133 VM

September 1997

update

In this issue

© Xephon plc 1997

VM Update

Published by
Xephon
27-35 London Road
Newbury
Berkshire RG14 1JL
England
Telephone: 01635 38342
From USA: 01144 1635 38342
E-mail: xephon@compuserve.com

North American office
Xephon/QNA
1301 West Highway 407, Suite 201-405
Lewisville, TX 75067
USA
Telephone: 940 455 7050

Australian office
Xephon/RSM
GPO Box 6258
Halifax Street
Adelaide, SA 5000
Australia
Telephone: 088 223 1391

Editorial panel
Articles published in *VM Update* are reviewed by our panel of experts. Members of the panel include John Illingworth (UK), Reinhard Meyer (Germany), Philippe Taymans (Belgium), Romney White (USA), Martin Wicks (UK), and Jim Vincent (USA).

Contributions
Articles published in *VM Update* are paid for at the rate of £170 ($250) per 1000 words for original material. To find out more about contributing an article, without any obligation, please contact us at any of the addresses above and we will send you a copy of our *Notes for Contributors*.

Editor
Trevor Eddolls

Disclaimer
Readers are cautioned that, although the information in this journal is presented in good faith, neither Xephon nor the organizations or individuals that supplied information in this journal give any warranty or make any representations as to the accuracy of the material it contains. Neither Xephon nor the contributing organizations or individuals accept any liability of any kind howsoever arising out of the use of such material. Readers should satisfy themselves as to the correctness and relevance to their circumstances of all advice, information, code, JCL, EXECs, and other contents of this journal before making any use of it.

Subscriptions and back-issues
A year's subscription to *VM Update*, comprising twelve monthly issues, costs £170.00 in the UK; $255.00 in the USA and Canada; £176.00 in Europe; £182.00 in Australasia and Japan; and £180.50 elsewhere. In all cases the price includes postage. Individual issues, starting with the January 1990 issue, are available separately to subscribers for £14.50 ($21.50) each including postage.

VM Update on-line
Code from *VM Update* can be downloaded from our Web site at http://www.xephon.com; you will need the user-id shown on your address label.

© Xephon plc 1997. All rights reserved. None of the text in this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior permission of the copyright owner. Subscribers are free to copy any code reproduced in this publication for use in their own installations, but may not sell such code or incorporate it in any commercial product. No part of this publication may be used for any form of advertising, sales promotion, or publicity without the written permission of the publisher. Copying permits are available from Xephon in the form of pressure-sensitive labels, for application to individual copies. A pack of 240 labels costs $36 (£24), giving a cost per copy of 15 cents (10 pence). To order, contact Xephon at any of the addresses above.

Printed in England.

Transferring files between VM hosts using a PC

The SEND and RECEIVE commands of OS/2 Communication Manager (CM/2), Personal Communications 3270, or the FTTERM terminal emulation do not allow more than one file to be transferred with one invocation of the command. So, if you want to transfer several files from one host to a PC and then up to another host VM system, it can be a tedious task, especially if you have lots of files to process.

This REXX procedure helps you to collect the files to be transferred by creating a batch file with all the necessary SEND/RECEIVE commands.

It is assumed that you have a PC running terminal sessions to the source and the target host systems. The source and/or target session can be CM/2, PC/3270, or FTTERM.

FTTERM is a DOS terminal emulation product that is often used when you are connected to a host ASCII adapter or 7171, or anything similar, via a modem and an asynchronous communication line.

SYNTAX

DOWNUP is called with the following parameters:

```
DOWNUP fn ft fm host1 host2</fm2> <art> <(platte>
```

or

```
DOWNUP FILE     host1 host2</fm2> <art> <(platte>
```

Fn ft fm specify the files to be transferred; wildcards are allowed. Alternatively you can use ‘FILE’ instead of ‘fn ft fm’. In the latter you must prepare a file (the file name is prompted by the procedure) in which you specify lines with ‘fn ft fm’.

Host1 is the session ID of the download host session (eg ‘A’); for FTTERM specify ‘FT’ instead. Host2 is the session ID of the target host session (eg ‘B’). Fm2 can be specified if you want the files to be copied with a different filemode on the target host (the default is the filemode of the source host).

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

With ‘art’ you define the transfer type: ‘A’ means ASCII CRLF and ‘B’ (which is the default) means BINARY NOCRLF.

The files to be transferred are temporarily saved on the PC in the root directory of drive C; you can, however, specify a different drive by coding ‘platte’.

Restrictions:

- Host1 and host2 may not both be FT.
- Art may not be ‘A’ if there is no write access to the source file because the file is temporarily unpacked.

? gives a help panel for the function.

INTERNALS AND PREREQUISITES

You must have a PC that is connected to both host systems concurrently by either Communications Manager, Personal Communications 3270, or the FTTERM terminal emulation product.

INSTALLATION SPECIFIC CONFIGURATION

Hardcoded values – DOWNUP FILES A, DOWN FILES A, or UP FILES A are used to hold the batch SEND/RECEIVE commands.

Variable value – ‘ftpath’ defines the path where the FTTERM terminal emulation product is installed on the PC.

