS THE AITM DIR ELEMENT
L 6,X'0'(,7) ADDRESS THE 1ST MODEL ELEMENT
LA 5,LIST POINT TO LIST

LOOP DS ØH
LA 4,3 SET COUNT TO 3

NEXT DS ØH
MVC Ø(8,5),Ø(6) MOVE CSMODELNAME TO LIST
CLI X'9F'(12),X'53' CICS 5.3.Ø ?
BNE PRE53Ø
MVC 12(4,5),X'2E'(6) MOVE TERMID TO LIST CICS53Ø
B GOON
PRE5Ø  DS  ØH
  MVC  12(4,5),X'2D'(6)  MOVE TERMID TO LIST PRE CICS5Ø
GOON  DS  ØH
  LR  8,5  SAVE REG5 VALUE
  L  7,X'10'(.7)  LOAD NEXT MODEL ELEMENT
  LTR  7,7  IS THERE A NEXT ENTRY ?
  BZ  SENDLIST  NO, PUT OUT LIST
  L  6,X'Ø'(.7)  ADDRESS NEXT MODEL DEFINITION
  LA  5,22(.5)  POINT TO NEXT HORIZ BUFFER LOC
  BCT  4,NEXT  PUT OUT 3 COLUMNS
  MVI  17(B),X'15'  MOVE NL FOR THIS LINE
  LA  5,18(,8)  POINT TO NEXT LINE LOC Ø
B  LOOP  
*******************************************************************
ERRMSG1 DS  ØH
  EXEC CICS SEND TEXT FROM(MSGDAT1) ERASE FREEKB
  B  RETURN
*******************************************************************
SENDLIST DS  ØH
  MVI  17(B),X'15'  MOVE FINAL NL TO BUFFER
  EXEC CICS ASSIGN SYSID(SYSID) PROGRAM(PROGRAM)
  MVC  TRANSID,EIBTRNID
  EXEC CICS SEND TEXT FROM(OUT) ERASE FREEKB PAGING
*******************************************************************
RETURN DS  ØH
  EXEC CICS RETURN
*******************************************************************
START DS  ØH
*******************************************************************
*    LOAD DFHEISTG USER FIELDS                                  *
*******************************************************************
  LA  5,LIST
  MVC  Ø(255,5),BLANKS
  MVC  CICSVER,=C'CICSVER='
  MVC  SYSIDENT,=C'SYSID= '
  MVC  TRANID,=C'TRANID='
  MVC  PROGNA,=C'PROGRAM='
  MVC  CSMODEL1,=C'AI-MODEL'
  MVC  TERMINA1,=C'TERMINAL'
  MVC  CSMODEL2,=C'AI-MODEL'
  MVC  TERMINA2,=C'TERMINAL'
  MVC  CSMODEL3,=C'AI-MODEL'
  MVC  TERMINA3,=C'TERMINAL'
  MVI  NL,X'15'
  MVI  NL1,X'15'
  MVI  NL2,X'15'
  MVI  NL3,X'15'
  MVI  NL4,X'15'
  MVI  FILLØ,C'-'
Yet another cold start next time – revisited

Since my article was published in *CICS Update* Issue 182, January 2001, I have corrected an error I made in the first version of the program.

This error regards the management of the CICS TS 1.3 global catalog. Instead of checking for four characters in the CICS TS 1.3 global catalog, the program must check for only one character because, in certain cases, CICS writes its applid shifted to the right by two or three characters so it hides a portion of the checking area.

The differences are shown below.

Old version:

```assembly
MVC FILL0+1,FILL0
MVI NL0,X'15'
MVC FILL1,FILL0
MVC HDRMSG,MSGDAT
B BEGIN

***********************************************************************
*          CONSTANTS                                                  *
***********************************************************************
MSGDAT   DC   CL79'AUTOINSTALL MODEL TO TERMINAL X-REFERENCE   '
MSGDAT1  DC   CL80'CICS VERSION/RELEASE NOT SUPPORTED !         '
BLANKS   DC   CL256''
END   CSMODEL
```

Claus Reis  
*CICS Systems Programmer*  
*Nuernberger Lebensversicherung AG (Germany)*  © Xephon 2001
New version:

```
Ø3 FILLER PIC X(19).
Ø3 GCD-KEYPOINT-TS13 PIC X(1).
   88 GCD-WARMKEYP-TS13 VALUE ' '.
   * WHAT YOU SEE   ' ' *
   * WHAT YOU WRITE X'Ø  *
   * WITH HEX ON     3' *
     88 GCD-EMERKEYP-TS13 VALUE 'E'.
   * WHAT YOU SEE   ' ' *
   * WHAT YOU WRITE X'Ø  *
   * WITH HEX ON     4' *
```

Editor’s note: we also received the following e-mail:

The article *Yet another cold start next time* contains an error in the code – the author has assumed that the field for the warm keypoint indicator in the CTS 1.3 GCD is 000003. This is because his Netid and Applid take up only 13 characters! Other sites might use the full 17 characters. So, the code should only check for 03 at offset 20 rather than 000003 at offset 18, because the 0000 may be part of the applid! as in NETIDXYZ.CICSABCD.

---

Gianluca Bonzano (Italy)
J P Lemon (UK) © Xephon 2001

---

CICS dynamic hardcopy printer assignment – part 2

This month we conclude the code that allows you to dynamically change the hardcopy assignment for your users.

MSYS160.BMS

```
MSYS16Ø DFHMSD TYPE=DSECT,LANG=COBOL,MODE=INOUT, -
   MAPATTS=(COLOR, -
     HILIGHT, -
     PS, -
     VALIDN), -
   STORAGE=AUTO, -
   TIOAPFX=NO
```
DFHMDF POS=(06,001),
LENGTH=079,
INITIAL='---------------------------------------------------------------',
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD1 DFHMDF POS=(07,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR1 DFHMDF POS=(07,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(07,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG1 DFHMDF POS=(07,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(07,018),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD2 DFHMDF POS=(08,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR2 DFHMDF POS=(08,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(08,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG2 DFHMDF POS=(08,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
DFHMDF POS=(10,018),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD5 DFHMDF POS=(11,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR5 DFHMDF POS=(11,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(11,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG5 DFHMDF POS=(11,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(11,018),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD6 DFHMDF POS=(12,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR6 DFHMDF POS=(12,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(12,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG6 DFHMDF POS=(12,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(12,018),
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD11  DFHMDF POS=(17,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR11  DFHMDF POS=(17,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(17,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG11  DFHMDF POS=(17,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(17,018),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

CMD12  DFHMDF POS=(18,001),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(UNPROT,BRT,FSET)

PAR12  DFHMDF POS=(18,003),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(18,008),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP)

ARG12  DFHMDF POS=(18,013),
LENGTH=004,
COLOR=DEFAULT,
HILIGHT=OFF,
ATTRB=(ASKIP,BRT)

DFHMDF POS=(18,018),
LENGTH=001,
COLOR=DEFAULT,
HILIGHT=OFF,
### OSYS160.PRO

```plaintext
***    Ffonam ***
IDENTIFICATION DIVISION.
PROGRAM-ID.  OSYS160.

* Kurzbeschreibung: TERMPRT-Dataset display and management *

* Transaction: SY16 bzw. SY17 (display only) *

* Hauptmaske: MSYS160 *

* Function: *

* Forward and backward browsing (scrolling) is possible within the displayed table. *

* Delete and Modify functions are provided for a line when the cursor is placed over it. After a line in the table is selected, it is changed to the appropriate line. *

* PGM-RESOURCE *

* Data *

<table>
<thead>
<tr>
<th>Typ</th>
<th>Sel</th>
<th>Ins</th>
<th>Upd</th>
<th>Del</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

* Copy-Files *

ENVIRONMENT DIVISION.
```

---

**NAME** | **TYPE** | **GROUP** | **DATE** | **TIME**
---|---|---|---|---
MSYS160 | PROGRAM | BASTPRT | 97.062 | 14.05.58
OSYS160 | PROGRAM | BASTPRT | 97.062 | 14.05.58
SY16 | TRANSACTION | BASTPRT | 97.062 | 16.15.15
SY17 | TRANSACTION | BASTPRT | Display only | 97.063 | 09.56.09
TERMPRT | FILE | BASTPRT1 | 97.297 | 13.35.32
TERMPRT | FILE | BASTPRT2 | 97.059 | 10.58.20
TERMPRT | FILE | BASTPRT3 | 97.063 | 09.57.36
WORKING-STORAGE SECTION.

