z390

File Access Method Guide

Automated Software Tools Corporation

VSAM support

The macros GET, PUT and POINT can be used for VSAM access with the parameter RPL=rpladdress. These are not documented here, please refer to the **z390_VSAM_User_Guide.**

Acronyms
BDW -- Block Descriptor Word
   Only used for RECFM=VB files.
   A 4-byte field of the form H'blocklength',H'0'
   The blocklength includes the length of the BDW.

RDW -- Record Descriptor Word
   Used for all variable files.
   A 4-byte field of the form H'recordlength',H'0'
   The recordlength includes the length of the RDW.
   The program must set this field correctly for output,
   and expect it on input.

CRLF -- Carriage Return, Line Feed
   A term for the hex string X'0D0A' (ASCII) or X'0D25' (EBCDIC).
   Now commonly used to terminate a text record.

Function Table

SVC		FUNCTION
DEC	HEX	
19	13	**OPEN**
19	13	**OPEN (VSE)**
20	14	**CLOSE**
20	14	**CLOSE (VSE)**
151	97	**GET**
152	98	**PUT**
153	99	**READ**
154	9A	**WRITE**
155	9B	**CHECK**
156	9C	**POINT**

Supporting Macros

DCB	Define a file
DCBD	DCB structure (DSECT)
DCBE	DCBE fields
DECBD	DECB structure (DSECT)
IHADCBE	DCBE structure (DSECT)
DTFSD	Define a file (VSE)
DTFPR	Define a file (VSE)

Sample Programs

The z390 folder rt\test contains many examples of File I/O usage and have the generic name of TESTDCB?.MLC.

DCB

z390 supports sequential and random access to files.

To access a file, a data control block (DCB) must be defined.

All programs that use the File I/O macros must include the DCBD DSECT,
so a typical program end might be:

```
DCBD
EQUREGS
END
```

Add the DECBD DSECT if you are using READ or WRITE.

Some of the parameters may be set before and/or after OPEN.
The individual descriptions indicate this.

```
label   DCB   DSORG=PS,
              EODAD=0,
              RECFM=FB,
              DDNAME=,
              MACRF=GM,
              SYNAD=0,
              BLKSIZE=0,
              LRECL=0,
              DCBE=,
              RECORD=0,
              DSNAME=0
```
EXLST is currently unsupported.

'label' is the file identifier, the DCB name, which is used in all
the file I/O macros.

```
DDNAME    DSECT=DCBDDNAM  Type=CL8  Default=CL8'label'
DSNAME    DSECT=DCBDSNAM  Type=A    No default
```
  Can these be set in the DCB prior to OPEN: Yes
  Can these be set in the DCB after OPEN   : No

  Only one of these parameters may be set in the DCB.
  If both parameters are set in the program prior to open,
  the DSNAME will take precedence.

When DDNAME is specified, a SET variable will point to the path and file that is to be opened.

Within a program the GETENV macro can be used to extract the SET variable. See the "Macro and Services Guide" for full details.

eg.
MYDCB DCB DDNAME=MYDATA, ...

For execution, set up a .BAT (Batch) file...
   SET MYDATA=drive:\path\file
   CALL EZ390 drive:\path\program parms

DSNAME is a label defined in the program which has the file spec.
The file spec must terminate with X'00' or be defined as a
double-quoted string within a standard C-type constant.

eg.
MYDCB DCB DSNAME=MYDATA, ...
     ...
MYDATA DC   C'drive:\path\file',X'00'
 or
MYDATA DC   C'"drive:\path\file"'

MACRF   DSECT=DCBMACRF Coded   Default=GM
  Can this be set in the DCB prior to OPEN: Yes, but not recommended
  Can this be set in the DCB after OPEN   : No

  MACRF=GM Get Move, use the GET macro to read a record
        Use for standard (QSAM) file read

  MACRF=PM Put Move, use the PUT macro to write a record
        Use for standard (QSAM) file write

  MACRF=R   Read Mode.
        Use READ/CHECK to read a record.
        Use POINT for positioning.

  MACRF=W   Write Mode.
        Use WRITE/CHECK to write or update a record.
        Use POINT for positioning.

MACRF=RW Update Mode.
        Use READ/WRITE/CHECK to read, write or update a record.
        Use POINT for positioning.

DSORG   DSECT=DCBDSORG Coded   Default=PS
   Can this be set in the DCB prior to OPEN: Yes, but not recommended
   Can this be set in the DCB after OPEN   : No

   DSORG=PS Physical Sequential
        The only option at present, can be omitted.