DOWNUP EXEC

```
/*********************************************************************/
/* Transfer files from one host to another via a PC                  */
/*********************************************************************/
/* Call:   DOWNUP fn ft fm host1 host2</fm2> <art> <(platte>         */
/*         DOWNUP FILE     host1 host2</fm2> <art> <(platte>         */
/*               fn, ft, fm           : source files (wildcards OK)  */
/*               FILE                 : specify files through a file */
/*                                    :   (the name is prompted)     */
/*                                    :   lines in file: fn ft fm    */
/*               host1                : PC session for download      */
/*                                    : 'FT' = FTTERM                */
/*               host2                : PC session for upload        */
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
/*                                  : 'FT' = FTTERM                 */
/*               fm2                : file mode at target host      */
/*               art = A            : ASCII CRLF                    */
/*                   = B  (default) : BINARY NOCRLF                 */
/*               platte   (def. C)  : PC drive for temporary files  */
/*********************************************************************/
trace off
backslash = '\'

/************ path for FTTERM on the PC (please change) *************/

ftpath = backslash'COMM'backslash'FTTERM'backslash

/*********************************************************************/

parse upper arg file .
if file = '?' then signal hilfe
if file = 'FILE' then parse upper arg . host1 host2 art '(' platte .
   else parse upper arg fn ft fm host1 host2 art '(' platte .
if fn = '' | ft = '' | fm = '' | host1 = '' | host2 = '' ,
   then signal hilfe
parse var host2 host2 '/' fm2
if art = '' then art = 'B'
if platte = '' then platte = 'C'
if art ¬= 'A' & art ¬= 'B' then signal hilfe
if art = 'A' then app = 'ASCII CRLF'
if art = 'B' then app = ''
if length(host1) ¬= 1 & host1 ¬= 'FT' then signal hilfe
if length(host2) ¬= 1 & host2 ¬= 'FT' then signal hilfe
if host1 = 'FT' & host2 = 'FT' then signal hilfe
if length(fm2) > 1 then signal hilfe
if length(platte) ¬= 1 then signal hilfe
quelle_ftterm = Ø
ziel_ftterm = Ø
if host1 = 'FT' then do
                        host1 = ''
                        quelle_ftterm = 1
                     end
                else host1 = host1':'
if host2 = 'FT' then do
                        host2 = ''
                        ziel_ftterm = 1
                     end
                else host2 = host2':'
'MAKEBUF'
if file = 'FILE' then do
   do until erc = Ø
      say 'Please specify file name (fn ft fm):'
      pull fn ft fm .
      if fm = '' then fm = 'A'
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
      address command 'ESTATE' fn ft fm
      erc = rc
      if erc ¬= Ø then say 'File' fn ft fm 'does not exist!'
   end
   'EXECIO * DISKR' fn ft fm '(FIFO FINIS'
end
else do
   'LISTFILE' fn ft fm '(STACK FIFO FORMAT'
end
anz = queued()
do i = 1 to anz
   pull xfn.i xft.i xfm.i . lrecl.i .
end
/*********************************************************************/
/* Source and target are NOT FTTERM                                  */
/*********************************************************************/
if ¬ (quelle_ftterm | ziel_ftterm) then do
   do i = 1 to anz
      if fm2 ¬= '' then xfm2 = fm2
                   else xfm2 = xfm.i
      if art = 'B' then appsend = app 'LRECL' lrecl.i
                   else appsend = app
      if art = 'A' then do
         'SET CMSTYPE HT'
         'COPYFILE' xfn.i xft.i xfm.i '= = = (UNPACK'
         if rc ¬= Ø & rc ¬= 32 then do
            say xfn.i xft.i xfm.i 'cannot be unpacked'
            signal ende
         end
         'SET CMSTYPE RT'
      end
      queue 'RECEIVE' platte':'xfn.i'.'left(xft.i,3) ,
            host1 || xfn.i xft.i xfm.i '('app
      queue 'SEND' platte':'xfn.i'.'left(xft.i,3) ,
            host2 || xfn.i xft.i xfm2 '('appsend
      queue 'DEL' platte':'xfn.i'.'left(xft.i,3)
   end
   n = 3 * anz
   address command 'ERASE DOWNUP FILES A'
   'EXECIO' n 'DISKW DOWNUP FILES A (FINIS'
   say 'Please switch to the PC command prompt and enter:'
   say 'RECEIVE' platte':DOWNUP.CMD' host1'DOWNUP FILES A (ASCII CRLF'
   say 'then enter:'
   say 'DOWNUP'
end
/*********************************************************************/
/* Source or target is FTTERM                                        */
/*********************************************************************/
else do
   do i = 1 to anz
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
   if art = 'A' then do
      'SET CMSTYPE HT'
      'COPYFILE' xfn.i xft.i xfm.i '= = = (UNPACK'
      if rc ¬= Ø & rc ¬= 32 then do
         say xfn.i xft.i xfm.i 'cannot be unpacked'
         signal ende
      end
      'SET CMSTYPE RT'
   end
   queue 'RECEIVE' platte':'ftpath || xfn.i'.'left(xft.i,3) ,
         host1 || xfn.i xft.i xfm.i '('app
end
n = anz + 1
if ziel_ftterm ,
   then do
      queue 'RECEIVE' platte':'ftpath'UP' ,
               host1'UP FILES A (ASCII CRLF'
      queue 'RECEIVE' platte':DELT.CMD' ,
               host1'DELT FILES A (ASCII CRLF'
      n = anz + 2
   end
   else queue 'RECEIVE' platte':UP.CMD' host1'UP FILES A (ASCII CRLF'
address command 'ERASE DOWN FILES A'
'EXECIO' n 'DISKW DOWN FILES A (FINIS'
do i = 1 to anz
   if fm2 ¬= '' then xfm2 = fm2
                else xfm2 = xfm.i
   if art = 'B' then appsend = app 'LRECL' lrecl.i
                else appsend = app
   queue 'SEND' platte':'ftpath || xfn.i'.'left(xft.i,3) ,
         host2 || xfn.i xft.i xfm2 '('appsend
   if quelle_ftterm then , /* source is FTTERM */
      queue 'DEL' platte':'ftpath || xfn.i'.'left(xft.i,3)
end
if quelle_ftterm then n = 2 * anz
                 else n = anz
address command 'ERASE UP FILES A'
'EXECIO' n 'DISKW UP FILES A (FINIS'
if ziel_ftterm then do
   do i = 1 to anz
      queue 'DEL' platte':'ftpath || xfn.i'.'left(xft.i,3)
   end
   /* if host2 ¬= '' then n = 2 * anz */
   if quelle_ftterm then n = 2 * anz
                    else n = anz
   address command 'ERASE DELT FILES A'
   'EXECIO' n 'DISKW DELT FILES A (FINIS'
end
if ziel_ftterm then do    /* target is FTTERM */
 /*
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
      say 'Please switch to a PC command prompt and enter:'
      say 'RECEIVE' platte':DOWN.CMD' host1'DOWN FILES A (ASCII CRLF'
      say 'then:'
      say 'DOWN'
      say 'Then switch to FTTERM (ALT F8 in FTTERM session) and enter:'
      say 'BATCH UP'
      say 'Then switch to PC command prompt and enter:'
      say 'DELT'
    */
    /* short cut: preliminary on the PC is file: FTDN.CMD */
      say 'Please switch to PC command prompt and enter:'
      say 'FTDN'
      say 'Then switch to FTTERM (ALT F8 in FTTERM session) and enter:'
      say 'BATCH UP'
      say 'Then switch to PC command prompt and press RETURN'
   end
   else do                                  /* source is FTTERM */
    /*
      say 'Please switch to FTTERM (ALT F8 in FTTERM-Session) and
enter:'
      say 'RECEIVE' platte':DOWN' host1'DOWN FILES A (ASCII CRLF'
      say 'folowing:'
      say 'BATCH DOWN'
      say 'Then switch to the OS/2 window and enter:'
      say platte':'ftpath'UP'
    */
    /* short cut: preliminaries on the PC are: FTUP.CMD and FTDN */
      say 'Switch to FTTERM (ALT F8 in FTTERM session) and enter:'
      say 'FTDN'
      say 'then:'
      say 'BATCH DOWN'
      say 'Then switch to PC command prompt and enter:'
      say 'FTUP'
   end
end

/*********************************************************************/
ende:
'DROPBUF'
exit
/*********************************************************************/
/* Help                                                              */
/*********************************************************************/
hilfe:
'VMFCLEAR'
address cms 'type downup  exec * 1 18'
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

PC BATCH FILES

On the PC you need the following batch files:

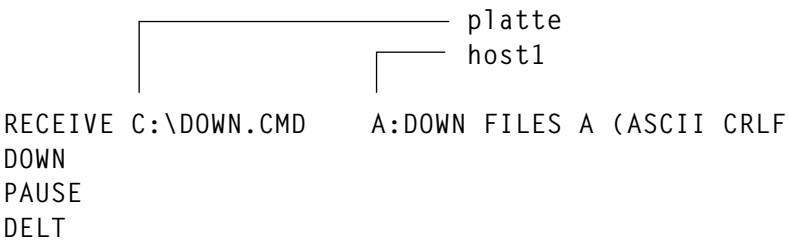

Figure 1: FTDN command

- FTDN in the FTTERM product path (this is a FTTERM batch file): receive down down files a (ascii crlf

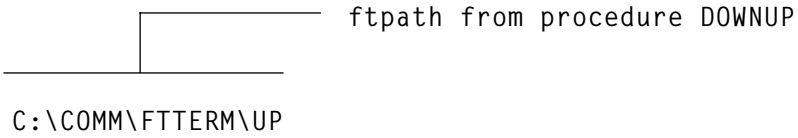

Figure 2: FTUP command

- FTDN.CMD (see Figure 1) and FTUP.CMD (see Figure2).

Dr Reinhard Meyer (Germany) © Xephon 1997

Named substrings

REXX has very powerful string handling functions, but it does not offer a way to construct data structures like C or COBOL do. The following small collection of REXX functions can be used as an extension to the language. They enable a programmer to access parts of a string based on names instead of exact positions and lengths – hence the title ‘named substrings’.

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

The named substring functions are based on something I call a ‘named substring record definition’ – a string comprising the following elements for every named substring: name, start, length. Where:

- Name is the name of the substring.
- Start is the starting position of the substring in the string.
- Length is the length of the substring.

An example of a named substring record definition is: ‘firstname, 1,20, lastname,21,20, address,41,30’.

It is of course a good idea to store such a record definition in a file rather than hide it in a procedure. To this end I wrote NSFILE EXEC, which reads a file containing a record definition and returns it in the format described above. This file may contain comments, which help in documenting an application.

Perhaps the most obvious usage of named substrings is in the handling of file records. As an example for this article I used a logfile which is written and read by several procedures at our installation. Using the named substring functions, the record layout of this file is defined once and used by all these procedures. So whenever the layout of this logfile changes, (hopefully) all I will have to do is adapt the record definition.

There are some more things that could be done using a named substring record definition, like a simple report generator or a field-based editor, but I haven’t found the time to write these functions yet. Maybe someone else will?

CONTENTS OF THE LOG FILE NS DATAFILE