*1 LEER       PIC X(1) VALUE SPACE.
*1 ZEILE      PIC S9(4) COMP.
*1 ZEILEN-INDEX PIC S9(4) COMP.
*1 KOMMANDO   PIC X.
 *
*1 COMBER.
   *5 DSZ-C       PIC X.
   *5 WAHL-C      PIC X.
   *5 FIRSTKEY-C  PIC X(4).
   *5 LASTKEY-C   PIC X(4).
   *5 PREVKEY-C   PIC X(4).
   *5 NEXTKEY-C   PIC X(4).
   *5 FILLER     OCCURS 13.
     *10 PAR-C    PIC X(4).
     *10 ARG-C    PIC X(4).

COPY DFHAID.
COPY DFHBMSCA.
COPY MSYS160.

*1 FILLER REDEFINES MSYS160I.
   *5 FILLER       PIC X(85).
   *5 FILLER     OCCURS 13.
     *10 CMDL     PIC S9(4) COMP.
     *10 CMDA     PIC X.
     *10 CMD      PIC X.
     *10 FILLER   PIC X(3).
     *10 PAR      PIC X(4).
     *10 FILLER   PIC X(3).
     *10 ARG      PIC X(4).

*1 TERMPRT-SATZ.
   *5 TERMPRT-TERM PIC X(4).
   *5 TERMPRT-SPACE PIC X(1).
   *5 TERMPRT-PRT  PIC X(4).
*1 WRK-APPLID.
   *5 FILLER     PIC X(4).
   *5 WRK-CICS   PIC X(4).

$INCLUDE CICS CODE

LINKAGE SECTION.

*1 DFHCOMMAREA PIC X(122).

PROCEDURE DIVISION.

STEUER SECTION.
* MOVE LOW-VALUE TO MSYS1600.
* EXEC CICS HANDLE ABEND
   LABEL(STEUER-ENDE)
   END-EXEC.
* EXEC CICS IGNORE CONDITION
   NOTFND
   ENDFILE
   END-EXEC.
* IF EIBCALEN = Ø
   MOVE SPACES TO COMBER
   MOVE "Ø" TO DSZ-C
ELSE
   MOVE DFHCOMMAREA TO COMBER.
* IF DSZ-C = "Ø"
   PERFORM MAP-AUFBAU
ELSE
   MOVE LOW-VALUES TO MSYS1600
   MOVE SPACES TO MELDUNGO
   PERFORM ANZEIGEN.
* EXEC CICS RETURN
  TRANSID(EIBTRNID)
  COMMAREA(COMBER)
  LENGTH(122)
  END-EXEC.
* STEUER-ENDE.
  PERFORM CLEAR.
  EXEC CICS RETURN
  END-EXEC.
  GOBACK.
****************************************************************
MAP-AUFBAU SECTION.
* EXEC CICS ASKTIME
  ABSTIME(DATUMO)
  END-EXEC.
EXEC CICS FORMATTIME
  ABSTIME(DATUMO)
  DDMYYY (DATUMO)
  DATESEP("/")
  END-EXEC.
EXEC CICS ASSIGN
  APPLID(WRK-APPLID)
  END-EXEC.
MOVE WRK-CICS TO ACICSO.
MOVE EIBTRMID TO TRMIDO.
*
IF EIBTRNID = "SY16"
   MOVE DFHUNIMD TO ERFARGA
   MOVE "( Commands : L = Delete / A = Modify )" TO ATEXTO
ELSE
   MOVE SPACE TO ATEXTO
   MOVE DFHBMASF TO ERFARGA.
   MOVE DFHUNIMD TO ERFPARA
   MOVE -1 TO ERFPARL.
*
MOVE SPACES TO TERMPRT-TERM.
PERFORM START-BROWSING.
*
IF CICS-NOTFND
   PERFORM BLAETTERN-SPERREN
   PERFORM LEERE-ZEILE
       VARYING ZEILEN-INDEX FROM 1 BY +1
       UNTIL (ZEILEN-INDEX > 13)
   MOVE "TABELLE-LEER" TO MELDUNGO
ELSE
   MOVE "ANFG" TO PREVKEY-C
   PERFORM VORWAERTS-BLAETTERN
   PERFORM END-BROWSING.
*
IF WAHL-C = "C"
   MOVE PAR-C (1) TO ERFPARO
   MOVE ARG-C (1) TO ERFARGO.
*
EXEC CICS SEND
   MAP("MSYS16Ø")
   ERASE
   CURSOR
END-EXEC.
*
MOVE "1" TO DSZ-C.
*
MAP-AUFBAU-ENDE.
EXIT.
****************************************************************
ANZEIGEN SECTION.
*
IF EIBAID = DFHPF3 OR DFHPF12 OR DFHCLEAR
   PERFORM CLEAR
   EXEC CICS RETURN
END-EXEC.
*
IF EIBAID = DFHPF8
IF NEXTKEY-C = "ENDE"
  MOVE "End of data" TO MELDUNGO
ELSE
  IF NEXTKEY-C = "NORM"
    MOVE LASTKEY-C TO TERMPRT-TERM
    PREVKEY-C
    PERFORM START-BROWSING
    PERFORM VORWAERTS-BLAETTTERN
    PERFORM END-BROWSING
    IF EIBTRNID = "SY16"
      MOVE PAR-C (1) TO ERFPARO
      MOVE SPACES TO ERFARGO
    ELSE
      CONTINUE
    ELSE
      MOVE NEXTKEY-C TO LASTKEY-C
      MOVE "NORM" TO NEXTKEY-C
  ELSE
    IF EIBAID = DFHPF7
      IF PREVKEY-C = "ANFG"
        MOVE "Start of data" TO MELDUNGO
      ELSE
        IF PREVKEY-C = "NORM"
          MOVE FIRSTKEY-C TO TERMPRT-TERM
          PERFORM START-BROWSING
          PERFORM RUECKWAERTS-BLAETTTERN
          PERFORM END-BROWSING
          IF EIBTRNID = "SY16"
            MOVE PAR-C (ZEILE) TO ERFPARO
            MOVE SPACES TO ERFARGO
          ELSE
            CONTINUE
          ELSE
            MOVE PREVKEY-C TO FIRSTKEY-C
            MOVE "NORM" TO PREVKEY-C
        ELSE
          IF EIBAID = DFHENTER
            EXEC CICS RECEIVE
            MAP("MSYS16Ø")
            END-EXEC
          * 
          IF EIBTRNID = "SY16"
            PERFORM EINGABE-AUSWERTEN
          ELSE
            PERFORM ANZEIGE-AUSWERTEN
          ELSE
            MOVE "Key not pressed" TO MELDUNGO.
          * 
          MOVE -1 TO ERFPARL.
          EXEC CICS SEND
* MAP("MSYS16O")
  CURSOR
  DATAONLY
  END-EXEC.
*
ANZEIGEN-ENDE.
EXIT.
****************************************************************
ANZEIGE-AUSWERTEN SECTION.
*
  IF ERFPARA = DFHBMEOF
    MOVE SPACES TO ERFPARI.
  *
  IF ERFPARI NOT = SPACES
    MOVE ERFPARI TO TERMPRT-TERM
    MOVE ERFPARI TO FIRSTKEY-C
    MOVE "NORM" TO PREVKEY-C
    PERFORM SEITE-AUFFRISCHEN.
ANZEIGE-AUSWERTEN-ENDE.
EXIT.
****************************************************************
EINGABE-AUSWERTEN SECTION.
*
  IF ERFPARA = DFHBMEOF
    MOVE SPACES TO ERFPARI.
  *
  IF ERFARGA = DFHBMEOF
    MOVE SPACES TO ERFARGI.
  *
  MOVE SPACE TO KOMMANDO.
  PERFORM CMD-SUCHEN
    VARYING ZEILEN-INDEX FROM 13 BY -1
    UNTIL (ZEILEN-INDEX < 1).
  *
  IF KOMMANDO = "L"
    PERFORM SATZ-LOESCHEN
    GO TO EINGABE-AUSWERTEN-ENDE.
  *
  IF KOMMANDO = "A"
    MOVE PAR-C (ZEILE) TO ERFPARO
    MOVE ARG-C (ZEILE) TO ERFARGO
    GO TO EINGABE-AUSWERTEN-ENDE.
  *
  IF KOMMANDO NOT = SPACE
    MOVE "INCORRECT-FUNCTION" TO MELDUNGO
    GO TO EINGABE-AUSWERTEN-ENDE.
  IF (ERFPARI NOT = SPACES) AND (ERFARGI = SPACES)
    MOVE ERFPARI TO TERMPRT-TERM
    MOVE ERFPARI TO FIRSTKEY-C
    MOVE "NORM" TO PREVKEY-C
    PERFORM SEITE-AUFFRISCHEN
GO TO EINGABE-AUSWERTEN-ENDE.
IF (ERFPARI = SPACES) OR (ERFARGI = SPACES)
  MOVE "ZU-WENIG-DATEN" TO MELDUNGO
ELSE
  PERFORM SATZ-ERFASSEN
  PERFORM SEITE-AUFFRISCHEN
  MOVE "DATEN-GESICHERT" TO MELDUNGO.

