APAR OA60203
DFSMSdfp OAM
Publication Update Document

The z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support (V2R3 version publication numbers: SC23-6866-30) to be updated as follows…

Note:

1. **Red text with strikethrough should be removed.**
2. **Blue text is new text that should be added.**
3. **Black text is existing text for reference.**
4. **'…' means the content here doesn't contain any new changes and will not be shown.**

Appendix F. "Auto-delete installation exit" -> "Writing the CBRHADUX exit" -> Page 574 Paragraph before subtitle "Input":

…
If the object name is not found in the verification table, the exit approves deletion. When the object is deleted, the name is written to a sequential data set called the notify data set, HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY ("XXXXXXXX" qualifier will be replaced by actually storage group name and padded with "X". Optional qualifier "YYYY" will be replaced by actually DB2SSID and padded with "Y" if the user chooses to do so, otherwise the data set name would be HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY). Fully qualified data set names contain both the object name and the collection name, allowing uniqueness across OSMC and concurrent I/O from the different tasks. A concatenation of these data sets (one for each group) provides input to other applications that need to synchronize their directories with the OAM object directories.
…

Page 574 Paragraph after subtitle "Output":
The output data set HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY ("XXXXXXXX" qualifier will be replaced by actually storage group name and padded with "X". Optional qualifier "YYYY" will be replaced by actually DB2SSID and padded with "Y" if the user chooses to do so, otherwise the data set name would be HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY), known as the NOTIFY data set, is defined in the DCBO in the dynamic working storage and also defined in local storage in the DCB with the DDNAME of CBRADUXO.
…