```
AVORS    1997-Ø3-12 1Ø:Ø8:25  ØØØØØØØ1 SQLTEST  U ØØ2 TØØ152
TITLE=UNLOAD HYP SBACK-TEST
AVORS    1997-Ø3-18 Ø8:Ø5:22  ØØØØØØØ2 SQLTEST  U ØØ2 TØØØ91
TITLE=UNLOAD RN8ØSTG3 (ALLE OWS_...)
AVORS    1997-Ø3-18 1Ø:Ø2:48  ØØØØØØØ3 SQLTEST  U ØØ2 TØØ175
TITLE=UNLOAD RN8ØSTG3 (ALLE OWS_...)
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

NSSAMP RECDEF

```
* <---------------------------------------------------------------->
* <  NSSAMP RECDEF                                                 >
* <                                                                >
* <  SAMPLE RECORD DEFINITION FOR NAMED SUBSTRING FUNCTIONS        >
* <                                                                >
* <---------------------------------------------------------------->
*
*
*
* FIELD  START LENGTH  DESCRIPTION
TSUID        1   8     TIMESTAMP: USERID
TSDATE      1Ø  1Ø     TIMESTAMP: DATE    YYYY-MM-DD
TSTIME      21   8     TIMESTAMP: TIME    HH:MM:SS
TASK        31   8     UNIQUE     TASK NUMBER
DATABASE    4Ø   8
TYPE        49   1     "B" -> FULL BACKUP (UARCHIVE)
*                      "F" -> FULL RESTORE
*                      "U" -> UNLOAD DATASPACE(S)
*                      "R" -> RELOAD TABLE(S)
JOB         51   3     JOB NUMBER
VOLID       55   6     TAPE VOLUME ID
TITLE       71  6Ø     JOB TITLE
```

