REXX/VSE Enhancements since 1997

The latest hardcopy version of the REXX/VSE Reference Manual SC33-6642-01 is dated December, 1996. Further enhancements are included only into the softcopy version of this manual. Here is a summary of recent REXX/VSE extensions, that are not contained in the hardcopy version of this manual:

- **ADDRESS POWER GETQE: new operand FCB to retrieve the name of the FCB-image phase**
- **ADDRESS POWER GETQE: new operand ASATOMCC to convert ASA print control characters into MCC**
- **ADDRESS POWER GETQE: new operand OPTB to receive settings of user-defined output operands**
- **ADDRESS POWER GETQE: new operand UCS to retrieve UCB information**
- **ADDRESS POWER PUTQE: new operand LONGREC to specify a record length of 128 bytes for POWER RDR queue entries**
- **ADDRESS POWER PUTQE: new operand JSEP to specify the number of job seperator pages**
- **ADDRESS POWER PUTQE: new operand FCB to specify name of the FCB-image phase**
- **ADDRESS POWER PUTQE: new operands NEWSEGM, JOBSUF to create segmented POWER output queue entries**
- **ADDRESS POWER PUTQE: new operands FLASH_NAME, FLASH_COUNT to support IBM 3800 printers**
- **ADDRESS POWER PUTQE: new operand SYSID to support system specification in a shared spooling environemnt**
- **ADDRESS POWER PUTQE: new operand FNO to specify the formname**
- **ADDRESS POWER PUTQE: new operand UCS to specify name of the UCB-image phase and UCB-options**
- **ADDRESS POWER PRELEASE: new option (NOGENCM to suppress generation of job completion messages**
- **SYSIPT record length extended from 80 bytes to 128 bytes**
- **Function FINDMSG: Reimplemented to reduce the amount of CPU time needed for its execution**
- **Function SENDMSG:**
  - **new operand HIGH to create messages to be displayed in highlighted form**
  - **allow keyword ALL as console name to send a message to all defined master consoles**
- **New function SORTSTEM to sort a stem variable**
- **New function PAUSEMSG to create a console message and wait for its reply**
- **Function DATE: new operands to specify separator characters for input and/or output format**
- **Function SYSVAR: new argument SYSCPUID**
- **Command EXECIO: new operands BYTES and STRTBYTE to handle bytestream-like library members**
- **Command EXECIO: operand RECSIZE also possible for EXECIO DISKR SYSIPT**
- **Command EXECIO: operand RECSIZE also possible for EXECIO DISKR SYSLST together with new operands ASA and MCC**
- **Command EXECIO: new operand SYSPCH to write to SYSPCH**
- **Command EXECIO: new operand UNDEF for SAM files**
- **New command VSAMIO to handle VSAM files**
- **New function SOCKET to allow communication via a network based on TCP/IP**
- **New SSL-support for the SOCKET function**
- **New field in ARXEOJTB table entries to consider single phase only during space checking of phases to be LOADed**
- **Stemnames with more than one dot**

The enhancements are available with the following system levels / PTFs:

For VSE/ESA 2.3:

Enhancement	VSE/ESA 2.3.x	2.3 PTF	2.3 APAR
GETQE FCB	2.3.1	UQ18313	PQ16345
GETQE ASATOMCC	2.3.2	UQ38549	PQ33431
GETQE OPTB		UQ56784	PQ51117
GETQE UCS	-	-	-