Page 577 "Sample auto-delete installation exit" replace the sample under this session with the new sample below:

```
CBRHADUX TITLE 'OSMC SAMPLE AUTO-DELETE INSTALLATION EXIT'              00050000
*****************************************************************      00100000
*                                                               *      00116600
```

```
* $MOD(CBRHADUX),COMP(OSMC),PROD(OAM) :                                *  00133300
*                                                                     *  00150000
* MODULE NAME:  CBRHADUX                                             *  00200000
*                                                                     *  00250000
* DESCRIPTIVE NAME:  OSMC Sample AUTO-DELETE Installation Exit       *  00262500
*                                                                     *  00275000
*********************************************************************  00317700
*PROPRIETARY V3 STATEMENT                                             *  00360500
*Licensed Material - Property of IBM                                  *  00367700
*"Restricted Materials of IBM"                                           00367800
*5650-ZOS                                                             *  00373100
*COPYRIGHT IBM CORP. 1989, 2020                                       *  00378500
*END PROPRIETARY V3 STATEMENT                                         *  00383900
*********************************************************************  00384100
*                                                                     *  00384300
*  NOTE: Read before implementing, this exit is used during OSMC     *  00384500
*        and does not perform an OSREQ functions during its           *  00384700
*        processing.                                                  *  00384900
*                                                                     *  00385100
*  This sample program, if installed as is, will prevent objects      *  00385300
*  from being deleted/                                                *  00385500
*                                                                     *  00385700
*  RETURNS:       12 - Do not delete any objects within this storage *  00385900
*                      group and do not call this exit again for this*  00386100
*                      instance of OSMC for this storage group.       *  00386300
*                                                                     *  00386500
*  This sample exit is provided as an example of different coding     *  00386700
*  techniques that may be used.  It should be modified to suit the    *  00386900
*  specific needs of each user.  The minimum modification that will   *  00387100
*  enable any of the function provided is to delete or comment out    *  00387300
*  the three lines of "bypass" code that return the "12" described    *  00387500
*  above.  (right after "-- THREE LINES OF BYPASS CODE --"  )         *  00387700
*                                                                     *  00387900
*  SUBROUTINE MCVERIFY is provided as an example of a technique that  *  00388100
*  could be used to check management class, but is not currently      *  00388300
*  invoked.  This technique involves coding the management class      *  00388500
*  names in the data storage area for the CBRHADUX.                   *  00388700
*                                                                     *  00388900
*  Search for keyword MMM to find the Minimum Mandatory               *  00389100
*  Modification/Review Points in the code.                            *  00389300
*                                                                     *  00389600
*  When running in a Multi-OAM configuration, a unique notify data    *  00389601
*  set name should be used for each OAM instance or different OAM     *  00389602
*  instances may try to write to the same data set concurrently and   *  00389603
*  error may occur. Search for keyword CCC to find the code that was  *  00389604
*  added to include the DB2SSID as part of the notify data set name.  *  00389605
*  The use of the DB2SSID as a qualifier in the data set name can     *  00389606
*  also be used in a classic OAM configuration.                       *  00389607
*                                                                     *  00389608
*********************************************************************  00390100
*  FUNCTION:                                                          *  00390300
*    Module CBRHADUX is used to verify whether or not an object       *  00390500
*    should be deleted.  This module is called during OSMC            *  00390700
*    processing when an object has been selected for deletion.        *  00390900
*                                                                     *  00391100
*  OPERATION:                                                         *  00430300
*  - Perform standard entry linkage.                                  *  00469500
*  - Set CBRHADUX return code to 12 indicating to OSMC that objects   *  00508700
*    are not to be deleted for this storage group and that the        *  00547900
*    CBRHADUX exit is not to be called again for this storage group   *  00587100
*    [default behavior]                                               *  00626300
*    (Read NOTE: READ BEFORE IMPLEMENTING EXIT for explanation of     *  00665500
*     default behavior)                                               *  00704700
```

```
*  - [NOTE] Write message to the operator stating that objects       *  00743900
*     will NOT be expired for this storage group [default behavior]  *  00783100
*  - If return code not set to 12 then                               *  00822300
*     | If this is the first call then                               *  00861500
*     | | - Initialize storage                                       *  00900700
*     | | - Set up Data Area to allow for reentrancy.                *  00939900
*     | | - Prepare Data Area for next call for same Storage Group   *  00979100
*     | | - Establish register save area in Data Area.               *  01018300
*     | Endif                                                        *  01057500
*     | If current Storage Group is GROUPXX [see note in code]       *  01096700
*     | | - Set CBRHADUX return code to 8 indicating to OSMC that    *  01135900
*     | | - objects are to be deleted for this storage group and that*  01175100
*     | |   the CBRHADUX exit is not to be called again for this     *  01214300
*     | |   storage group.                                           *  01253500
*     | |   (as an example of bypassing processing by the CBRHADUX   *  01292700
*     | |    exit for a given storage group)                         *  01331900
*     | Else                                                         *  01371100
*     | | If this invocation is not the end of a storage group       *  01410300
*     | | (indicated by the ADUDONE flag)                            *  01449500
*     | | | If this is the first invocation (i.e. verify table has   *  01488700
*     | | | not yet been built) then                                 *  01527900
*     | | | | - Open input dataset and build a local table in storage*  01567100
*     | | | |  if needed                                             *  01606300
*     | | | | - Open output dataset                                  *  01645500
*     | | | Endif                                                    *  01684700
*     | | | If no errors encountered so far then                     *  01723900
*     | | | | If object name and collection name of current object   *  01763100
*     | | | |  do not exist in the in the Verify Table Then          *  01802300
*     | | | | | Add a record to the notify dataset                   *  01841500
*     | | | | Endif                                                    *  01880700
*     | | | Endif                                                    *  01919900
*     | | Endif                                                      *  01959100
*     | Endif                                                        *  01998300
*     | If not the end of storage group (indicated by ADUDONE flag)  *  02037500
*     | and an error return code of 8 or greater has been set then   *  02076700
*     | | Write message to the operator indicating the storage group *  02115900
*     | | reason code and return code                                *  02155100
*     | Endif                                                        *  02194300
*     | If return code is 8 (indicating that CBRHADUX will not be    *  02233500
*     | called again)                                                *  02272700
*     | | - Free the verify table (if it exists)                     *  02311900
*     | | - Close the output dataset                                 *  02351100
*     | | - Free data area                                           *  02390300
*     | Endif                                                        *  02429500
*     Endif                                                          *  02468700
*  - Perform standard exit linkage                                   *  02507900
*                                                                    *  02550000
*  NOTES:                                                            *  02600000
*                                                                    *  02650000
*     DEPENDENCIES: This routine is called on a group basis.  That   *  02700000
*                   is, the first and every call of a single         *  02750000
*                   instance of this routine is for the same storage *  02800000
*                   group.  This allows a return code indicating the *  02850000
*                   routine is not interested in a certain group     *  02900000
*                   (i.e. GROUPXX) and that the output data          *  02950000
*                   set can be segregated by group allowing for      *  03000000
*                   processing overlap (i.e. more than one task      *  03050000
*                   can have an instance of this routine because     *  03100000
*                   each task is writing to a separate data set).    *  03150000
*                                                                    *  03200000
*                   The end of cycle call does not include the name  *  03250000
*                   of an object to be deleted.                      *  03300000
*                                                                    *  03350000
```

```
*                                                                     *  03400000
*     CHARACTER-CODE-DEPENDENCIES: EBCDIC Character Set               *  03450000
*                                                                     *  03500000
*     RESTRICTIONS: None                                              *  03550000
*                                                                     *  03600000
*     REGISTER CONVENTIONS:                                           *  03650000
*                                                                     *  03700000
*               Standard Entry Linkage                                *  03750000
*                                                                     *  03800000
*               R0 and R1 used for system service invocation          *  03850000
*               R2 and R3 are work registers                          *  03900000
*               R4  contains the ADUP address                         *  03950000
*               R5 through R10 are work registers                     *  04000000
*               R11 contains the address of the dynamic area          *  04050000
*               R12 is the base register                              *  04100000
*                                                                     *  04150000
*  MODULE TYPE: User Exit                                             *  04175000
*                                                                     *  04200000
*    PROCESSOR: Assembler                                       @P2C*  04250000
*                                                                     *  04300000
*    ATTRIBUTES:                                                      *  04350000
*     LOCATION: LINKLIB                                               *  04400000
*     STATE:    PROBLEM                                               *  04450000
*     TYPE:     REENTRANT                                             *  04500000
*                                                                     *  04650000
*  INPUT:                                                             *  04700000
*      SYMBOLIC NAME: CBRADUP                                         *  04750000
*      DESCRIPTION:   Control block containing the parameters for     *  04800000
*                     object deletion. (Contained in MACRO CBRADUP)   *  04850000
*                                                                     *  04900000
*      SYMOBLIC NAME: Verify Data Set [OPTIONAL]                      *  04907100
*      DSN NAME:      HLQ.OBJECT.DELETE.VERIFY                        *  04914200
*      DESCRIPTION:   This is the input data set to the CBRHADUX      *  04921300
*                     which is used to determine that an object       *  04928400
*                     should not be expired.  The criteria            *  04935500
*                     is based on Object Name and Collection Name     *  04942600
*                     which is based upon the input to this module.   *  04949700
*                     This Data set also has a table created in       *  04956800
*                     storage for which comparisons shall be made.    *  04963900
*      ATTRIBUTES:                                                    *  04971000
*         ORGANIZATION:   PS                                          *  04978100
*         RECORD FORMAT:  FB                                          *  04985200
*         RECORD LENGTH:  89                                          *  04992300
*         BLOCK SIZE:     8900                                        *  04999400
*      RECORD FORMAT:                                                 *  05006500
*         0-43  :   Object Name        CHAR(44)                       *  05013600
*         44    :   Blank                                             *  05020700
*         45-89 :   Collection Name    CHAR(44)                       *  05027800
*                                                                     *  05034900
*      NOTES:         Be sure to modify the (HLQ) high level          *  05042000
*                     qualifier and adjust accordingly to the         *  05049100
*                     private catalog that is accessible. An          *  05056200
*                     in-storage table will be built for these names  *  05063300
*                     and will be formatted as follows:               *  05070400
*                                                                     *  05077500
*          VERIFY TABLE:                                              *  05084600
*               0-3    :  Number of entries in table                 *  05091700
*               4-7    :  Pointer to next table                      *  05098800
*               9-4095 :  Records from Input data set                *  05105900
*                                                                     *  05113000
*          NOTE: For last Table in Chain, the pointer to the next     *  05120100
*                table is 00000000 (signifies end of chain)           *  05127200
*                                                                     *  05134300
```

```
*  OUTPUT:                                                          *  05141400
*      SYMBOLIC NAME: ADUUFLD                                       *  05148500
*      DESCRIPTION:   CBRADUP parameter list is updated to hold a   *  05155600
*                     pointer to the dynamic data area for reentrant*  05162700
*                     calls for the same storage group.             *  05169800
*                                                                   *  05176900
*      SYMBOLIC NAME: Notify Data Set                               *  05184000
*      DSN NAME:      HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY             *  05191100
*      DESCRIPTION:   This is the output data set that contains the *  05198200
*                     Object Name along with the Collection Name of *  05205300
*                     each object deleted for the Storage group,    *  05212400
*                     which replaces the XXXXXXXX in the data set   *  05219500
*                     name, padded with the character X. The few    *  05226600
*                     sections marked with MMM correlate to the     *  05233700
*                     output data set name.                         *  05240800
*      NOTE:          To include the DB2SSID in the data set name as*  05240801
*                     a qualifier, uncomment all the lines marked   *  05240802
*                     with CCC. The DSN NAME will become:           *  05240803
*                     HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY        *  05240804
*                     The DB2SSID will replace the YYYY in the data *  05240805
*                     set name, padded with the character Y.        *  05240806
*                                                                   *  05240807
*      ATTRIBUTES:                                                  *  05247900
*         ORGANIZATION:   PS                                        *  05255000
*         RECORD FORMAT:  FB                                        *  05262100
*         RECORD LENGTH:  89                                        *  05269200
*         BLOCK SIZE:     8900                                      *  05276300
*      RECORD FORMAT:                                               *  05283400
*         0-43  :   Object Name        CHAR(44)                     *  05290500
*         44    :   Blank                                           *  05297600
*         45-89 :   Collection Name    CHAR(44)                     *  05304700
*                                                                   *  05311800
*   NOTE:  If this dataset is preallocated prior to the invocation  *  05318900
*          of this routine, the DCB declares of the dataset should  *  05326000
*          not be altered or removed.  Otherwise, modification to   *  05333100
*          the DCB statements can cause unexpected results.         *  05340200
*                                                                   *  05347300
*   NOTE:  Be sure to modify the VERIFY and NOTIFY data set names   *  05354400
*          prior to assembling and linking this module.  The        *  05361500
*          following labels need to be modified to reflect the High *  05368600
*          Level Qualifier (HLQ) you would like to be used.         *  05375700
*                 1)  D1                                            *  05389500
*                 2)  NTFYDSN                                       *  05390300
*                 3)  VRFYDSN                                       *  05391400
*                                                                   *  05392600
*    RETURN CODES = 0  Allows for expiration of the object to       *  05393200
*                      continue and to continue calling the user    *  05393800
*                      exit for OSMC processing.                    *  05394400
*                   4  Do not allow expiration of this object for   *  05395000
*                      this instance of OSMC for this storage group.*  05395600
*                   8  Expire all objects in this storage group for *  05396200
*                      this instance of OSMC but do not call the    *  05396800
*                      user exit again.                             *  05397400
*                   12 Do not allow for expiration of objects for   *  05398000
*                      this OSMC instance and this storage group and*  05398600
*                      call the user exit again.                    *  05399200
*                                                                   *  05400000
* CHANGE ACTIVITY:                                                  *  05450000
*                                                                   *  05750000
*      $L0=OAM,110,082687,TUCWV:  INITIAL RELEASE                   *  05758300
*      $L1=JDP3227,320,890523,TUCHTT:  RELEASE 1                    *  05766600
*      $D1=JDP3227,320,890523,TUCLJS:  COLLECTION NAMES             *  05774900
*      $L2=PRESCOTT,331,901112,TUCLJS:  PRESCOTT SUPPORT            *  05783200
```

```
*      $L3=OAMR1C,R1C,090311,TUCDMS:  RAS Improvements               *  05791500
*      $L4=OAMR21,R21,110215,TUCBJF:  RAS Improvements               *  05795700
*      $P1=K210268,R21,110714,TUCBJF: RAS Improvements               *  05797800
*      $P2=130050,R23,160817,TUCAED: Typo in prolog                  *  05797801
*      $00=OA60203,R23,200821,TUCJFX: ADD DB2SSID (OPTIONAL) IN      *  05797802
*                                     NOTIFY DS NAME                 *  05797803
********************************************************************  05797804
         TITLE 'CBRHADUX WORKING STORAGE'                               05800500
*--------------------------------------------------------------------*  05801000
*                                                                    *  05801500
*        CBRHADUX WORKING STORAGE                                    *  05802000
*                                                                    *  05802500
*--------------------------------------------------------------------*  05803000
DATAREA  DSECT ,                                                        05803500
SAVE1    DS    18F                SAVE AREA                             05804000
SAVE14   DS    1F                 R14 SAVE AREA FOR SUBROUTINES         05804500
FLAGS    DS    X                  FLAG AREA                             05805000
TABUILT  EQU   X'80'              VERIFY TABLE HAS BEEN BUILT           05805500
ODCBO    EQU   X'40'              NOTIFY DATA SET IS OPEN               05806000
VRECSPTR DS    F                  ADDRESS OF THE IST 4K BLOCK OF        05806500
RETTXT   DS    AL2                                                      05807000
         DS    AL2                                                      05807500
         DS    AL2                                                      05808000
         DS    CL8                                                      05808500
*                                 OBJECT NAMES READ FROM THE VERIFY     05809000
*                                 DATA SET                              05809500
RETCODE  DS    F                  INTERNAL RETURN CODE                  05810000
REASCODE DS    F                                                        05810500
S99RBPTR DS    F                  ADDRESS OF SVC99 RB                   05811000
RENTAREA DS    0F                 REENTRANT COPY OF STATIC DEFINED      05811500
*                                 CONTROL BLOCKS                        05812000
OPENL    OPEN   (,),MF=L                                                05812500
DCBS     DS    0F                                                       05813000
DCBO     DCB  DDNAME=CBRADUXO,MACRF=(PM),OPTCD=W,LRECL=89,BLKSIZE=8900,+05813500
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN                          05814000
DCBI     DCB  DDNAME=CBRADUXI,MACRF=(GM),LRECL=89,BLKSIZE=8900,        +05814500
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN,EODAD=REODAD             05815000
DYNRB    DS    0F                                                       05815500
         DC    AL1(20)            RB LENGTH                             05816000
         DC    AL1(01)            DSNAME ALLOCATION                     05816500
         DC    X'C0'              FLAGS1 - NO EXIST ALLOC               05817000
         DC    X'0' FLAGS1                                              05817500
         DC    F'0'               ERROR CODES                           05818000
TXTPTR   DC    A(TXTPTRV)         TEXT UNIT POINTERS                    05818500
         DC    F'0'               RESERVED                              05819000
*        DC    X'D1'              WAIT FOR VOLS,UNITS,DSNS AND MOUNTS   05819500
         DC    X'00'                                                    05820000
         DC    AL3(0)                                                   05820500
TXTPTRV  DS    0F                                                       05821000
TXTDSN   DC    A(DSNTXT)                                                05821500
TXTDDN   DC    A(DDNMTXT)                                               05822000
TXTSTAT  DC    A(SHRTXT)                                                05822500
TXTDISP  DC    A(KEEPTXT)                                               05823000
TXTTRK   DC    A(0)                                                     05823500
TXTPRIM  DC    A(0)                                                     05824000
TXTSEC   DC    A(0)                                                     05824500
TXTRET   DC    A(0)                                                     05825000
TXTUNIT  DC    A(UNITTXT)                                               05825500
         DC    A(CLOSTXT)                                               05826000
         DC    X'80'                                                    05826500
         DC    AL3(0)                                                   05827000
DSNTXTN  DC    AL2(2)                                                   05827500
         DC    AL2(1)                                                   05828000
```

```
         DC    AL2(ENDDSNN-NTFYDSN)                                      05828500
*                                                                      * 05829000
*  Please note that modifying NTFYDS2 can cause unexpected results,    * 05829500
*  the XXXXXXXX is overlaid by the current storage group name.         * 05830000
*  the YYYY is overlaid by the current DB2SSID if the CCC line is      * 05830001
*  uncommented.                                                        * 05830250
*                                                                      * 05830500
NTFYDSN  DC    C'HLQ.'                                                MMM 05831000
*NTFYDB2  DC    C'YYYY.'                                               CCC 05831250
NTFYDS2  DC    C'XXXXXXXX.OBJECT.DELETE.NOTIFY'                          05831500
ENDDSNN  EQU   *                                                        05832000
*                                                                       05832500
CONVERT  DS    D                  CONVERSION AREA FOR WTD              05833000
         DS    D                  EXTENSION OF WORK AREA               05833500
HEXWORK  DS    D                  CHARACTER CONVERSION FOR WTD         05834000
*----------------------------------------------------------------------* 05834500
*                                                                      * 05835000
*        SINGLE LINE WTO PARAMETER LIST                                * 05835500
*                                                                      * 05836000
*----------------------------------------------------------------------* 05836500
WTOLIST  WTO   TEXT=((,DE)),                                           +05837000
               MF=L                                                     05837500
WTOLISTL EQU   *-WTOLIST          SIZE OF WTO MACRO EXPANDED           05838000
         SPACE 2                                                        05838500
*----------------------------------------------------------------------* 05839000
*                                                                      * 05839500
*        SINGLE LINE WTO TEXT LINE                                     * 05840000
*                                                                      * 05840500
*----------------------------------------------------------------------* 05841000
STATDISP EQU   *-DATAREA          DISPLACEMENT IN DATA AREA            05841500
STATLINE DS    0F                 STATUS CONTROL LINE MODEL            05842000
         DS    AL2                LENGTH OF MESSAGE                    05842500
         DS    C'CBRHADUX USER EXIT: SG ('  START OF WTO MSG           05843000
STATSG   DS    CL8                STORAGE GROUP NAME                   05843500
         DS    C') completed RC='  LABEL FOR RETURN CODE              05844000
STATRET  DS    CL8                RETURN CODE                          05844500
         DS    C' RSN='           LABEL FOR REASON CODE                05845000
STATREAS DS    CL8                REASON CODE                          05845500
STATLEN  EQU   *-STATLINE         LENGTH OF STATUS LINE                05846000
*                                                                       05846500
DATALEN  EQU   *-DATAREA                                               05847000
         EJECT                                                         05850000
         TITLE 'CBRHADUX DSECT DEFINITIONS'                            05857100
*----------------------------------------------------------------------*  05864200
*                                                                     *  05871300
*        CBRADUP - PARAMETER LIST DSECT DEFINITION                    *  05878400
*                                                                     *  05885500
*---------------------------------------------------------------------*  05892600
         CBRADUP                                                       05899700
         SPACE 2                                                       05906800
*----------------------------------------------------------------------*  05913900
*                                                                     *  05921000
*        MANAGEMENT CLASS DSECT TO MAP TO MCTAB                       *  05928100
*                                                                     *  05935200
*---------------------------------------------------------------------*  05942300
TAB      DSECT ,                                                       05949400
MCLEN    DS    H                                                       05956500
MCNAME   DS    CL30                                                    05963600
TABLEN   EQU   *-TAB                                                   05970700
         EJECT                                                         05977800
         TITLE 'CBRHADUX - VERIFY FOR OBJECT DELETION EXIT'            05984900
********************************************************************** 05992000
*                                                                     * 05999100
```

```
*        CBRHADUX ENTRY POINT - MAINLINE PROCESSING                   * 06006200
*                                                                     * 06013300
********************************************************************** 06020400
CBRHADUX CSECT                    BEGINNING OF CBRHADUX                06027500
* CBRHADUX AMODE 31                  FOR 31 BIT ADDRESSING             06034600
* CBRHADUX RMODE ANY                                                   06041700
ADUX     DS    0H                 ENTRY POINT                          06050000
         USING *,R15              ALLOW FOR IMPLICIT ADDRESSING        06100000
         B     PASTID             BRANCH AROUND ID                     06150000
         DC    CL8'CBRHADUX'      EYECATCHER FOR EXIT                  06187500
         DC    CL8'&SYSDATE'      DATE IN THE EYECATCHER               06225000
PASTID   DS    0H                 BRANCH AROUND LABEL                  06262500
         STM   R14,R12,12(R13)    SAVE CALLERS REGS                    06300000
         LR    R12,R15            SET BASE REG                         06350000
         DROP  R15                                                     06400000
         USING ADUX,R12           IMPLICIT ADDRESSABILITY              06450000
         LR    R4,R1              GET INPUT PARAMETER LIST             06500000
         USING CBRADUP,R4         ADDRESSABILITY ON PARM LIST          06526900
         SYSSTATE AMODE64=NO ACSENV=P                                  06540300