NSFILE EXEC

```
/* ================================================================== */
/* Name        :  NSFILE   EXEC                                       */
/* ================================================================== */
/* Application :  named substrings                                    */
/*                                                                    */
/* Usage       :  Procedure                                           */
/*                                                                    */
/* Arguments   :  record_definition_fileid                            */
/*                                                                    */
/* Result      :  record_definition                                   */
/*                                                                    */
/* Function    :  Named Substrings: read a record definition file     */
/*                                                                    */
/* ================================================================== */
arg ifid

recdef=''
'pipe <' ifid '| nlocate 1-1 /*/ | stem in.'
do i=1 to in.Ø
   parse var in.i name pos len .
   recdef=recdef name','pos','len
end

return recdef
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

NSREAD EXEC

```
/* ================================================================== */
/* Name        :  NSREAD   EXEC                                       */
/* ================================================================== */
/* Application :  named substrings                                    */
/*                                                                    */
/* Usage       :  Function                                            */
/*                                                                    */
/* Arguments   :  oldrec,recdef,'fieldlist'[,prefix])                 */
/*                                                                    */
/* Result      :  Ø                                                   */
/*                                                                    */
/* Function    :  Named substrings: Read  Function                    */
/*                                                                    */
/* This function is used to read named substrings from a string.      */
/* Based on the record definition (argument recdef), the variables    */
/* specified in argument fieldlist are set in the calling EXEC        */
/* (via vmfe2e) to the values extracted from argument oldrec.         */
/* If the optional argument prefix is specified, the variables        */
/* are prefixed with it. The argument fieldlist can be specified      */
/* as an asterisk to read all substrings in oldrec.                   */
/* ================================================================== */
arg oldrec,recdef,fieldlist,prefix

fnames=''
nfields=words(recdef)
do i=1 to nfields
   parse value word(recdef,i) with fname ',' p  ',' l
   fnames=fnames fname
   pos.fname=p
   len.fname=l
end
reclen=pos.fname   + len.fname - 1

if fieldlist = '*' then fieldlist=fnames

fields=''
do i=1 to words(fieldlist)
   fields=fields prefix || word(fieldlist,i)
end

do i=1 to words(fieldlist)
   f=word(fieldlist,i)
   interpret word(fields,i)'="'substr(oldrec,pos.f,len.f)'"'
end

'vmfe2e set' fields

return Ø
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

NSWRITE EXEC

```
/* ================================================================== */
/* Name        :  NSWRITE  EXEC                                       */
/* ================================================================== */
/* Application :  named substrings                                    */
/*                                                                    */
/* Usage       :  Function                                            */
/*                                                                    */
/* Arguments   :  oldrec,recdef,'fieldlist'[,prefix])                 */
/*                                                                    */
/* Result      :  newrec                                              */
/*                                                                    */
/* Function    :  Named substrings: Write Function                    */
/*                                                                    */
/* This function returns the argument oldrec (a string) in which      */
/* the named substrings specified in the argument fieldlist           */
/* have been replaced with the values of the corresponding variables  */
/* in the calling EXEC.                                               */
/* The argument fieldlist can be specified                            */
/* as an asterisk to replace all substrings.                          */
/* ================================================================== */
arg oldrec,recdef,fieldlist,prefix

fnames=''
nfields=words(recdef)
do i=1 to nfields
   parse value word(recdef,i) with fname ',' p  ',' l
   fnames=fnames fname
   pos.fname=p
   len.fname=l
end
reclen=pos.fname   + len.fname - 1

if fieldlist = '*' then fieldlist=fnames

fields=''
do i=1 to words(fieldlist)
   fields=fields prefix || word(fieldlist,i)
end

'vmfe2e get' fields

oldrec=left(oldrec,reclen)
do i=1 to words(fieldlist)
   f=word(fieldlist,i)
   oldrec=overlay(value(word(fields,i)),oldrec,pos.f,len.f)
end

return oldrec
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

If many extract operations have to be done, generating a parse string with this function and then using the REXX parse statement to do the extracting is a lot faster than using NSREAD EXEC.

```
/* ================================================================== */
/* Name        :  NSPARSE  EXEC                                       */
/* ================================================================== */
/* Application :  named substrings                                    */
/*                                                                    */
/* Usage       :  Function                                            */
/*                                                                    */
/* Arguments   :  recdef,'fieldlist'[,prefix])                        */
/*                                                                    */
/* Result      :  Parse String                                        */
/*                                                                    */
/* Function    :  Named substrings: generate Parse String             */
/*                                                                    */
/* This function returns a string which can be used with the          */
/* REXX parse statement to extract substrings from a string.          */
/* The argument recdef is a named substring record definition, the    */
/* argument fieldlist is a list of fields that are to be extracted    */
/* with the parse, and the argument prefix can be used to optionally  */
/* prefix the variable names with a string.                           */
/* The argument fieldlist can be specified                            */
/* as an asterisk to extract all substrings.                          */
/* ================================================================== */
arg recdef,fieldlist,prefix

fnames=''
nfields=words(recdef)
do i=1 to nfields
   parse value word(recdef,i) with fname ',' p  ',' l
   fnames=fnames fname
   pos.fname=p
   len.fname=l
end
reclen=pos.fname   + len.fname - 1

if fieldlist = '*' then fieldlist=fnames

ps=''
do i=1 to words(fieldlist)
   f=word(fieldlist,i)
   ps=ps pos.f  prefix || f pos.f+len.f '.'
end

return ps
```

 © 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

SAMPLE EXEC

```
/* ================================================================== */
/* sample EXEC to demonstrate named substring functions               */
/* ================================================================== */

/* read a record definition file                                      */
recdef=nsfile('nssamp recdef a')
say 'contents of recdef:' recdef
say

/* generate a parse string                                            */
parsestring=nsparse(recdef,'*','w_')
say 'contents of parsestring:' parsestring
say

/* read in a datafile                                                 */
'pipe < ns datafile a | stem records.'