EINGABE-AUSWERTEN-ENDE.
EXIT.
****************************************************************
CMD-SUCHEN SECTION.
*
INSPECT CMD (ZEILEN-INDEX) REPLACING ALL "_" BY SPACE.
*
IF (CMD (ZEILEN-INDEX) NOT = SPACES) AND
  (CMDA (ZEILEN-INDEX) NOT = DFHBMEOF)
  MOVE CMD (ZEILEN-INDEX) TO KOMMANDO
  MOVE ZEILEN-INDEX TO ZEILE.
*
  MOVE "_" TO CMD (ZEILEN-INDEX).
*
CMD-SUCHEN-ENDE.
EXIT.
****************************************************************
SATZ-ERFASSEN SECTION.
*
  MOVE ERFPARI  TO TERMPRT-TERM.
  MOVE ERFARGI  TO TERMPRT-PRT.
*
EXEC CICS DELETE
  DATASET("TERMPRT")
  RIDFLD(TERMPRT-TERM)
END-EXEC.
*
EXEC CICS WRITE
  DATASET("TERMPRT")
  RIDFLD(TERMPRT-TERM)
  FROM(TERMPRT-SATZ)
END-EXEC.
*
  MOVE SPACES TO ERFARGO.
  IF FIRSTKEY-C = SPACES
    MOVE TERMPRT-TERM TO FIRSTKEY-C.
  *
SATZ-ERFASSEN-ENDE.
EXIT.
****************************************************************
SEITE-AUFFRISCHEN SECTION.
*
IF FIRSTKEY-C = SPACES
  PERFORM LEERE-ZEILE
  VARYING ZEILEN-INDEX FROM 1 BY +1
  UNTIL (ZEILEN-INDEX > 13)
  PERFORM BLAETTERN-SPERREN
  GO TO SEITE-AUFFRISCHEN-ENDE.
MOVE FIRSTKEY-C TO TERMPRT-TERM
PERFORM START-BROWSING
IF CICS-NOTFND
  MOVE "Terminal not available" TO MELDUNG
  GO TO SEITE-AUFFRISCHEN-ENDE.
PERFORM VORWAERTS-BLAETTERN.
PERFORM END-BROWSING.
*
SEITE-AUFFRISCHEN-ENDE.
EXIT.
****************************************************************
START-BROWSING SECTION.
*
  EXEC CICS STARTBR
      DATASET("TERMPRT")
      RIDFLD(TERMPRT-TERM)
  END-EXEC.
  MOVE EIBRCODE TO CICS-CODE.
*
START-BROWSING-ENDE.
EXIT.
****************************************************************
END-BROWSING SECTION.
*
  EXEC CICS ENDBR
      DATASET("TERMPRT")
  END-EXEC.
*
END-BROWSING-ENDE.
EXIT.
****************************************************************
VORWAERTS-BLAETTERN SECTION.
*
  EXEC CICS READNEXT
      DATASET("TERMPRT")
      RIDFLD(TERMPRT-TERM)
      INTO(TERMPRT-SATZ)
  END-EXEC.
  MOVE EIBRCODE TO CICS-CODE.
*
  MOVE TERMPRT-TERM TO FIRSTKEY-C.
*
  PERFORM VB-NAECHSTER-SATZ
      VARYING ZEILEN-INDEX FROM 1 BY +1
UNTIL (ZEILEN-INDEX > 13) OR CICS-ENDFILE.

*  
IF (ZEILEN-INDEX > 13)
MOVE "NORM" TO NEXTKEY-C
MOVE "NORM" TO PREVKEY-C
ELSE
IF CICS-ENDFILE
MOVE "ENDE" TO NEXTKEY-C
MOVE "End of data" TO MELDUNGO
ELSE
MOVE TERMPRT-TERM TO NEXTKEY-C.
*

IF PREVKEY-C NOT = "ANFG"
MOVE "NORM" TO PREVKEY-C.
*

PERFORM LEERE-ZEILE
  VARYING ZEILEN-INDEX FROM ZEILEN-INDEX
       BY +1
       UNTIL (ZEILEN-INDEX > 13).
*

VORWAERTS-BLAETTERN-ENDE.
EXIT.
****************************************************************
VB-NAECHSTER-SATZ SECTION.
*
MOVE TERMPRT-TERM TO LASTKEY-C.
PERFORM MAP-ZEILE.
*
EXEC CICS READNEXT
  DATASET("TERMPRT")
  RIDFLD(TERMPRT-TERM)
  INTO(TERMPRT-SATZ)
END-EXEC.
MOVE EIBRCODE TO CICS-CODE.
*
VB-NAECHSTER-SATZ-ENDE.
EXIT.
****************************************************************
RUECKWAERTS-BLAETTERN SECTION.
*
EXEC CICS READPREV
  DATASET("TERMPRT")
  RIDFLD(TERMPRT-TERM)
  INTO(TERMPRT-SATZ)
END-EXEC.
MOVE EIBRCODE TO CICS-CODE.
*