********************************************************************** 06553800
*                                                                     * 06557600
*    Comment the following lines to allow for object deletion, by     * 06561400
*      default objects will not be deleted and a WTO is made.         * 06565200
*                  -- THREE LINES OF BYPASS CODE --                   * 06569000
*                                                                     * 06572800
********************************************************************** 06576600
*                                                                     * 06580400
         LA    R2,ADUFAILN        LOAD RETURN CODE 12 INTO R2          06584200
       WTO 'CBRHADUX USER EXIT: RC=12 OSMC bypassed object expiration' 06586100
         B     NOR13              BYPASS ALL CODE EXCEPT EXIT CODE     06588000
*                                                                     * 06591800
********************************************************************** 06595600
         L     R11,ADUUFLD        GET ADDRESS OF DYNAMIC AREA          06600000
         LTR   R11,R11            IS THIS THE FIRST CALL               06650000
         BNZ   NOGMAIN            NO, DO NOT GETMAIN                   06700000
         GETMAIN RU,LV=DATALEN,SP=0  GET DYNAMIC AREA                  06750000
         LR    R11,R1             GET STORAGE ADDRESS FOR DYNAMIC AREA 06800000
         USING DATAREA,R11        SET DYNAMIC AREA ADDRESSABILITY FOR  06850000
*                                  REENTRANCY                          06900000
         SLR   R1,R1              GET A ZERO                           06950000
         ST    R1,RETCODE         ZERO THE RETURN CODE                 07000000
         ST    R1,REASCODE        ZERO THE REASON CODE                 07025000
         ST    R1,VRECSPTR        ZERO THE IN-STORAGE NAMES FIELD      07050000
         STC   R1,FLAGS           ZERO THE FLAGS BYTE                  07100000
         ST    R11,ADUUFLD        SAVE DYNAMIC AREA FOR NEXT CALL      07150000
NOGMAIN  DS    0H                 BRANCH AROUND FOR NO GETMAIN         07200000
         ST    R13,SAVE1+4        BACKCHAIN SAVE AREAS                 07500000
         LA    R2,SAVE1           GET SAVE AREA ADDRESS                07550000
         ST    R2,8(,R13)         FORWARDCHAIN SAVE AREAS              07600000
         LR    R13,R2             ESTABLISH SAVE AREA ADDRESS          07650000
         SR    R2,R2              CLEAR FOR ZERO                       07700000
         ST    R2,RETCODE         ZERO THE INTERNAL RETURN CODE        07750000
         ST    R2,REASCODE        ZERO THE INTERNAL REASON CODE        07752300
********************************************************************** 07754600
*  The following comparison is made to provide an example, of how to * 07756900
*  allow deletions for a particular storage group.  The following    * 07759200
*  four lines can be removed if this does not pertain to your needs. * 07761500
********************************************************************** 07763800
         CLC   ADUSGNAM,=CL8'GROUPXX' DO NOT PROCESS FURTHER IF    MMM  07766100
         BNE   NOTGRPXX           GROUPXX FOUND DO NOT PROCESS         07768400
         LA    R2,ADU8RS01        LOAD REASON CODE FOR GROUPXX         07770700
         ST    R2,REASCODE        STORE REASON CODE                    07773000
         LA    R2,ADUFAILC        INDICATE DELETE IS OKAY BUT          07775300
```

```
         ST    R2,RETCODE          DO NOT CALL ADUX AGAIN               07778400
         B     ENDADUX             FOR THIS CYCLE                       07785500
NOTGRPXX DS    0H                                                       07792700
         CLI   ADUFUNC,ADUDONE     IS THIS AN END OF CYCLE CALL         07800000
         BE    DOFREE              YES, GO CLEANUP                      07850000
         TM    FLAGS,TABUILT       HAS VERIFY TABLE BEEN BUILT          07900000
         BO    DOVERIFY            YES, DO NOT RE-OPEN DATA SET         07950000
         BAL   R14,OPENDS          FIRST CALL OPEN INPUT AND OUTPUT     08000000
         OI    FLAGS,TABUILT       INDICATE TABLE BUILT FOR NEXT CALL   08050000
*                                  VRECSPTR WILL BE ZERO IF NO          08100000
*                                  ENTRIES IN TABLE                     08150000
DOVERIFY DS    0H                                                       08200000
         L     R2,RETCODE          GET RETURN CODE                      08250000
         LTR   R2,R2               DO NOT CONTINUE IF NON-ZERO          08300000
         BNZ   ENDADUX             END ADUX IF NON-ZERO CODE            08350000
         L     R10,VRECSPTR        GET ADDRESS OF IN-CORE VERIFY RECS   08400000
         LTR   R10,R10             IS THERE ANY VERIFY TABLE            08450000
         BZ    NOVERIFY            IF ZERO NO VERIFY NEEDED             08500000
         BAL   R14,VERIFY          CALL VERIFY ROUTINE                  08550000
NOVERIFY DS    0H                  FREEMAIN TABLE                       08600000
         L     R2,RETCODE          GET THE RETURN CODE                  08650000
         LTR   R2,R2               IF NOT ZERO VERIFY FAILED            08700000
         BNZ   NONOTIFY            NO NOTIFY IF VERIFY FAILED           08750000
         BAL   R14,NOTIFY          ADD OBJECT TO NOTIFY DATA SET        08800000
NONOTIFY DS    0H                  DO NOT NOTIFY IF NON-ZERO CODE       08850000
ENDADUX  DS    0H                  END OF CBRADUX                       08900000
*********************************************************************** 08908300
*  If you would not like the CBRHADUX module to display return code  *  08916600
*  messages, the please comment out the following statement.         *  08924900
*********************************************************************** 08933200
         BAL   R14,ADUXOUT         CALL WTO FOR ADUX STATUS             08941500
         LA    R3,ADUFAILC         RETURN CODE INDICATING NO            08950000
*                                  RECALL OF ADUX FOR                   09000000
*                                  THIS CYCLE                           09050000
         C     R3,RETCODE          IF ADUX IS TO BE CALLED AGAIN        09100000
         BH    NOFREE              DO NOT FREE AND CLOSE                09150000
DOFREE   DS    0H                  FREE THE VERIFY TABLE AND CLOSE      09200000
*                                  THE OUTPUT DATA SET                  09250000
         BAL   R14,FREETAB         FREEMAIN THE VERIFY TABLE            09300000
         BAL   R14,CLOSEDS         CLOSE AND FREE OUTPUT DATA SET       09350000
         L     R13,SAVE1+4         GET CALLERS SAVE AREA ADDRESS        09400000
         L     R2,RETCODE          GET RETURN CODE                      09450000
         SLR   R1,R1               GET A ZERO                           09500000
         ST    R1,ADUUFLD          CLEAR THE USER FIELD TO AVOID        09550000
*                                  INADVERTENT USE OF FREEMAINED        09600000
*                                  STORAGE                              09650000
         FREEMAIN RU,LV=DATALEN,SP=0,A=(R11)  FREE DYNAMIC AREA         09700000
         B     NOR13               BYPASS R13 RESTORE                   09750000
NOFREE   DS    0H                  BYPASS FREEMAIN                      09800000
         L     R13,4(R13)          RESTORE CALLERS SAVE AREA            09850000
NOR13    DS    0H                  BYPASS R13 RESTORE IF FREEMAIN PATH  09900000
         L     R14,12(R13)         GET RETURN ADDRESS                   09950000
         LR    R15,R2              GET RETURN CODE                      10000000
         LM    R0,R12,20(R13)      RESTORE CALLERS REGS                 10050000
         BSM   0,R14               RETURN TO CALLER                     10100000
         EJECT                                                          10150000
         TITLE 'MCVERIFY - TEST MANAGEMENT CLASS FOR DELETION'          10151000
MCVERIFY DS    0H                  MC VERIFY SUBROUTINE                 10152000
********************************************************************** 10154000
*                                                                    * 10154400
* SUBROUTINE:  MCVERIFY                                              * 10154800
*                                                                    * 10155200
* FUNCTION:    Search the in storage management class table to       * 10155600
```

```
*              determine if this object can be deleted. if the input  * 10156000
*              management class is listed in the table then no        * 10156400
*              objects with this management class are to be deleted.  * 10156800
*                                                                     * 10157200
* OPERATION:                                                          * 10157600
*     1.  Load total number of Management Class Name Entries.         * 10158000
*     2.  Load address of the MC table.                               * 10158400
*     3.  If ADUMCNAM matches current MC table entry then set a       * 10158800
*         return code of 4 and a reason code of 1, do not delete      * 10159200
*         object but continue to process storage group.               * 10159600
*     4.  Return to calling module.                                   * 10160000
*                                                                     * 10160400
* CALLED BY:   Add a call from main line if desired                   * 10160800
*                                                                     * 10161200
* CALLS:       None                                                   * 10161600
*                                                                     * 10162000
* NOTE: This subroutine is provided as an example only and is not     * 10162400
*       called from the main line code.  Modify the main line code    * 10162800
*       as needed to invoke this routine if desired.                  * 10163200
*                                                                     * 10163600
********************************************************************** 10166000
         SPACE 1                                                      10168000
         ST    R14,SAVE14          SAVE RETURN ADDRESS               10170000
         LA    R11,MCCNT           LOAD NUMBER OF TABLE ENTRIES      10172000
         LA    R10,MCTAB           LOAD ADDRESS OF TABLE IN R10      10174000
         USING TAB,R10             USE DSECT                         10176000
COMPMC   CLC   MCNAME,ADUMCNAM     COMPARE MC NAMES                  10178000
         BE    VMCMATCH                                              10180000
         LA    R10,TABLEN(,R10)                                      10182000
         BCT   R11,COMPMC                                            10184000
VMCMATCH DS    0H                  MC NAME IS IN VERIFY TABLE        10186000
         LA    R1,ADU4RS01         LOAD REASON CODE OF 1             10186600
         ST    R1,REASCODE         STORE REASON CODE                 10187200
         LA    R1,ADUNODEL         GOOD COMPARE, DO NOT DELETE       10188000
         ST    R1,RETCODE          SET DO NOT DELETE RETURN CODE     10190000
ENDMC    DS    0H                  END MC VERIFY SUBROUTINE          10192000
         L     R14,SAVE14          GET SAVED R14                     10194000
         BR    R14                 RETURN TO MAINLINE                10196000
         EJECT                                                       10198000
         TITLE 'OPENDS - OPEN INPUT/OUTPUT DATASETS'                 10199000
OPENDS   DS    0H                  OPENDS SUBROUTINE                 10200000
********************************************************************** 10250000
*                                                                   * 10300000
* SUBROUTINE:  OPENDS                                               * 10350000
*                                                                   * 10400000
* FUNCTION:    Open the input data set and create the verify table.  * 10437500
*              Then open the output data set for object deletion    * 10475000
*              notification.                                        * 10512500
*                                                                   * 10550000
* OPERATION:                                                        * 10560700
*     1.  Prepare Data Area for Input and Output control blocks,    * 10571400
*         along with the necessary SVC 99 RB.                       * 10582100
*.    2.  Begin input (Verify) data set processing.                 * 10592800
*     3.  Verify that the input data set was allocated successfully, * 10603500
*         on failure set RC=12 and REAS=1. (ADUX not called again)  * 10614200
*     4.  Verify that the input data set is catalogued, on failure  * 10624900
*         set RC=12 and REAS=2. (ADUX not called again)             * 10635600
*     5.  Perform an OPEN on the Verify Data Set using OPEN MACRO.  * 10646300
*     6.  Verify input data set was opened successfully, on failure * 10657000
*         set RC=12 and REAS=3. (ADUX not called again)             * 10667700
*     7.  Obtain storage to build the Verify table, using GETMAIN   * 10678400
*         MACRO using Subpool=0.                                    * 10689100
*     8.  Verify GETMAIN succeeded, on failure set RC=12, REAS=4    * 10699800
```

```
*         (ADUX not called again)                                  * 10710500
*     9.  Initialize the initial verify table and store a pointer to  * 10721200
*         the table in the Data Area.                              * 10731900
*     10. Read a record from input data set using GET MACRO, on    * 10742600
*         failure call ADUXSYN and end input processing and read   * 10753300
*         until end of file.                                       * 10764000
*     11. Increment and store counter for the number of records.   * 10774700
*     12. If counter equals MAXNAMES then branch to obtain a new   * 10785400
*         block of storage for Verify Table, otherwise go to step 10. * 10796100
*         A. Obtain storage to get a new block for Verify table using * 10806800
*            GETMAIN MACRO with subpool=0.                         * 10817500
*         B. Verify GETMAIN succeeded, on failure RC=12 and REAS=5    * 10828200
*            (ADUX not called again)                               * 10838900
*         C. Initialize the new block of storage for Verify Table.    * 10849600
*     13. Upon end of file condition close the dataset using CLOSE    * 10860300
*         MACRO.                                                   * 10871000
*     14. Verify no errors have occurred while processing input before* 10881700
*         beginning output data set processing, on failure branch to  * 10892400
*         end of input/output data set processing.                 * 10903100
*     15. Begin output data set (NOTIFY) processing by updating the   * 10913800
*         output data set name with Storage Group name.            * 10924500
*     16. Optionally update the output data set name with DB2SSID     * 10924501
*     17. Update Text Unit pointers for the Notify data set.       * 10935200
*     18. Perform DYNALLOC for output data set and verify no errors   * 10945900
*         occurred, on failure set RC=12 and REAS=6                * 10956600
*     19. Attempt to open the output data set using the OPEN MACRO,   * 10967300
*         on failure set RC=12 and REAS=7 (ADUX not called again)  * 10978000
*     20. Return to main line code.                                * 10988700
*                                                                  * 11000000
* REGISTER CONVENTIONS:                                           * 11037500
*     R8  - DCB Pointer                                            * 11075000
*     R11 - Data Area Pointer                                      * 11112500
*                                                                  * 11150000
****************************************************************** 11200000
         SPACE 2                                                    11250000
         ST    R14,SAVE14          SAVE RETURN ADDRESS              11300000
         LA    R2,RENTAREA         GET TARGET ADDRESS FOR MOVE      11350000
         LA    R3,MOVELN           GET THE LENGTH OF THE MOVE       11400000
         LA    R6,STATAREA         GET SOURCE ADDRESS OF THE MOVE   11450000
         LR    R7,R3               GET LENGTH OF THE MOVE AND       11500000
*                                  PAD WITH ZEROS                   11550000
         MVCL  R2,R6               COPY CONTROL BLOCKS TO           11600000
*                                  DYNAMIC AREA FROM STATIC AREA    11650000
         LA    R2,DYNRB            GET DYNAMIC RB ADDRESS           11700000
         ST    R2,S99RBPTR         STORE IN DYNAMIC RB POINTER FIELD 11750000
         OI    S99RBPTR,X'80'      SET END OF LIST FLAG             11800000
         LA    R2,TXTPTRV          GET DYNAMIC VERSION OF TXTPTR    11850000
         ST    R2,TXTPTR           STORE IN SVC 99 RB DYNAMIC VERSION 11900000
         LA    R2,0                GET ZERO FOR DDNAME TEXT UNIT PTR 11908300
         ST    R2,TXTDDN           PUT IT IN THE TXT UNIT PTR       11916600
         MVC   RETTXT(6),RETSTAT   MOVE FROM STATIC TO DYNAMIC      11924900
         LA    R2,RETTXT           LET SYSTEM DETERMINE DDNAME      11933200
         ST    R2,TXTRET                                            11941500
         LA    R1,S99RBPTR         SETUP REG 1 FOR DYNAMIC ALLOC.   11970700
         DYNALLOC                  ALLOCATE INPUT DATA SET          12000000
         LA    R2,4                GET DYNAMIC ENVIRONMENT ERROR    12050000
         CR    R15,R2              WAS DYNAMIC ALLOC. OKAY          12100000
         BL    DOOPENI             YES, OPEN VERIFY DATA SET        12150000
         BE    CHKNODS             IF ERROR CHECK IF NO DATA SET    12175000
         LA    R15,ADUCRS01        SET REASON CODE OF 1             12200000
         ST    R15,REASCODE        STORE REASON CODE                12225000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX   12250000
         ST    R15,RETCODE         SET RETURN CODE                  12300000
```

```
         B     ENDOPEN             END ADUX                                12350000
CHKNODS  DS    0H                  ENVIRONMENT ERROR                       12400000
         LH    R15,=X'1708'        GET NO CATALOGED DS FAIL CODE           12450000
         CH    R15,DYNRB+4         FAILED BECAUSE DS NOT CATALOGED         12500000
         BE    NOOPENI             SO PROCEED WITHOUT VERIFY               12550000
         LA    R15,ADUCRS02        SET REASON CODE OF 2                    12566600
         ST    R15,REASCODE        STORE REASON CODE                       12583200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX          12600000
         ST    R15,RETCODE         SET RETURN CODE                         12650000
         B     ENDOPEN             END ADUX                                12700000
DOOPENI  DS    0H                  ALLOCATED, DO OPEN                      12750000
         LA    R1,OPENL            GET OPEN LIST ADDRESS                   12800000
         LA    R8,DCBI             GET INPUT DCB ADDRESS                   12850000
         USING IHADCB,R8                                                   12950000
         MVC   DCBDDNAM,RETTXT+6   MOVE SYSTEM GENERATED DDN INTO DCB      12966600
         OPEN  ((R8),INPUT),MF=(E,(1)) OPEN INPUT DCB                      12983200
         TM    DCBOFLGS,DCBOFOPN   WAS DS OPENED PROPERLY                  13000000
         BO    OPENI               BUILD VERIFY TABLE                      13050000
         LA    R15,ADUCRS03        LOAD REASON CODE OF 3                   13066600
         ST    R15,REASCODE        STORE REASON CODE                       13083200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX          13100000
         ST    R15,RETCODE         SET RETURN CODE                         13150000
         B     ENDOPEN             END ADUX                                13200000
*                                                                          13212500
*        Build Verify Table                                                13225000
*                                                                          13237500
OPENI    DS    0H                                                          13250000
         GETMAIN RU,LV=4096,SP=0   GET BLOCK FROM SUBPOOL ZERO             13300000
*                                                                          13309000
         LTR   R15,R15             VERIFY GETMAIN WAS SUCCESSFUL           13318000
         BZ    BTABBEG             BRANCH TO BUILD INITIAL TABLE           13327000
         LA    R15,ADUCRS04        LOAD REASON CODE OF 4                   13336000
         ST    R15,REASCODE        STORE REASON CODE                       13345000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX          13354000
         ST    R15,RETCODE         SET RETURN CODE                         13363000
         B     ENDOPEN             END OPEN DATA SET                       13372000
*                                                                          13381000
BTABBEG  LR    R5,R1               GET THE VERIFY TABLE ADDRESS            13390000
         SLR   R6,R6               GET A ZERO                              13400000
*                                   ALSO, USE AS A RECORD COUNTER          13450000
         ST    R6,0(,R5)           INDICATE IN-CORE TABLE NOT YET SET      13500000
         ST    R6,4(,R5)           INDICATE IN-CORE TABLE NOT YET SET      13550000
         LA    R9,8(,R5)           GET PAST RECORD COUNTER                 13600000
         ST    R5,VRECSPTR         SAVE TABLE ADDRESS                      13650000
GETINPUT DS    0H                                                          13700000
         LA    R1,DCBI             GET INPUT DCB ADDRESS                   13750000
         LR    R0,R9               ADDRESS OF RECORD                       13800000
         GET   (1),(0)             READ A RECORD                           13850000
         L     R2,RETCODE          CHECK FOR SYNAD ENTRY                   13900000
         LTR   R2,R2               IF NON-ZERO SYNAD ENTERED               13950000
         BNZ   REODAD              CLOSE DATA SET AND END                  14000000
         LA    R9,89(R9)           GET TO NEXT SLOT IN VERIFY TBL          14050000
         LA    R6,1(0,R6)          INCREMENT NUMBER OF RECORDS             14100000
         ST    R6,0(0,R5)          UPDATE COUNTER IN BLOCK                 14150000
         C     R6,MAXNAMES         SEE IF MAX NAMES IN TABLE YET           14200000
         BE    GNEXTBLK            IF SO, GET A NEW BLOCK                  14250000
         B     GETINPUT            READ UNTIL EODOD                        14300000
GNEXTBLK DS    0H                                                          14350000
         GETMAIN RU,LV=4096,SP=0   GET BLOCK FROM SUBPOOL ZERO             14400000
*                                                                          14409000
         LTR   R15,R15             VERIFY GETMAIN WAS SUCCESSFUL           14418000
         BZ    BTABNEXT            BRANCH TO BUILD INITIAL TABLE           14427000
         LA    R15,ADUCRS05        LOAD REASON CODE OF 5                   14436000
```

```
         ST    R15,REASCODE        STORE REASON CODE                    14445000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       14454000
         ST    R15,RETCODE         SET RETURN CODE                      14463000
         B     ENDOPEN             END OPEN DATA SET                    14472000
*                                                                       14481000
BTABNEXT ST    R1,4(0,R5)          CHAIN TO CURRENT TABLE               14490000
         LR    R5,R1               GET THE VERIFY TABLE ADDRESS         14500000
         SLR   R6,R6               CLEAR TO ZERO                        14550000
         ST    R6,0(,R5)           INDICATE IN-CORE TABLE NOT YET SET   14600000
         ST    R6,4(,R5)           INDICATE IN-CORE TABLE NOT YET SET   14650000
         LA    R9,8(,R5)           GET PAST RECORD COUNTER              14700000
         B     GETINPUT            READ UNTIL EODOD                     14750000
REODAD   DS    0H                  END OF DATA ON READ                  14800000
         LA    R2,DCBI             GET INPUT DCB POINTER                14850000
         LA    R1,OPENL            GET AREA FOR CLOSE LIST              14900000
         CLOSE ((R2)),MF=(E,(1))   CLOSE INPUT DCB                      14950000
         FREEPOOL DCBI             RELEASE BUFFER POOL                  14975000
NOOPENI  DS    0H                  NO VERIFY DATA SET                   15000000
         L     R2,RETCODE          CHECK FOR ERROR                      15050000
         LTR   R2,R2               IF NON-ZERO ERROR OCCURRED           15100000
         BNE   ENDOPEN             END PROCESSING                       15150000
*                                                                       15160000
* BEGIN OUTPUT DATASET PROCCESSING ADDING DB2SSID TO DS NAME IF CCC     15165000
* LINES ARE UNCOMMENTED                                                 15170000
*                                                                       15170001
*        LA    R6,NTFYDB2          LOAD ADDRESS OF DB2SSID IN       CCC 15170002
*                                   NOTIFY DATA SET NAME                 15170003
*        LA    R8,ADUDSSID         LOAD ADDRESS OF DB2SSID FROM     CCC 15170005
*                                   CBRADUP                              15170006
*        CLC   0(1,R8),=CL1' '     COMPARE 1ST CHAR SSID WITH BLANK CCC 15170007
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170008
*        MVC   0(1,R6),0(R8)       MOVE 1ST CHAR SSID INTO THE      CCC 15170009
*                                   NOTIFY DATA SET NAME                 15170010
*        CLC   1(1,R8),=CL1' '     COMPARE 2ND CHAR SSID WITH BLANK CCC 15170011
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170012
*        MVC   1(1,R6),1(R8)       MOVE 2ND CHAR SSID INTO THE      CCC 15170013
*                                   NOTIFY DATA SET NAME                 15170014
*        CLC   2(1,R8),=CL1' '     COMPARE 3RD CHAR SSID WITH BLANK CCC 15170015
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170016
*        MVC   2(1,R6),2(R8)       MOVE 3RD CHAR SSID INTO THE      CCC 15170017
*                                   NOTIFY DATA SET NAME                 15170018
*        CLC   3(1,R8),=CL1' '     COMPARE 4TH CHAR SSID WITH BLANK CCC 15170019
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170020
*        MVC   3(1,R6),3(R8)       MOVE 4TH CHAR SSID INTO THE      CCC 15170021
*                                   NOTIFY DATA SET NAME                 15170022
*ENDSSID  DS    0H                                                  CCC 15170023
*                                                                       15170800
* BEGIN OUTPUT DATASET PROCCESSING                                      15191600
*                                                                       15212400
         LA    R6,NTFYDS2          LOAD ADDRESS OF SG NAME IN VERIFY    15233200
*                                   DATA SET NAME                        15254000
         L     R7,ADUSGLEN         LOAD LENGTH OF SG NAME               15274800
         LA    R8,ADUSGNAM         LOAD ADDRESS OF SG NAME FROM         15295600
*                                   CBRADUP                              15316400
         LR    R9,R7               PREPARE FOR MVCL                     15337200
         MVCL  R6,R8               MOVE SG NAME INTO THE VERIFY         15358000
*                                   DATA SET NAME                        15378800
*                                                                       15400000
* UPDATE TEXT UNIT POINTERS FOR THE NOTIFY DATA SET                     15450000
*                                                                       15500000
         LA    R3,0                GET ZERO FOR DDNAME TEXT UNIT PTR    15516600
         ST    R3,TXTDDN           PUT IT IN THE TXT UNIT PTR           15533200
         LA    R3,DSNTXTN          GET NOTIFY DSNAME TEXT UNIT PTR      15550000
```

```
         ST    R3,TXTDSN           PUT IT IN THE TXT UNIT PTR          15600000
*                                  INSTEAD OF VERIFY DSNAME            15650000
         LA    R3,MODTXT           GET STAT=SHR TEXT UNIT PTR          15800000
         ST    R3,TXTSTAT          STORE NEW TEXT UNIT PTR             15850000
         LA    R3,CTLGTXT          GET DISP=,KEEP TEXT UNIT PTR        15900000
         ST    R3,TXTDISP          STORE NEW TEXT UNIT PTR             15950000
         LA    R3,TRKTXT           GET TRACK ALLOATION TU PTR          16000000
         ST    R3,TXTTRK           STORE NEW TEXT UNIT PTR             16050000
         LA    R3,PRIMTXT          GET PRIMARY AMOUNT TU PTR           16100000
         ST    R3,TXTPRIM          STORE NEW TEXT UNIT PTR             16150000
         LA    R3,SECTXT           GETSECONDARY AMOUNT TU PTR          16200000
         ST    R3,TXTSEC           STORE NEW TEXT UNIT PTR             16250000
         MVC   RETTXT(6),RETSTAT   MOVE FROM STATIC TO DYNAMIC         16262500
         LA    R3,RETTXT           LET SYSTEM DETERMINE DDNAME         16275000
         ST    R3,TXTRET                                               16287500
         LA    R1,S99RBPTR         SET UP FOR SVC 99                   16300000
         DYNALLOC                  FREE THE DATA SET                   16350000
         LTR   R15,R15             IF DYNALLOC OKAY                    16400000
         BZ    OPENO               OPEN NOTIFY DATA SET                16450000
         LA    R15,ADUCRS06        LOAD REASON CODE OF 6               16466600
         ST    R15,REASCODE        STORE REASON CODE                   16483200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX      16500000
         ST    R15,RETCODE         SET RETURN CODE                     16550000
         B     ENDOPEN             END ADUX                            16600000
OPENO    DS    0H                  OPEN OUTPUT DATA SET                16650000
         LA    R1,OPENL            GET OPEN LIST ADDRESS               16700000
         LA    R8,DCBO             GET INPUT DCB ADDRESS               16750000
         MVC   DCBDDNAM,RETTXT+6   MOVE SYSTEM GENERATED DDN INTO DCB  16775000
         OPEN  ((R8),OUTPUT),MF=(E,(1)) OPEN OUTPUT DCB                16800000
         USING IHADCB,R8                                               16850000
         TM    DCBOFLGS,DCBOFOPN   WAS DS OPENED PROPERLY              16900000
         BNO   NOOPENO             SET NO NOTIFY TABLE                 16950000
         OI    FLAGS,ODCBO         INDICATE NOTIFY DATA SET IS OPEN    17000000
         B     ENDOPEN             END DATA SET OPENING                17050000
NOOPENO  DS    0H                  OUTPUT DATA SET DID NOT OPEN        17100000
         LA    R2,ADUCRS07         LOAD REASON CODE OF 7               17116600
         ST    R2,REASCODE         STORE REASON CODE                   17133200
         LA    R2,ADUFAILN         SET FAILING RETURN CODE             17150000
         ST    R2,RETCODE          DO NOT RECALL ADUX                  17200000
ENDOPEN  DS    0H                  END OF OPENDS SUBROUTINE            17250000
         L     R14,SAVE14          GET RETURN ADDRESS                  17300000
         BR    R14                 RETURN TO MAINLINE                  17350000
         EJECT                                                         17400000
VERIFY   DS    0H                  VERIFY SUBROUTINE                   17450000
*********************************************************************** 17500000
*                                                                     * 17512500
* SUBROUTINE:  VERIFY                                                 * 17525000
*                                                                     * 17537500
* FUNCTION:    Search the Verify Table, built by OPENDS.  If the      * 17550000
