Sample Exit to use JCL UPSI switches with COBOL/VSE and LE/VSE

With this unsupported modification to the supplied CEEBXITA LE/VSE Assembler user exit program you can use the JCL // UPSI switches with COBOL/VSE programs running under LE/VSE 1.4.

The modification has been tested using VSE/ESA 2.5.1 and LE/VSE 1.4.1 and COBOL/VSE 1.1.1 programs with complete success. **No support or service of this sample is implied or acknowledged and use of it is completely at the user's discretion and responsibility.** Use of this sample on higher language environment levels is possible but is not supported and is at the user's responsibility.

This JCL will Assemble the sample user exit and then LINKEDT a new CEEBXITA PHASE.
By default it is installed as an environment wide assembler exit.
To activate the user exit this new CEEBXITA PHASE MUST be the one used at runtime for COBOL/VSE programs and the LE/VSE UPSI runtime option in CEEDOPT MUST have OVR specified. Please tailor the JCL before executing.

```
* $$ JOB JNM=CEEBXITA,CLASS=S,DISP=D,LDEST=(*,userid) <== Please change
// JOB CEEBXITA  ASSEMBLE CEEBXITA SAMPLE USER EXIT
// SETPARM LEBASE='PRD2.SCEEBASE'  <== Set to LE install lib.sublib
// SETPARM USRLIB='user.testlib'   <== set to output lib.sublib.
/* *****************************************************************
/*  PLEASE NOTE           IMPORTANT                                *
/* --------------------------------------------------------------  *
/*                                                                 *
/* This is an unsupported modification to the supplied CEEBXITA    *
/* LE/VSE Assembler user exit program to provide support for the   *
/* JCL // UPSI switches with COBOL/VSE programs running under      *
/* LE/VSE 1.4.1 and above.                                         *
/*                                                                 *
/* This sample has been tested using VSE/ESA 2.5.1, LE/VSE 1.4.1   *
/* and COBOL/VSE 1.1.1 programs successfully.                      *
/*                                                                 *
/* No support or service of this sample is implied or acknowledged *
/* and use of it is completely at the users discretion and         *
/* responsibility.                                                 *
/*                                                                 *
/* This JCL will Assemble the sample user exit and then catalog    *
/* a CEEBXITA.OBJ and finally linkedit a CEEBXITA.PHASE. To        *
/* activate this exit, the user can choose to either include the   *
/* CEEBXITA.OBJ into application load modules they wish to use     *
/* the JCL // UPSI card or ensure that the CEEBXITA.PHASE produced *
/* by this JCL is the version used at execution time. It should    *
/* be noted that any CEEBXITA.OBJ included in the main application *
/* load module will over-ride any CEEBXITA.PHASE at execution      *
/* time.                                                           *
/*                                                                 *
/* Restrictions :                                                  *
/*                                                                 *
/*  This assembler user exit is not compatible with the Debug      *
/* Tool for VSE DL/I exit, EQADLIXA, as supplied with Debug Tool   *
/* for VSE/ESA 1.1.1 as CEEBXITA.PHASE. Either this sample exit is *
/* used at execution time, or the Debug Tool exit is used. They    *
/* cannot both be used in the same application at execution time.  *
/*                                                                 *
/* Please tailor this JCL where appropriate before executing.      *
/*                                                                 *
/* *****************************************************************
// PAUSE Please ensure JCL has been tailored before executing.
// LIBDEF *,SEARCH=(PRD1.MACLIB,&LEBASE)
// LIBDEF *,CATALOG=&USRLIB
// DLBL IJSYSPH,'CEEBXITA.SYSPCH.FILE',0,SD
// EXTENT SYSPCH,volid,1,0,xxxx,yy           <== Please change
ASSGN SYSPCH,DISK,VOL=volid,SHR              <== Please change
// OPTION DECK
// EXEC ASMA90,SIZE=(ASMA90,90K),PARM='EX(LBX(EDECKXIT))'
         PUNCH 'CATALOG CEEBXITA.OBJ R=Y'
         TITLE 'LE/VSE Assembler User Exit under VSE'
*/*******************************************************************/
*/*                                                                 */
*/*    LICENSED MATERIALS - PROPERTY OF IBM                         */
*/*                                                                 */
*/*    5686-066  (C) COPYRIGHT IBM CORP. 1991,  2000                */
