

**Rexx/VSE at CCI**  
**Presented by Michael Rosinger**  
**WAVV 2003**  
**Winston-Salem, NC**

# Rexx/VSE at CCI

## **Rexx/VSE**

- Do not have FAQs or other AO product
- Only implementation is Rexx/VSE
- Batch execution as a PROC
- Interpreted, but very efficient
- Systems Programmer's best friend...
- Not really a "REXXpert", but I have fun trying...

3

## **Sample #1 Dynamic Batch FTP**

- The problem:
  - Input VSAM file has customer number embedded in the dataset name
  - Output ASCII file needs to have the date and time embedded in the filename
  - Target folder for output file is customer number
- Rexx is only tool available to solve problem

4

## Sample #1 Dynamic Batch FTP

- Expect and edit one argument (customer #)
- Pipe SYSIPT commands to BSI batch FTP client via REXXIPT
- Generate JCL statements via ADDRESS JCL
- Execute batch FTP via ADDRESS LINK\*

\*must be added to REXX EOJ Return Table

5

## Dynamic Batch FTP...

```
arg crdtr
say 'REXX0024: creditor number in is->' crdtr '<-'
if length(crdtr)<>6 | verify(crdtr,'0123456789')<>0 then
do
  say 'REXX0024: invalid creditor number specified as input'
  say 'REXX0024: expecting numeric string of exactly six digits'
  say 'REXX0024: terminating...'
  exit 8
end
/* get time and parse out the colons for use in file name */
tm = time()
tm = substr(tm,1,2)||substr(tm,4,2)||substr(tm,7,2)
say 'REXX0024: parsed time is->' tm '<-'
/* build FTP commands in stem variable inp. */
inp.0 = 9
inp.1 = 'OPEN ftpserver nnnn'
inp.2 = 'USER ?????'
inp.3 = 'PASS ?????'
inp.4 = 'SYST'
inp.5 = 'CWD '||crdtr
inp.6 = 'INPUT VSAM FEEDBCK'
. . . . .
```

## ... Dynamic Batch FTP

```
. . . . .
inp.7 = 'TYPE A'
inp.8 = 'STOR '||crdtr||'_FB_'||date('S')||'_'||tm||'.TXT'
inp.9 = 'QUIT'
say 'REXX0024: output from ftp...'

address jcl
call rexxipt inp.
"// OPTION LOG"
"// LIBDEF PHASE,SEARCH=VENDLIB.IPTOOLS"
"// LIBDEF SOURCE,SEARCH=(VENDLIB.IPCONFIG,VENDLIB.IPTOOLS)"
"// DLBL FEEDBCK,'#'||crdtr||".FB.FILE',,VSAM,CAT=IFECCAT"
"// SETPFIX LIMIT=128K"
address link "BSTTFTPC"
y = outtrap('output.')
do i = 1 to output.0
    say output.i
end

return rc
```

## Dynamic Batch FTP The JCL

```
// LIBDEF PROC,SEARCH=CCILIB.PROD
// EXEC REXX=REXX0024,PARM='012345'
/*
/* generated JCL for FTP step would appear like this...
/*
// OPTION LOG
// LIBDEF PHASE,SEARCH=VENDLIB.IPTOOLS
// LIBDEF SOURCE,SEARCH=(VENDLIB.IPCONFIG,VENDLIB.IPTOOLS)
// DLBL FEEDBCK,'#012345.FB.FILE',,VSAM,CAT=IFECCAT
// SETPFIX LIMIT=128K
// EXEC BSTTFTPC
OPEN ftpserver nnnn
USER ????
PASS ????
SYST
CWD 012345
INPUT VSAM FEEDBCK
TYPE A
STOR 012345_FB_20030426_142400.TXT
QUIT
/*
```

## **Sample #2 Highlighted Console Message**

- The problem:
  - Need way to put a lengthy, highlighted message on the console requiring operator response
  - “// PAUSE” is limited to total of 72 characters - so message text can only be 63 characters

9

## **Sample #2 Highlighted Console Message**

- Input message text from SYSIPT
- Receive multiple lines of text
- Concatenate lines into one message
- Truncate to maximum of 122 characters
- Display message on console via pausemsg() function and wait for operator response