PUTQE LONGREC	2.3.2	UQ24696	PQ22015
PUTQE JSEP	2.3.0		
PUTQE FCB	2.3.1	UQ13366	PQ11893
PUTQE NEWSEGM, JOBSUF	2.3.2	UQ24696	PQ22015
PUTQE FLASH_NAME, FLASH_COUNT	2.3.2	UQ38549	PQ33431
PUTQE SYSID		UQ41250	PQ36310
PUTQE FNO		UQ56784	PQ51117
PUTQE UCS	-	-	-
PRELEASE (NOGENCM	-	-	-
SYSIPT extension to 128 bytes	2.3.1	UQ18313	PQ16345
FINDMSG reimplentation	2.3.2	UQ24696	PQ22015
SENDMSG HIGH	2.3.1	UQ18313	PQ16345
SENDMSG ALL		UQ53048	PQ47519
SORTSTEM	2.3.1	UQ18313	PQ16345
SORTSTEM RANGE		UQ53048	PQ47519
PAUSEMSG		UQ45008	PQ39736
DATE separator	2.3.2	UQ24696	PQ22015
SYSVAR SYSCPUID	-	-	-
EXECIO string	2.3.0		
EXECIO DISKR SYSIPT RECSIZE	2.3.2	UQ27680	PQ24952
EXECIO DISKR SYSLST RECSIZE	-	-	-
EXECIO SYSPCH	-	-	-
EXECIO UNDEF	-	-	-
VSAMIO	-	-	-
SOCKET	2.3.2	UQ37224	PQ31258
SOCKET: GETCLIENTID, GIVESOCKET, TAKESOCKET		UQ56784	PQ51117
SOCKET: TRANSLATE		UQ53048	PQ47519
new ARXEOJTB field	-	-	-

For VSE/ESA 2.4:

Enhancement	VSE/ESA 2.4.x	2.4 PTF	2.4 APAR
GETQE FCB	2.4.0		
GETQE ASATOMCC	2.4.2	UQ38550	PQ33431
GETQE OPTB		UQ56785	PQ51117
GETQE UCS	.	UQ61688	PQ56280
PUTQE LONGREC	2.4.0	UQ24697	PQ22016
PUTQE JSEP	2.4.0		
PUTQE FCB	2.4.0		
PUTQE NEWSEGM, JOBSUF	2.4.0	UQ24697	PQ22016
PUTQE FLASH_NAME, FLASH_COUNT	2.4.2	UQ38550	PQ33431
PUTQE SYSID	2.4.2	UQ41251	PQ36310
PUTQE FNO		UQ56785	PQ51117
PUTQE UCS	.	UQ61688	PQ56280
PRELEASE (NOGENCM	.	UQ61688	PQ56280
SYSIPT extension to 128 bytes	2.4.0	.	.
FINDMSG reimplentation	2.4.0	.	.
SENDMSG HIGH	2.4.0	.	.

SENDMSG ALL		UQ53049	PQ47519
SORTSTEM	2.4.0	.	.
SORTSTEM RANGE		UQ53049	PQ47519
PAUSEMSG		UQ45018	PQ39746
DATE separator	2.4.0	UQ24697	PQ22016
SYSVAR SYSCPUID	.	UQ61688	PQ56280
EXECIO string	2.4.0		
EXECIO DISKR SYSIPT RECSIZE	2.4.0	UQ27683	PQ24982
EXECIO DISKR SYSLST RECSIZE	-	-	-
EXECIO SYSPCH	-	-	-
EXECIO UNDEF	-	-	-
VSAMIO	-	-	-
SOCKET	2.4.2	UQ37225	PQ31258
SOCKET: GETCLIENTID, GIVESOCKET, TAKESOCKET		UQ44970	PQ39623
SOCKET: TRANSLATE		UQ53049	PQ47519
SOCKET - SSL-support	-	-	-
new ARXEOJTB field	.	UQ61688	PQ56280
Stemname with multiple dots	-	-	-

For VSE/ESA 2.5:

Enhancement	VSE/ESA 2.5.x	2.5 PTF	2.5 APAR
GETQE FCB	2.5.0		
GETQE ASATOMCC	2.5.0	.	.
GETQE OPTB		UQ56786	PQ51117
GETQE UCS	.	UQ61686	PQ56301
PUTQE LONGREC	2.5.0	.	.
PUTQE JSEP	2.5.0		
PUTQE FCB	2.5.0		
PUTQE NEWSEGM, JOBSUF	2.5.0	.	.
PUTQE FLASH_NAME, FLASH_COUNT	2.5.0	.	.
PUTQE SYSID	2.5,0	.	.
PUTQE FNO		UQ56786	PQ51117
PUTQE UCS	.	UQ61686	PQ56301
PRELEASE (NOGENCM	.	UQ61686	PQ56301
SYSIPT extension to 128 bytes	2.5.0	.	.
FINDMSG reimplentation	2.5.0	.	.
SENDMSG HIGH	2.5.0	.	.
SENDMSG ALL	2.5.2	UQ53055	PQ47519
SORTSTEM	2.5.0	.	.
SORTSTEM RANGE	2.5.2	UQ53055	PQ47519
PAUSEMSG	2.5.1	UQ47043	PQ41345
DATE separator	2.5.0	.	.
SYSVAR SYSCPUID	.	UQ61686	PQ56301
EXECIO string	2.5.0		
EXECIO DISKR SYSIPT RECSIZE	2.5.0	.	.
EXECIO DISKR SYSLST RECSIZE	-	-	-
EXECIO SYSPCH	-	-	-

EXECIO UNDEF	-	-	-
VSAMIO	-	-	-
SOCKET	2.5.0	.	.
SOCKET: GETCLIENTID, GIVESOCKET, TAKESOCKET	2.5.1	UQ47043	PQ41345
SOCKET: TRANSLATE	2.5.2	UQ53055	PQ47519
SOCKET - SSL-support	-	-	-
new ARXEOJTB field	.	UQ61686	PQ56301
Stemname with multiple dots	-	-	-

For VSE/ESA 2.6:

Enhancement	VSE/ESA 2.6.x	2.6 PTF	2.6 APAR
GETQE FCB	2.6.0		
GETQE ASATOMCC	2.6.0	.	.
GETQE OPTB	2.6.0		
GETQE UCS	2.6.1	UQ61687	PQ56301
PUTQE LONGREC	2.6.0	.	.
PUTQE JSEP	2.6.0		
PUTQE FCB	2.6.0		
PUTQE NEWSEGM, JOBSUF	2.6.0	.	.
PUTQE FLASH_NAME, FLASH_COUNT	2.6.0	.	.
PUTQE SYSID	2.6,0	.	.
PUTQE FNO	2.6.0		
PUTQE UCS	2.6.1	UQ61687	PQ56301
PRELEASE (NOGENCM	2.6.1.	UQ61687	PQ56301
SYSIPT extension to 128 bytes	2.6.0	.	.
FINDMSG reimplentation	2.6.0	.	.
SENDMSG HIGH	2.6.0	.	.
SENDMSG ALL	2.6.0		
SORTSTEM	2.6.0	.	.
SORTSTEM RANGE	2.6.0		
PAUSEMSG	2.6.0		
DATE separator	2.6.0	.	.
SYSVAR SYSCPUID	2.6.1	UQ61687	PQ56301
EXECIO string	2.6.0		
EXECIO DISKR SYSIPT RECSIZE	2.6.0	.	.
EXECIO DISKR SYSLST RECSIZE	-	-	-
EXECIO SYSPCH	-	-	-
EXECIO UNDEF	-	-	-
VSAMIO	2.6.0		
SOCKET	2.6.0	.	.
SOCKET: GETCLIENTID, GIVESOCKET, TAKESOCKET	2.6.0		
SOCKET: TRANSLATE	2.6.0		
SOCKET - SSL-support	-	-	-
new ARXEOJTB field	2.6.1	UQ61687	PQ56301
Stemname with multiple dots	-	-	-

For VSE/ESA 2.7:

Enhancement	VSE/ESA 2.7.x	2.7 PTF	2.7 APAR
GETQE FCB	2.7.0		
GETQE ASATOMCC	2.7.0	.	.
GETQE OPTB	2.7.0		
GETQE UCS	2.7.0		
PUTQE LONGREC	2.7.0	.	.
PUTQE JSEP	2.7.0		
PUTQE FCB	2.7.0		
PUTQE NEWSEGM, JOBSUF	2.7.0	.	.
PUTQE FLASH_NAME, FLASH_COUNT	2.7.0	.	.
PUTQE SYSID	2.7,0	.	.
PUTQE FNO	2.7.0		
PUTQE UCS	2.7.0		
PRELEASE (NOGENCM	2.7.0		
SYSIPT extension to 128 bytes	2.7.0	.	.
FINDMSG reimplentation	2.7.0	.	.
SENDMSG HIGH	2.7.0	.	.
SENDMSG ALL	2.7.0		
SORTSTEM	2.7.0	.	.
SORTSTEM RANGE	2.7.0		
PAUSEMSG	2.7.0		
DATE separator	2.7.0	.	.
SYSVAR SYSCPUID	2.7.0		
EXECIO string	2.7.0		
EXECIO DISKR SYSIPT RECSIZE	2.7.0	.	.
EXECIO DISKR SYSLST RECSIZE	2.7.0		
EXECIO SYSPCH	2.7.0		
EXECIO UNDEF	2.7.0		
VSAMIO	2.7.0		
SOCKET	2.7.0	.	.
SOCKET: GETCLIENTID, GIVESOCKET, TAKESOCKET	2.7.0		
SOCKET: TRANSLATE	2.7.0		
SOCKET - SSL-support	2.7.0		
new ARXEOJTB field	2.7.0		
Stemname with multiple dots	2.7.0		

Detailed descriptions of the enhancements are following:

ADDRESS POWER GETQE: new operand FCB

A new operand is available for command ADDRESS POWER GETQE LST:

```
FCB fcb_var
```

It is optional. "fcb_var" specifies a variable where you want to receive the name of the FCB-image phase VSE/POWER is to use for printing the related LST-output.

ADDRESS POWER GETQE: new operand ASATOMCC

A new operand is available for command ADDRESS POWER GETQE LST:

```
ASATOMCC
```

It is optional and only valid together with CTRLREC. Usually GETQE offers ASA-controlled data records unchanged. But you can request ASA to machine control conversion using keyword ASATOMCC. In this case you get for every ASA data record two machine control records: - a first one doing the forms control operation - a second one writing the actual data immediately

ADDRESS POWER GETQE: new operand OPTB

On ADDRESS POWER PUTQE, the operand "OPTB ostem" is enhanced:

> **It is also possible to specify the keyword id instead of the keyword for a user-defined output operand, for example OPTBSTEM.1 = '001F=HUGO'. In this case all following output operand specifications in "ostem" must use this keyword id description and not the keyword!**

New operand OPTB has been added to ADDRESS POWER GETQE:

> **OPTB ostem:**
> **"ostem" specifies the REXX stem to receive settings of user-defined output operands. The values are returned in the form**
> **KEYWORDID={value|(value,...)}**
> **in "ostem.n", for example**
> **OPTBSTEM.1 = '001F=ABC'**

ADDRESS POWER GETQE: new operand UCS

A new operand is available for command ADDRESS POWER GETQE LST:

```
UCS ucb_var
```

It is optional. "ucb_var" specifies a variable where you want to receive the UCB (universal character set buffer) information assigned to a LST queue entry. The information is returned in the format

(*phasenam*,*op*)

where *phasenam* is the name of the UCB-image phase loaded into the UCB of the printer, and *op* is either 'F ', 'C ', or 'FC'. An 'F' indicates that the UCB is to be loaded with the folding operation code causing lowercase letters to be printed in uppercase, and 'C' prevents data checks from being generated because of print-line mismatches with the UCB. If none of the two options are set, two blanks are set. If no UCB-image phase has been defined for a LST queue entry, consists of 8 blanks.

An additional return code -36 is possible, meaning "Operand UCS is invalid".