*/*    ALL RIGHTS RESERVED.                                         */
*/*                                                                 */
*/*   US Government Users Restricted Rights - Use, duplication or   */
*/*   disclosure restricted by GSA ADP Schedule Contract with IBM   */
*/*   Corp.                                                         */
*/*                                                                 */
*/*******************************************************************/
CEEBXITA CSECT
CEEBXITA AMODE  ANY
CEEBXITA RMODE  ANY
         ENTRY  CEEBXITA
*/*-----------------------------------------------------------------*/
*/* CEEBXITA - Run-Time Assembler User Exit under VSE.              */
*/*                                                                 */
*/* Function code:                                                  */
*/*    1 - first enclave initialization                             */
*/*    2 - first enclave termination                                */
*/*    3 - nested enclave initialization                            */
*/*    4 - nested enclave termination                               */
*/*    5 - PROCESS TERMINATION                                      */
*/*                                                                 */
*/* Register Usage:                                                 */
*/*    Input                                                        */
*/*       R1  - A(A(Assembler User Exit Plist cntrl blk)) - CEEAUE  */
*/*             NOTE: CEEAUE  contains the address of a 256 Byte    */
*/*                   work area initialized to zero which is used   */
*/*                   as the CEEBXITA's Save Area                   */
*/*       R15 - Entry Point Address                                 */
*/*       R14 - Return Address                                      */
*/*       R12 - Pointer to CAA                                      */
*/*    Output                                                       */
*/*       CEEAUE  altered                                           */
*/*    Work Regs                                                    */
*/*       R3 - Code Base                                            */
*/*       R2 - Base for User Exit Control Block (CEEAUE)            */
*/*                                                                 */
*/* History:                                                        */
*/*                                                                 */
*/*  19/02/99 GWH  Modify exit to use // UPSI settings in place of  */
*/*            the LE/VSE UPSI runtime option.                      */
*/*  10/03/03 GWH  Update for supported LE/VSE levels.              */
*/*                (Currently, LE/VSE 1.4.1 and above)              */
*/*                                                                 */
*/*-----------------------------------------------------------------*/
         SPACE 2
         USING CEEBXITA,R3        Code Base
         USING CEEAUE,R2         User Exit Control Block
         SPACE 1
         STM   14,12,12(13)       Save Registers in caller's S.A.
         LR    R3,R15             Set Module Base
         B     AROUND
         DC    CL8'CEEBXITA'      Eye-catcher for Debugging
         DC    CL8'V1.R4.M1'
         DC    CL8'12.03.03'
*
AROUND   DS    0H
```

```
         L     R2,0(,R1)          Get the parm passed
         L     R10,CEEAUE_A_WORK    Obtain the workarea
*                                 Chain the Save Areas
         ST    R10,8(,R13)        Old points to new
         ST    R13,4(,R10)        New SaveArea points to old
         LR    R13,R10            R13 points to new SaveArea
         USING XITASTOR,R13
*
         L    R4,CEEAUE_FUNC        Get the function code.
         C    R4,FRST_INIT_CODE   First EnclaveInitialization Exit?
         BE   FRST_INIT_EXIT      Yes. Branch to it?
*
         C    R4,FRST_TERM_CODE   First Enclave Termination Exit?
         BE   FRST_TERM_EXIT      Yes. Branch to it?
*
         C    R4,NSTD_INIT_CODE   Nested Enclave Initialization Exit?
         BE   NSTD_INIT_EXIT      Yes. Branch to it?
*
         C    R4,NSTD_TERM_CODE   Nested Enclave Termination Exit?
         BE   NSTD_TERM_EXIT      Yes. Branch to it?
*
         C    R4,PROC_TERM_CODE   Process Termination Exit?
         BE   PROC_TERM_EXIT      Yes. Branch to it?
*
         B    RETURN              It is neither Init nor Term
*                                 ...exit code. Simply Return
         EJECT
*/*==================================================================*/
*/*      First enclave initialization exit.                          */