RECFM   DSECT=DCBRECFM Coded   Default=FB
   Can this be set in the DCB prior to OPEN: Yes, but not recommended
   Can this be set in the DCB after OPEN   : No

   RECFM=F   Fixed
        Translation: None

        Input:
          Records are read LRECL at a time.

        Output:
          Records are written LRECL at a time.

   RECFM=FB Fixed Blocked
        Translation: None

        Input and MACRF=GM:
          Records are read LRECL at a time out of BLKSIZE.

        Input and MACRF=R or RW:
          The whole block is read.

        Output and MACRF=PM:
          Records are written LRECL at a time into BLKSIZE.
          CLOSE may write a short block.

        Output and MACRF=W or RW:
          The whole block is written.

RECFM=V   Variable
    Translation: None

    Input:
      Each record is prefixed by the RDW.
      The receiving area must be big enough for the largest
      RDW+record.
      If the RDW indicates a record size larger than LRECL,
      abend S013 will occur.

    Output:
      Each record must be prefixed by the RDW.

RECFM=VB Variable Blocked
    Translation: None

    Input and MACRF=GM:
      Each record is prefixed by the RDW.
      The receiving area must be big enough for the largest
      RDW+record.
      If the RDW indicates a record size larger than LRECL,
      abend S013 will occur.

    Input and MACRF=R or RW:
      The whole block is read, prefixed by the BDW.
      The receiving area must be big enough for the largest
      BDW+block.
      If the BDW indicates a block size larger than BLKSIZE,
      abend S013 will occur.

    Output and MACRF=PM:
      Each record must be prefixed by the RDW.
      CLOSE will write the last block.
      Each block written will have the BDW automatically
      inserted at the start.

    Output and MACRF=W or RW:
      Each block must be prefixed by the BDW.

RECFM=FT Fixed ASCII text
    Translation: In ASCII mode: None

: For non-ASCII input
  ASCII chars are translated to EBCDIC after
  being read.

: For non-ASCII mode output
  EBCDIC chars are translated to ASCII before
  being written, this is an internal function
  and does not affect the record in storage.
  Non-EBCDIC chars may be translated to X'00'.

Input:
  The file is assumed to be in conventional ASCII format,
  with each record ending in CRLF.
  CRLF is never read as part of the record.
  If the record is shorter than LRECL, then trailing
  blanks are inserted.
  EODAD is invoked when all bytes have been read.

Output:
  CRLF is inserted at the end of each record after
  trailing blanks are stripped.

RECFM=VT Variable ASCII text
  Translation: In ASCII mode: None
        : For non-ASCII input
          ASCII chars are translated to EBCDIC after
          being read.

        : For non-ASCII mode output
          EBCDIC chars are translated to ASCII before
          being written, this is an internal function
          and does not affect the record in storage.
          Non-EBCDIC chars may be translated to X'00'.

  Input:
    The file is assumed to be in conventional ASCII format,
    with each record ending in CRLF.
    CRLF is never read as part of the record.
    Each record is prefixed with the RDW.
    EODAD is invoked when all bytes have been read.
    The receiving area must be big enough for the largest
    RDW+record.

If the RDW indicates a record size larger than LRECL,
abend S013 will occur.

Output:
Each record must be prefixed with the RDW.
CRLF is inserted at the end of each record after
trailing blanks are stripped.
Note: The RDW is not written.

RECORD   DSECT=DCBREC   Type=A   Default=0 (undefined)
  Can this be set in the DCB prior to OPEN: Yes
  Can this be set in the DCB after OPEN   : Yes

  Supplies the default I/O area which can be overridden on the
  GET/PUT/READ/WRITE Macros.
  If omitted, then the I/O area must be specified on the
  GET/PUT/READ/WRITE Macros.

LRECL   DSECT=DCBLRECLF Type=F   Default=0
  Can this be set in the DCB prior to OPEN: Yes
  Can this be set in the DCB after OPEN   : Yes

  The field DCBLRECL (Type=H) is retained for compatibility.
  Maximum value is 2G-1.

  For RECFM=F or FB, sets the record size.
  For all other RECFM, sets the maximum record size.

  For RECFM=F, LRECL=0 is valid, provided that BLKSIZE is set.
  For RECFM=FB, LRECL must be a multiple of BLKSIZE.
  For variable records, include 4 for the RDW.

BLKSIZE   DSECT=DCBBLKSIF Type=F   Default=0
  Can this be set in the DCB prior to OPEN: Yes
  Can this be set in the DCB after OPEN   : Yes

  The field DCBBLKSI (Type=H) is retained for compatability.
  Maximum value is 2G-1.