ADDRESS POWER PUTQE: new operand LONGREC

The operand LONGREC is optionally added to the RDR operands:

LONGREC indicates that the record length of a POWER RDR queue entry is 128 bytes. If not specified, the default length is 80 bytes.

ADDRESS POWER PUTQE: new operand JSEP

PUTQE offers a new PUN/LST Operand: JSEP jsep_pages is the number of desired job separator pages. "jsep_pages" is a number between 0 and 9. For more information about JSEP see manual "VSE/POWER Administration and Operation".

ADDRESS POWER PUTQE: new operand FCB

A new operand is defined for command ADDRESS POWER PUTQE:

```
         FCB phasename
```

It is optional.
For "phasename" specify the name of the FCB-image phase which VSE/POWER is to use for printing the related job output.
Syntax sample:

```
           'PUTQE LST JOBNAME PUTQTEST FCB $$BFCBWM'
```

ADDRESS POWER PUTQE: new operands NEWSEGM, JOBSUF

The operands JOBSUF and NEWSEGM are optionally added to the PUN/LST operands:

```
  |-FIRST-JOBNAME-nam-JOBNUM-numvar-----------------------------|
  |                                 |-NEWSEGM--svar-|           |
  |                                                             |
  |-NEXT-JOBNAME-nam-JOBNUM-num---------------------------------|
  |                             |-JOBSUF-jsuf-| |-NEWSEGM-svar-||
  |                                                             |
  |-LAST-JOBNAME-nam-JOBNUM-num---------------------------------|
                               |-JOBSUF-jsuf-|
```

JOBSUF "jsuf"

> **"jsuf" specifies the segment number of the POWER output queue entry where the given data is to be appended**

NEWSEGM "segmvar"

> **indicates that after append of the given data to the specified output queue entry a new segment is to be started. "segmvar" specifies a variable used to return the number of the old segment, where the given data has been appended.**

new return code:

-29 No data is specified together with NEWSEGM. This does not work.

Sample:

```
   ADDRESS POWER
     'PUTQE LST JOBNAME PUTQTEST JOBNUM jn FIRST' ,
     'CLASS Q STEM file1.'
     'PUTQE LST JOBNAME PUTQTEST JOBNUM' jn 'NEXT' ,
     'CLASS Q STEM  file2. NEWSEGM segment_var'
     segment_var = segment_var + 1
     /* start of new segment */
     'PUTQE LST JOBNAME PUTQTEST JOBNUM' jn ' LAST' ,
     'CLASS Q STEM file3. JOBSUF ' segment_var
```

ADDRESS POWER PUTQE: new operands FLASH_COUNT, FLASH_NAME

PUTQE offers new LST Operands: FLASH_COUNT fnum, FLASH_NAME fnam that apply to IBM 3800 only.

FLASH_COUNT "fnum"

> **"fnum" is a number from 0 to 255. It specifies the number of copies to be flashed with the overlay. If you specify a count without a FLASH_NAME, then the forms-overlay frame loaded at the time of printing is used.**
> **If you specify a count of 0, then the operator is prompted to load the requested forms-overlay frame, but the overlay is not flashed.**

FLASH_NAME "fnam"

"fnam" is the one-to-four-character name of the forms-overlay frame bto be used by the printer. If you specify an overlay name without a count, all copies are flashed.

ADDRESS POWER PUTQE: new operand SYSID

PUTQE offers a new operand:

SYSID "n"

This operand applies to shared spooling. "n" is either the character N or a number between 1 and 9.

Specify "SYSID N" if the output is to be available on any of the sharing systems. Specify a digit between 1 and 9, if your output is to be available on a certain one of your sharing systems. For "n", give the number with which the systems VSE/POWER was initialized (by SYSID=n in the VSE/POWER generation macro).