MOVE TERMPRT-TERM TO LASTKEY-C.
*

PERFORM RB-NAECHSTER-SATZ
   VARYING ZEILEN-INDEX FROM 13 BY -1
   UNTIL (ZEILEN-INDEX < 1) OR
         CICS-ENDFILE.
*
   IF (ZEILEN-INDEX < 1)
      MOVE "NORM" TO PREVKEY-C
   ELSE
      IF CICS-ENDFILE
         MOVE "ANFG" TO PREVKEY-C
      MOVE "Start of data" TO MELDUNGO
   ELSE
      MOVE TERMPRT-TERM TO PREVKEY-C.
*
   MOVE "NORM" TO NEXTKEY-C.
   COMPUTE ZEILE = ZEILEN-INDEX + 1.
*
   PERFORM LEERE-ZEILE
      VARYING ZEILEN-INDEX FROM ZEILEN-INDEX
         BY -1
         UNTIL (ZEILEN-INDEX < 1).
*
   RUECKWAERTS-BLAETTERN-ENDE.
   EXIT.
****************************************************************
RB-NAECHSTER-SATZ SECTION.
*
   MOVE TERMPRT-TERM TO FIRSTKEY-C.
   PERFORM MAP-ZEILE.
*
   EXEC CICS READPREV
      DATASET("TERMPRT")
      RIDFLD(TERMPRT-TERM)
      INTO(TERMPRT-SATZ)
   END-EXEC.
   MOVE EIBRCODE TO CICS-CODE.
*
   RB-NAECHSTER-SATZ-ENDE.
   EXIT.
****************************************************************
MAP-ZEILE SECTION.
*
   MOVE SPACE TO CMD (ZEILEN-INDEX).
*
   IF EIBTRNID = "SY16"
      MOVE "_" TO CMD (ZEILEN-INDEX)
      MOVE DFHUNIMD TO CMDA (ZEILEN-INDEX)
   ELSE
      MOVE DFHBMASF TO CMDA (ZEILEN-INDEX).
*
MOVE TERMPRT-TERM TO PAR (ZEILEN-INDEX)  
PAR-C (ZEILEN-INDEX).  
MOVE TERMPRT-PRT TO ARG (ZEILEN-INDEX)  
ARG-C (ZEILEN-INDEX).  
*  
MAP-ZEILE-ENDE.  
EXIT.  
***************************************************************************  
LEERE-ZEILE SECTION.  
*  
MOVE SPACE TO CMD (ZEILEN-INDEX).  
MOVE DFHBMASF TO CMDA (ZEILEN-INDEX).  
*  
MOVE SPACES TO PAR (ZEILEN-INDEX)  PAR-C (ZEILEN-INDEX)  
ARG (ZEILEN-INDEX)  ARG-C (ZEILEN-INDEX).  
*  
LEERE-ZEILE-ENDE.  
EXIT.  
***************************************************************************  
SATZ-LOESCHEN SECTION.  
*  
MOVE PAR-C (ZEILE) TO ERFPARO  
MOVE ARG-C (ZEILE) TO ERFARGO  
MOVE PAR-C (ZEILE) TO TERMPRT-TERM.  
*  
EXEC CICS DELETE  
DATASET("TERMPRT")  
RIDFLD(TERMPRT-TERM)  
END-EXEC.  
*  
IF FIRSTKEY-C = LASTKEY-C  
MOVE SPACES TO FIRSTKEY-C LASTKEY-C.  
PERFORM SEITE-AUFFRISCHEN  
MOVE "SATZ-GELOESCHT" TO MELDUNGO.  
*  
SATZ-LOESCHEN-ENDE.  
EXIT.  
***************************************************************************  
BLAETTERN-SPERREN SECTION.  
*  
MOVE "ANFG" TO PREVKEY-C.  
MOVE "ENDE" TO NEXTKEY-C.  
MOVE SPACES TO FIRSTKEY-C LASTKEY-C.  
*  
BLAETTERN-SPERREN-ENDE.  
EXIT.  
***************************************************************************  
CLEAR SECTION.  
*  
EXEC CICS SEND CONTROL  
ERASE
You will find ‘INCLUDE CICSCODE’ in the OSYS160 program. We do our programming on our VM system and include files from there. All you have to do is replace this include card with the dataset CICSCODE.INC. This contains certain CICS control characters and return codes.

The OSYS160 program also contains two copies. These are actually the standard IBM copybooks that are distributed with CICS 4.1 and are only modified because we use quotes instead of the apostrophe in our COBOL programs. In case you do not already have these copybooks on your system I have included them also. Note that I have changed them so that the hexadecimal characters are not in character form (which causes problems when transferring between PCs and mainframes) but are entered in the hexadecimal format VALUE X"nn". The two copy books are

CICSCODE INC

*D-------------------------------------------------------------*
*D    Reference   : EIBRCODE - Returncode of the last executed CICS/VS-Commands *
*D    Function    : Pick up the "EIBRCODE" for explicit Error processing in CICS programs *
*D-------------------------------------------------------------*
Ø1 CICS-CODE.
Ø5 FILLER PIC X.
88 CICS-OK  VALUE LOW-VALUE.
88 CICS-NOTOPEN  VALUE X"0C".
88 CICS-NOTFND  VALUE X"81".
88 CICS-ENDFILE  VALUE X"0F".
88 CICS-ENDDATA  VALUE X"01".
88 CICS-MAPFAIL  VALUE X"04".
88 CICS-INVREQ  VALUE X"E0".
88 CICS-QIDERR  VALUE X"02".
88 CICS-DUPKEY  VALUE X"84".
88 CICS-DUPREC  VALUE X"82".
Ø5 FILLER PIC X(5).

DFHAID.COP

* 5685-083
* COPYRIGHT = NONE
*
Ø1 DFHAID.
 Ø2 DFNULL PIC X VALUE IS X"ØØ".
 Ø2 DFHENTER PIC X VALUE IS X"7D".
 Ø2 DFHCLEAR PIC X VALUE IS X"6D".
 Ø2 DFHCLRIP PIC X VALUE IS X"6A".
 Ø2 DFHPEN PIC X VALUE IS "=".
 Ø2 DFHOPID PIC X VALUE IS "W".
 Ø2 DFHMSRE PIC X VALUE IS "X".
 Ø2 DFHSTRF PIC X VALUE IS X"88".
 Ø2 DFHTRIG PIC X VALUE IS QUOTE.
 Ø2 DFHPA1 PIC X VALUE IS "%".
 Ø2 DFHPA2 PIC X VALUE IS ">".
 Ø2 DFHPA3 PIC X VALUE IS ",".
 Ø2 DFHPF1 PIC X VALUE IS "1".
 Ø2 DFHPF2 PIC X VALUE IS "2".
 Ø2 DFHPF3 PIC X VALUE IS "3".
 Ø2 DFHPF4 PIC X VALUE IS "4".
 Ø2 DFHPF5 PIC X VALUE IS "5".
 Ø2 DFHPF6 PIC X VALUE IS "6".
 Ø2 DFHPF7 PIC X VALUE IS "7".
 Ø2 DFHPF8 PIC X VALUE IS "8".
 Ø2 DFHPF9 PIC X VALUE IS "9".
 Ø2 DFHPF10 PIC X VALUE IS X"7A".
 Ø2 DFHPF11 PIC X VALUE IS X"7B".
 Ø2 DFHPF12 PIC X VALUE IS X"7C".
 Ø2 DFHPF13 PIC X VALUE IS "A".
 Ø2 DFHPF14 PIC X VALUE IS "B".
 Ø2 DFHPF15 PIC X VALUE IS "C".
 Ø2 DFHPF16 PIC X VALUE IS "D".
 Ø2 DFHPF17 PIC X VALUE IS "E".
 Ø2 DFHPF18 PIC X VALUE IS "F".
 Ø2 DFHPF19 PIC X VALUE IS "G".
 Ø2 DFHPF20 PIC X VALUE IS "H".
 Ø2 DFHPF21 PIC X VALUE IS "I".
 Ø2 DFHPF22 PIC X VALUE IS X"4A".
 Ø2 DFHPF23 PIC X VALUE IS X"4B".
 Ø2 DFHPF24 PIC X VALUE IS "<".