*              object name is in the table then send a return code    * 17562500
*              indicating that auto-deletion should not occur.        * 17575000
*                                                                     * 17587500
* OPERATION:                                                          * 17600000
*     1.  Load address of the table pointer from Data Area.           * 17612500
*     2.  Retrieve the number of entries from Verify Table.           * 17625000
*     3.  If number of entries = 0 then end VERIFY.                   * 17637500
*     4.  While Names and collections to verify                       * 17650000
*         A. If ADUONAME = row in Verify Table, then check Collection * 17662500
*         B. Read next record from Verify Table.                      * 17675000
*         C. If entries = 0 then move to next Verify Table.           * 17687500
*         D. If ADUCLNAM = row in Verify Table, then set RC=4 and     * 17700000
*            REAS=2 and do not delete object.                         * 17712500
*     5.  Branch back to main line code.                              * 17725000
```

```
*                                                                     * 17737500
********************************************************************** 17750000
         SPACE 1                                                      17800000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                17850000
         L     R5,VRECSPTR         GET FIRST TABLE ADDRESS            17900000
COMPLOOP DS    0H                  COMPARE LOOP                       17950000
         L     R3,0(,R5)           GET COUNT OF ENTRIES               18000000
         LA    R2,8(,R5)           GET ADDRESS OF FIRST NAME          18050000
         LTR   R3,R3               IF EMPTY FILE                      18066600
         BZ    ENDVERF             GO TO ENDVERF                      18083200
COMPNAME DS    0H                  NAME COMPARE LOOP                  18100000
         CLC   0(44,R2),ADUONAME   COMPARE NAME IN VERIFY TABLE       18150000
*                                  AGAINST OBJECT BEING DELETED       18200000
         BE    VCOLNAM             IF MATCH, CHECK COLLECTION NAME    18250000
DIFCOLNM LA    R2,89(,R2)          GET ADDRESS OF NEXT NAME           18300000
         BCT   R3,COMPNAME         DECREMENT COUNTER BY 1             18350000
         L     R5,4(,R5)           GET NEXT TABLE SECTION ADDRESS     18400000
         LTR   R5,R5               IF THERE IS A NEXT SECTION         18450000
         BZ    ENDVERF             END VERIFY IF NOT                  18500000
         B     COMPLOOP            COMPARE NEXT NAME                  18550000
VCOLNAM  DS    0H                  OBJ MATCH, CHECK COL NAME          18560000
         CLC   45(44,R2),ADUCLNAM  OBJ MATCH, SAME COL NAME?          18570000
         BE    VMATCH              IF MATCH, DO NOT DELETE            18580000
         B     DIFCOLNM            NO MATCH, GET NEXT OBJ NAME        18590000
VMATCH   DS    0H                  OBJECT NAME IS IN VERIFY TABLE     18600000
         LA    R1,ADU4RS02         LOAD REASON CODE OF 2              18616600
         ST    R1,REASCODE         STORE REASON CODE                  18633200
         LA    R1,ADUNODEL         GOOD COMPARE, DO NOT DELETE        18650000
         ST    R1,RETCODE          SET DO NOT DELETE RETURN CODE      18700000
ENDVERF  DS    0H                  END VERIFY SUBROUTINE              18750000
         L     R14,SAVE14          GET SAVED R14                      18800000
         BR    R14                 RETURN TO MAINLINE                 18850000
         EJECT                                                        18900000
NOTIFY   DS    0H                  NOTIFY SUBROUTINE                  18950000
********************************************************************** 19000000
*                                                                     * 19016600
* SUBROUTINE:  NOTIFY                                                 * 19033200
*                                                                     * 19049800
* FUNCTION:    Write the object and collection names to the notify    * 19066400
*              data set.  This data set can be read by another        * 19083000
*              application for removal of object information.         * 19099600
*                                                                     * 19116200
* OPERATION:                                                          * 19132800
*     1.  Load address of the output DCB.                             * 19149400
*     2.  Load address of Object name and collection name output line * 19166000
*         for output to NOTIFY data set.                              * 19182600
*     3.  Write record to data set using PUT MACRO.                   * 19199200
*     4.  Branch back to main line processing.                        * 19215800
*                                                                     * 19232400
********************************************************************** 19250000
         SPACE 1                                                      19300000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                19350000
         LA    R1,DCBO             GET OUTPUT DCB ADDRESS             19400000
         LA    R0,ADUOBJCL         GET OBJECT NAME/COLL. ADDRESS      19450000
         PUT   (1),(0)             WRITE NAME TO NOTIFY DATA SET      19500000
         L     R14,SAVE14          GET SAVED R14                      19550000
         BR    R14                 RETURN TO MAINLINE                 19600000
         EJECT                                                        19650000
FREETAB  DS    0H                  FREEMAIN VERIFY TABLE SUBROUTINE   19700000
********************************************************************** 19750000
*                                                                     * 19761500
* SUBROUTINE:  FREETAB                                                * 19773000
*                                                                     * 19784500
```

```
* FUNCTION:    Free the in-storage Verify Table.                     * 19796000
*                                                                    * 19807500
* OPERATION:                                                         * 19819000
*     1.  Load address of the Verify Table Head.(VRECSPTR)           * 19830500
*     2.  While (tables exist [Pointer(R5) != 0])                    * 19842000
*         A.  Free the table storage by using the FREEMAIN macro.    * 19853500
*         B.  Load the next table address from current table.        * 19865000
*     3. Return to main line code.                                   * 19876500
*                                                                    * 19888000
********************************************************************** 19900000
         SPACE 1                                                      19950000
         ST    R14,SAVE14          SAVE RETURN ADDRESS               20000000
         L     R5,VRECSPTR         GET FIRST TABLE ADDRESS           20050000
FREELOOP DS    0H                  LOOP THROUGH CHAINED TABLES       20100000
         LTR   R5,R5               IS THERE A TABLE ADDRESS          20200000
         BZ    ENDFREE             IF NOT, END FREEMAIN LOOP         20250000
         L     R3,4(,R5)           GET NEXT TABLE ADDRESS            20275000
         FREEMAIN RU,LV=4096,A=(R5) FREE TABLE SECTION                20300000
         LR    R5,R3               ADDRESS NEXT SECTION TO FREE      20350000
         B     FREELOOP            FREEMAIN NEXT SECTION             20400000
ENDFREE  DS    0H                  END OF TABLE FREEMAIN LOOP        20450000
         L     R14,SAVE14          GET SAVED R14                     20500000
         BR    R14                 RETURN TO MAINLINE                20550000
         EJECT                                                       20600000
CLOSEDS  DS    0H                  CLOSE DATA SET SUBROUTINE         20650000
********************************************************************** 20700000
*                                                                  * 20750000
* SUBROUTINE:  CLOSEDS                                             * 20800000
*                                                                  * 20850000
* FUNCTION:    Closes the Notify data set.                         * 20858300
*                                                                  * 20866600
* OPERATION:                                                       * 20874900
*     1.  Check to see if Notify data set is currently open.       * 20883200
*     2.  Prepare for close macro.                                 * 20891500
*         A. Load address of the Open List. (R1)                   * 20899800
*         B. Load address of the Notify DCB. (R2)                  * 20908100
*     3.  Perform a close on the Notify data set using the CLOSE   * 20916400
*         macro.                                                   * 20924700
*     4.  Free the DCB buffer pool using the FREEPOOL macro.       * 20933000
*     5.  Return to main line code.                                * 20941300
*                                                                  * 20950000
******************************************************************** 21000000
         SPACE 1                                                     21050000
         ST    R14,SAVE14          SAVE RETURN ADDRESS              21100000
         TM    FLAGS,ODCBO         CHECK TO SEE IF OUTPUT DS IS OPEN 21150000
         BNO   NOCLOSE             DO NOT ISSUE CLOSE IF NOT OPEN   21200000
         LA    R1,OPENL            GET AREA FOR CLOSE LIST          21250000
         LA    R2,DCBO             GET DCB ADDRESS                  21300000
         CLOSE ((R2)),MF=(E,(1))   CLOSE OUTPUT DCB NOTIFY DSN      21333300
         FREEPOOL DCBO             RELEASE OUTPUT BUFFER POOL       21366600
NOCLOSE  DS    0H                  DO NOT CLOSE                     21400000
         L     R14,SAVE14          GET SAVED R14                    21450000
         BR    R14                 RETURN TO MAINLINE               21500000
         EJECT                                                      21550000
ADUXSYN  DS    0H                  BEGINNING OF WTO SUBROUTINE      21600000
******************************************************************** 21650000
*                                                                  * 21700000
* SUBROUTINE:  ADUXSYN                                             * 21750000
*                                                                  * 21800000
* FUNCTION:    Handles input and output errors during file operations.* 21825000
*                                                                  * 21850000
* OPERATION:                                                       * 21875000
*     1.  Set a RC=12 and REAS=8 (ADUX not called again) due to I/O * 21900000
```

```
*         error.                                                      * 21925000
*     2.  Store the return code for ADUX.                             * 21950000
*     3.  Return to calling module.                                   * 21975000
*                                                                     * 22000000
********************************************************************* 22050000
         SPACE 1                                                      22100000
         LA    R2,ADUCRS08         LOAD REASON CODE OF 8              22116600
         ST    R2,REASCODE         STORE REASON CODE                  22133200
         LA    R2,ADUFAILN         SET FAIL CODE TO NO DELETE AND     22150000
*                                  DO NOT CALL ADUX FOR THIS SG       22183300
         ST    R2,RETCODE          STORE RETURN CODE                  22216600
         BR    R14                 RETURN TO MAINLINE                 22250000
         EJECT                                                        22300000
ADUXOUT  DS    0H                                                     22300600
********************************************************************* 22301200
*                                                                     * 22301800
* SUBROUTINE:  ADUXOUT                                               * 22302400
*                                                                     * 22303000
* FUNCTION:    Writes to the console to show if an error occurred     * 22303600
*              while the CBRHADUX was processing.                     * 22304200
*                                                                     * 22304800
* OPERATION:                                                          * 22305400
*     1.  Determine whether the ADUXOUT should be performed based on  * 22306000
*         ADUFUNC and return code severity.                           * 22306600
*     2.  Copy from static to dynamic memory, the WTO macro list      * 22307200
*         definition along with the print line label.                 * 22307800
*     3.  Move current Storage group name to WTO message line.        * 22308400
*     4.  Convert the current return code and move to the WTO message * 22309000
*         line.                                                       * 22309600
*     5.  Convert the current reason code and move to the WTO message * 22310200
*         line.                                                       * 22310800
*     6.  Prepare to perform the WTO macro.                           * 22311400
*         A. Set the address of the message. (R5)                     * 22312000
*         B. Clear register 0 to zero.                                * 22312600
*     7.  Write to operator using the WTO macro.                      * 22313200
*     8.  Return to main line code.                                   * 22313800
*                                                                     * 22314400
* NOTE: If you change the return code severity to allow anything      * 22315000
*       below 8, can cause unexpected results, such as the WTO        * 22315600
*       could become full.  Modify the value of RETMIN to adjust the  * 22316200
*       severity of messages to be displayed.                         * 22316800
*                                                                     * 22317400
********************************************************************* 22318000
         SPACE 1                                                      22318600
         ST    R14,SAVE14          SAVE RETURN ADDRESS                22319200
*                                                                     22319800
         CLI   ADUFUNC,ADUDONE     IS THIS AN END OF CYCLE CALL       22320400
         BE    OUTDONE             DO NOT PERFORM WTO ON EQUAL        22321000
         L     R5,RETCODE          LOAD CURRENT RETURN CODE           22321600
         C     R5,RETMIN           IF RETURN CODE < RETMIN THEN       22322200
         BL    OUTDONE              ENDADUX OUT                       22322800
         MVC   WTOLIST(WTOLISTL),WTOMODEL  COPY FROM MODULE STORAGE   22323400
*                                          TO WORKING STORAGE         22324000
         MVC   STATLINE(STATLEN),STATMODL  COPY FROM MODULE STORAGE   22324600
*                                          TO WORKING STORAGE         22325200
         MVC   STATSG(8),ADUSGNAM  MOVE SG TO STATUS LINE             22325800
         L     R5,RETCODE          LOAD CURRENT RETURN CODE           22326400
         CVD   R5,CONVERT          CONVERT TO PACK FORMAT             22327000
         UNPK  HEXWORK(8),CONVERT+4(4)  UNPACK FOR CONVERSION         22327600
         OI    HEXWORK+7,X'F0'     FORMAT LAST BYTE TO EBCDIC         22328200
         MVC   STATRET(8),HEXWORK  MOVE TO STATUS LINE                22328800
         L     R5,REASCODE         LOAD CURRENT REASON CODE           22329400
         CVD   R5,CONVERT          CONVERT TO PACK FORMAT             22330000
```

```
         UNPK  HEXWORK(8),CONVERT+4(4)  UNPACK FOR CONVERSION           22330600
         OI    HEXWORK+7,X'F0'     FORMAT LAST BYTE TO EBCDIC          22331200
         MVC   STATREAS(8),HEXWORK MOVE TO STATUS LINE                 22331800
*                                                                      22332400
*        PERFORM WRITE TO OPERATOR REQUEST                             22333000
*                                                                      22333600
         LR    R5,R11              LOAD WORKING STORAGE AREA           22334200
         LA    R6,STATDISP         LOAD OFFSET OF STATUS LINE          22334800
         AR    R5,R6               FIND DISPLACEMENT TO STATUS         22335400
*                                  LINE FROM WORKING STORAGE           22336000
         SR    R0,R0               CLEAR REGISTER 0                    22336600
         WTO   TEXT=(R5),                                             +22337200
               MF=(E,WTOLIST)      PERFORM WTO TO CONSOLE              22337800
OUTDONE  L     R14,SAVE14          RESTORE RETURN ADDRESS              22338400
         BR    R14                 RETURN TO MAINLINE                  22339000
         EJECT                                                         22339600
********************************************************************** 22340200
*                                                                    * 22340800
*        LITERAL DEFINITIONS                                         * 22341400
*                                                                    * 22342000
********************************************************************** 22342600
         LTORG ,                   ORGANIZE LITERALS                   22343200
R0       EQU   0                                                       22350000
R1       EQU   1                                                       22400000
R2       EQU   2                                                       22450000
R3       EQU   3                                                       22500000
R4       EQU   4                                                       22550000
R5       EQU   5                                                       22600000
R6       EQU   6                                                       22650000
R7       EQU   7                                                       22700000
R8       EQU   8                                                       22750000
R9       EQU   9                                                       22800000
R10      EQU   10                                                      22850000
R11      EQU   11                                                      22900000
R12      EQU   12                                                      22950000
R13      EQU   13                                                      23000000
R14      EQU   14                                                      23050000
R15      EQU   15                                                      23100000
*--------------------------------------------------------------------* 23100100
*                                                                    * 23100200
*        MAX RETURN CODE TO DETERMINE WHAT SEVERITY OF MESSAGE       * 23100300
*         SHOULD BE REPORTED                                         * 23100400
*                                                                    * 23100500
*--------------------------------------------------------------------* 23100600
RETMIN   DC    F'8'                HIGHEST RETURN CODE THAT THE     MMM 23100700
*                                  ADUXOUT SHOULD WTO MESSAGES FOR      23100800
*--------------------------------------------------------------------* 23100900
*                                                                    * 23101000
*        REASON CODES FOR DIAGNOSTIC REASONS                         * 23101100
*                                                                    * 23101200
*--------------------------------------------------------------------* 23101300
*                                                                      23101400
* RETURN CODE ADUNODEL = RETCODE = 4                                   23101500
*                                                                      23101600
ADU4RS01 EQU   1    MANAGEMENT CLASS REJECTED OBJECT TO BE DELETED    23101700
ADU4RS02 EQU   2    OBJECT RESIDES IN VERIFY DATASET, NO DELETION     23101800
*                                                                      23101900
* RETURN CODE ADUFAILC = RETCODE = 8                                   23102000
*                                                                      23102100
ADU8RS01 EQU   1    DELETE ALL OBJECTS FOR THIS STORAGE GROUP AND     23102200
*                   DO NOT CALL EXIT AGAIN                           23102300
*                                                                      23102400
* RETURN CODE ADUFAILN = RETCODE = 12                                 23102500
```

```
*                                                                      23102600
ADUCRS01 EQU   1    DYNALLOC FAILED FOR VERIFY DATASET                 23102700
ADUCRS02 EQU   2    VERIFY DATASET NOT CATOLOGED                       23102800
ADUCRS03 EQU   3    VERIFY DATASET FAILED TO OPEN                      23102900
ADUCRS04 EQU   4    GETMAIN FAILED FOR INITIAL ON VERIFY TABLE         23103000
ADUCRS05 EQU   5    GETMAIN FAILED FOR NEXT ON VERIFY TABLE            23103100
ADUCRS06 EQU   6    DYNALLOC FAILED FOR NOTIFY DATASET                 23103200
ADUCRS07 EQU   7    NOTIFY DATASET FAILED TO OPEN                      23103300
ADUCRS08 EQU   8    I/O ERROR OCCURRED DURING GETMAIN                  23103400
*--------------------------------------------------------------------* 23103500
*                                                                    * 23103600
*        SINGLE LINE WTO PARAMETER LIST                              * 23103700
*                                                                    * 23103800
*--------------------------------------------------------------------* 23103900
WTOMODEL WTO   TEXT=((,DE)),                                         +23104000
               MF=L                                                   23104100
         SPACE 2                                                      23104200
*--------------------------------------------------------------------* 23104300
*                                                                    * 23104400
*        SINGLE LINE WTO MODEL LINE                                  * 23104500
*                                                                    * 23104600
*--------------------------------------------------------------------* 23104700
STATMODL DC    0F'0'               STATUS LINE MODEL FOR WORKING     23104800
*                                  STORAGE                           23104900
         DC    AL2(STATMLEN-2)     LENGTH OF MESSAGE FOR WTO MACRO   23105000
         DC    C'CBRHADUX USER EXIT: SG ('  START OF WTO MSG        23105100
         DC    C'        '         PLACE HOLDER FOR STORAGE GROUP    23105200
         DC    C') completed RC='  LABEL FOR RETURN CODE            23105300
         DC    C'        '         PLACE HOLDER FOR RETURN CODE      23105400
         DC    C' RSN='            LABEL FOR REASON CODE             23105500
         DC    C'        '         PLACE HOLDER FOR REASON CODE      23105600
STATMLEN EQU   *-STATMODL          LENGTH OF STATUS LINE             23105700
*                                                                     23105800
*--------------------------------------------------------------------* 23105900
*                                                                    * 23106000
*        MANAGEMENT CLASS VERIFICATION                               * 23106100
*                                                                    * 23106200
*--------------------------------------------------------------------* 23106300
MCTAB    DC    C'MCNODEL1'         MANAGEMENT CLASS TABLE            23112500
         DC    C'MCNODEL2'                                           23125000
MCCNT    EQU   (*-MCTAB)/8                                           23137500
MAXNAMES DC    F'44'               MAXIMUM NUMBER OF NAMES IN        23168700
*                                  A 4K BLOCK OF VERIFY TABLE        23200000
*--------------------------------------------------------------------* 23202000
*                                                                    * 23204000
*        DD STATEMENT DECLARATIONS                                   * 23206000
*                                                                    * 23208000
*--------------------------------------------------------------------* 23210000
RETSTAT  DC    AL2(85)                                               23212500
         DC    AL2(1)                                                23225000
         DC    AL2(8)                                                23237500
DDNMTXT  DS    0F                                                    23250000
         DC    AL2(1)                                                23300000
         DC    AL2(1)                                                23350000
         DC    AL2(ENDDDN-STRTDDN)                                   23400000
STRTDDN  DC    C'CBRADUXI'                                           23450000
ENDDDN   EQU *                                                      23500000
MODTXT   DC    AL2(4)              STATUS=MOD                        23850000
         DC    AL2(1)                                                23900000
         DC    AL2(1)                                                23950000
         DC    AL1(2)                                                24000000
SHRTXT   DC    AL2(4)              STATUS=SHR                        24050000
         DC    AL2(1)                                                24100000
```

```
         DC    AL2(1)                                                  24150000
         DC    AL1(8)                                                  24200000
TRKTXT   DC    AL2(7)              TRACK ALLOCATION IF NOT OLD         24250000
         DC    AL2(0)                                                  24300000
PRIMTXT  DC    AL2(10)                                                 24350000
         DC    AL2(1)                                                  24400000
         DC    AL2(3)                                                  24450000
         DC    AL3(10)             10 TRACKS PRIMARY ALLOCATION        24500000
SECTXT   DC    AL2(11)                                                 24550000
         DC    AL2(1)                                                  24600000
         DC    AL2(3)                                                  24650000
         DC    AL3(10)             10 TRACKS SECONDARY ALLOCATION      24700000
CTLGTXT  DC    AL2(5)              DISP=CATLG                          24750000
         DC    AL2(1)                                                  24800000
         DC    AL2(1)                                                  24850000
         DC    AL1(2)                                                  24900000
KEEPTXT  DC    AL2(5)              DISP=KEEP                           24950000
         DC    AL2(1)                                                  25000000
         DC    AL2(1)                                                  25050000
         DC    AL1(8)                                                  25100000
UNITTXT  DC    AL2(21)                                                 25150000
         DC    AL2(1)                                                  25200000
         DC    AL2(5)                                                  25250000
         DC    C'SYSDA'                                                25300000
CLOSTXT  DC    AL2(28)             FREE DATA SET WHEN CLOSED           25350000
         DC    AL2(0)                                                  25400000
DSNTXT   DC    AL2(2)                                                  25450000
         DC    AL2(1)                                                  25500000
         DC    AL2(ENDDSN-VRFYDSN)                                     25550000
VRFYDSN  DC    C'HLQ.OBJECT.DELETE.VERIFY'                             25600000
ENDDSN   EQU   *                                                       25650000
STATAREA DS    0F                  STATIC CBS TO BE MOVED              25700000
         OPEN   (,),MF=L                                               25750000
DCBSTAT  DS    0F                                                      25800000
*   NOTE:  IF THIS DATASET IS PREALLOCATED PRIOR TO THE INVOCATION   *  25807100
*          OF THIS ROUTINE, THE DCB DECLARES OF THE DATASET SHOULD   *  25814200
*          BE REMOVED FROM THIS ROUTINE.  OTHERWISE, THE ALLOCATION  *  25821300
*          HERE WILL OVERRIDE THE PREALLOCATION OF THE DATASET,      *  25828400
*          CAUSING UNEXPECTED OUTPUT.                                *  25835500
*                                                                    *  25842600
         DCB  DDNAME=CBRADUXO,MACRF=(PM),OPTCD=W,LRECL=89,BLKSIZE=8900,+25871300
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN                         25900000
         DCB  DDNAME=CBRADUXI,MACRF=(GM),LRECL=89,BLKSIZE=8900,       +25950000
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN,EODAD=REODAD            26000000
DYNRBC   DS    0F                                                      26050000
         DC    AL1(20)             RB LENGTH                           26100000
         DC    AL1(01)             DSNAME ALLOCATION                   26150000
         DC    X'C0'               FLAGS1 - NO EXIST ALLOC             26200000
         DC    X'0' FLAGS1                                             26250000
         DC    F'0'                ERROR CODES                         26300000
         DC    A(0)                TEXT UNIT POINTERS                  26350000
         DC    F'0'                RESERVED                            26400000
         DC    X'00'               WAIT FOR VOLS,UNITS,DSNS AND MOUNTS 26450000
*        DC    X'D1'               WAIT FOR VOLS,UNITS,DSNS AND MOUNTS 26500000
         DC    AL3(0)                                                  26550000
TXTPTRC  DS    0F                                                      26600000
         DC    A(DSNTXT)                                               26650000
         DC    A(DDNMTXT)                                              26700000
         DC    A(SHRTXT)                                               26750000
         DC    A(KEEPTXT)                                              26800000
         DC    A(0)                TRACK TEXT UNIT FOR OUTPUT DS       26850000
         DC    A(0)                PRIMARY TEXT UNIT FOR OUTPUT DS     26900000
         DC    A(0)                SECONDARY TEXT UNIT FOR OUTPUT DS   26950000
```

```
          DC    A(0)                RETURN DDNAME FOR OUTPUT DS          26983300
          DC    A(UNITTXT)          UNIT TEXT UNIT FOR OUTOUT DS         27016600
          DC    A(CLOSTXT)          FREE (UN-ALLOCATE) AT CLOSE          27050000
          DC    X'80'                                                    27100000
          DC    AL3(0)                                                   27150000
          DC    AL2(2)                                                   27200000
          DC    AL2(1)                                                   27250000
          DC    AL2(ENDD1-D1)                                            27300000
D1        DC    C'HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY'                MMM 27350000
* COMMENT OUT THE ABOVE LINE WHEN UNCOMMENTING THE LINE BELOW           27350001
* D1       DC    C'HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY'           CCC 27350002
ENDD1     EQU   *                                                      27400000
MOVELN    EQU   *-STATAREA                                             27450000
          EJECT                                                        27500000
          DCBD    DSORG=QS                                           30033300
          EJECT                                                      30049900
          IEFZB4D2                                                   30066600
          END CBRHADUX                                               30116600
```