ADDRESS POWER PUTQE: new operand FNO

New operand FNO has been added to ADDRESS POWER PUTQE LST:
FNO formnam

specifies the formname. "formnam" consists of one to four alphameric characters.

new return code:

-35 Operand FNO is invalid

ADDRESS POWER PUTQE: new operand UCS

A new operand is available for command ADDRESS POWER PUTQE LST:

```
   -UCS-+-phasenam------------+-.
        |                     |
        .-(phasenam-+-----+-)-.
                    +-,F--+
                    +-,C--+
                    .-,FC-.
```

It is optional. "phasenam" specifies the name of the UCB-image phase loaded into the UCB (universal character set buffer) of the printer. The following options are available:

F

to indicate that the UCB is to be loaded with the folding operation code causing lowercase letters to be printed in uppercase.

C

to prevent data checks from being generated because of print-line mismatches with the UCB.

An additional return code -36 is possible, meaning "Operand UCS is invalid".

ADDRESS POWER PRELEASE RDR without job completion messages

If you are not interested in POWER-generated job completion messages, add string (NOGENCM to your PRELEASE RDR command (for example: ADDRESS POWER "R RDR,MYJOB (NOGENCM").

Reimplementation of function FINDMSG

The implementation of FINDMSG is changed to reduce the amount of CPU time needed for its execution. The message action table is no longer loaded into the REXX stack; instead it is loaded in another part of main storage. Usually existing REXX procedures using FINDMSG still run with this new version of FINDMSG. Only the following aspects have to be considered, when using the new version of FINDMSG:

- **REXX stack is no longer used; thus resetting the stack using the DELSTACK command after FINDMSG processing has finished is no longer necessary.**
- **Some of the old Error Codes of FINDMSG are no longer used, and new Error Codes (variable RC) have been added:**

16	**unused**
24	**unused**
29	**unused**
36	**Invalid TIME**
44	**unused**
48	**message action table not loaded**
52	**Problem loading library member**
56	**Library member not found**
60	**Storage problem**
64	**Invalid OPTION**
68	**Format error within message action table**

In addition, FINDMSG now sets more special REXX variables; it now covers the same MDB Variables as function GETMSG.

Function SENDMSG: new operand 'HIGH'

SENDMSG offers a 4th operand "type" and a new keyword "ALL" as second operand :

```
 ___________________________________________________________
.                                                           .
· >>--SENDMSG(message,consname,cart,type)--------------><  ·
.___________________________________________________________.
```

consname

VSE console name to be used as destination for the message.
The default name is 'REXX'.
Specify 'ALL' to send the message to all master consoles active in your VSE/ESA system.

type

If HIGH, the message is displayed in highlighted form.
If omitted, the message is displayed in non-highlighted form.

Example of Function Call:
Sending a highlighted message to all active master consoles is done in the following example:

```
fc = SENDMSG('Hello','ALL',,'HIGH')
```

New function SORTSTEM

This function sorts a stem variable. Sort criteria is the standard "<"-function applied to EBCDIC-strings. SORTSTEM returns function code 0 that replaces the function call.

SORTSTEM has the following format:

```
 __________________________________________________________
|                                                          |
| >>--SORTSTEM(stemname,zone,sortorder,range)----------><  |
|__________________________________________________________|
```

stemname

is the stem to be sorted. For example, if 'svar.' is specified as stemname, variable svar.0 has to contain the number n of strings to be sorted, and svar.1, svar.2, ... svar.n have to contain the strings to be sorted.

zone

specifies, after the keyword ZONE, a beginning and an ending position within the strings to be sorted identifying the sort criteria. Due to the implemented sort algorithm ("heapsort") the order of "equal" strings may be changed.
If not specified, the entire strings are used during comparison. If string lengths do not match, the shorter string is padded with blanks.

sortorder

defines the order of sorting
ASCENDING means sorting in ascending order.
This is the default.
DESCENDING means sorting in descending order.

range

specifies, after the keyword RANGE, a beginning and an ending index identifying the part of the stem to be sorted.