DFHBMSCA.COP

* 5685-083
* COPYRIGHT = NONE
*
*
01  DFHBMSCA.
02  DFHBMPEM  PICTURE X  VALUE IS 'X"19"'.
02  DFHBMPNL  PICTURE X  VALUE IS 'X"15"'.
02  DFHBMPFF  PICTURE X  VALUE IS 'X"0C"'.
02  DFHBMPCR  PICTURE X  VALUE IS 'X"00"'.
02  DFHBMASK  PICTURE X  VALUE IS '0'.
02  DFHBMUNP  PICTURE X  VALUE IS 'X"40"'.
02  DFHBMUNN  PICTURE X  VALUE IS 'X"50"'.
02  DFHBMPRO  PICTURE X  VALUE IS 'X"60"'.
02  DFHMBRKY  PICTURE X  VALUE IS 'H'.
02  DFHBMDFR  PICTURE X  VALUE IS '/'.
02  DFHBMASF  PICTURE X  VALUE IS '1'.
02  DFHBMASB  PICTURE X  VALUE IS '8'.
02  DFHBMEOF  PICTURE X  VALUE IS 'X"80"'.
02  DFHBMCUR  PICTURE X  VALUE IS 'X"02"'.
02  DFHBMEC  PICTURE X  VALUE IS 'X"82"'.
02  DFHBMDFLG  PICTURE X.
  88  DFHERASE  VALUES ARE 'X"80"', 'X"82"'.
  88  DFHCURSR  VALUES ARE 'X"02"', 'X"82"'.
02  DFHBMDET  PICTURE X  VALUE IS 'X"FF"'.
02  DFHBMPSO-BIN  PIC 9(4) COMP VALUE 3599.
  * ABOVE VALUE 3599 = 'X"0E0F"' ADDED BY APAR PN24842
02  FILLER  REDEFINES  DFHBMPSO-BIN.
03  DFHBMPSO  PICTURE X.
03  DFHBMPSI  PICTURE X.
  02  DFHSA  PICTURE X  VALUE IS 'X"28"'.
  02  DFHCOLOR  PICTURE X  VALUE IS 'X"42"'.
  02  DFHPS  PICTURE X  VALUE IS 'X"43"'.
  02  DFHHLT  PICTURE X  VALUE IS 'X"41"'.
  02  DFH3270  PICTURE X  VALUE IS 'X"C0"'.
  02  DFHVAL  PICTURE X  VALUE IS 'A'.
  02  DFHOUTLN  PICTURE X  VALUE IS 'B'.
  02  DFHBLTREN  PICTURE X  VALUE IS 'X"46"'.
  02  DFHALL  PICTURE X  VALUE IS 'X"00"'.
  02  DFHERRR  PICTURE X  VALUE IS 'X"3F"'.
  02  DFHDFDF  PICTURE X  VALUE IS 'X"FF"'.
  02  DFHDFCOL  PICTURE X  VALUE IS 'X"00"'.
  02  DFHBLUE  PICTURE X  VALUE IS '1'.
  02  DFHRED  PICTURE X  VALUE IS '2'.
  02  DFHPINK  PICTURE X  VALUE IS '3'.
  02  DFHGREEN  PICTURE X  VALUE IS '4'.
  02  DFHTURQ  PICTURE X  VALUE IS '5'.
  02  DFHYELLOW  PICTURE X  VALUE IS '6'.
  02  DFHRNEUTR  PICTURE X  VALUE IS '7'.
  02  DFHBASE  PICTURE X  VALUE IS 'X"00"'.
  02  DFHDFHI  PICTURE X  VALUE IS 'X"00"'.
  02  DFHBLINK  PICTURE X  VALUE IS '1'.
  02  DFHREVRS  PICTURE X  VALUE IS '2'.

Simple tool to manage the data extracted from CICS CSD in a DB2 environment

When a data EXTRACT function from a CSD CICS is executed with DFH0FORC utility program – EXTRACT GROUP(group name) OBJECTS USERPROGRAM(DFH0FORC) – we obtain a sequential file that can be loaded into DB2 tables.

When an extract is carried out for a CICS group (or groups), the output
file can contain the definitions of various types of RDO resource (programs, files, typeterminals, transactions, profiles, etc), and every resource has its own format. Therefore, in order to correctly manage the import into a database environment using a standard DB2 utility to create tables and to load the tables, it is a good idea to divide the files depending on their RDO resource type.

Every record written by the Extract function is organized into columns that correspond to the DB2 columns, but every CSD resource has a different format, therefore it must be processed according to its record type.

In order to simplify this operation I have developed a REXX EXEC (CICSDB2) that carries out the following functions:

- It reads the output of the extract function.
- It writes a sysin file (DB2/SQL statements) for every RDO resource type in order to execute the DB2 Create Table utility program.
- It writes a sysin file (DB2/SQL statements) for every RDO resource type in order to execute the DB2 Load Table utility program.
- It writes a sequential file (input file) for every RDO resource type in order to execute the DB2 Load Table utility program.
- It writes and submits a job for the execution of DB2 Create Tablespace, Create Table, and Load Table utilities.

This utility has been used in the following environment:

- OS/390 1.3 and OS/390 2.6.
- CICS/ESA 4.1.0.
- DB2 4.1.0 and DB2 5.1.0.

CICSDB2 REXX EXEC

```rexx
/* REXX */
/* C-List CICSDB2. 
Called by job batch. 
Tool to manage the data extracted from CICS CSD in a
```
DB2 environment.
The functions are:
- read the output of the CICS extract function;
- write a sysin file (DB2/SQL statements) to execute the DB2 Create Tablespace utility program;
- write a sysin file (DB2/SQL statements) for every RDO resource type in order to execute the DB2 Create Table utility program;
- write a sysin file (DB2/SQL statements) for every RDO resource type in order to execute the DB2 Load Table utility program;
- write a sequential file (input file) for every RDO resource type in order to execute the DB2 Load Table utility program;
- write and submit a job for the execution of DB2 Create Tablespace, DB2 Create Table and DB2 Load Table utility; */

```
Trace ?o
PARSE ARG filein
tfile = userid()||'.CICSSVIL.DFHCSB.DB2.'
sfile = userid()||'.CICSSVIL.DFHCSB.DB2.SYSIN(' ADDRESS TSO
  dd=OUTTRAP(dd.)
  "ALLOC DA ('filein') F(FILEIN) SHR REUSE"
  dd=OUTTRAP('OFF')
  if rc ≠ Ø then do
typfunc = 'Allocfilein'
Call CX_Error_func
  say mess
  Exit
  End
  dd=OUTTRAP(dd.)
  "EXECIO * DISKR FILEIN (STEM recinp. FINIS"
  dd=OUTTRAP('OFF')
  if rc ≠ 'OFF' then do
typfunc = 'Readfilein'
Call CX_Error_func
  say mess
  Exit
  End

f1=Ø;f2=Ø;f3=Ø;f4=Ø;f5=Ø;f6=Ø;f7=Ø;f8=Ø;f9=Ø;f10=Ø;f11=Ø;f12=Ø;f13=Ø
Do i=1 to recinp.Ø
  trec = substr(recinp.i,1,4)
  if trec = 'CONN' then do
    f1 = f1 + 1
    Call Prepare_Sysin_Conn
    Iterate
  End
  if trec = 'FILE' then do
    f2 = f2 + 1
    Call Prepare_Sysin_File
    Iterate
  End
  if trec = 'LSRP' then do
    f3 = f3 + 1