The z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support (V2R4 version publication numbers: SC23-6866-40) to be updated as follows…

Note:

1. **Red text with strikethrough should be removed.**
2. **Blue text is new text that should be added.**
3. **Black text is existing text for reference.**
4. **'…' means the content here doesn't contain any new changes and will not be shown.**

Appendix F. "Auto-delete installation exit" -> "Writing the CBRHADUX exit" -> Page 562 Paragraph before subtitle "Input":

…
If the object name is not found in the verification table, the exit approves deletion. When the object is deleted, the name is written to a sequential data set called the notify data set, HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY ("XXXXXXXX" qualifier will be replaced by actually storage group name and padded with "X". Optional qualifier "YYYY" will be replaced by actually DB2SSID and padded with "Y" if the user chooses to do so, otherwise the data set name would be HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY). Fully qualified data set names contain both the object name and the collection name, allowing uniqueness across OSMC and concurrent I/O from the different tasks. A concatenation of these data sets (one for each group) provides input to other applications that need to synchronize their directories with the OAM object directories.
…

Page 563 Paragraph after subtitle "Output":

The output data set HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY ("XXXXXXXX" qualifier will be replaced by actually storage group name and padded with "X". Optional qualifier "YYYY" will be replaced by actually DB2SSID and padded with "Y" if the user chooses to do so, otherwise the data set name would be HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY), known as the NOTIFY data set, is defined in the DCBO in the dynamic working storage and also defined in local storage in the DCB with the DDNAME of CBRADUXO.