Examples of Function Calls:

```
  fc = SORTSTEM(input.)
  fc = SORTSTEM(svar.,'ZONE 9 14','DESCENDING','RANGE 11 24')
```

Error Codes (variable RC):

04 Invalid number of arguments
08 Invalid ZONE
12 Invalid STEM variable
16 Storage problem
20 Stem handling problem
24 Invalid STEM.0 setting
28 Invalid sort order
32 Invalid RANGE

New function PAUSEMSG

This function issues a console message and waits for an operator reply. If successful, PAUSEMSG returns the operator reply, otherwise a null string.

PAUSEMSG has the following format:

```
 _________________________________________________________
.                                                         .
. >>--PAUSEMSG(message)--------------------------------><  .
._________________________________________________________.
```

message

Any string of 1 to 122 bytes. A string longer than that will be truncated to 122 bytes.

Examples of Function Calls:

```
  reply = PAUSEMSG(message)
  Say PAUSEMSG('Please enter your name')
```

Error Codes (variable RC):

04 Invalid number of arguments
08 WTOR error
12 nullstring as argument

DATE Separator

Function Date has been enhanced by a fourth and fifth operand specifying separator characters for input and output dates. Any single nonalphanumeric character is valid. Input separator characters and output separator characters apply to the following formats and have the following default values:

Format Name	Format Structure	Default Separator Value
European	dd/mm/yy	'/'
Normal	dd mon yyyy	' '

Ordered	yy/mm/dd	'/'
Standard	yyyymmdd	''
Usa	mm/dd/yy	'/'

Note: Null is a valid value for input_separator_char and output_separator_char.

Here are some examples, assuming today is 13 March 1992:

```
  DATE('E')                            ->    '13/03/92'
  DATE('E',,,'+')                      ->    '13+03+92'
  DATE('E','081698','U',,'')           ->    '16/08/98'
  DATE('S')                            ->    '19920313'
  DATE('S',,)                          ->    '19920313'
  DATE('S',,,'-')                      ->    '1992-03-13'
  DATE('U')                            ->    '03/13/92'
  DATE('U','1.Feb.1998,'N','+','.')    ->    '02+01+98'
  DATE('U','1998-08-16','S','','-')    ->    '081698'
```

Function SYSVAR: new argument SYSCPUID

SYSVAR('SYSCPUID') stores the CPUID of your VSE system in the variable SYSCPUID.

Command EXECIO: new operands BYTES and STRTBYTE

A description of the EXECIO bytestream processing capabilites for library members is contained in the manual VSE/ESA V2R3 Enhancements SC33-6629-01, and in later versions of the REXX/VSE Reference manual and the REXX/VSE User's Guide.

SYSIPT processing: new record length

The record length for SYSIPT data processed with REXX is extended from 80 bytes to 128 bytes. Thus, REXX function REXXIPT now pads or truncates to 128 bytes within a record, and BLKSIZE and RECSIZE values of EXECIO DISKR SYSIPT are now 128.

Command EXECIO DISKR SYSIPT: new operand RECSIZE

If a different record length than the default 128 bytes is desired when reading records from SYSIPT, specify RECLEN n with 0 < n <= 128.

Sample: 'EXECIO * DISKR SYSIPT (stem record. RECSIZE 80'

Command EXECIO DISKW SYSLST: greater record length

It is possible to write SYSLST records with a record length greater than 121.

```
|   |-SYSLST--+----------------------------------------------------+-----------------|
    |         |   (1) +-NOCC-+                                     |                 |
    |         +-(-----+------+--+----------------+--+-----------+--+                 |
    |                 +-CC---+  +-STEM--var_name-+  +-RECSIZE-n-+                    |
    |                 +-MCC--+                                                       |
    |                 +-ASA--+                                                       |
    |                                                                                |
```

For SYSLST with ASA or MCC option specified, a record greater than 256 bytes is truncated with a return code rc=1.

For SYSLST with CC option specified, a record greater than 121 bytes is truncated with a return code rc=1.