```
if trec = 'MAPS' then do
  f4 = f4 + 1
  Call Prepare_Sysin_Maps
  Iterate
End

if trec = 'PART' then do
  f5 = f5 + 1
  Call Prepare_Sysin_Part
  Iterate
End

if trec = 'PROF' then do
  f6 = f6 + 1
  Call Prepare_Sysin_Prof
  Iterate
End

if trec = 'PROG' then do
  f7 = f7 + 1
  Call Prepare_Sysin_Prog
  Iterate
End

if trec = 'PTNR' then do
  f8 = f8 + 1
  Call Prepare_Sysin_Ptnr
  Iterate
End

if trec = 'SESS' then do
  f9 = f9 + 1
  Call Prepare_Sysin_Sess
  Iterate
End

if trec = 'TERM' then do
  f10 = f10 + 1
  Call Prepare_Sysin_Term
  Iterate
End

if trec = 'TRAN' then do
  f11 = f11 + 1
  Call Prepare_Sysin_Tran
  Iterate
End

if trec = 'TRCL' then do
  f12 = f12 + 1
  Call Prepare_Sysin_Trcl
  Iterate
End

if trec = 'TYPE' then do
  f13 = f13 + 1
  Call Prepare_Sysin_Type
Iterate
End
Call Prepare_Sysin_Tables
Do w=1 to 13
   sysin.w = ''
End
if f1 > Ø then do
   nrec = f1
   conn.Ø=f1; do x=1 to conn.Ø; rec.x = conn.x; end
   Wfile = tfile||'CONN'
   Call Write_type_file
   sysin.1 = sfile||'CONN'
end
if f2 > Ø then do
   nrec = f2
   file.Ø=f2; do x=1 to file.Ø; rec.x = file.x; end
   Wfile = tfile||'FILE'
   Call Write_type_file
   sysin.2 = sfile||'FILE'
end
if f3 > Ø then do
   nrec = f3
   lsrp.Ø=f3; do x=1 to lsrp.Ø; rec.x = lsrp.x; end
   Wfile = tfile||'LSRP'
   Call Write_type_file
   sysin.3 = sfile||'LSRP'
end
if f4 > Ø then do
   nrec = f4
   maps.Ø=f4; do x=1 to maps.Ø; rec.x = maps.x; end
   Wfile = tfile||'MAPS'
   Call Write_type_file
   sysin.4 = sfile||'MAPS'
end
if f5 > Ø then do
   nrec = f5
   part.Ø=f5; do x=1 to part.Ø; rec.x = part.x; end
   Wfile = tfile||'PART'
   Call Write_type_file
   sysin.5 = sfile||'PART'
end
if f6 > Ø then do
   nrec = f6
   prof.Ø=f6; do x=1 to prof.Ø; rec.x = prof.x; end
   Wfile = tfile||'PROF'
   Call Write_type_file
   sysin.6 = sfile||'PROF'
end
if f7 > Ø then do
nrec = f7
prog.Ø=f7; do x=1 to prog.Ø; rec.x = prog.x; end
Wfile = tfile||'PROG'
Call Write_type_file
sysin.7 = sfile||'PROG')
end
if f8 > Ø then do
nrec = f8
ptnr.Ø=f8; do x=1 to ptnr.Ø; rec.x = ptnr.x; end
Wfile = tfile||'PTNR'
Call Write_type_file
sysin.8 = sfile||'PTNR')
end
if f9 > Ø then do
nrec = f9
sess.Ø=f9; do x=1 to sess.Ø; rec.x = sess.x; end
Wfile = tfile||'SESS'
Call Write_type_file
sysin.9 = sfile||'SESS')
end
if f10 > Ø then do
nrec = f10
term.Ø=f10; do x=1 to term.Ø; rec.x = term.x; end
Wfile = tfile||'TERM'
Call Write_type_file
sysin.10 = sfile||'TERM')
end
if f11 > Ø then do
nrec = f11
tran.Ø=f11; do x=1 to tran.Ø; rec.x = tran.x; end
Wfile = tfile||'TRAN'
Call Write_type_file
sysin.11 = sfile||'TRAN')
end
if f12 > Ø then do
nrec = f12
trcl.Ø=f12; do x=1 to trcl.Ø; rec.x = trcl.x; end
Wfile = tfile||'TRCL'
Call Write_type_file
sysin.12 = sfile||'TRCL')
end
if f13 > Ø then do
nrec = f13
type.Ø=f13; do x=1 to type.Ø; rec.x = type.x; end
Wfile = tfile||'TYPE'
Call Write_type_file
sysin.13 = sfile||'TYPE')
end
"Newstack"
Queue '//'userid()'C JOB (LTYZ11ØØ),CLASS=S,MSGCLASS=X,MSGLEVEL=(1,1),'