*/*==================================================================*/
FRST_INIT_EXIT DS   0H
         SPACE 1
*
*  To activate the abend codes table, uncomment the next 2 lines
*
*        LA    R4,ABEND_CODES     Get addr(abend codes table)
*        ST    R4,CEEAUE_A_AB_CODES Store it in slot
*
* *********************************************************************
*
*  NOTE :
*
*    To bypass the use of the // UPSI JCL card, uncomment the following
* branch instruction.
*
* *********************************************************************
*        B     RETURN                 use LE/VSE UPSI option
* *********************************************************************
*
*  The following code will replace the LE/VSE UPSI runtime option with
* contents of the JCL // UPSI byte. This will disable to use of the
* LE/VSE UPSI runtime option as a parameter over-ride, CEEUOPT or as
* a default option (CEEDOPT).
*
         MVC   UPSIOPN,=C'UPSI('      initialize runtime option
         MVI   UPSICLO,C')'
         LA    R8,14                  save length of option
         STH   R8,NEWOLEN
         COMRG REG=(8)                address part. comreg.
         USING COMREG,R8
         MVC   COMRUPSI(1),UPSI       retrieve current UPSI byte
         DROP  R8
*
* Convert COMREG UPSI byte into LE UPSI runtime option format
*
         MVI   NEWUPSI,C'0'           initialize UPSI to zeros
         MVC   NEWUPSI+1(L'NEWUPSI-1),NEWUPSI
         XR    R8,R8
         ICM   R8,8,COMRUPSI          get comreg UPSI byte
         LA    R9,NEWUPSI             address the new UPSI area
SHIFT    EQU   *
         ALR   R8,R8                  interrogate bit and shift
         BC    12,NOTON               zero found, dont change
         MVI   0(R9),C'1'             insert C'1' in new UPSI
NOTON    EQU   *
         BC    10,DONE                no more 1's so finish
         LA    R9,1(R9)               point to next byte
         B     SHIFT                  check next bit
DONE     EQU   *
         LA    R9,NEWOPTS             address new UPSI option
         ST    R9,ANEWOPTS            save
         LA    R9,ANEWOPTS            address the UPSI option addr
         ST    R9,CEEAUE_A_OPTIONS    save in parm list for LE/VSE
         B     RETURN                 return to LE/VSE
         SPACE 1
*/*=================================================================*/
*/*      First enclave termination exit.                            */
*/*=================================================================*/
         SPACE 1
*
FRST_TERM_EXIT DS   0H
         B     RETURN
         SPACE 1
*/*=================================================================*/
*/*      Nested enclave initialization exit.                        */
*/*=================================================================*/
         SPACE 1
*
NSTD_INIT_EXIT DS   0H
         B     RETURN
         SPACE 1
*/*=================================================================*/
*/*      Nested enclave termination exit.                           */
*/*=================================================================*/
         SPACE 1
*
NSTD_TERM_EXIT DS   0H
         B     RETURN
*
         SPACE 1
*/*=================================================================*/
*/*      Process termination exit.                                  */
*/*=================================================================*/
         SPACE 1
*
PROC_TERM_EXIT DS   0H
*        B     RETURN
*
         SPACE 1
*/*=================================================================*/
*/*               R E T U R N     T O    C A L L E R                */
*/*=================================================================*/
RETURN   DS    0H
         L     R13,4(,R13)        Get A(caller's save area)
         LM    R14,R12,12(R13)    Restore caller's regs
         BR    R14                Go home
         EJECT
*              CONSTANTS AND WORKAREAS
*
FRST_INIT_CODE  DC  F'1'          First enclave initialization code
FRST_TERM_CODE  DC  F'2'          First enclave termination code
NSTD_INIT_CODE  DC  F'3'          Nested enclave initialization code
NSTD_TERM_CODE  DC  F'4'          Nested enclave termination code
PROC_TERM_CODE  DC  F'5'          Process termination code
*
*
```

```
*   The first field in the following table is a count
*   of how many abend codes are not to be trapped by
*   the LE/VSE error handler.
*
*   If you want the LE/VSE error handler NOT to trap:
*      - a system ss cancel code, enter X'000000ss' into the table
*      - a system ii interruption code, enter X'800000ii' into the
*        table
*      - a user uuuu abend, enter F'uuuu' into the table