  LRECL and BLKSIZE cannot both be zero.
  For RECFM=F sets the record size if LRECL=0.
  For RECFM=FB, sets the block size and LRECL must be a

multiple of BLKSIZE.
For RECFM=VB, BLKSIZE must be at least 4 greater than LRECL.
For all other RECFM, sets the maximum block size.

DCBE   DSECT=DCBDCBE Type=A   Default=0 (undefined)
   Can this be set in the DCB prior to OPEN: Yes
   Can this be set in the DCB after OPEN   : Yes

   The DCBE is a control block defined by the DCBE macro.
   When defined, the addresses of the end-of-file (EODAD) and the I/O
   error routine (SYNAD) may be defined.

   When set, these addresses override the DCB EODAD and SYNAD parameters.

   The DSECT IHADCBE maps the DCBE control block.

   The DCBE macro is DCBE EODAD=,SYNAD= with both parameters defaulting
   to 0 (undefined).

EODAD   DSECT=DCBEODAD Type=A   Default=0 (undefined)
   Can this be set in the DCB prior to OPEN: Yes
   Can this be set in the DCB after OPEN   : Yes

   The address of the end-of-file routine.
   This may be overidden if DCBE is coded.

   If a further GET is done after end-of-file, then the program
   is terminated.

SYNAD   DSECT=DCBSYNAD Type=A   Default=0 (undefined)
   Can this be set in the DCB prior to OPEN: Yes
   Can this be set in the DCB after OPEN   : Yes

   The address of the uncorrectable I/O error routine.
   This may be overridden if DCBE is coded.

DTFSD

VSE only.
DTFSD maps to the standard DCB as follows:

```
label   DTFSD BLKSIZE=n,EOFADDR=

label   DCB   DSORG=PS,
              EODAD=,    Maps from EOFADDR
              RECFM=F,
              DDNAME=,   label if it exists, otherwise blanks
              MACRF=RW,
              SYNAD=0,
              BLKSIZE=,  Maps from BLKSIZE
              LRECL=,    Maps from BLKSIZE
              DCBE=0,
              RECORD=0,
              DSNAME=0
```

DTFPR

VSE only.
DTFPR maps to the standard DCB as follows:

```
label   DTFPR BLKSIZE=n

label   DCB   DSORG=PS,
              EODAD=0,
              RECFM=F,
              DDNAME=,   label if it exists, otherwise blanks
              MACRF=RW,
              SYNAD=0,
              BLKSIZE=,  Maps from BLKSIZE
              LRECL=,    Maps from BLKSIZE
              DCBE=0,
              RECORD=0,
              DSNAME=0
```

I/O Macros syntax

General syntax definitions:
   dcbname   the label on the DCB macro.
   reg   general register, avoid 0, 1, 14 or 15.
   type   INPUT, OUTPUT or UPDATE (default is INPUT).
   ioarea   label of the I/O area for the record to be read/written.
   decbname   internal or external DECB for controlling READ/WRITE.
   rel   relative record number.
   rba   relative byte address.

Continuation characters have been omitted.

OPEN

Open one or more files.

Syntax:
Open one file for INPUT
  OPEN dcbname,MF=E

Open one file for non-INPUT
  OPEN (dcbname,type),MF=E

Open multiple files
  OPEN (dcbname1,type,dcbname2,type),MF=E

The same, using register notation
  OPEN ((reg),type),MF=E
  OPEN ((reg1),type,(reg2),type),MF=E

The List form may be used with MF=L to generate a parameter list.
The last entry has the top bit on.
Parameters of the form (regn) are not permitted with MF=L.

Register Usage:
  R0 = Flags
  R1 = DCB addresses

Abends:
  S013 OPEN failed and no SYNAD exit provided

OPEN (VSE)

VSE only.
Open one or more files for UPDATE.

Syntax:
    OPEN dtfname
    OPEN (dtfname1,dtfname2,...)

Register Usage:
    R0 = Flags
    R1 = DTF addresses

Abends:
    S013 OPEN failed and no SYNAD exit provided

CLOSE

Close one or more files.

Syntax:
Close one file
    CLOSE dcbname,MF=E

Close multiple files
    CLOSE (dcbname1,,dcbname2),MF=E

The same, using register notation
    CLOSE ((reg1)),MF=E
    CLOSE ((reg1),,(reg2)),MF=E

The List form may be used with MF=L to generate a parameter list.
The last entry has the top bit on.
Parameters of the form (regn) are not permitted with MF=L.