Queue '//' REGION=8M, TYPRUN=HOLD, NOTIFY='userid()
Queue '//'*
Queue '//' Create DB2 Tablespace/tables and Load DB2 tables.'
Queue '//'*
Queue '//'DB2PROC JCLLIB ORDER=(DSNS.DB2.PROCLIB)' 
Queue '//'JOBLIB DD DSN=SYS1.DSNS.DB2.SDSNLOAD, DISP=SHR
Queue '//'TABCRE EXEC PGM=IKJEFTØ1,DYNAMNBR=20*
Queue '//'SYSTSPRT DD SYOUT='*
Queue '//'SYSTSIN DD '*
Queue 'DSN SYSTEM(DSNS)' 
Queue 'RUN PROGRAM(DSNTIAD) PLAN(DSNTIA41)' -
Queue 'LIB('DSNS.DB2.RUNLIB.LOAD')"
Queue '//'SYSPRINT DD SYOUT='*
Queue '//'SYSUDUMP DD SYOUT='*
Queue '//'SYSSIN DD DSN='sfile'TABLES), DISP=SHR'
Do w=1 to 13
  if sysin.w = '' then nop
  else Queue '//' DD DSN='sysin.w', DISP=SHR'
End
Queue '//'*
Queue '//'*
Do w=1 to 13
  if sysin.w = '' then nop
  else do
    Queue '//'*
    Queue '//'LOAD"w" EXEC
    DSNUPROD,PARM='DSNS,EM.CICSRDO."w",COND=(Ø,NE,TABCRE)'
    linput = length(sysin.w) - 1
    fsysin = substr(sysin.w,1,linput)||'#)'
    tf = substr(fsysin,(linput-3),4)
    finput = tfile||tf
    Queue '//SYSRECØØ DD DISP=SHR, DSN='finput
    Queue '//SYSIN DD DISP=SHR, DSN='fsysin
  end
End
Queue "$$
Address tso "submit * end($$)"
Delstack
say time() 'Utility CREATE Tablespace/Tables and LOAD Tables is running.'
Exit
Prepare_Sysin_Conn:
if f1 = 1 then do
  tf = 'CONN'
  Wfile = sfile||tf||''
  Call Alloc_sysinf
  "NEWSTACK"
  Queue 'CREATE TABLE PRA1ØS.TABCONN'
  Queue ' (TYPERES CHAR(4) NOT NULL,' 
  Queue ' CONNAM CHAR(8) NOT NULL,'
Queue '                GROUP    CHAR(8)        NOT NULL,'
Queue '                DESCR    CHAR(58) NOT NULL WITH DEFAULT ,
Queue '                NETNAME  CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                INDSYS   CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                REMSYS   CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                REMNAM   CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                REMSYNET CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                ACMETHOD CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                PROTOCOL CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                CONNTYPE CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                SINGLESE CHAR(3)  NOT NULL WITH DEFAULT ,
Queue '                DATASTRE CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                RECFORM  CHAR(2)  NOT NULL WITH DEFAULT ,
Queue '                QUEUELIM CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                MAXOTIME CHAR(4)  NOT NULL WITH DEFAULT ,
Queue '                AUTOCONN CHAR(3)  NOT NULL WITH DEFAULT ,
Queue '                INSERVIC CHAR(3)  NOT NULL WITH DEFAULT ,
Queue '                SECURNAM CHAR(8)  NOT NULL WITH DEFAULT ,
Queue '                ATTCHSEC CHAR(10) NOT NULL WITH DEFAULT ,
Queue '                BINDSEC  CHAR(3)  NOT NULL WITH DEFAULT ,
Queue '                USEDEFUS CHAR(3)  NOT NULL WITH DEFAULT ,
Queue '                PSRECOV  CHAR(10) NOT NULL WITH DEFAULT)'
Queue '               IN PRD1ØS.CICSRDO;'
Queue 'CREATE  INDEX PRA1ØS.XTABCONN'
Queue '                 ON PRA1ØS.TABCONN'
Queue '                     (CONNAM   ASC)'
Queue '                 USING STOGROUP PRG1ØS'
Queue '                 PRIQTY 12'
Queue '                 ERASE NO'
Queue '                 BUFFERPOOL BP1'
Queue '                 CLOSE NO ;'
Call Write_sysinf
"DELSTACK"
Wfile = sfile||tf||'#)'
Call Alloc_sysinf
"NEWSTACK"
Queue '     LOAD DATA RESUME YES LOG YES INDDN '
Queue '      SYSRECØØ INTO TABLE PRA1ØS.TABCONN'
Queue '   PROTOCOL POSITION(115) CHAR(4),   '
Queue '   CONNTYPE POSITION(119) CHAR(8),   '
Queue '   SINGLESE POSITION(127) CHAR(3),   '
Queue '   DATASTRE POSITION(138) CHAR(8),   '
Queue '   RECFORM POSITION(138) CHAR(2),   '
Queue '   QUEUELIM POSITION(140) CHAR(4),   '
Queue '   MAXOTIME POSITION(144) CHAR(4),   '
Queue '   AUTOCONN POSITION(148) CHAR(3),   '
Queue '   INSERVIC POSITION(151) CHAR(3),   '
Queue '   SECURNAM POSITION(154) CHAR(8),   '
Queue '   ATTCHESEC POSITION(162) CHAR(10),   '
Queue '   BINDSEC POSITION(172) CHAR(3),   '
Queue '   USEDEFUS POSITION(175) CHAR(3),   '
Queue '   PSRECOV POSITION(178) CHAR(10)   '
Queue ' )'
Queue
Call Write_sysinf
"DELSSTACK"
End

conn.f1 = recinp.i
Return
Prepare_Sysin_File:
if f2 = 1 then do
tf = 'FILE'
Wfile = sfile||tf'||''
Call Alloc_sysinf
"NEWSTACK"
Queue 'CREATE TABLE PRA10S.TABFILE'
Queue '   (TYPERES  CHAR(4)        NOT NULL,   '
Queue '   FILENAM  CHAR(8)        NOT NULL,   '
Queue '   GROUP    CHAR(8)        NOT NULL,   '
Queue '   DESCR    CHAR(58) NOT NULL WITH DEFAULT ,   '
Queue '   PASSWORD CHAR(8)  NOT NULL WITH DEFAULT ,   '
Queue '   LSRPID   CHAR(1)  NOT NULL WITH DEFAULT ,   '
Queue '   DSNSHR   CHAR(10) NOT NULL WITH DEFAULT ,   '
Queue '   STRINGS  CHAR(3)  NOT NULL WITH DEFAULT ,   '
Queue '   NSRGROUP CHAR(8)  NOT NULL WITH DEFAULT ,   '
Queue '   REMSYS   CHAR(4)  NOT NULL WITH DEFAULT ,   '
Queue '   REMNAM   CHAR(8)  NOT NULL WITH DEFAULT ,   '
Queue '   RECSIZE  CHAR(5)  NOT NULL WITH DEFAULT ,   '
Queue '   KEYLEN   CHAR(3)  NOT NULL WITH DEFAULT ,   '
Queue '   STATUS   CHAR(9)  NOT NULL WITH DEFAULT ,   '
Queue '   OPENTIME CHAR(8)  NOT NULL WITH DEFAULT ,   '
Queue '   DISP     CHAR(5)  NOT NULL WITH DEFAULT ,   '
Queue '   DATABUF  CHAR(5)  NOT NULL WITH DEFAULT ,   '
Queue '   INDEXBUF CHAR(5)  NOT NULL WITH DEFAULT ,   '
Queue '   TABLE    CHAR(4)  NOT NULL WITH DEFAULT ,   '
Queue '   MAXNUMR  CHAR(8)  NOT NULL WITH DEFAULT ,   '
Queue '   RECFORM CHAR(1)  NOT NULL WITH DEFAULT ,   '
Queue ' ADD CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' BROWSE CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' DELETE CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' READ CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' UPDATE CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' JOURNAL CHAR(2) NOT NULL WITH DEFAULT ,'
Queue ' JNLR CHAR(10) NOT NULL WITH DEFAULT ,'
Queue ' JNLSYNC CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' JNLU CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' JNL CHAR(6) NOT NULL WITH DEFAULT ,'
Queue ' JNLSYNCW CHAR(3) NOT NULL WITH DEFAULT ,'
Queue ' RECOVERY CHAR(11) NOT NULL WITH DEFAULT ,'
Queue ' FWDRECOV CHAR(2) NOT NULL WITH DEFAULT ,'
Queue ' BKTYPE CHAR(7) NOT NULL WITH DEFAULT ,'
Queue ' RESSECN CHAR(6) NOT NULL WITH DEFAULT )'
Queue ' IN PRDS.CICSRDO ;'
Queue ' CREATE INDEX PRA10S.XTABFILE'
Queue ' ON PRA10S.TABFILE'
Queue ' (FILENAME ASC)'
Queue ' USING STOGROUP PRG10S ,'
Queue ' PRIOTY 12 ,'
Queue ' ERASE NO ,'
Queue ' BUFFERPOOL BP1 ,'
Queue ' CLOSE NO ;'