For SYSLST without option CC specified, a record greater than 120 bytes is truncated with a return code rc=1.

If you want to use a different record length for SYSLST, you can specify operand RECSIZE n with 0<N<=256

EXECIO SYSPCH

In addition to SYSLST, EXECIO supports writing to SYSPCH:

```
 |-SYSPCH--+------------------------------------+---------------------------------|
|         |   (1) +-NOCC-+                     |                                 |
|         +-(-----+------+--+----------------+-+                                 |
                  +-CC---+  +-STEM--var_name-+
```

SYSPCH records are written with BLKSIZE 81, RECSIZE 81, and RECFORM FIXUNB.

EXECIO ... (... RECFORM UNDEF ...

EXECIO supports undefined record format for SAM files on disk with operand RECFORM UNDEF.

New command VSAMIO

VSAMIO controls the input and output (I/O) of information to and from a VSAM file. More information is found in chapter 10 of the REXX/VSE Reference Manual.

New function SOCKET

The famous SOCKET Application Program Interface for inter-process peer-to-peer communication between applications, either on the local system or spread in a distributed, TCP/IP based network environment, is available for the REXX/VSE language. It is a general-use, wide spread and very flexible programming interface having its roots in the original Berkeley Software Distribution (BSD) socket implementation for UNIX. The Socket API allowing to receive data from a network and send data to a network is based on an open/close/read/write paradigm similar to reading from and writing to any other device. The REXX/VSE implementation provides a SOCKET function with the same syntax and semantic as the REXX SOCKET functions available on the VM and OS/390 platform. It maps the socket calls from the C programming language to the REXX programming language. An applicable TCP/IP setup is required for your VSE system. More information is found in the REXX/VSE Reference Manual.

SSL-support for SOCKET function

The SOCKET subfunctions INITIALIZE, ACCEPT, CONNECT, and TAKESOCKET are enhanced to provide the possibility to write SSL-enabled socket applications in REXX. More information is found in the REXX/VSE Reference Manual.

LINK and LINKPGM Host Command Environment: new ARXEOJTB entry field

When specifying within ARXEOJTB that a called phase is to be LOADed into program area, you can choose between 2 options of space checking within the program area: Either all phases are to be considered that start with the first 4 bytes of the given phasename (option 'A' or blank, for instance overlay program MSHP) or only the size of the single given phasename is checked (option 'S').

Here are sample parts of an ARXEOJTB table making use of the new field:

```
ARXEOJTB_ENTRY_3   EQU *             /* ARXEOJTB Entry 3             */
                   DC CL8'MSHP    '  /* Synonym used in ADDRESS LINK */
                   DC CL8'MSHP    '  /* Name of phase                */
                   DC AL4(0)         /* Must be Zero                  */
                   DC CL1'YES'       /* Phase Loaded in program area */
                   DC CL1'ALL'       /* Consider all phases  @56301UB*/
                   DC CL6' '         /* Reserved             @56301UB*/
```

```
ARXEOJTB_ENTRY_6   EQU *             /* ARXEOJTB Entry 6             */
                   DC CL8'DITTO   '  /* Synonym used in ADDRESS LINK */
                   DC CL8'DITTO   '  /* Name of phase                */
                   DC AL4(0)         /* Must be Zero                 */
                   DC CL1'YES'       /* Phase Loaded in program area */
                   DC CL1'SINGLE'    /* Consider this phase  @56301UB*/
                   DC CL6' '         /* Reserved             @56301UB*/
```

Stemnames with multiple dots

The following REXX commands and REXX functions allow specification of stemnames with multiple dots:

- **ADDRESS VSE VSAMIO**
- **ADDRESS POWER GETQE**
- **ADDRESS POWER PUTQE**
- **OUTTRAP()**
- **REXXIPT()**
- **SORTSTEM()**
- **GETMSG()**

Remarks

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Comments and Questions

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

vseesa@de.ibm.com