*
ABEND_CODES DS 0H
*
           DC  A(((ABEND_CODES_END-ABEND_CODES)/4)-1)
*
*          DC  F'uuu'             LE/VSE won't trap user uuu abends
*          DC  X'000000ss'        LE/VSE won't trap system ss cancel
*                                 codes
*          DC  X'800000ii'        LE/VSE won't trap system ii
*                                 interruption codes
*
ABEND_CODES_END DS 0H
*
                     SPACE 1
*===================================================================
*         Parameter list passed to the User Exit.
*         Addressed by Reg 1 = A(A(CEEAUE))
*===================================================================
CEEAUE               DSECT 0D
CEEAUE_LEN           DS  F        Len of user exit Ctl Blk
CEEAUE_FUNC          DS  F        Function Code
*                                 1 ==> Initialization Exit
*                                 2 ==> Termination Exit
*                                 3 ==> Init Exit for nested enclave
*                                 4 ==> Term Exit for nested enclave
*                                 5 ==> Process termination Exit
CEEAUE_RETC          DS  F        Return or ABEND code
CEEAUE_RSNC          DS  F        Return or ABEND Reason Code
*
CEEAUE_FLAGS         DS  0F       Flags
CEEAUE_FLAG1         DS  X
CEEAUE_ABTERM        EQU X'80'    0 = Normal termination
*                                 1 = Abnormal termination
CEEAUE_ABND          EQU X'40'    0 = terminate pgm with Return/Reason
*                                 1 = terminate pgm with ABEND/Reason
CEEAUE_DUMP          EQU X'20'    0 = If CXIT_ABND is 1, ABEND without
*                                     a dump
*                                 1 = IF CXIT_ABND is 1, ABEND  w/ dump
                     SPACE 1
CEEAUE_FLAG2         DS  X        Reserved
CEEAUE_FLAG3         DS  X        Reserved
CEEAUE_FLAG4         DS  X        Reserved
                     SPACE 1
CEEAUE_A_CC_PLIST    DS  A        Ptr to Reg 1 upon invocation
*                                 ...Initialization Exit Only
CEEAUE_A_WORK        DS  A        Ptr to 256-byte work area
CEEAUE_A_OPTIONS     DS  A        A(A(Run-Time Options string))
CEEAUE_USERWD        DS  F        User word. Retained thru Termination
CEEAUE_A_AB_CODES    DS  A        Ptr to list of abend codes that
*                                 LE/VSE error handler shouldn't trap
*                                  format of table is:
*                                    fullword: # of entries
*                                    fullword: error code
*                                    fullword: error code
*                                      ...
*                                  Use X'000000ss' for system cancel
*                                                  codes
*                                  Use X'800000ii' for system interrupt
*                                                  codes
*                                      F'uuuu'     for user abends
CEEAUE_FBCODE        DS  A        Feedback code
CEEAUE_PAGE          DS  F        Minimum value for page allocations
                     SPACE 1
*/*=================================================================*/
*/*      E N D     O F     P A R A M E T E R    L I S T             */
*/*=================================================================*/
                     SPACE 1
XITASTOR DSECT
SAVEAREA             DS  CL72
COMRUPSI             DS  CL1
ANEWOPTS             DS  F
NEWOPTS              DS  0CL16
NEWOLEN              DS  H
UPSIOPN              DS  CL5
NEWUPSI              DS  CL8
UPSICLO              DS  CL1
                     SPACE 1
         MAPCOMR
CEEBXITA CSECT
R0       EQU   0
R1       EQU   1
R2       EQU   2
R3       EQU   3
R4       EQU   4
R5       EQU   5
R6       EQU   6
R7       EQU   7
R8       EQU   8
R9       EQU   9
R10      EQU   10
R11      EQU   11
R12      EQU   12
R13      EQU   13
R14      EQU   14
R15      EQU   15
         END
/*
CLOSE SYSPCH,xxx                  <== Please change
// IF $RC GT 4 THEN
// GOTO $EOJ
// DLBL IJSYSIN,'CEEBXITA.SYSPCH.FILE',0,SD
// EXTENT SYSIPT,volid            <=== Please change
ASSGN SYSIPT,DISK,VOL=volid,SHR   <=== Please change
// EXEC LIBR,SIZE=256K,PARM='ACC S=&USRLIB'
/*
CLOSE SYSIPT,xxx                  <=== Please change
/*
* Linkedit CEEBXITA Load Module
// LIBDEF *,SEARCH=(&USRLIB,&LEBASE)
// OPTION CATAL
// LIBDEF PHASE,CATALOG=&USRLIB
 INCLUDE CEE$BXIT
/*
// EXEC LNKEDT,SIZE=256K,PARM='MSHP'
/*
/&
* $$ EOJ
```

Trademarks

Comments and Questions

Comments or questions on this documentation are welcome. Please send your comments to:

zvse@de.ibm.com