…

Page 565 "Sample auto-delete installation exit" replace the sample under this session with the new sample below:

```
CBRHADUX TITLE 'OSMC SAMPLE AUTO-DELETE INSTALLATION EXIT'              00050000
*********************************************************************** 00100000
*                                                                     * 00116600
* $MOD(CBRHADUX),COMP(OSMC),PROD(OAM) :                               * 00133300
*                                                                     * 00150000
* MODULE NAME:  CBRHADUX                                              * 00200000
*                                                                     * 00250000
* DESCRIPTIVE NAME:  OSMC Sample AUTO-DELETE Installation Exit        * 00262500
*                                                                     * 00275000
*********************************************************************** 00317700
*PROPRIETARY V3 STATEMENT                                             * 00360500
*Licensed Material - Property of IBM                                  * 00367700
*"Restricted Materials of IBM"                                          00367800
*5650-ZOS                                                             * 00373100
*COPYRIGHT IBM CORP. 1989, 2020                                       * 00378500
*END PROPRIETARY V3 STATEMENT                                         * 00383900
*********************************************************************** 00384100
*                                                                     * 00384300
*  NOTE: Read before implementing, this exit is used during OSMC      * 00384500
*        and does not perform an OSREQ functions during its           * 00384700
*        processing.                                                  * 00384900
*                                                                     * 00385100
*  This sample program, if installed as is, will prevent objects      * 00385300
*  from being deleted/                                                * 00385500
*                                                                     * 00385700
*  RETURNS:       12 - Do not expire any objects within this storage * 00385900
*                      group and do not call this exit again for this* 00386100
*                      instance of OSMC for this storage group.       * 00386300
*                                                                     * 00386500
*  This sample exit is provided as an example of different coding     * 00386700
*  techniques that may be used.  It should be modified to suit the    * 00386900
*  specific needs of each user.  The minimum modification that will   * 00387100
*  enable any of the function provided is to delete or comment out    * 00387300
*  the three lines of "bypass" code that return the "12" described    * 00387500
*  above.  (right after "-- THREE LINES OF BYPASS CODE --"  )         * 00387700
*                                                                     * 00387900
*  SUBROUTINE MCVERIFY is provided as an example of a technique that  * 00388100
*  could be used to check management class, but is not currently      * 00388300
*  invoked.  This technique involves coding the management class      * 00388500
*  names in the data storage area for the CBRHADUX.                   * 00388700
*                                                                     * 00388900
*  Search for keyword MMM to find the Minimum Mandatory               * 00389100
*  Modification/Review Points in the code.                            * 00389300
*                                                                     * 00389600
*  When running in a Multi-OAM configuration, a unique notify data    * 00389601
*  set name should be used for each OAM instance or different OAM     * 00389602
*  instances may try to write to the same data set concurrently and   * 00389603
```

```
*  error may occur. Search for keyword CCC to find the code that was *  00389604
*  added to include the DB2SSID as part of the notify data set name. *  00389605
*  The use of the DB2SSID as a qualifier in the data set name can    *  00389606
*  also be used in a classic OAM configuration.                      *  00389607
*                                                                    *  00389608
**********************************************************************  00390100
*  FUNCTION:                                                         *  00390300
*    Module CBRHADUX is used to verify whether or not an object      *  00390500
*    should be deleted.  This module is called during OSMC           *  00390700
*    processing when an object has been selected for deletion.       *  00390900
*                                                                    *  00391100
*  OPERATION:                                                        *  00430300
*  - Perform standard entry linkage.                                 *  00469500
*  - Set CBRHADUX return code to 12 indicating to OSMC that objects  *  00508700
*    are not to be deleted for this storage group and that the       *  00547900
*    CBRHADUX exit is not to be called again for this storage group  *  00587100
*    [default behavior]                                              *  00626300
*    (Read NOTE: READ BEFORE IMPLEMENTING EXIT for explanation of    *  00665500
*     default behavior)                                              *  00704700
*  - [NOTE] Write message to the operator stating that objects       *  00743900
*     will NOT be expired for this storage group [default behavior]  *  00783100
*  - If return code not set to 12 then                               *  00822300
*     | If this is the first call then                               *  00861500
*     | | - Initialize storage                                       *  00900700
*     | | - Set up Data Area to allow for reentrancy.               *  00939900
*     | | - Prepare Data Area for next call for same Storage Group   *  00979100
*     | | - Establish register save area in Data Area.              *  01018300
*     | Endif                                                        *  01057500
*     | If current Storage Group is GROUPXX [see note in code]       *  01096700
*     | | - Set CBRHADUX return code to 8 indicating to OSMC that    *  01135900
*     | | - objects are to be deleted for this storage group and that*  01175100
*     | |   the CBRHADUX exit is not to be called again for this     *  01214300
*     | |   storage group.                                           *  01253500
*     | |   (as an example of bypassing processing by the CBRHADUX   *  01292700
*     | |    exit for a given storage group)                         *  01331900
*     | Else                                                         *  01371100
*     | | If this invocation is not the end of a storage group       *  01410300
*     | | (indicated by the ADUDONE flag)                            *  01449500
*     | | | If this is the first invocation (i.e. verify table has   *  01488700
*     | | | not yet been built) then                                 *  01527900
*     | | | | - Open input dataset and build a local table in storage*  01567100
*     | | | |  if needed                                             *  01606300
*     | | | | - Open output dataset                                  *  01645500
*     | | | Endif                                                    *  01684700
*     | | | If no errors encountered so far then                     *  01723900
*     | | | | If object name and collection name of current object   *  01763100
*     | | | |  do not exist in the in the Verify Table Then          *  01802300
*     | | | | | Add a record to the notify dataset                   *  01841500
*     | | | | Endif                                                    *  01880700
*     | | | Endif                                                    *  01919900
*     | | Endif                                                      *  01959100
*     | Endif                                                        *  01998300
*     | If not the end of storage group (indicated by ADUDONE flag)  *  02037500
*     | and an error return code of 8 or greater has been set then   *  02076700
*     | | Write message to the operator indicating the storage group *  02115900
*     | | reason code and return code                                *  02155100
*     | Endif                                                        *  02194300
*     | If return code is 8 (indicating that CBRHADUX will not be    *  02233500
*     | called again)                                                *  02272700
*     | | - Free the verify table (if it exists)                     *  02311900
*     | | - Close the output dataset                                 *  02351100
*     | | - Free data area                                           *  02390300
*     | Endif                                                        *  02429500
```

```
*     Endif                                                          *  02468700
*  - Perform standard exit linkage                                   *  02507900
*                                                                    *  02550000
*  NOTES:                                                            *  02600000
*                                                                    *  02650000
*     DEPENDENCIES: This routine is called on a group basis.  That   *  02700000
*                   is, the first and every call of a single         *  02750000
*                   instance of this routine is for the same storage *  02800000
*                   group.  This allows a return code indicating the *  02850000
*                   routine is not interested in a certain group     *  02900000
*                   (i.e. GROUPXX) and that the output data          *  02950000
*                   set can be segregated by group allowing for      *  03000000
*                   processing overlap (i.e. more than one task      *  03050000
*                   can have an instance of this routine because     *  03100000
*                   each task is writing to a separate data set).    *  03150000
*                                                                    *  03200000
*                   The end of cycle call does not include the name  *  03250000
*                   of an object to be deleted.                      *  03300000
*                                                                    *  03350000
*                                                                    *  03400000
*     CHARACTER-CODE-DEPENDENCIES: EBCDIC Character Set              *  03450000
*                                                                    *  03500000
*     RESTRICTIONS: None                                             *  03550000
*                                                                    *  03600000
*     REGISTER CONVENTIONS:                                          *  03650000
*                                                                    *  03700000
*               Standard Entry Linkage                               *  03750000
*                                                                    *  03800000
*               R0 and R1 used for system service invocation         *  03850000
*               R2 and R3 are work registers                         *  03900000
*               R4  contains the ADUP address                        *  03950000
*               R5 through R10 are work registers                    *  04000000
*               R11 contains the address of the dynamic area         *  04050000
*               R12 is the base register                             *  04100000
*                                                                    *  04150000
*  MODULE TYPE: User Exit                                            *  04175000
*                                                                    *  04200000
*    PROCESSOR: Assembler                                            *  04250000
*                                                                    *  04300000
*    ATTRIBUTES:                                                     *  04350000
*     LOCATION: LINKLIB                                              *  04400000
*     STATE:    PROBLEM                                              *  04450000
*     TYPE:     REENTRANT                                            *  04500000
*                                                                    *  04650000
*  INPUT:                                                            *  04700000
*      SYMBOLIC NAME: CBRADUP                                        *  04750000
*      DESCRIPTION:   Control block containing the parameters for    *  04800000
*                     object deletion. (Contained in MACRO CBRADUP)  *  04850000
*                                                                    *  04900000
*      SYMOBLIC NAME: Verify Data Set [OPTIONAL]                     *  04907100
*      DSN NAME:      HLQ.OBJECT.DELETE.VERIFY                       *  04914200
*      DESCRIPTION:   This is the input data set to the CBRHADUX     *  04921300
*                     which is used to determine that an object      *  04928400
*                     should not be expired.  The criteria           *  04935500
*                     is based on Object Name and Collection Name    *  04942600
*                     which is based upon the input to this module.  *  04949700
*                     This Data set also has a table created in      *  04956800
*                     storage for which comparisons shall be made.   *  04963900
*      ATTRIBUTES:                                                   *  04971000
*         ORGANIZATION:   PS                                         *  04978100
*         RECORD FORMAT:  FB                                         *  04985200
*         RECORD LENGTH:  89                                         *  04992300
*         BLOCK SIZE:     8900                                       *  04999400
```

```
*      RECORD FORMAT:                                                 *  05006500
*         0-43  :   Object Name        CHAR(44)                      *  05013600
*         44    :   Blank                                            *  05020700
*         45-89 :   Collection Name    CHAR(44)                      *  05027800
*                                                                     *  05034900
*      NOTES:         Be sure to modify the (HLQ) high level         *  05042000
*                     qualifier and adjust accordingly to the        *  05049100
*                     private catalog that is accessible. An         *  05056200
*                     in-storage table will be built for these names *  05063300
*                     and will be formatted as follows:              *  05070400
*                                                                    *  05077500
*          VERIFY TABLE:                                             *  05084600
*               0-3    :  Number of entries in table                 *  05091700
*               4-7    :  Pointer to next table                      *  05098800
*               9-4095 :  Records from Input data set                *  05105900
*                                                                    *  05113000
*          NOTE: For last Table in Chain, the pointer to the next    *  05120100
*                table is 00000000 (signifies end of chain)          *  05127200
*                                                                    *  05134300
*  OUTPUT:                                                           *  05141400
*      SYMBOLIC NAME: ADUUFLD                                        *  05148500
*      DESCRIPTION:   CBRADUP parameter list is updated to hold a    *  05155600
*                     pointer to the dynamic data area for reentrant *  05162700
*                     calls for the same storage group.              *  05169800
*                                                                    *  05176900
*      SYMBOLIC NAME: Notify Data Set                                *  05184000
*      DSN NAME:      HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY              *  05191100
*      DESCRIPTION:   This is the output data set that contains the  *  05198200
*                     Object Name along with the Collection Name of  *  05205300
*                     each object deleted for the Storage group,     *  05212400
*                     which replaces the XXXXXXXX in the data set    *  05219500
*                     name, padded with the character X. The few     *  05226600
*                     sections marked with MMM correlate to the      *  05233700
*                     output data set name.                          *  05240800
*      NOTE:          To include the DB2SSID in the data set name as *  05240801
*                     a qualifier, uncomment all the lines marked    *  05240802
*                     with CCC. The DSN NAME will become:            *  05240803
*                     HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY         *  05240804
*                     The DB2SSID will replace the YYYY in the data  *  05240805
*                     set name, padded with the character Y.         *  05240806
*                                                                    *  05240807
*      ATTRIBUTES:                                                   *  05247900
*         ORGANIZATION:   PS                                         *  05255000
*         RECORD FORMAT:  FB                                         *  05262100
*         RECORD LENGTH:  89                                         *  05269200
*         BLOCK SIZE:     8900                                       *  05276300
*      RECORD FORMAT:                                                *  05283400
*         0-43  :   Object Name        CHAR(44)                      *  05290500
*         44    :   Blank                                            *  05297600
*         45-89 :   Collection Name    CHAR(44)                      *  05304700
*                                                                    *  05311800
*   NOTE:  If this dataset is preallocated prior to the invocation   *  05318900
*          of this routine, the DCB declares of the dataset should   *  05326000
*          not be altered or removed.  Otherwise, modification to    *  05333100
*          the DCB statements can cause unexpected results.          *  05340200
*                                                                    *  05347300
*   NOTE:  Be sure to modify the VERIFY and NOTIFY data set names    *  05354400
*          prior to assembling and linking this module.  The         *  05361500
*          following labels need to be modified to reflect the High  *  05368600
*          Level Qualifier (HLQ) you would like to be used.          *  05375700
*                 1)  D1                                             *  05389500
*                 2)  NTFYDSN                                        *  05390300
*                 3)  VRFYDSN                                        *  05391400
```

```
*                                                                     *  05392600
*    RETURN CODES = 0  Allows for expiration of the object to         *  05393200
*                      continue and to continue calling the user      *  05393800
*                      exit for OSMC processing.                      *  05394400
*                   4  Do not allow expiration of this object for     *  05395000
*                      this instance of OSMC for this storage group.  *  05395600
*                   8  Expire all objects in this storage group for   *  05396200
*                      this instance of OSMC but do not call the      *  05396800
*                      user exit again.                               *  05397400
*                   12 Do not expire any objects within this storage  *  05398000
*                      group and do not call this exit again for this*  05398600
*                      instance of OSMC for this storage group.       *  05399200
*                                                                     *  05400000
* CHANGE ACTIVITY:                                                    *  05450000
*                                                                     *  05750000
*      $L0=OAM,110,082687,TUCWV:  INITIAL RELEASE                     *  05758300
*      $L1=JDP3227,320,890523,TUCHTT:  RELEASE 1                      *  05766600
*      $D1=JDP3227,320,890523,TUCLJS:  COLLECTION NAMES               *  05774900
*      $L2=PRESCOTT,331,901112,TUCLJS:  PRESCOTT SUPPORT              *  05783200
*      $L3=OAMR1C,R1C,090311,TUCDMS:  RAS Improvements                *  05791500
*      $L4=OAMR21,R21,110215,TUCBJF:  RAS Improvements                *  05795700
*      $P1=K210268,R21,110714,TUCBJF: RAS Improvements                *  05797800
*      $P2=130050,R23,160817,TUCAED: Typo in prolog                   *  05797801
*      $P3=244338,R24,180330,TUCAED: Encoding compatibility           *  05797802
*      $P4=298255,R24,190204,TUCAED: Correct CBRADUP and CBRHADUX     *  05797803
*                                    comments for return codes        *  05797804
*      $00=OA60203,R23,200821,TUCJFX: ADD DB2SSID (OPTIONAL) IN       *  05797805
*                                     NOTIFY DS NAME                  *  05797806
***********************************************************************  05797807
         TITLE 'CBRHADUX WORKING STORAGE'                               05800500
*---------------------------------------------------------------------*  05801000
*                                                                     *  05801500
*        CBRHADUX WORKING STORAGE                                     *  05802000
*                                                                     *  05802500
*---------------------------------------------------------------------*  05803000
DATAREA  DSECT ,                                                        05803500
SAVE1    DS    18F                 SAVE AREA                          05804000
SAVE14   DS    1F                  R14 SAVE AREA FOR SUBROUTINES      05804500
FLAGS    DS    X                   FLAG AREA                          05805000
TABUILT  EQU   X'80'               VERIFY TABLE HAS BEEN BUILT        05805500
ODCBO    EQU   X'40'               NOTIFY DATA SET IS OPEN            05806000
VRECSPTR DS    F                   ADDRESS OF THE IST 4K BLOCK OF     05806500
RETTXT   DS    AL2                                                    05807000
         DS    AL2                                                    05807500
         DS    AL2                                                    05808000
         DS    CL8                                                    05808500
*                                  OBJECT NAMES READ FROM THE VERIFY  05809000
*                                  DATA SET                           05809500
RETCODE  DS    F                   INTERNAL RETURN CODE               05810000
REASCODE DS    F                                                      05810500
S99RBPTR DS    F                   ADDRESS OF SVC99 RB                05811000
RENTAREA DS    0F                  REENTRANT COPY OF STATIC DEFINED   05811500
*                                  CONTROL BLOCKS                     05812000
OPENL    OPEN   (,),MF=L                                               05812500
DCBS     DS    0F                                                     05813000
DCBO     DCB  DDNAME=CBRADUXO,MACRF=(PM),OPTCD=W,LRECL=89,BLKSIZE=8900,+05813500
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN                        05814000
DCBI     DCB  DDNAME=CBRADUXI,MACRF=(GM),LRECL=89,BLKSIZE=8900,       +05814500
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN,EODAD=REODAD           05815000
DYNRB    DS    0F                                                     05815500
         DC    AL1(20)             RB LENGTH                          05816000
         DC    AL1(01)             DSNAME ALLOCATION                  05816500
         DC    X'C0'               FLAGS1 - NO EXIST ALLOC            05817000
```

```
         DC    X'0' FLAGS1                                              05817500
         DC    F'0'                ERROR CODES                          05818000
TXTPTR   DC    A(TXTPTRV)          TEXT UNIT POINTERS                   05818500
         DC    F'0'                RESERVED                             05819000
*        DC    X'D1'               WAIT FOR VOLS,UNITS,DSNS AND MOUNTS  05819500
         DC    X'00'                                                    05820000
         DC    AL3(0)                                                   05820500
TXTPTRV  DS    0F                                                       05821000
TXTDSN   DC    A(DSNTXT)                                                05821500
TXTDDN   DC    A(DDNMTXT)                                               05822000
TXTSTAT  DC    A(SHRTXT)                                                05822500
TXTDISP  DC    A(KEEPTXT)                                               05823000
TXTTRK   DC    A(0)                                                     05823500
TXTPRIM  DC    A(0)                                                     05824000
TXTSEC   DC    A(0)                                                     05824500
TXTRET   DC    A(0)                                                     05825000
TXTUNIT  DC    A(UNITTXT)                                               05825500
         DC    A(CLOSTXT)                                               05826000
         DC    X'80'                                                    05826500
         DC    AL3(0)                                                   05827000
DSNTXTN  DC    AL2(2)                                                   05827500
         DC    AL2(1)                                                   05828000
         DC    AL2(ENDDSNN-NTFYDSN)                                     05828500
*                                                                     * 05829000
*  Please note that modifying NTFYDS2 can cause unexpected results,   * 05829500
*  the XXXXXXXX is overlaid by the current storage group name.        * 05830000
*  the YYYY is overlaid by the current DB2SSID if the CCC line is     * 05830250
*  uncommented.                                                       * 05830251
*                                                                     * 05830500
NTFYDSN  DC    C'HLQ.'                                               MMM 05831000
NTFYDB2  DC    C'YYYY.'                                              CCC 05831250
NTFYDS2  DC    C'XXXXXXXX.OBJECT.DELETE.NOTIFY'                         05831500
ENDDSNN  EQU   *                                                       05832000
*                                                                      05832500
CONVERT  DS    D                   CONVERSION AREA FOR WTD              05833000
         DS    D                   EXTENSION OF WORK AREA               05833500
HEXWORK  DS    D                   CHARACTER CONVERSION FOR WTD         05834000
*--------------------------------------------------------------------* 05834500
*                                                                    * 05835000
*        SINGLE LINE WTO PARAMETER LIST                              * 05835500
*                                                                    * 05836000
*--------------------------------------------------------------------* 05836500
WTOLIST  WTO   TEXT=((,DE)),                                          +05837000
               MF=L                                                    05837500
WTOLISTL EQU   *-WTOLIST           SIZE OF WTO MACRO EXPANDED          05838000
         SPACE 2                                                       05838500
*--------------------------------------------------------------------* 05839000
*                                                                    * 05839500
*        SINGLE LINE WTO TEXT LINE                                   * 05840000
*                                                                    * 05840500
*--------------------------------------------------------------------* 05841000
STATDISP EQU   *-DATAREA           DISPLACEMENT IN DATA AREA           05841500
STATLINE DS    0F                  STATUS CONTROL LINE MODEL           05842000
         DS    AL2                 LENGTH OF MESSAGE                   05842500
         DS    C'CBRHADUX USER EXIT: SG ('  START OF WTO MSG           05843000
STATSG   DS    CL8                 STORAGE GROUP NAME                  05843500
         DS    C') completed RC='  LABEL FOR RETURN CODE               05844000
STATRET  DS    CL8                 RETURN CODE                         05844500
         DS    C' RSN='            LABEL FOR REASON CODE               05845000
STATREAS DS    CL8                 REASON CODE                         05845500
STATLEN  EQU   *-STATLINE          LENGTH OF STATUS LINE               05846000
*                                                                      05846500
DATALEN  EQU   *-DATAREA                                               05847000
```

```
         EJECT                                                          05850000
         TITLE 'CBRHADUX DSECT DEFINITIONS'                             05857100
*------------------------------------------------------------------*    05864200
*                                                                  *    05871300
*        CBRADUP - PARAMETER LIST DSECT DEFINITION                 *    05878400
*                                                                  *    05885500
*------------------------------------------------------------------*    05892600
         CBRADUP                                                        05899700
         SPACE 2                                                        05906800
*------------------------------------------------------------------*    05913900
*                                                                  *    05921000
*        MANAGEMENT CLASS DSECT TO MAP TO MCTAB                    *    05928100
*                                                                  *    05935200
*------------------------------------------------------------------*    05942300
TAB      DSECT ,                                                        05949400
MCLEN    DS    H                                                        05956500
MCNAME   DS    CL30                                                     05963600
TABLEN   EQU   *-TAB                                                   05970700
         EJECT                                                          05977800
         TITLE 'CBRHADUX - VERIFY FOR OBJECT DELETION EXIT'             05984900
******************************************************************** 05992000
*                                                                  *   05999100
*        CBRHADUX ENTRY POINT - MAINLINE PROCESSING                *   06006200
*                                                                  *   06013300
******************************************************************** 06020400
CBRHADUX CSECT                     BEGINNING OF CBRHADUX               06027500
* CBRHADUX AMODE 31                   FOR 31 BIT ADDRESSING            06034600
* CBRHADUX RMODE ANY                                                   06041700
ADUX     DS    0H                  ENTRY POINT                         06050000
         USING *,R15               ALLOW FOR IMPLICIT ADDRESSING       06100000
         B     PASTID              BRANCH AROUND ID                    06150000
         DC    CL8'CBRHADUX'       EYECATCHER FOR EXIT                 06187500
         DC    CL8'&SYSDATE'       DATE IN THE EYECATCHER              06225000
PASTID   DS    0H                  BRANCH AROUND LABEL                 06262500
         STM   R14,R12,12(R13)     SAVE CALLERS REGS                   06300000
         LR    R12,R15             SET BASE REG                        06350000
         DROP  R15                                                     06400000
         USING ADUX,R12            IMPLICIT ADDRESSABILITY             06450000
         LR    R4,R1               GET INPUT PARAMETER LIST            06500000
         USING CBRADUP,R4          ADDRESSABILITY ON PARM LIST         06526900
         SYSSTATE AMODE64=NO ACSENV=P                                  06540300
******************************************************************** 06553800
*                                                                  *   06557600