say '-----  demonstrate the parse function ------'
do i=1 to records.0
   interpret 'parse value records.i with' parsestring
   say
   say 'contents of record:' records.i
   say 'contents of variable w_database:' w_database
   say 'contents of variable w_tsdate  :' w_tsdate
   say 'contents of variable w_volid   :' w_volid
end

say '-----  demonstrate the read function ------'
record=records.1
say
say
say 'original contents of record:' record
say
x=nsread(record,recdef,'tsdate database')
say 'contents of variable database:' database
say 'contents of variable tsdate  :' tsdate

say '-----  demonstrate the write function ------'
database='SQLPROD'
tsdate='1999-12-31'
record=nswrite(record,recdef,'tsdate database')
say
say 'new contents of record     :' record
say
return
```

EXAMPLE RESULTS

```
contents of recdef:  TSUID,1,8 TSDATE,10,10 TSTIME,21,8 TASK,31,8
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
DATABASE,40,8
TYPE,49,1 JOB,51,3 VOLID,55,6 TITLE,71,60

contents of parsestring:  1 W_TSUID 9 . 10 W_TSDATE 20 . 21 W_TSTIME 29
. 31 W_T
ASK 39 . 40 W_DATABASE 48 . 49 W_TYPE 50 . 51 W_JOB 54 . 55 W_VOLID 61 .
71 W_TI
TLE 131 .

-----  demonstrate the parse function ------

contents of record: AVORS    1997-03-12 10:08:25  00000001 SQLTEST  U
002 T00152
          TITLE=UNLOAD HYP SBACK-TEST
contents of variable w_database: SQLTEST
contents of variable w_tsdate  : 1997-03-12
contents of variable w_volid   : T00152

contents of record: AVORS    1997-03-18 08:05:22  00000002 SQLTEST  U
002 T00091
          TITLE=UNLOAD RN80STG3 (ALLE OWS_...)
contents of variable w_database: SQLTEST
contents of variable w_tsdate  : 1997-03-18
contents of variable w_volid   : T00091

contents of record: AVORS    1997-03-18 10:02:48  00000003 SQLTEST  U
002 T00175
          TITLE=UNLOAD RN80STG3 (ALLE OWS_...)
contents of variable w_database: SQLTEST
contents of variable w_tsdate  : 1997-03-18
contents of variable w_volid   : T00175
-----  demonstrate the read function ------


original contents of record: AVORS    1997-03-12 10:08:25  00000001
SQLTEST  U 0
02 T00152          TITLE=UNLOAD HYP SBACK-TEST

contents of variable database: SQLTEST
contents of variable tsdate  : 1997-03-12
-----  demonstrate the write function ------

new contents of record     : AVORS    1999-12-31 10:08:25  00000001
SQLPROD  U 0
02 T00152          TITLE=UNLOAD HYP SBACK-TEST

Ready;
```

© Xephon 1997

 © 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

Multiplatform command scheduler – part 3

This month we continue the code that allows multiplatform command scheduling.

