



ERXVSAM

VSAM Processing Support with REXX

Version 1.0

August 17, 2011



Table of Contents

I	ERXVSAM – VSAM Processing Support with REXX	3
I.1	Overview.....	3
I.1.1	Licence.....	3
I.2	Invocation.....	3
I.2.1	Syntax.....	3
I.2.2	Return Codes.....	4
I.2.3	REXX Control Variable.....	4
I.2.4	REXX Result Variable	4
I.3	Examples.....	4
I.3.1	Example 1 (with preallocated file (DDVSAM DDname).....	4
I.3.2	Example 2 (with explicit dataset name)	4
I.3.3	Example 3 (delete two records).....	4
I.3.4	Batch Implementations.....	5
I.3.4.1	Sample implementation as JCL (via IRXJCL).....	5
I.3.4.2	Sample implementation with Batch TSO (via IKJEFT01, batch TSO)	5
I.3.4.3	RXVSAM member of DD:SYSEXEC.....	5



1 ERXVSAM – VSAM Processing Support with REXX

1.1 Overview

The ERXVSAM function provides REXX support for basic VSAM processing. It provides read/write/delete record set processing for a VSAM (ESDS, KSDS) dataset. Depending on the operation, the stack or REXX stem variables (STEM keyword) serve as input or output records.

1.1.1 Licence

ERXVSAM is a licensed service; it is available for a 14-day free trial use.

1.2 Invocation

1.2.1 Syntax

```

frc = ERXVSAM (
    READ | WRITE | DELETE,
    dsname | DD:ddname,
    KEY_FIRST | KEY_LAST | KEY_EQ | KEY_GE | RBA_EQ,
    key | rba (not for KEY_FIRST | KEY_LAST),
    * | recordcount,
    [STEM stem] )
    
```

<p>dsname DD:ddname</p>	<p>The identification of the file to be processed. If <i>ddname</i> is specified, the associated DDname must be preallocated.</p>
<p>READ WRITE DELETE</p>	<p>The command to be performed</p>
<p>KEY_FIRST KEY_LAST KEY_EQ KEY_GE RBA_EQ</p>	<p>The initial positioning for the processing to be performed</p>
<p>key rba</p>	<p>The key or relative byte address for the initial positioning. No entry is required (nor permitted) for KEY_FIRST relative KEY_LAST.</p>
<p>* recordcount</p>	<p>The number of records to be processed. * = all records (must be entered as '*' for direct value).</p>
<p>STEM stem</p>	<p>The name of the stem variable where the associated records are located. <i>stem.0</i> contains the number of records. The stack will be used if this keyword is omitted.</p>



1.2.2 Return Codes

ERXVSAM sets the return value (<frc> above) to indicate the processing results:

0	OK
8	Record not found
12	Open error
16	Locate error (key/RBA not found)
20	Read/write error
24	Argument length error
28	Invalid argument
32	Argument missing

1.2.3 REXX Control Variable

The following variable controls the ERXVSAM processing

_TRACE Enable program tracing (input); 1 = enable, 0 = disable (default).

Note:

_TRACE is used only for debugging purposes.

1.2.4 REXX Result Variable

The following REXX variable will be set if a processing error occurs

_RPLERRCD VSAM feedback (set for VSAM error) as hexadecimal value.

1.3 Examples

1.3.1 Example 1 (with preallocated file (DDVSAM DDname))

```
PUSH 'MYKEY line'
frc = ERXVSAM('WRITE', 'DD:DDVSAM', 'KEY_GE', 'MYKEY', '*')
IF frc = 0 THEN SAY 'RC:'frc 'ERRCD: '_RPLERRCD
```

1.3.2 Example 2 (with explicit dataset name)

```
frc = ERXVSAM('READ', 'SOFAR.TEST.ESDS', 'RBA_EQ', 0, '*', 'STEM', 'R.')
IF frc = 0 THEN DO
  DO i = 1 TO R.0
    SAY R.i
  END
END
ELSE SAY 'RC:'frc 'ERRCD: '_RPLERRCD
```

1.3.3 Example 3 (delete two records)

```
frc = ERXVSAM('DELETE', 'SOFAR.TEST.KSDS', 'KEY_FIRST', 2)
IF frc <> 0 THEN DO
  SAY 'RC:'frc 'ERRCD: '_RPLERRCD
END
```



1.3.4 Batch Implementations

1.3.4.1 Sample implementation as JCL (via IRXJCL)

```
//XJCL      EXEC PGM=IRXJCL, PARM='RXVSAM'
//STEPLIB  DD DSN=SO FAR.LOAD, DISP=SHR
//SYSEXEC  DD DSN=SO FAR.EXEC, DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//DDVSAM   DD DSN=SO FAR.TEST.KSDS, DISP=SHR
```

1.3.4.2 Sample implementation with Batch TSO (via IKJEFT01, batch TSO)

```
//BT SO      EXEC PGM=IKJEFT01, PARM='RXVSAM'
//STEPLIB  DD DSN=SO FAR.LOAD, DISP=SHR
//SYSEXEC  DD DSN=SO FAR.EXEC, DISP=SHR
//SYSUADS  DD DSN=SYS1.UADS, DISP=SHR
//SYSLBC   DD DSN=SYS1.BROADCAST, DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD DUMMY
//SYSPRINT DD SYSOUT=*
//DDVSAM   DD DSN=SO FAR.TEST.KSDS, DISP=SHR
```

Note:

- The STEPLIB must be changed to specify the library that contains the ERXVSAM load module.
- The SYSPROC must be changed to specify the library that contains the REXX exec to be performed.

1.3.4.3 RXVSAM member of DD:SYSEXEC

```
/* REXX */
cmd = 'READ'
key = 'KEY2'
frc = ERXVSAM(cmd, 'DD:DDVSAM', 'KEY_GE', key, '*')
IF frc = 0 THEN DO
  DO WHILE QUEUED() > 0
    PARSE PULL rec
    SAY rec
  END
END
ELSE SAY cmd frc
```