*    Comment the following lines to allow for object deletion, by  *   06561400
*      default objects will not be deleted and a WTO is made.      *   06565200
*                  -- THREE LINES OF BYPASS CODE --                *   06569000
*                                                                  *   06572800
******************************************************************** 06576600
*                                                                  *   06580400
         LA    R2,ADUFAILN         LOAD RETURN CODE 12 INTO R2         06584200
       WTO 'CBRHADUX USER EXIT: RC=12 OSMC bypassed object expiration' 06586100
         B     NOR13               BYPASS ALL CODE EXCEPT EXIT CODE    06588000
*                                                                  *   06591800
******************************************************************** 06595600
         L     R11,ADUUFLD         GET ADDRESS OF DYNAMIC AREA         06600000
         LTR   R11,R11             IS THIS THE FIRST CALL              06650000
         BNZ   NOGMAIN             NO, DO NOT GETMAIN                  06700000
         GETMAIN RU,LV=DATALEN,SP=0  GET DYNAMIC AREA                  06750000
         LR    R11,R1              GET STORAGE ADDRESS FOR DYNAMIC AREA 06800000
         USING DATAREA,R11         SET DYNAMIC AREA ADDRESSABILITY FOR  06850000
*                                   REENTRANCY                         06900000
         SLR   R1,R1               GET A ZERO                          06950000
         ST    R1,RETCODE          ZERO THE RETURN CODE                07000000
```

```
         ST    R1,REASCODE         ZERO THE REASON CODE                  07025000
         ST    R1,VRECSPTR         ZERO THE IN-STORAGE NAMES FIELD       07050000
         STC   R1,FLAGS            ZERO THE FLAGS BYTE                   07100000
         ST    R11,ADUUFLD         SAVE DYNAMIC AREA FOR NEXT CALL       07150000
NOGMAIN  DS    0H                  BRANCH AROUND FOR NO GETMAIN          07200000
         ST    R13,SAVE1+4         BACKCHAIN SAVE AREAS                  07500000
         LA    R2,SAVE1            GET SAVE AREA ADDRESS                 07550000
         ST    R2,8(,R13)          FORWARDCHAIN SAVE AREAS               07600000
         LR    R13,R2              ESTABLISH SAVE AREA ADDRESS           07650000
         SR    R2,R2               CLEAR FOR ZERO                        07700000
         ST    R2,RETCODE          ZERO THE INTERNAL RETURN CODE         07750000
         ST    R2,REASCODE         ZERO THE INTERNAL REASON CODE         07752300
*********************************************************************** 07754600
*  The following comparison is made to provide an example, of how to *  07756900
*  allow deletions for a particular storage group.  The following    *  07759200
*  four lines can be removed if this does not pertain to your needs. *  07761500
*********************************************************************** 07763800
         CLC   ADUSGNAM,=CL8'GROUPXX' DO NOT PROCESS FURTHER IF    MMM  07766100
         BNE   NOTGRPXX            GROUPXX FOUND DO NOT PROCESS          07768400
         LA    R2,ADU8RS01         LOAD REASON CODE FOR GROUPXX          07770700
         ST    R2,REASCODE         STORE REASON CODE                     07773000
         LA    R2,ADUFAILC         INDICATE DELETE IS OKAY BUT           07775300
         ST    R2,RETCODE          DO NOT CALL ADUX AGAIN                07778400
         B     ENDADUX             FOR THIS CYCLE                        07785500
NOTGRPXX DS    0H                                                        07792700
         CLI   ADUFUNC,ADUDONE     IS THIS AN END OF CYCLE CALL          07800000
         BE    DOFREE              YES, GO CLEANUP                       07850000
         TM    FLAGS,TABUILT       HAS VERIFY TABLE BEEN BUILT           07900000
         BO    DOVERIFY            YES, DO NOT RE-OPEN DATA SET          07950000
         BAL   R14,OPENDS          FIRST CALL OPEN INPUT AND OUTPUT      08000000
         OI    FLAGS,TABUILT       INDICATE TABLE BUILT FOR NEXT CALL    08050000
*                                  VRECSPTR WILL BE ZERO IF NO           08100000
*                                  ENTRIES IN TABLE                      08150000
DOVERIFY DS    0H                                                        08200000
         L     R2,RETCODE          GET RETURN CODE                       08250000
         LTR   R2,R2               DO NOT CONTINUE IF NON-ZERO           08300000
         BNZ   ENDADUX             END ADUX IF NON-ZERO CODE             08350000
         L     R10,VRECSPTR        GET ADDRESS OF IN-CORE VERIFY RECS    08400000
         LTR   R10,R10             IS THERE ANY VERIFY TABLE             08450000
         BZ    NOVERIFY            IF ZERO NO VERIFY NEEDED              08500000
         BAL   R14,VERIFY          CALL VERIFY ROUTINE                   08550000
NOVERIFY DS    0H                  FREEMAIN TABLE                        08600000
         L     R2,RETCODE          GET THE RETURN CODE                   08650000
         LTR   R2,R2               IF NOT ZERO VERIFY FAILED             08700000
         BNZ   NONOTIFY            NO NOTIFY IF VERIFY FAILED            08750000
         BAL   R14,NOTIFY          ADD OBJECT TO NOTIFY DATA SET         08800000
NONOTIFY DS    0H                  DO NOT NOTIFY IF NON-ZERO CODE        08850000
ENDADUX  DS    0H                  END OF CBRADUX                        08900000
*********************************************************************** 08908300
*  If you would not like the CBRHADUX module to display return code  *  08916600
*  messages, the please comment out the following statement.         *  08924900
*********************************************************************** 08933200
         BAL   R14,ADUXOUT         CALL WTO FOR ADUX STATUS              08941500
         LA    R3,ADUFAILC         RETURN CODE INDICATING NO             08950000
*                                  RECALL OF ADUX FOR                    09000000
*                                  THIS CYCLE                            09050000
         C     R3,RETCODE          IF ADUX IS TO BE CALLED AGAIN         09100000
         BH    NOFREE              DO NOT FREE AND CLOSE                 09150000
DOFREE   DS    0H                  FREE THE VERIFY TABLE AND CLOSE       09200000
*                                  THE OUTPUT DATA SET                   09250000
         BAL   R14,FREETAB         FREEMAIN THE VERIFY TABLE             09300000
         BAL   R14,CLOSEDS         CLOSE AND FREE OUTPUT DATA SET        09350000
         L     R13,SAVE1+4         GET CALLERS SAVE AREA ADDRESS         09400000
```

```
         L     R2,RETCODE          GET RETURN CODE                       09450000
         SLR   R1,R1               GET A ZERO                            09500000
         ST    R1,ADUUFLD          CLEAR THE USER FIELD TO AVOID         09550000
*                                  INADVERTENT USE OF FREEMAINED         09600000
*                                  STORAGE                               09650000
         FREEMAIN RU,LV=DATALEN,SP=0,A=(R11)  FREE DYNAMIC AREA         09700000
         B     NOR13               BYPASS R13 RESTORE                    09750000
NOFREE   DS    0H                  BYPASS FREEMAIN                       09800000
         L     R13,4(R13)          RESTORE CALLERS SAVE AREA             09850000
NOR13    DS    0H                  BYPASS R13 RESTORE IF FREEMAIN PATH   09900000
         L     R14,12(R13)         GET RETURN ADDRESS                    09950000
         LR    R15,R2              GET RETURN CODE                       10000000
         LM    R0,R12,20(R13)      RESTORE CALLERS REGS                  10050000
         BSM   0,R14               RETURN TO CALLER                      10100000
         EJECT                                                           10150000
         TITLE 'MCVERIFY - TEST MANAGEMENT CLASS FOR DELETION'          10151000
MCVERIFY DS    0H                  MC VERIFY SUBROUTINE                  10152000
*********************************************************************** 10154000
*                                                                     * 10154400
* SUBROUTINE:  MCVERIFY                                               * 10154800
*                                                                     * 10155200
* FUNCTION:    Search the in storage management class table to        * 10155600
*              determine if this object can be deleted. if the input  * 10156000
*              management class is listed in the table then no        * 10156400
*              objects with this management class are to be deleted.  * 10156800
*                                                                     * 10157200
* OPERATION:                                                          * 10157600
*     1.  Load total number of Management Class Name Entries.         * 10158000
*     2.  Load address of the MC table.                               * 10158400
*     3.  If ADUMCNAM matches current MC table entry then set a       * 10158800
*         return code of 4 and a reason code of 1, do not delete      * 10159200
*         object but continue to process storage group.               * 10159600
*     4.  Return to calling module.                                   * 10160000
*                                                                     * 10160400
* CALLED BY:   Add a call from main line if desired                   * 10160800
*                                                                     * 10161200
* CALLS:       None                                                   * 10161600
*                                                                     * 10162000
* NOTE: This subroutine is provided as an example only and is not     * 10162400
*       called from the main line code.  Modify the main line code    * 10162800
*       as needed to invoke this routine if desired.                  * 10163200
*                                                                     * 10163600
*********************************************************************** 10166000
         SPACE 1                                                        10168000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                   10170000
         LA    R11,MCCNT           LOAD NUMBER OF TABLE ENTRIES          10172000
         LA    R10,MCTAB           LOAD ADDRESS OF TABLE IN R10          10174000
         USING TAB,R10             USE DSECT                             10176000
COMPMC   CLC   MCNAME,ADUMCNAM     COMPARE MC NAMES                      10178000
         BE    VMCMATCH                                                  10180000
         LA    R10,TABLEN(,R10)                                          10182000
         BCT   R11,COMPMC                                                10184000
VMCMATCH DS    0H                  MC NAME IS IN VERIFY TABLE            10186000
         LA    R1,ADU4RS01         LOAD REASON CODE OF 1                 10186600
         ST    R1,REASCODE         STORE REASON CODE                     10187200
         LA    R1,ADUNODEL         GOOD COMPARE, DO NOT DELETE           10188000
         ST    R1,RETCODE          SET DO NOT DELETE RETURN CODE         10190000
ENDMC    DS    0H                  END MC VERIFY SUBROUTINE              10192000
         L     R14,SAVE14          GET SAVED R14                         10194000
         BR    R14                 RETURN TO MAINLINE                    10196000
         EJECT                                                           10198000
         TITLE 'OPENDS - OPEN INPUT/OUTPUT DATASETS'                    10199000
OPENDS   DS    0H                  OPENDS SUBROUTINE                     10200000
```

```
*********************************************************************** 10250000
*                                                                     * 10300000
* SUBROUTINE:  OPENDS                                                 * 10350000
*                                                                     * 10400000
* FUNCTION:    Open the input data set and create the verify table.   * 10437500
*              Then open the output data set for object deletion      * 10475000
*              notification.                                          * 10512500
*                                                                     * 10550000
* OPERATION:                                                          * 10560700
*     1.  Prepare Data Area for Input and Output control blocks,      * 10571400
*         along with the necessary SVC 99 RB.                         * 10582100
*.    2.  Begin input (Verify) data set processing.                   * 10592800
*     3.  Verify that the input data set was allocated successfully,  * 10603500
*         on failure set RC=12 and REAS=1. (ADUX not called again)    * 10614200
*     4.  Verify that the input data set is catalogued, on failure    * 10624900
*         set RC=12 and REAS=2. (ADUX not called again)               * 10635600
*     5.  Perform an OPEN on the Verify Data Set using OPEN MACRO.    * 10646300
*     6.  Verify input data set was opened successfully, on failure   * 10657000
*         set RC=12 and REAS=3. (ADUX not called again)               * 10667700
*     7.  Obtain storage to build the Verify table, using GETMAIN     * 10678400
*         MACRO using Subpool=0.                                      * 10689100
*     8.  Verify GETMAIN succeeded, on failure set RC=12, REAS=4      * 10699800
*         (ADUX not called again)                                     * 10710500
*     9.  Initialize the initial verify table and store a pointer to  * 10721200
*         the table in the Data Area.                                 * 10731900
*     10. Read a record from input data set using GET MACRO, on       * 10742600
*         failure call ADUXSYN and end input processing and read      * 10753300
*         until end of file.                                          * 10764000
*     11. Increment and store counter for the number of records.      * 10774700
*     12. If counter equals MAXNAMES then branch to obtain a new      * 10785400
*         block of storage for Verify Table, otherwise go to step 10. * 10796100
*         A. Obtain storage to get a new block for Verify table using * 10806800
*            GETMAIN MACRO with subpool=0.                            * 10817500
*         B. Verify GETMAIN succeeded, on failure RC=12 and REAS=5    * 10828200
*            (ADUX not called again)                                  * 10838900
*         C. Initialize the new block of storage for Verify Table.    * 10849600
*     13. Upon end of file condition close the dataset using CLOSE    * 10860300
*         MACRO.                                                      * 10871000
*     14. Verify no errors have occurred while processing input before* 10881700
*         beginning output data set processing, on failure branch to  * 10892400
*         end of input/output data set processing.                    * 10903100
*     15. Begin output data set (NOTIFY) processing by updating the   * 10913800
*         output data set name with Storage Group name.               * 10924500
*     16. Optionally update the output data set name with DB2SSID     * 10924501
*     17. Update Text Unit pointers for the Notify data set.          * 10935200
*     18. Perform DYNALLOC for output data set and verify no errors   * 10945900
*         occurred, on failure set RC=12 and REAS=6                   * 10956600
*     19. Attempt to open the output data set using the OPEN MACRO,   * 10967300
*         on failure set RC=12 and REAS=7 (ADUX not called again)     * 10978000
*     20. Return to main line code.                                   * 10988700
*                                                                     * 11000000
* REGISTER CONVENTIONS:                                               * 11037500
*     R8  - DCB Pointer                                               * 11075000
*     R11 - Data Area Pointer                                         * 11112500
*                                                                     * 11150000
*********************************************************************** 11200000
         SPACE 2                                                        11250000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                  11300000
         LA    R2,RENTAREA         GET TARGET ADDRESS FOR MOVE          11350000
         LA    R3,MOVELN           GET THE LENGTH OF THE MOVE           11400000
         LA    R6,STATAREA         GET SOURCE ADDRESS OF THE MOVE       11450000
         LR    R7,R3               GET LENGTH OF THE MOVE AND           11500000
*                                  PAD WITH ZEROS                       11550000
```

```
         MVCL  R2,R6               COPY CONTROL BLOCKS TO               11600000
*                                  DYNAMIC AREA FROM STATIC AREA        11650000
         LA    R2,DYNRB            GET DYNAMIC RB ADDRESS               11700000
         ST    R2,S99RBPTR         STORE IN DYNAMIC RB POINTER FIELD    11750000
         OI    S99RBPTR,X'80'      SET END OF LIST FLAG                 11800000
         LA    R2,TXTPTRV          GET DYNAMIC VERSION OF TXTPTR        11850000
         ST    R2,TXTPTR           STORE IN SVC 99 RB DYNAMIC VERSION   11900000
         LA    R2,0                GET ZERO FOR DDNAME TEXT UNIT PTR    11908300
         ST    R2,TXTDDN           PUT IT IN THE TXT UNIT PTR           11916600
         MVC   RETTXT(6),RETSTAT   MOVE FROM STATIC TO DYNAMIC          11924900
         LA    R2,RETTXT           LET SYSTEM DETERMINE DDNAME          11933200
         ST    R2,TXTRET                                                11941500
         LA    R1,S99RBPTR         SETUP REG 1 FOR DYNAMIC ALLOC.       11970700
         DYNALLOC                  ALLOCATE INPUT DATA SET              12000000
         LA    R2,4                GET DYNAMIC ENVIRONMENT ERROR        12050000
         CR    R15,R2              WAS DYNAMIC ALLOC. OKAY              12100000
         BL    DOOPENI             YES, OPEN VERIFY DATA SET            12150000
         BE    CHKNODS             IF ERROR CHECK IF NO DATA SET        12175000
         LA    R15,ADUCRS01        SET REASON CODE OF 1                 12200000
         ST    R15,REASCODE        STORE REASON CODE                    12225000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       12250000
         ST    R15,RETCODE         SET RETURN CODE                      12300000
         B     ENDOPEN             END ADUX                             12350000
CHKNODS  DS    0H                  ENVIRONMENT ERROR                    12400000
         LH    R15,=X'1708'        GET NO CATALOGED DS FAIL CODE        12450000
         CH    R15,DYNRB+4         FAILED BECAUSE DS NOT CATALOGED      12500000
         BE    NOOPENI             SO PROCEED WITHOUT VERIFY            12550000
         LA    R15,ADUCRS02        SET REASON CODE OF 2                 12566600
         ST    R15,REASCODE        STORE REASON CODE                    12583200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       12600000
         ST    R15,RETCODE         SET RETURN CODE                      12650000
         B     ENDOPEN             END ADUX                             12700000
DOOPENI  DS    0H                  ALLOCATED, DO OPEN                   12750000
         LA    R1,OPENL            GET OPEN LIST ADDRESS                12800000
         LA    R8,DCBI             GET INPUT DCB ADDRESS                12850000
         USING IHADCB,R8                                                12950000
         MVC   DCBDDNAM,RETTXT+6   MOVE SYSTEM GENERATED DDN INTO DCB   12966600
         OPEN  ((R8),INPUT),MF=(E,(1)) OPEN INPUT DCB                   12983200
         TM    DCBOFLGS,DCBOFOPN   WAS DS OPENED PROPERLY               13000000
         BO    OPENI               BUILD VERIFY TABLE                   13050000
         LA    R15,ADUCRS03        LOAD REASON CODE OF 3                13066600
         ST    R15,REASCODE        STORE REASON CODE                    13083200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       13100000
         ST    R15,RETCODE         SET RETURN CODE                      13150000
         B     ENDOPEN             END ADUX                             13200000
*                                                                       13212500
*        Build Verify Table                                             13225000
*                                                                       13237500
OPENI    DS    0H                                                       13250000
         GETMAIN RU,LV=4096,SP=0   GET BLOCK FROM SUBPOOL ZERO          13300000
*                                                                       13309000
         LTR   R15,R15             VERIFY GETMAIN WAS SUCCESSFUL        13318000
         BZ    BTABBEG             BRANCH TO BUILD INITIAL TABLE        13327000
         LA    R15,ADUCRS04        LOAD REASON CODE OF 4                13336000
         ST    R15,REASCODE        STORE REASON CODE                    13345000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       13354000
         ST    R15,RETCODE         SET RETURN CODE                      13363000
         B     ENDOPEN             END OPEN DATA SET                    13372000
*                                                                       13381000
BTABBEG  LR    R5,R1               GET THE VERIFY TABLE ADDRESS         13390000
         SLR   R6,R6               GET A ZERO                           13400000
*                                  ALSO, USE AS A RECORD COUNTER        13450000
         ST    R6,0(,R5)           INDICATE IN-CORE TABLE NOT YET SET   13500000
```

```
         ST    R6,4(,R5)           INDICATE IN-CORE TABLE NOT YET SET   13550000
         LA    R9,8(,R5)           GET PAST RECORD COUNTER              13600000
         ST    R5,VRECSPTR         SAVE TABLE ADDRESS                   13650000
GETINPUT DS    0H                                                       13700000
         LA    R1,DCBI             GET INPUT DCB ADDRESS                13750000
         LR    R0,R9               ADDRESS OF RECORD                    13800000
         GET   (1),(0)             READ A RECORD                        13850000
         L     R2,RETCODE          CHECK FOR SYNAD ENTRY                13900000
         LTR   R2,R2               IF NON-ZERO SYNAD ENTERED            13950000
         BNZ   REODAD              CLOSE DATA SET AND END               14000000
         LA    R9,89(R9)           GET TO NEXT SLOT IN VERIFY TBL       14050000
         LA    R6,1(0,R6)          INCREMENT NUMBER OF RECORDS          14100000
         ST    R6,0(0,R5)          UPDATE COUNTER IN BLOCK              14150000
         C     R6,MAXNAMES         SEE IF MAX NAMES IN TABLE YET        14200000
         BE    GNEXTBLK            IF SO, GET A NEW BLOCK               14250000
         B     GETINPUT            READ UNTIL EODOD                     14300000
GNEXTBLK DS    0H                                                       14350000
         GETMAIN RU,LV=4096,SP=0   GET BLOCK FROM SUBPOOL ZERO          14400000
*                                                                       14409000
         LTR   R15,R15             VERIFY GETMAIN WAS SUCCESSFUL        14418000
         BZ    BTABNEXT            BRANCH TO BUILD INITIAL TABLE        14427000
         LA    R15,ADUCRS05        LOAD REASON CODE OF 5                14436000
         ST    R15,REASCODE        STORE REASON CODE                    14445000
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       14454000
         ST    R15,RETCODE         SET RETURN CODE                      14463000
         B     ENDOPEN             END OPEN DATA SET                    14472000
*                                                                       14481000
BTABNEXT ST    R1,4(0,R5)          CHAIN TO CURRENT TABLE               14490000
         LR    R5,R1               GET THE VERIFY TABLE ADDRESS         14500000
         SLR   R6,R6               CLEAR TO ZERO                        14550000
         ST    R6,0(,R5)           INDICATE IN-CORE TABLE NOT YET SET   14600000
         ST    R6,4(,R5)           INDICATE IN-CORE TABLE NOT YET SET   14650000
         LA    R9,8(,R5)           GET PAST RECORD COUNTER              14700000
         B     GETINPUT            READ UNTIL EODOD                     14750000
REODAD   DS    0H                  END OF DATA ON READ                  14800000
         LA    R2,DCBI             GET INPUT DCB POINTER                14850000
         LA    R1,OPENL            GET AREA FOR CLOSE LIST              14900000
         CLOSE ((R2)),MF=(E,(1))   CLOSE INPUT DCB                      14950000
         FREEPOOL DCBI             RELEASE BUFFER POOL                  14975000
NOOPENI  DS    0H                  NO VERIFY DATA SET                   15000000
         L     R2,RETCODE          CHECK FOR ERROR                      15050000
         LTR   R2,R2               IF NON-ZERO ERROR OCCURRED           15100000
         BNE   ENDOPEN             END PROCESSING                       15150000
*                                                                       15160000
* BEGIN OUTPUT DATASET PROCCESSING ADDING DB2SSID TO DS NAME IF CCC     15165000
* LINES ARE UNCOMMENTED                                                 15170000
*                                                                       15170001
*        LA    R6,NTFYDB2          LOAD ADDRESS OF DB2SSID IN       CCC 15170002
*                                   NOTIFY DATA SET NAME                 15170003
*        LA    R8,ADUDSSID         LOAD ADDRESS OF DB2SSID FROM     CCC 15170005
*                                   CBRADUP                              15170006
*        CLC   0(1,R8),=CL1' '     COMPARE 1ST CHAR SSID WITH BLANK CCC 15170007
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170008
*        MVC   0(1,R6),0(R8)       MOVE 1ST CHAR SSID INTO THE      CCC 15170009
*                                   NOTIFY DATA SET NAME                 15170010
*        CLC   1(1,R8),=CL1' '     COMPARE 2ND CHAR SSID WITH BLANK CCC 15170011
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170012
*        MVC   1(1,R6),1(R8)       MOVE 2ND CHAR SSID INTO THE      CCC 15170013
*                                   NOTIFY DATA SET NAME                 15170014
*        CLC   2(1,R8),=CL1' '     COMPARE 3RD CHAR SSID WITH BLANK CCC 15170015
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170016
*        MVC   2(1,R6),2(R8)       MOVE 3RD CHAR SSID INTO THE      CCC 15170017
*                                   NOTIFY DATA SET NAME                 15170018
```

```
*        CLC   3(1,R8),=CL1' '     COMPARE 4TH CHAR SSID WITH BLANK CCC 15170019
*        BE    ENDSSID             IF EQUAL GOTO ENDSSID            CCC 15170020
*        MVC   3(1,R6),3(R8)       MOVE 4TH CHAR SSID INTO THE      CCC 15170021
*                                   NOTIFY DATA SET NAME                 15170022
*ENDSSID  DS    0H                                                   CCC 15170023
*                                                                       15170800
* BEGIN OUTPUT DATASET PROCCESSING                                      15191600
*                                                                       15212400
         LA    R6,NTFYDS2          LOAD ADDRESS OF SG NAME IN VERIFY    15233200
*                                   DATA SET NAME                        15254000
         L     R7,ADUSGLEN         LOAD LENGTH OF SG NAME               15274800
         LA    R8,ADUSGNAM         LOAD ADDRESS OF SG NAME FROM         15295600
*                                   CBRADUP                              15316400
         LR    R9,R7               PREPARE FOR MVCL                     15337200
         MVCL  R6,R8               MOVE SG NAME INTO THE VERIFY         15358000
*                                   DATA SET NAME                        15378800
*                                                                       15400000
* UPDATE TEXT UNIT POINTERS FOR THE NOTIFY DATA SET                     15450000
*                                                                       15500000
         LA    R3,0                GET ZERO FOR DDNAME TEXT UNIT PTR    15516600
         ST    R3,TXTDDN           PUT IT IN THE TXT UNIT PTR           15533200
         LA    R3,DSNTXTN          GET NOTIFY DSNAME TEXT UNIT PTR      15550000
         ST    R3,TXTDSN           PUT IT IN THE TXT UNIT PTR           15600000
*                                   INSTEAD OF VERIFY DSNAME             15650000
         LA    R3,MODTXT           GET STAT=SHR TEXT UNIT PTR           15800000
         ST    R3,TXTSTAT          STORE NEW TEXT UNIT PTR              15850000
         LA    R3,CTLGTXT          GET DISP=,KEEP TEXT UNIT PTR         15900000
         ST    R3,TXTDISP          STORE NEW TEXT UNIT PTR              15950000
         LA    R3,TRKTXT           GET TRACK ALLOATION TU PTR           16000000
         ST    R3,TXTTRK           STORE NEW TEXT UNIT PTR              16050000
         LA    R3,PRIMTXT          GET PRIMARY AMOUNT TU PTR            16100000
         ST    R3,TXTPRIM          STORE NEW TEXT UNIT PTR              16150000
         LA    R3,SECTXT           GETSECONDARY AMOUNT TU PTR           16200000
         ST    R3,TXTSEC           STORE NEW TEXT UNIT PTR              16250000
         MVC   RETTXT(6),RETSTAT   MOVE FROM STATIC TO DYNAMIC          16262500
         LA    R3,RETTXT           LET SYSTEM DETERMINE DDNAME          16275000
         ST    R3,TXTRET                                                16287500
         LA    R1,S99RBPTR         SET UP FOR SVC 99                    16300000
         DYNALLOC                  FREE THE DATA SET                    16350000
         LTR   R15,R15             IF DYNALLOC OKAY                     16400000
         BZ    OPENO               OPEN NOTIFY DATA SET                 16450000
         LA    R15,ADUCRS06        LOAD REASON CODE OF 6                16466600
         ST    R15,REASCODE        STORE REASON CODE                    16483200
         LA    R15,ADUFAILN        ERROR CAUSES NO RETURN TO ADUX       16500000
         ST    R15,RETCODE         SET RETURN CODE                      16550000
         B     ENDOPEN             END ADUX                             16600000
OPENO    DS    0H                  OPEN OUTPUT DATA SET                 16650000
         LA    R1,OPENL            GET OPEN LIST ADDRESS                16700000
         LA    R8,DCBO             GET INPUT DCB ADDRESS                16750000
         MVC   DCBDDNAM,RETTXT+6   MOVE SYSTEM GENERATED DDN INTO DCB   16775000
         OPEN  ((R8),OUTPUT),MF=(E,(1)) OPEN OUTPUT DCB                 16800000
         USING IHADCB,R8                                                16850000
         TM    DCBOFLGS,DCBOFOPN   WAS DS OPENED PROPERLY               16900000
         BNO   NOOPENO             SET NO NOTIFY TABLE                  16950000
         OI    FLAGS,ODCBO         INDICATE NOTIFY DATA SET IS OPEN     17000000
         B     ENDOPEN             END DATA SET OPENING                 17050000
NOOPENO  DS    0H                  OUTPUT DATA SET DID NOT OPEN         17100000
         LA    R2,ADUCRS07         LOAD REASON CODE OF 7                17116600
         ST    R2,REASCODE         STORE REASON CODE                    17133200
         LA    R2,ADUFAILN         SET FAILING RETURN CODE              17150000
         ST    R2,RETCODE          DO NOT RECALL ADUX                   17200000
ENDOPEN  DS    0H                  END OF OPENDS SUBROUTINE             17250000
         L     R14,SAVE14          GET RETURN ADDRESS                   17300000
```

```
         BR    R14                 RETURN TO MAINLINE                   17350000
         EJECT                                                          17400000
VERIFY   DS    0H                  VERIFY SUBROUTINE                    17450000
*********************************************************************** 17500000
*                                                                     * 17512500
* SUBROUTINE:  VERIFY                                                 * 17525000
*                                                                     * 17537500
* FUNCTION:    Search the Verify Table, built by OPENDS.  If the      * 17550000
*              object name is in the table then send a return code    * 17562500
*              indicating that auto-deletion should not occur.        * 17575000