```
RUNQUEST:
TEXT = QUETEXT
@X = WORD(QUETEXT,1)
IF CLKRULES & CMS,
   THEN 'REXXWRTR' CTLQUE 'QUE' CTLMOD 'Ø',
         TAGX||COLON QUEDATE QUETIME QUEFACT
IF CLKRULES & TSO,
   THEN 'REXXWRTR' CTLQUE 'QUE A Ø',
         TAGX||COLON QUEDATE QUETIME QUEFACT
IF CLKRULES & (OS2 | DOS),
   THEN 'EXECIO 1 DISKW' CTLQUE'.QUE Ø',
         TAGX||COLON QUEDATE QUETIME QUEFACT
IF ¬CLKRULES
   THEN SELECT
    WHEN @X = 'IF' THEN INTERPRET QUETEXT
    WHEN @X = 'REXX' THEN INTERPRET SUBWORD(QUETEXT,2)
    WHEN @X = 'CP' THEN INTERPRET "'"SUBWORD(QUETEXT,2)"'"
    WHEN @X = 'CMS' THEN INTERPRET "'"SUBWORD(QUETEXT,2)"'"
    WHEN @X = 'MSG' THEN INTERPRET "'"QUETEXT"'"
    WHEN @X = 'SAY' THEN INTERPRET QUETEXT
    /* CMS ONE LINERS MUST NOT BE IN QUOTES */
    OTHERWISE IF QUESEQ = PCESEQ THEN QUETEXT; ELSE INTERPRET QUETEXT
    END
RUNCODE = RC
/* SAY '' (QUETAG) '=' RUNCODE (TO SEE VALS) */
INTERPRET (QUETAG) '=' RUNCODE
RUNSEQ = RUNSEQ + 1
RUNQUEUE = QUETAG||COLON,
        QUEDATE QUETIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
RUNLIST.RUNSEQ = RUNQUEUE
RUNLIST = RUNLIST QUESEQ'/'RUNSEQ
                         /* SAVE QUESEQ NUMBER OF EACH COMMAND WE RUN */
IF TAGX ¬= '' THEN INTERPRET QUETAG'.Ø=RUNQUEUE'
WRITEQUE:
/*WHEN QUEFACT IS Ø ONLY EXECUTE COMMAND ONCE. STAR MAKES IT A
COMMENT.*/
IF QUEDAYS ¬= Ø THEN QUEDROP = ''
IF QUEDAYS = Ø & QUECYCLE = '' THEN DO
    QUEDROP = '*'
    RUNDATE = DATE('O')
    RUNTIME = TIME()
    END
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
IF QUEDAYS = Ø & QUECYCLE ¬= '' THEN DO
    IF QUEDATE < DATE('O') THEN DO
        QUEDROP = '*'
        RUNDATE = DATE('O')
        RUNTIME = TIME()
        END
    END
REQDATE = RUNDATE
IF QUELOOP ¬= '' THEN DO
    IF LEFT(NXTTIME,5) <= QUESTOP
       THEN DO
        PARSE VALUE RUNTIME WITH RUNTIME '-' Z
        RUNTIME = RUNTIME||'-'||QUESTART||'-'||QUESTOP
        END
       ELSE DO     /* > */
        NXTTIME = QUESTART||':ØØ'
        REQDATE = NXTDATE
        END
    END
IF PCESEQ ¬= QUESEQ THEN QUETEXT = ','
IF QUECYCLE ¬= '' & NXTDYS = Ø & CHKDATE = RUNDATE THEN
    CLKQUEUE = QUEDROP||TAGX||COLON,
            REQDATE NXTTIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
   ELSE
    CLKQUEUE = QUEDROP||TAGX||COLON,
            NXTDATE NXTTIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
CALL WRITEREC
SIGNAL READQUE
EOF:
IF WRTSEQ > Ø & ¬ONEPASS THEN SIGNAL BEGINQUE
IF ¬QUIET & RUNSEQ ¬= Ø THEN DO X = 1 BY 1 UNTIL X >=  RUNSEQ
 /* SAY 'CLKQUEUE -' RUNLIST.X */
    PARSE VALUE WORD(RUNLIST,X) WITH TXTQUE '/' TXTRUN
    IF DATATYPE(TXTRUN) = 'NUM'
       THEN DO
        IF DATATYPE(TXT.TXTQUE.Ø) = 'NUM' & TXT.TXTQUE.Ø > 1
           THEN DO Y = 1 FOR TXT.TXTQUE.Ø /* UNLOAD ACTUAL TEXT LINES */
            IF Y = 1
               THEN SAY 'CLKQUEUE -' SUBWORD(RUNLIST.TXTRUN,1,7)','
               ELSE SAY '          ' TXT.TXTQUE.Y
            END
           ELSE SAY 'CLKQUEUE -' RUNLIST.TXTRUN
        END
    END
/* RESET THE WORKAREAS IN CASE WE GO AROUND AGAIN. */
DROP TXT. RUNLIST.; RUNLIST = ''
DATTIMID = 'ON' DATE('U') 'AT' LEFT(TIME(),5) 'FOR ID' XUSERID
IF ¬QUIET THEN DO
    IF RUNSEQ ¬= Ø THEN SAY 'CLKQUEUE - RAN' RUNSEQ 'QUEUED COMMAND(S)',
        DATTIMID'.'
    IF RUNSEQ = Ø THEN SAY 'CLKQUEUE - NO QUEUED COMMANDS WERE
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
SELECTED',
        DATTIMID'.'
    END
RUNTOT = RUNTOT + RUNSEQ
RUNSEQ = Ø
IF SLPMINS = Ø | SLPCYCLS = Ø THEN SIGNAL EXIT
CNTCYCLS = CNTCYCLS + 1
IF CNTCYCLS < SLPCYCLS THEN DO
    IF ¬QUIET THEN SAY 'CLKQUEUE - SLEEPING FOR' SLPMINS,
            'MINUTE(S) AT' TIME() 'ON' DATE('U')||'.'
    IF CMS THEN 'EXECIO Ø CP (NOTYPE STRING SLEEP' SLPMINS 'MIN'
    IF TSO THEN 'MVSSLEEP' SLPMINS 'MIN *QUIET'
    IF OS2 THEN CALL 'SLEEP' SLPMINS 'MIN *QUIET'
    SIGNAL BEGINQUE
    END
IF QUIET THEN SIGNAL EXIT
IF CNTCYCLS > 1 THEN SAY 'CLKQUEUE - RAN A TOTAL OF' RUNTOT,
        'QUEUED COMMAND(S).'
IF PASSCNT > 1 THEN SAY 'CLKQUEUE -' PASSCNT,
        'PASSES OF "'||CTLQUE CTLTYP'" FILE MADE IN',
         CNTCYCLS 'QUEUEING CYCLES.'
EXIT:
EXIT ØØØ
FIND: PROCEDURE
PARSE UPPER ARG STR,FND
POS = WORDPOS(FND,STR)
RETURN POS
DOC:
/*BEGTYPE
REXXNAME: CLKQUEUE
FUNCTION: TO QUEUE VM/TSO/MVS/OS2/DOS COMMANDS AT THEIR APPOINTED TIMES.
   1) RUNS VM/TSO/MVS/OS2/DOS COMMANDS BASED ON "DATE" AND "TIME".
   2) RUNS CAN BE ONCE OR REQUEUED EVERY N NUMBER OF DAYS.
   3) CAN RERUN COMMANDS EVERY N HOURS, MINUTES, OR SECONDS.
   4) RERUNS CAN BE MADE TO STOP AT A CERTAIN TIME OR DAY.
   5) REPORTS ACTUAL RUN DATE, TIME, AND RET-CODE AFTER EACH RUN.