Register Usage:
    R1 = DCB addresses

Abends:
    S013 CLOSE failed and no SYNAD exit provided

CLOSE (VSE)

VSE only.
Close one or more files.

Syntax:
  CLOSE (dtfname1,dtfname2,...)

Register Usage:
  R1 = DTF addresses

Abends:
  S013 CLOSE failed and no SYNAD exit provided

GET

Read a sequential record.

Syntax:
Read a record into an I/O area supplied on DCB RECORD=ioarea
  GET dcbname

Read a record into an I/O area
  GET dcbname,ioarea

The same, using register notation
  GET (reg)
  GET (reg1),(reg2)

Register Usage:
  R0 = ioarea
  R1 = DCB address

PUT

Write a sequential record.

Syntax:
Write a record from an I/O area supplied on the DCB RECORD=ioarea
  PUT dcbname

Write a record from an I/O area
  PUT dcbname,ioarea

The same, using register notation
  PUT (reg)
  PUT (reg1),(reg2)

Register Usage:
  R0 = ioarea
  R1 = DCB address

READ

Read a block from a file.

Notes:
  1) If EODAD or SYNAD result from the READ, they will only be
     processed by the CHECK macro.

  2) If the decbname parameter is specified in register notation,
     no internal DECB is generated. It is the programmer's
     responsibilty to create and address the 16-byte DECB.

Syntax:
Read a block using an internal DECB, into an I/O area
supplied on the DCB RECORD=ioarea
  READ decbname,,dcbname

Read a block using an internal DECB into an I/O area.
The I/O area must be at least as large as DCB BLKSIZE
  READ decbname,,dcbname,ioarea

The same, using register notation
    READ (reg1),,(reg2)
    READ (reg1),,(reg2),(reg3)

Register Usage:
    R0 = DCB and ioarea addresses...saved
    R1 = DECB address
    R15= DCB address

WRITE

Write a block to a file.

Notes:
    1) If SYNAD results from the WRITE, it will only be processed
       by the CHECK macro.

    2) If the decbname parameter is specified in register notation,
       no internal DECB is generated. It is the programmer's
       responsibility to create and address the 16-byte DECB.

Syntax:
Write a block using an internal DECB, from an I/O area
supplied on the DCB RECORD=ioarea
    WRITE decbname,,dcbname

Write a block using an internal DECB from an I/O area
    WRITE decbname,,dcbname,ioarea

The same, using register notation
    WRITE (reg1),,(reg2)
    WRITE (reg1),,(reg2),(reg3)

Register Usage:
    R0 = DCB and ioarea addresses...saved
    R1 = DECB address
    R15= DCB address

CHECK

Process EODAD or SYNAD on READ.
Process SYNAD on WRITE.

Syntax:
  CHECK decbname

The same, using register notation
  CHECK (reg)

Register Usage:
  R1 = DECB address

Abends:
  S013 READ/WRITE failed and no SYNAD exit provided

POINT

Position pointer for next READ or WRITE

Notes:
  1) When register notation is used for rba or rel,
     the register points to a field containing the value.

  2) rel is a fullword, maximum value 2,147,483,647 (2G)
     rel is multiplied by BLKSIZE to get the rba.

  3) rba is signed 64-bit, maximum value...very big.

Syntax:
Point to a record using relative record number
  POINT dcbname,rel

Point to a record using relative byte address
  POINT dcbname,,RBA=rba

The same, using register notation
  POINT (reg1),(reg2)
  POINT (reg1),,RBA=(reg2)

Register Usage:
    R0 = rel or RBA
    R1 = DCB address
    R15= Blocksize

Change Summary

June 10, 2011
    CLOSE ((reg1)) format added
    CLOSE MF=E/L documented
    OPEN MF=E/L documented

November 24, 2008
    EXLST unsupported comment

January 18, 2008
    Correction to description of CRLF.
    Removed references to OPEN (reg) and CLOSE ((reg))
    Added abend sections.

September 28, 2007
    Added VSE macros OPEN, CLOSE, DTFSD, DTFPR
    Added mention of GETENV to extract environment variables.
    Note about macros which are also used by VSAM.

July 10, 2007
    All macros now list possible general register usage.
    BLKSIZE and LRECL can be fullword or halfword.
    DCBE parameter and macro, and DSECT IHADCBE documented.

Trademarks

IBM, CICS, VSAM and VSE are registered trademarks of International Business Machines Corporation.

Credits

Author: Melvyn Maltz
Shipping date: June 10, 2011
z390 version : V1.5.04
zCICS version: V10