*                                                                     * 17587500
* OPERATION:                                                          * 17600000
*     1.  Load address of the table pointer from Data Area.           * 17612500
*     2.  Retrieve the number of entries from Verify Table.           * 17625000
*     3.  If number of entries = 0 then end VERIFY.                   * 17637500
*     4.  While Names and collections to verify                       * 17650000
*         A. If ADUONAME = row in Verify Table, then check Collection * 17662500
*         B. Read next record from Verify Table.                      * 17675000
*         C. If entries = 0 then move to next Verify Table.           * 17687500
*         D. If ADUCLNAM = row in Verify Table, then set RC=4 and     * 17700000
*            REAS=2 and do not delete object.                         * 17712500
*     5.  Branch back to main line code.                              * 17725000
*                                                                     * 17737500
*********************************************************************** 17750000
         SPACE 1                                                        17800000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                  17850000
         L     R5,VRECSPTR         GET FIRST TABLE ADDRESS              17900000
COMPLOOP DS    0H                  COMPARE LOOP                         17950000
         L     R3,0(,R5)           GET COUNT OF ENTRIES                 18000000
         LA    R2,8(,R5)           GET ADDRESS OF FIRST NAME            18050000
         LTR   R3,R3               IF EMPTY FILE                        18066600
         BZ    ENDVERF             GO TO ENDVERF                        18083200
COMPNAME DS    0H                  NAME COMPARE LOOP                    18100000
         CLC   0(44,R2),ADUONAME   COMPARE NAME IN VERIFY TABLE         18150000
*                                  AGAINST OBJECT BEING DELETED         18200000
         BE    VCOLNAM             IF MATCH, CHECK COLLECTION NAME      18250000
DIFCOLNM LA    R2,89(,R2)          GET ADDRESS OF NEXT NAME             18300000
         BCT   R3,COMPNAME         DECREMENT COUNTER BY 1               18350000
         L     R5,4(,R5)           GET NEXT TABLE SECTION ADDRESS       18400000
         LTR   R5,R5               IF THERE IS A NEXT SECTION           18450000
         BZ    ENDVERF             END VERIFY IF NOT                    18500000
         B     COMPLOOP            COMPARE NEXT NAME                    18550000
VCOLNAM  DS    0H                  OBJ MATCH, CHECK COL NAME            18560000
         CLC   45(44,R2),ADUCLNAM  OBJ MATCH, SAME COL NAME?            18570000
         BE    VMATCH              IF MATCH, DO NOT DELETE              18580000
         B     DIFCOLNM            NO MATCH, GET NEXT OBJ NAME          18590000
VMATCH   DS    0H                  OBJECT NAME IS IN VERIFY TABLE       18600000
         LA    R1,ADU4RS02         LOAD REASON CODE OF 2                18616600
         ST    R1,REASCODE         STORE REASON CODE                    18633200
         LA    R1,ADUNODEL         GOOD COMPARE, DO NOT DELETE          18650000
         ST    R1,RETCODE          SET DO NOT DELETE RETURN CODE        18700000
ENDVERF  DS    0H                  END VERIFY SUBROUTINE                18750000
         L     R14,SAVE14          GET SAVED R14                        18800000
         BR    R14                 RETURN TO MAINLINE                   18850000
         EJECT                                                          18900000
NOTIFY   DS    0H                  NOTIFY SUBROUTINE                    18950000
*********************************************************************** 19000000
*                                                                     * 19016600
* SUBROUTINE:  NOTIFY                                                 * 19033200
*                                                                     * 19049800
* FUNCTION:    Write the object and collection names to the notify    * 19066400
*              data set.  This data set can be read by another        * 19083000
*              application for removal of object information.         * 19099600
```

```
*                                                                     * 19116200
* OPERATION:                                                          * 19132800
*     1.  Load address of the output DCB.                             * 19149400
*     2.  Load address of Object name and collection name output line * 19166000
*         for output to NOTIFY data set.                              * 19182600
*     3.  Write record to data set using PUT MACRO.                   * 19199200
*     4.  Branch back to main line processing.                        * 19215800
*                                                                     * 19232400
********************************************************************** 19250000
         SPACE 1                                                      19300000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                19350000
         LA    R1,DCBO             GET OUTPUT DCB ADDRESS             19400000
         LA    R0,ADUOBJCL         GET OBJECT NAME/COLL. ADDRESS      19450000
         PUT   (1),(0)             WRITE NAME TO NOTIFY DATA SET      19500000
         L     R14,SAVE14          GET SAVED R14                      19550000
         BR    R14                 RETURN TO MAINLINE                 19600000
         EJECT                                                        19650000
FREETAB  DS    0H                  FREEMAIN VERIFY TABLE SUBROUTINE   19700000
********************************************************************** 19750000
*                                                                     * 19761500
* SUBROUTINE:  FREETAB                                                * 19773000
*                                                                     * 19784500
* FUNCTION:    Free the in-storage Verify Table.                      * 19796000
*                                                                     * 19807500
* OPERATION:                                                          * 19819000
*     1.  Load address of the Verify Table Head.(VRECSPTR)            * 19830500
*     2.  While (tables exist [Pointer(R5) != 0])                     * 19842000
*         A.  Free the table storage by using the FREEMAIN macro.     * 19853500
*         B.  Load the next table address from current table.         * 19865000
*     3. Return to main line code.                                    * 19876500
*                                                                     * 19888000
********************************************************************** 19900000
         SPACE 1                                                      19950000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                20000000
         L     R5,VRECSPTR         GET FIRST TABLE ADDRESS            20050000
FREELOOP DS    0H                  LOOP THROUGH CHAINED TABLES        20100000
         LTR   R5,R5               IS THERE A TABLE ADDRESS           20200000
         BZ    ENDFREE             IF NOT, END FREEMAIN LOOP          20250000
         L     R3,4(,R5)           GET NEXT TABLE ADDRESS             20275000
         FREEMAIN RU,LV=4096,A=(R5) FREE TABLE SECTION                20300000
         LR    R5,R3               ADDRESS NEXT SECTION TO FREE       20350000
         B     FREELOOP            FREEMAIN NEXT SECTION              20400000
ENDFREE  DS    0H                  END OF TABLE FREEMAIN LOOP         20450000
         L     R14,SAVE14          GET SAVED R14                      20500000
         BR    R14                 RETURN TO MAINLINE                 20550000
         EJECT                                                        20600000
CLOSEDS  DS    0H                  CLOSE DATA SET SUBROUTINE          20650000
********************************************************************** 20700000
*                                                                     * 20750000
* SUBROUTINE:  CLOSEDS                                                * 20800000
*                                                                     * 20850000
* FUNCTION:    Closes the Notify data set.                            * 20858300
*                                                                     * 20866600
* OPERATION:                                                          * 20874900
*     1.  Check to see if Notify data set is currently open.          * 20883200
*     2.  Prepare for close macro.                                    * 20891500
*         A. Load address of the Open List. (R1)                      * 20899800
*         B. Load address of the Notify DCB. (R2)                     * 20908100
*     3.  Perform a close on the Notify data set using the CLOSE      * 20916400
*         macro.                                                      * 20924700
*     4.  Free the DCB buffer pool using the FREEPOOL macro.          * 20933000
*     5.  Return to main line code.                                   * 20941300
*                                                                     * 20950000
```

```
*********************************************************************** 21000000
         SPACE 1                                                        21050000
         ST    R14,SAVE14          SAVE RETURN ADDRESS                  21100000
         TM    FLAGS,ODCBO         CHECK TO SEE IF OUTPUT DS IS OPEN    21150000
         BNO   NOCLOSE             DO NOT ISSUE CLOSE IF NOT OPEN       21200000
         LA    R1,OPENL            GET AREA FOR CLOSE LIST              21250000
         LA    R2,DCBO             GET DCB ADDRESS                      21300000
         CLOSE ((R2)),MF=(E,(1))   CLOSE OUTPUT DCB NOTIFY DSN          21333300
         FREEPOOL DCBO             RELEASE OUTPUT BUFFER POOL           21366600
NOCLOSE  DS    0H                  DO NOT CLOSE                         21400000
         L     R14,SAVE14          GET SAVED R14                        21450000
         BR    R14                 RETURN TO MAINLINE                   21500000
         EJECT                                                          21550000
ADUXSYN  DS    0H                  BEGINNING OF WTO SUBROUTINE          21600000
*********************************************************************** 21650000
*                                                                     * 21700000
* SUBROUTINE:  ADUXSYN                                                * 21750000
*                                                                     * 21800000
* FUNCTION:    Handles input and output errors during file operations.* 21825000
*                                                                     * 21850000
* OPERATION:                                                          * 21875000
*     1.  Set a RC=12 and REAS=8 (ADUX not called again) due to I/O   * 21900000
*         error.                                                      * 21925000
*     2.  Store the return code for ADUX.                             * 21950000
*     3.  Return to calling module.                                   * 21975000
*                                                                     * 22000000
*********************************************************************** 22050000
         SPACE 1                                                        22100000
         LA    R2,ADUCRS08         LOAD REASON CODE OF 8                22116600
         ST    R2,REASCODE         STORE REASON CODE                    22133200
         LA    R2,ADUFAILN         SET FAIL CODE TO NO DELETE AND       22150000
*                                  DO NOT CALL ADUX FOR THIS SG         22183300
         ST    R2,RETCODE          STORE RETURN CODE                    22216600
         BR    R14                 RETURN TO MAINLINE                   22250000
         EJECT                                                          22300000
ADUXOUT  DS    0H                                                       22300600
*********************************************************************** 22301200
*                                                                     * 22301800
* SUBROUTINE:  ADUXOUT                                                * 22302400
*                                                                     * 22303000
* FUNCTION:    Writes to the console to show if an error occurred     * 22303600
*              while the CBRHADUX was processing.                     * 22304200
*                                                                     * 22304800
* OPERATION:                                                          * 22305400
*     1.  Determine whether the ADUXOUT should be performed based on  * 22306000
*         ADUFUNC and return code severity.                           * 22306600
*     2.  Copy from static to dynamic memory, the WTO macro list      * 22307200
*         definition along with the print line label.                 * 22307800
*     3.  Move current Storage group name to WTO message line.        * 22308400
*     4.  Convert the current return code and move to the WTO message * 22309000
*         line.                                                       * 22309600
*     5.  Convert the current reason code and move to the WTO message * 22310200
*         line.                                                       * 22310800
*     6.  Prepare to perform the WTO macro.                           * 22311400
*         A. Set the address of the message. (R5)                     * 22312000
*         B. Clear register 0 to zero.                                * 22312600
*     7.  Write to operator using the WTO macro.                      * 22313200
*     8.  Return to main line code.                                   * 22313800
*                                                                     * 22314400
* NOTE: If you change the return code severity to allow anything      * 22315000
*       below 8, can cause unexpected results, such as the WTO        * 22315600
*       could become full.  Modify the value of RETMIN to adjust the  * 22316200
*       severity of messages to be displayed.                         * 22316800
```

```
*                                                                     * 22317400
********************************************************************** 22318000
         SPACE 1                                                       22318600
         ST    R14,SAVE14          SAVE RETURN ADDRESS                 22319200
*                                                                      22319800
         CLI   ADUFUNC,ADUDONE     IS THIS AN END OF CYCLE CALL        22320400
         BE    OUTDONE             DO NOT PERFORM WTO ON EQUAL         22321000
         L     R5,RETCODE          LOAD CURRENT RETURN CODE            22321600
         C     R5,RETMIN           IF RETURN CODE < RETMIN THEN        22322200
         BL    OUTDONE              ENDADUX OUT                        22322800
         MVC   WTOLIST(WTOLISTL),WTOMODEL  COPY FROM MODULE STORAGE    22323400
*                                          TO WORKING STORAGE          22324000
         MVC   STATLINE(STATLEN),STATMODL  COPY FROM MODULE STORAGE    22324600
*                                          TO WORKING STORAGE          22325200
         MVC   STATSG(8),ADUSGNAM  MOVE SG TO STATUS LINE              22325800
         L     R5,RETCODE          LOAD CURRENT RETURN CODE            22326400
         CVD   R5,CONVERT          CONVERT TO PACK FORMAT              22327000
         UNPK  HEXWORK(8),CONVERT+4(4)  UNPACK FOR CONVERSION          22327600
         OI    HEXWORK+7,X'F0'     FORMAT LAST BYTE TO EBCDIC          22328200
         MVC   STATRET(8),HEXWORK  MOVE TO STATUS LINE                 22328800
         L     R5,REASCODE         LOAD CURRENT REASON CODE            22329400
         CVD   R5,CONVERT          CONVERT TO PACK FORMAT              22330000
         UNPK  HEXWORK(8),CONVERT+4(4)  UNPACK FOR CONVERSION          22330600
         OI    HEXWORK+7,X'F0'     FORMAT LAST BYTE TO EBCDIC          22331200
         MVC   STATREAS(8),HEXWORK MOVE TO STATUS LINE                 22331800
*                                                                      22332400
*        PERFORM WRITE TO OPERATOR REQUEST                             22333000
*                                                                      22333600
         LR    R5,R11              LOAD WORKING STORAGE AREA           22334200
         LA    R6,STATDISP         LOAD OFFSET OF STATUS LINE          22334800
         AR    R5,R6               FIND DISPLACEMENT TO STATUS         22335400
*                                   LINE FROM WORKING STORAGE           22336000
         SR    R0,R0               CLEAR REGISTER 0                    22336600
         WTO   TEXT=(R5),                                             +22337200
               MF=(E,WTOLIST)      PERFORM WTO TO CONSOLE              22337800
OUTDONE  L     R14,SAVE14          RESTORE RETURN ADDRESS              22338400
         BR    R14                 RETURN TO MAINLINE                  22339000
         EJECT                                                         22339600
********************************************************************** 22340200
*                                                                     * 22340800
*        LITERAL DEFINITIONS                                          * 22341400
*                                                                     * 22342000
********************************************************************** 22342600
         LTORG ,                   ORGANIZE LITERALS                   22343200
R0       EQU   0                                                       22350000
R1       EQU   1                                                       22400000
R2       EQU   2                                                       22450000
R3       EQU   3                                                       22500000
R4       EQU   4                                                       22550000
R5       EQU   5                                                       22600000
R6       EQU   6                                                       22650000
R7       EQU   7                                                       22700000
R8       EQU   8                                                       22750000
R9       EQU   9                                                       22800000
R10      EQU   10                                                      22850000
R11      EQU   11                                                      22900000
R12      EQU   12                                                      22950000
R13      EQU   13                                                      23000000
R14      EQU   14                                                      23050000
R15      EQU   15                                                      23100000
*---------------------------------------------------------------------* 23100100
*                                                                     * 23100200
*        MAX RETURN CODE TO DETERMINE WHAT SEVERITY OF MESSAGE        * 23100300
```

```
*         SHOULD BE REPORTED                                          * 23100400
*                                                                     * 23100500
*--------------------------------------------------------------------* 23100600
RETMIN   DC    F'8'               HIGHEST RETURN CODE THAT THE     MMM 23100700
*                                 ADUXOUT SHOULD WTO MESSAGES FOR      23100800
*--------------------------------------------------------------------* 23100900
*                                                                     * 23101000
*        REASON CODES FOR DIAGNOSTIC REASONS                          * 23101100
*                                                                     * 23101200
*--------------------------------------------------------------------* 23101300
*                                                                       23101400
* RETURN CODE ADUNODEL = RETCODE = 4                                   23101500
*                                                                       23101600
ADU4RS01 EQU   1    MANAGEMENT CLASS REJECTED OBJECT TO BE DELETED    23101700
ADU4RS02 EQU   2    OBJECT RESIDES IN VERIFY DATASET, NO DELETION     23101800
*                                                                       23101900
* RETURN CODE ADUFAILC = RETCODE = 8                                   23102000
*                                                                       23102100
ADU8RS01 EQU   1    DELETE ALL OBJECTS FOR THIS STORAGE GROUP AND     23102200
*                   DO NOT CALL EXIT AGAIN                            23102300
*                                                                       23102400
* RETURN CODE ADUFAILN = RETCODE = 12                                  23102500
*                                                                       23102600
ADUCRS01 EQU   1    DYNALLOC FAILED FOR VERIFY DATASET                23102700
ADUCRS02 EQU   2    VERIFY DATASET NOT CATOLOGED                      23102800
ADUCRS03 EQU   3    VERIFY DATASET FAILED TO OPEN                     23102900
ADUCRS04 EQU   4    GETMAIN FAILED FOR INITIAL ON VERIFY TABLE        23103000
ADUCRS05 EQU   5    GETMAIN FAILED FOR NEXT ON VERIFY TABLE           23103100
ADUCRS06 EQU   6    DYNALLOC FAILED FOR NOTIFY DATASET                23103200
ADUCRS07 EQU   7    NOTIFY DATASET FAILED TO OPEN                     23103300
ADUCRS08 EQU   8    I/O ERROR OCCURRED DURING GETMAIN                 23103400
*--------------------------------------------------------------------* 23103500
*                                                                     * 23103600
*        SINGLE LINE WTO PARAMETER LIST                               * 23103700
*                                                                     * 23103800
*--------------------------------------------------------------------* 23103900
WTOMODEL WTO   TEXT=((,DE)),                                         +23104000
               MF=L                                                    23104100
         SPACE 2                                                       23104200
*--------------------------------------------------------------------* 23104300
*                                                                     * 23104400
*        SINGLE LINE WTO MODEL LINE                                   * 23104500
*                                                                     * 23104600
*--------------------------------------------------------------------* 23104700
STATMODL DC    0F'0'              STATUS LINE MODEL FOR WORKING        23104800
*                                 STORAGE                              23104900
         DC    AL2(STATMLEN-2)    LENGTH OF MESSAGE FOR WTO MACRO      23105000
         DC    C'CBRHADUX USER EXIT: SG ('  START OF WTO MSG          23105100
         DC    C'        '        PLACE HOLDER FOR STORAGE GROUP       23105200
         DC    C') completed RC='  LABEL FOR RETURN CODE               23105300
         DC    C'        '        PLACE HOLDER FOR RETURN CODE         23105400
         DC    C' RSN='           LABEL FOR REASON CODE                23105500
         DC    C'        '        PLACE HOLDER FOR REASON CODE         23105600
STATMLEN EQU   *-STATMODL         LENGTH OF STATUS LINE                23105700
*                                                                       23105800
*--------------------------------------------------------------------* 23105900
*                                                                     * 23106000
*        MANAGEMENT CLASS VERIFICATION                                * 23106100
*                                                                     * 23106200
*--------------------------------------------------------------------* 23106300
MCTAB    DC    C'MCNODEL1'        MANAGEMENT CLASS TABLE               23112500
         DC    C'MCNODEL2'                                             23125000
MCCNT    EQU   (*-MCTAB)/8                                            23137500
```

```
MAXNAMES DC    F'44'              MAXIMUM NUMBER OF NAMES IN           23168700
*                                 A 4K BLOCK OF VERIFY TABLE           23200000
*--------------------------------------------------------------------* 23202000
*                                                                    * 23204000
*        DD STATEMENT DECLARATIONS                                   * 23206000
*                                                                    * 23208000
*--------------------------------------------------------------------* 23210000
RETSTAT  DC    AL2(85)                                                23212500
         DC    AL2(1)                                                 23225000
         DC    AL2(8)                                                 23237500
DDNMTXT  DS    0F                                                     23250000
         DC    AL2(1)                                                 23300000
         DC    AL2(1)                                                 23350000
         DC    AL2(ENDDDN-STRTDDN)                                    23400000
STRTDDN  DC    C'CBRADUXI'                                            23450000
ENDDDN   EQU *                                                       23500000
MODTXT   DC    AL2(4)             STATUS=MOD                          23850000
         DC    AL2(1)                                                 23900000
         DC    AL2(1)                                                 23950000
         DC    AL1(2)                                                 24000000
SHRTXT   DC    AL2(4)             STATUS=SHR                          24050000
         DC    AL2(1)                                                 24100000
         DC    AL2(1)                                                 24150000
         DC    AL1(8)                                                 24200000
TRKTXT   DC    AL2(7)             TRACK ALLOCATION IF NOT OLD         24250000
         DC    AL2(0)                                                 24300000
PRIMTXT  DC    AL2(10)                                                24350000
         DC    AL2(1)                                                 24400000
         DC    AL2(3)                                                 24450000
         DC    AL3(10)            10 TRACKS PRIMARY ALLOCATION        24500000
SECTXT   DC    AL2(11)                                                24550000
         DC    AL2(1)                                                 24600000
         DC    AL2(3)                                                 24650000
         DC    AL3(10)            10 TRACKS SECONDARY ALLOCATION      24700000
CTLGTXT  DC    AL2(5)             DISP=CATLG                          24750000
         DC    AL2(1)                                                 24800000
         DC    AL2(1)                                                 24850000
         DC    AL1(2)                                                 24900000
KEEPTXT  DC    AL2(5)             DISP=KEEP                           24950000
         DC    AL2(1)                                                 25000000
         DC    AL2(1)                                                 25050000
         DC    AL1(8)                                                 25100000
UNITTXT  DC    AL2(21)                                                25150000
         DC    AL2(1)                                                 25200000
         DC    AL2(5)                                                 25250000
         DC    C'SYSDA'                                               25300000
CLOSTXT  DC    AL2(28)            FREE DATA SET WHEN CLOSED           25350000
         DC    AL2(0)                                                 25400000
DSNTXT   DC    AL2(2)                                                 25450000
         DC    AL2(1)                                                 25500000
         DC    AL2(ENDDSN-VRFYDSN)                                    25550000
VRFYDSN  DC    C'HLQ.OBJECT.DELETE.VERIFY'                            25600000
ENDDSN   EQU   *                                                     25650000
STATAREA DS    0F                 STATIC CBS TO BE MOVED              25700000
         OPEN   (,),MF=L                                             25750000
DCBSTAT  DS    0F                                                     25800000
*   NOTE:  IF THIS DATASET IS PREALLOCATED PRIOR TO THE INVOCATION   * 25807100
*          OF THIS ROUTINE, THE DCB DECLARES OF THE DATASET SHOULD   * 25814200
*          BE REMOVED FROM THIS ROUTINE.  OTHERWISE, THE ALLOCATION  * 25821300
*          HERE WILL OVERRIDE THE PREALLOCATION OF THE DATASET,      * 25828400
*          CAUSING UNEXPECTED OUTPUT.                                * 25835500
*                                                                    * 25842600
         DCB  DDNAME=CBRADUXO,MACRF=(PM),OPTCD=W,LRECL=89,BLKSIZE=8900,+25871300
```

```
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN                              25900000
         DCB  DDNAME=CBRADUXI,MACRF=(GM),LRECL=89,BLKSIZE=8900,            +25950000
               DSORG=PS,RECFM=FB,SYNAD=ADUXSYN,EODAD=REODAD                 26000000
DYNRBC   DS    0F                                                           26050000
         DC    AL1(20)             RB LENGTH                                26100000
         DC    AL1(01)             DSNAME ALLOCATION                        26150000
         DC    X'C0'               FLAGS1 - NO EXIST ALLOC                  26200000
         DC    X'0' FLAGS1                                                  26250000
         DC    F'0'                ERROR CODES                              26300000
         DC    A(0)                TEXT UNIT POINTERS                       26350000
         DC    F'0'                RESERVED                                 26400000
         DC    X'00'               WAIT FOR VOLS,UNITS,DSNS AND MOUNTS      26450000
*        DC    X'D1'               WAIT FOR VOLS,UNITS,DSNS AND MOUNTS      26500000
         DC    AL3(0)                                                       26550000
TXTPTRC  DS    0F                                                           26600000
         DC    A(DSNTXT)                                                    26650000
         DC    A(DDNMTXT)                                                   26700000
         DC    A(SHRTXT)                                                    26750000
         DC    A(KEEPTXT)                                                   26800000
         DC    A(0)                TRACK TEXT UNIT FOR OUTPUT DS            26850000
         DC    A(0)                PRIMARY TEXT UNIT FOR OUTPUT DS          26900000
         DC    A(0)                SECONDARY TEXT UNIT FOR OUTPUT DS        26950000
         DC    A(0)                RETURN DDNAME FOR OUTPUT DS              26983300
         DC    A(UNITTXT)          UNIT TEXT UNIT FOR OUTOUT DS             27016600
         DC    A(CLOSTXT)          FREE (UN-ALLOCATE) AT CLOSE              27050000
         DC    X'80'                                                        27100000
         DC    AL3(0)                                                       27150000
         DC    AL2(2)                                                       27200000
         DC    AL2(1)                                                       27250000
         DC    AL2(ENDD1-D1)                                                27300000
D1       DC    C'HLQ.XXXXXXXX.OBJECT.DELETE.NOTIFY'                   MMM 27350000
* COMMENT OUT THE ABOVE LINE WHEN UNCOMMENTING THE LINE BELOW               27350001
* D1       DC    C'HLQ.YYYY.XXXXXXXX.OBJECT.DELETE.NOTIFY'              CCC 27350002
ENDD1    EQU   *                                                          27400000
MOVELN   EQU   *-STATAREA                                                 27450000
         EJECT                                                           27500000
         DCBD    DSORG=QS                                                30033300
         EJECT                                                           30049900
         IEFZB4D2                                                        30066600
         END CBRHADUX                                                    30116600
```