   6) RUNS CAN BE BASED ON PRIOR RETURN CODE SETTINGS USING IFS.
   7) UNLIKE SMART & PROFS ANY CMS/TSO OR EXEC/CLIST COMMAND CAN BE RUN.
   8) HAS SIMPLE ON-LINE DOCUMENTATION, AND FULL SET OF ERROR MESSAGES.
   9) RUNS SPECIAL TIME QUEUE SEQUENCES BY CALLING ITSELF.
   1Ø) CODE TIME BASED "IF" STATEMENTS TO HANDLE SPECIAL LOGICAL NEEDS.
HOWTORUN: ENTER COMMAND AS SHOWN BELOW.
          CLKQUEUE < &CTLQUE|* < &SLPMINS < &SLPCYCLS > > <*QUIET> >
          <      >  -MEANS THAT FIELDS WITHIN ARE OPTIONAL.
              |     -MEANS SELECT ONE OF TWO OR MORE OPTIONS.
           &CTLQUE  -ENTER THE NAME OF THE CLKQUEUE FILE.
                     THE FILETYPE MUST BE "CLKQUEUE".  IF "*" OR NO
                     PARAMETERS ARE ENTERED THEN THE DEFAULT FILENAME
                     OF "CLKQUEUE" IS USED.  IN "MVS" THIS CORRESPONDS
                     TO THE MEMBER NAME IN A DATASET NAMED...
          &SLPMINS  -ENTER THE NUMBER OF MINUTES CLKQUEUE SHOULD
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
                    SLEEP AFTER EACH PASS THROUGH A QUEUE CLKQUEUE
                    FILE.  IF "SLPMINS" IS ZERO OR NOT CODED THEN
                    CLKQUEUE WILL PROCESS THE QUEUE CLKQUEUE FILE
                  ONLY ONCE. THE MAXIMUM NUMBER OF SLEEP MINUTES IS 99.
          &SLPCYCLS -ENTER THE NUMBER OF TIMES CLKQUEUE SHOULD
                    SLEEP THEN REPROCESS THE QUEUE CLKQUEUE FILE
                    BEFORE IT STOPS RUNNING. THE DEFAULT IS ZERO.
            *QUIET   OPTIONAL KEYWORD THAT TELLS CLKQUEUE TO
                    SUSPEND NORMAL MESSAGE PRINTING.  ERROR
                    MESSAGES ARE STILL PRINTED.
          *ONEPASS   OPTIONAL KEYWORD.  IF CLKQUEUE QUEUES ONE
                    OR MORE REQUESTS IT WILL REPROCESS THE INPUT
                    CLOCK REQUESTS UNTIL IT FINDS NOTHING TO
                    QUEUE.  THIS OPTION TURNS THIS INPUT FILE
                    REPROCESSING LOGIC OFF.
          *CLKRULES  OPTIONAL KEYWORD.  MEANS THAT INSTEAD OF
                    EXECUTING QUEUED COMMANDS THEY ARE WRITTEN
                    TO A FILE OF THE SAME FILENAME, BUT WITH THE
                    FILETYPE OF "QUE".  THE FORMAT OF EACH OUTPUT
                    RECORD IS:  QUETAG QUEDATE QUETIME QUEFACT.
          *IFT(FT)   OPTIONAL KEYWORD.  USE THIS TO TELL CLKQUEUE
                    TO USE AN INPUT FILE FILETYPE OF YOUR CHOICE,
                    INSTEAD OF THE DEFAULT FILETYPE.
DESCRIBE: THIS EXEC WILL READ A FILE WITH THE FILETYPE OF CLKQUEUE
          SEARCHING FOR EXPIRED DATE AND TIMES.   THE FILENAME IS
          THE  USERS  CHOICE,   BUT  IF  NO  FILENAME  IS  ENTERED
          THE DEFAULT IS CLKQUEUE. ONCE QUEUE DATE AND TIMES HAVE
        QUEUED, NEW DATE AND TIMES ARE CREATED, THE ASSOCIATED
        COMMANDS ARE EXECUTED, AND INFO AND RETCODES SET BY THE COMMANDS
        ARE WRITTEN TO THE CLKQUEUE CONTROL FILE.  A FIELD
        CALLED THE QUEUE FACTOR IS USED TO CALCULATE WHEN TO
        SCHEDULE THE NEXT RUN, UNLESS IT'S ZERO, IN WHICH CASE IT
        IS RUN ONCE AND DROPPED.
QUEUEFMT: THIS PROGRAM USES A SCHEDULING CLKQUEUE FILE NAMED
          &QUECTL "CLKQUEUE" AS INPUT.  NOTE THAT &QUECTL IS A VARIABLE
          FOR WHICH "CLKQUEUE" IS DEFAULT.  IT SHOULD BE ON YOUR A DISK.
          IT CONTAINS THE RECORDS DEFINING WHAT TO SCHEDULE AND WHEN.
          BELOW IS A DEFINITION OF THE FIELDS IN EACH SCHEDULING
          RECORD.  RECORDS STARTING WITH AN ASTERISK(*) ARE TREATED
          AS COMMENTS, AND A RECORD STARTING WITH EOF SIGNALS THE
          END OF THE FILE.
  QUETAG QUEDATE QUETIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
          QUETAG  - TAG CAN UNIQUELY IDENTIFY THE COMMAND BEING RUN.
                    THE QUETAG IS ALSO USED AS A REXX VARIABLE INTO
                    WHICH THE QUETEXT RETURN CODE IS PLACED. TO SEE
                    THE VALUES ASSOCIATED WITH ANY OF THE PREVIOUSLY
                    RUN QUEUE ENTRIES PLEASE NOTE THAT THE ENTIRE
                    RUN TIME QUEUE RECORD IS PUT INTO THE ZERO(Ø)
                    INDEX VARIABLE OF QUETAG (IE QUETAG.Ø).
                    FOR EXAMPLE TO CHECK THE DATE THAT CMDX WAS
```

 © 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
                EXECUTED ENTER:
                    IF WORD(CMDX.Ø,5) = "9Ø/12/25" THEN "XMASLGC"
      QUEDATE - DATE OF NEXT RUN.  FORMAT = YY/MM/DD.
      QUETIME - TIME OF NEXT RUN.  FORMAT = HH/MM/SS.
      QUEFACT - INCREMENT FACTOR FOR CALCULATING NEXT RUN DATE/TIME.
           WHEN QUEFACT IS Ø COMMAND IS RUN ONLY ONCE.
           FMT = &DD|MONTH|YEAR<.H|M|S&NNN<*<XXX|HH:MM><.YY/MM/DD>>>