Call Write_sysinf
"DELSTACK"
Wfile = sfile||tf||'#)'
Call Alloc_sysinf
"NEWSTACK"
Queue ' LOAD DATA RESUME YES LOG YES INDDN ,'
Queue ' SYSREC00 INTO TABLE PRA10S.TABFILE '
Queue ' (',
Queue ' TYPERES POSITION(1) CHAR(4) ,'
Queue ' FILENAM POSITION(5) CHAR(8) ,'
Queue ' GROUP POSITION(13) CHAR(8) ,'
Queue ' DESCR POSITION(21) CHAR(58) ,'
Queue ' DSNAM POSITION(79) CHAR(44) ,'
Queue ' PASSWORD POSITION(123) CHAR(8) ,'
Queue ' LSRPID POSITION(131) CHAR(1) ,'
Queue ' DSNSHR POSITION(132) CHAR(10) ,'
Queue ' STRINGS POSITION(142) CHAR(3) ,'
Queue ' NSRGROUP POSITION(145) CHAR(8) ,'
Queue ' REMSYS POSITION(153) CHAR(4) ,'
Queue ' REMNAM POSITION(157) CHAR(8) ,'
Queue ' RECSIZE POSITION(165) CHAR(5) ,'
Queue ' KEYLEN POSITION(170) CHAR(3) ,'
Queue ' STATUS POSITION(173) CHAR(9) ,'
Queue ' OPENTIME POSITION(182) CHAR(8) ,'
Queue ' DISP POSITION(190) CHAR(5) ,
Queue '          DATABUF POSITION(195) CHAR(5),          '  
Queue '          INDEXBUF POSITION(200) CHAR(5),          '  
Queue '          TABLE POSITION(205) CHAR(4),          '  
Queue '          MAXNUMR POSITION(209) CHAR(8),          '  
Queue '          RECFORM POSITION(217) CHAR(1),          '  
Queue '          ADD POSITION(218) CHAR(3),          '  
Queue '          BROWSE POSITION(221) CHAR(3),          '  
Queue '          DELETE POSITION(224) CHAR(3),          '  
Queue '          READ POSITION(227) CHAR(3),          '  
Queue '          UPDATE POSITION(230) CHAR(3),          '  
Queue '          JOURNAL POSITION(233) CHAR(2),          '  
Queue '          JNLR POSITION(235) CHAR(10),          '  
Queue '          JNLSYNCR POSITION(245) CHAR(3),          '  
Queue '          JNLU POSITION(248) CHAR(3),          '  
Queue '          JNLA POSITION(251) CHAR(6),          '  
Queue '          JNLSYNCW POSITION(257) CHAR(3),          '  
Queue '          RECOVERY POSITION(260) CHAR(11),          '  
Queue '          FWDRECOV POSITION(271) CHAR(2),          '  
Queue '          BKTYPE POSITION(273) CHAR(7),          '  
Queue '          RESSECN POSITION(280) CHAR(6)          '  
Queue '          )'  
Queue  
Call Write_sysinf  
"DELSTACK"  
End  
file.f2 = recinp.i  
Return  
Prepare_Sysin_Lsrp:  
if f3 = 1 then do  
    tf = 'LSRP'  
    Wfile = sfile||tf||')'  
    Call Alloc_sysinf  
    "NEWSTACK"  
    Queue 'CREATE TABLE PRA10.S.TABLSRP'  
    Queue '          (TYPERES  CHAR(4) NOT NULL,'  
    Queue '          LSRPNAM  CHAR(8) NOT NULL,'  
    Queue '          GROUP    CHAR(8) NOT NULL,'  
    Queue '          DESCRA    CHAR(58) NOT NULL WITH DEFAULT,'  
    Queue '          POOLID   CHAR(1) NOT NULL WITH DEFAULT,'  
    Queue '          MAXKEYL  CHAR(3) NOT NULL WITH DEFAULT,'  
    Queue '          SHRLIM   CHAR(3) NOT NULL WITH DEFAULT,'  
    Queue '          STRINGS  CHAR(3) NOT NULL WITH DEFAULT,'  
    Queue '          D512     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D1K      CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D2K      CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D4K      CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D8K      CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D12K     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D16K     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D20K     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D25K     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D30K     CHAR(5) NOT NULL WITH DEFAULT,'  
    Queue '          D40K     CHAR(5) NOT NULL WITH DEFAULT,'  

Queue '               D24K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               D28K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               D32K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I512     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I1K      CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I2K      CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I4K      CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I8K      CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I12K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I16K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I20K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I24K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I28K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               I32K     CHAR(5)  NOT NULL WITH DEFAULT ,'
Queue '               HSD4K    CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD8K    CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD12K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD16K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD20K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD24K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD28K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSD32K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI4K    CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI8K    CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI12K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI16K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI20K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI24K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI28K   CHAR(8)  NOT NULL WITH DEFAULT ,'
Queue '               HSI32K   CHAR(8) NOT NULL WITH DEFAULT )'
Queue '       IN PRD1ØS.CICSRDO;'
Queue 'CREATE  INDEX PRA1ØS.XTABLSRP'
Queue '                 ON PRA1ØS.TABLSRP'
Queue '                     (LSRPNAM  ASC)'
Queue '                 USING STOGROUP PRG1ØS'
Queue '                 PRIQTY 12'
Queue '                 ERASE NO'
Queue '                 BUFFERPOOL BP1'
Queue '                 CLOSE NO ;'
Queue
Call Write_sysinf
"DELSTACK"
Wfile = sfile||tf||'#')
Call Alloc_sysinf
"NEWSTACK"

Editor’s note: this article will be concluded in next month’s issue.

Espedito Morvillo
Systems Programmer (Italy)  © Xephon 2001
CICS news

Compuware has started shipping the E-Business Edition of its Abend-AID fault management tool, designed to help save time in test and production environments and speed the integration of legacy systems and e-business applications.

It provides developers with diagnostic information that helps pinpoint problems and suggests corrective actions to resolve those problems. It’s said to function as “built-in expertise” for programmers for MQSeries in batch, IMS, and CICS environments. It enables programmers and developers to detect, analyze, and diagnose problems in applications that use MQSeries CICS Web Interface.

It maps out MQSeries so developers can identify any errors that might occur on those applications being integrated with MQSeries.

For further information contact: Compuware, 31440 Northwestern Highway, Farmington Hills, MI 48334-2564, USA. Tel: (248) 737 7300. URL: http://www.compuware.com/products/abendaid/.

* * *

Tivoli has announced its Tivoli Business Systems Manager (TBSM), replacing Tivoli Manager for OS/390, while the Distributed Edition component replaces Tivoli Global Enterprise Manager (GEM). The combination of both in one product, we’re told, provides an end-to-end enterprise management system.

It adds CICSPlex/SM as a source of discovery of CICS regions, running under OS/390, VSE, or OS/2, as well as discovering CICS files and transactions. Exception monitoring will be provided though the usage of system availability monitoring and realtime analysis.

As for the Distributed Edition, it supports existing GEM instrumentation architecture including heartbeat events via Application Policy Management (APM) events. It also supports thresholding via APM threshold events, automatic LOB creation via Application Management Specification (AMS) definition, and the ability to create customer instrumentation via Tivoli Module Builder, Tivoli Module Designer, and Tivoli Quickstart Wizard.

Also, it provides the ability to manage business system components on Windows NT, AIX, HP-UX, and Solaris via Tivoli Instrumentation Services and can invoke Tivoli tasks by mapping task libraries to APM-created software components.

It can exploit Tivoli Distributed Monitors, including the ability to map generic distributed monitors to a software component, and supports CICS and DB2 instrumentation to monitor and control Distributed Edition applications on OS/390. Functions common to both include the use of all usability function previously available in Tivoli Manager for OS/390, creation of lines of business via drag-and-drop, the ability to create LOB views containing both distributed and OS/390 resources, and a pre-packaged NetView application monitoring interface supported with DB2/CICS instrumentation.

For further information contact your local IBM representative. URL: http://www.tivoli.com/products/index/business_systems/.