10

## Highlighted Console Message

```
/* REXXMSG */
/* use new pausemsg() function from Ursula to place message in from */
/* SYSIPT on console and then wait for operator response */
/* */
'Execio * Diskr sysipt (Stem cardin.)
If cardin.0 < 1 then
  Do
    call assgn 'STDOUT','SYSLOG' /* REROUTE STDOUT TO SYSLOG */
    Say 'REXXMSG: enter at least 1 line of text via SYSIPT'
    call assgn 'STDOUT','SYSLST' /* RESET STDOUT TO SYSLST */
    exit 2
  End
/* */
msg_text = ''
Do i = 1 To cardin.0
  msg_text = msg_text||' '||strip(cardin.i,'B')
End
msg_text = strip(msg_text,'B') /* strip blanks from complete msg */
If length(msg_text) > 122 then msg_text = substr(msg_text,1,122)
y = pausemsg(msg_text)

exit 0
```

## Highlighted Console Message - The JCL

```
// LIBDEF PROC,SEARCH=CCILIB.PROD
// EXEC REXX=REXXMSG
REMINDER: Load tapes for weekend:
4->250, 2->251. Please use
initialized/long tapes.
Make sure write-protect is OFF!
/*

X1 0045 // JOB OPSMSG          REMINDER FOR WEEKEND
        DATE 04/25/2003, CLOCK 12/48/37
X1-0045 REMINDER: Load tapes for weekend: 4->250, 2->251. Please
        use initialized/long tapes. Make sure write-protect is OFF!
```

**Sample #3  
Generate Dynamic  
// DATE card**

- The problem:
  - Date on report(s) must be last day of processed month
  - Sometimes report jobstep begins after midnight (first day, next month)
  - No job scheduler software to control this
  - Did not want to make program change (due to other complications)

13

**Sample #3  
Generate Dynamic  
// DATE card**

- Get current date
- If DD is "01", compute date for yesterday
- Determine date to use
- Construct "// DATE" card
- Execute "// DATE" via ADDRESS JCL

14

## Generate Dynamic // DATE card...

```
/* ***** */
/* SETDATE: This proc will proceed EOM program DMS620 and issue a */
/* // DATE to the job stream via Address JCL. This will */
/* hopefully solve the problem of the next month's date */
/* showing up on the statistics reports. */
/* If the DD of the current date is 1, compute the date */
/* for yesterday and then use that date; otherwise, use */
/* the current date "as is". */
/* ***** */
CALL ASSGN 'STDOUT','SYSLOG' /* REROUTE STDOUT TO SYSLOG */
SAY 'SETDATE: Checking the date ('||date('N')||') for change...'
CALL ASSGN 'STDOUT','SYSLSST' /* RESET STDOUT TO SYSLSST */

today_b = date('B') /* get base date */
today_u = date('U') /* get USA format */
say 'today B is' today_b
say 'today U is' today_u
. . . . .
```

## ... Generate Dynamic // DATE card

```
. . . . .
if substr(today_u,4,2) = '01' then
do
yesterday_b = today_b - 1
yesterday_u = date('U',yesterday_b,'B')
new_date = yesterday_u
end
else
new_date = today_u

/* now add the four digit year to the new date */
new_date_u = date('N',new_date,'U')
new_date = substr(new_date,1,6)||substr(new_date_u,8)

/* issue the // DATE command with the new date */
cmd = '// DATE '||new_date
ADDRESS JCL cmd
call assgn 'STDOUT','SYSLOG' /* REROUTE STDOUT TO SYSLOG */
say 'SETDATE: according to date submitting->' cmd
call assgn 'STDOUT','SYSLSST' /* RESET STDOUT TO SYSLSST */

exit
```



## Generate Dynamic // DATE card - The JCL

```
/* exec proc SETDATE to generate a // DATE for report DMS620
// LIBDEF PROC,SEARCH=CCILIB.PROD
// EXEC REXX=SETDATE
/*
// DATE mm/dd/yyyy <== date card generated by proc
/*
/* report program(s) follow...
/*
```

## For More Information...

- Rexx/VSE home page:

- <http://www-1.ibm.com/servers/eserver/zseries/os/vse/support/rexx/rexxhome.html>

- Various Rexx links from IBM:

- <http://www-1.ibm.com/servers/eserver/zseries/os/vse/support/rexx/rexxinfo.html>