           &DD REPRESENTS THE NUMBER OF DAYS BEFORE COMMAND GETS
           REQUEUED.  FOR INSTANCE, SEVEN(7) MEANS RERUN COMMAND
         EVERY 7 DAYS. USE A PERIOD(.) TO INCLUDE A SECOND FIELD THAT
           SETS A CLKQUEUE TIME FACTOR.  ENTER 'MONTH' OR 'YEAR' TO
           REQUEUE MONTHLY AND YEARLY.   TO QUEUE A COMMAND USING A
           TIME FACTOR ENTER H FOR HOURS, M FOR MINUTES, OR S FOR
           SECONDS FOLLOWED BY THE TIME AMOUNT "NNN".  FOR A THREE
           MINUTE DELAY BETWEEN EXECUTIONS TO RUN EVERY DAY ENTER:
                Ø1.MØØ3
           "XXX|HH:MM" REPRESENTS THE NUMBER OF TIMES THE TIME
           FACTOR SHOULD BE INCREMENTED OR THE HH:MM TIME
           THAT THE COMMAND SHOULD STOP RUNNING.  BOTH VALUES
           TOGETHER (NS&XS) CAN'T GO BEYOND MIDNIGHT.  TO RUN
           A COMMAND BETWEEN 8AM AND 4PM EVERY HOUR, ONCE
           PER WEEK CODE:
                Ø8:ØØ:ØØ Ø7.H1*9  OR  Ø8:ØØ:ØØ Ø7.H1*16:ØØ
           ".YY/MM/DD" REPRESENTS THE STOP OR END DATE FOR
           THE QUEUE LOGIC.  TO MAKE THE ABOVE TEST CASE
           END ON APRIL 9TH 199Ø FOLLOW THE EXAMPLE SHOWN
           BELOW:
                Ø8:ØØ:ØØ Ø7.H1*16:ØØ.9Ø/Ø4/Ø9
           NOTE:  CALCULATED STOP TIMES ARE RESET AFTER
           EACH EXECUTION OF THE LOOP OPTION (IE *XXX), UNLIKE
           EXPLICIT STOP TIMES (IE HH:MM).
RUNDATE - DATE COMMAND WAS LAST RUN.  A Ø PLACE HOLDER IS REQUIRED.
RUNTIME - TIME COMMAND WAS LAST RUN.  A Ø PLACE HOLDER IS REQUIRED.
RUNCODE - RETCODE OF COMMAND LAST RUN.  A Ø PLACE HOLDER IS NEEDED.
           THIS VALUE CAN BE LOOKED AT WITH "IF" CODE BECAUSE
           AFTER EVERY RUN RETURN CODE IS PUT INTO ITS QUETAG VALUE.
      QUETEXT - TEXT OF COMMAND TO RUN.  ALL COMMANDS ARE ALLOWED.
          OPTIONALLY, "IF" STATEMENTS CAN EXECUTE
          QUEUED COMMANDS BY TESTING THE RETURN CODES OF PRIOR ONES.
           AN EXAMPLE IS SHOWN BELOW TO SEND MSG TO OPERATOR.
           Ø IF CMD1 ¬= Ø THEN "CP MSG OP CMD1 AUTOLOG FAILED!"
           IN CONDITIONALS USE "CLKQ" TO REUSE THIS LOGIC
           WITH A UNIQUE OR SPECIALIZED QUEUE COMMAND SET.
           IF THE WANTED COMMAND WILL NOT FIT ON THE QUEUE
           RECORD THE QUETEXT INFO MAY BE CODED ONTO THE
           NEXT LINE BY PUTTING A COMMA AFTER THE RUNCODE.
           NOTE, THE QUEUE PARAMETERS MUST ALL BE ON THE
           SAME LINE, AND IF CONTINUATIONS ARE DONE NO
           QUETEXT FIELDS CAN BE ON THE QUEUE INPUT LINE.
           SUBSEQUENT CONTINUATIONS OF THE QUETEXT
           FIELD MUST FOLLOW THE RULES OF REXX, AND ALL
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
                    CONTINUATION LINES, EXCEPT THE LAST, MUST END
                    WITH A COMMA(,) OR SEMI-COLON(;).
                    EX... IF CMDZ = 16 THEN "CLKQ EOYCYCL" ELSE NOP
CTLEXAMP: BELOW ARE SAMPLE CLKQUEUE RECORDS.
 CMD1 88/01/22 14:30:00 01 00/00/00 00:00:00 0 DIRLOG RSCS
 CMD2 88/01/22 23:59:00 01.M10 00/00/00 00:00:00 0 CP QUERY RSCS
 88/01/22 23:59:00 01.M10 00/00/00 00:00:00 0 IF CMD2=45 THEN MSG OP
RSCS DOWN!
 CMD4 88/01/22 23:59:00 01.M10 0 0 0 IF CMD2 = 45 THEN DIRLOG RSCS
 CMDX: 88/01/22 23:59:00 01.M10 00/00/00 00:00:00 0,
       IF CMD1 ¬= 0 &,
          CMD2 ¬= 0 THEN DO;
           "MSG OP ************************";
           "MSG OP UNABLE TO RECOVER...";
           END
 EOF
CMDEXAMP: BELOW IS A REQUEST FOR CLKQUEUE TO KEEP TRYING TO QUEUE
          COMMANDS IN A FILE NAMED "TASKLIST CLKQUEUE" EVERY FIFTEEN
          MINUTES FOR 24 HOURS STRAIGHT.
          CLKQUEUE   TASKLIST   15   96
BATCHFMT: CLKQUEUE RUNS WELL WHEN SUBMITTED TO JES AS A BATCH JOB.  AN
     EXAMPLE OF THE JCL FOLLOWS, AND THE SYNTAX REMAINS THE SAME.
     REMEMBER, YOU MUST PUT YOUR CLKQUEUE CONTROLS INTO A DATASET CALLED
     'YOURID.CLKQUEUE(CLKQUEUE)', AND SUBMIT IT FROM YOUR MATCHING TSO
     ID, FOR THIS JCL TO WORK.
          //CLOKSTEP EXEC PGM=IKJEFT01,DYNAMNBR=30,REGION=4096K,
          // PARM=('%CLKQUEUE 3 10')
          //+*********************************************************
          //+ FOR DOCUMENTATION ENTER '%CLKQUEUE ?' IN THE PARM FIELD.
          //+*********************************************************
          //SYSEXEC  DD   DSN=MIRVI.REXX,DISP=SHR
          //SYSTSPRT DD   SYSOUT=*
          //SYSTSIN  DD   DUMMY
ENDTYPE*/
/*********************************************************************/
TEXT = ''
DO SEQ = 1,
   UNTIL POS('ENDTYPE*/',TEXT) > 0 | SEQ >= SOURCELINE()
    IF TEXT == ''
       THEN IF POS('/*BEGTYPE',WORD(SOURCELINE(SEQ),1)) = 0
           THEN ITERATE
    TEXT = SOURCELINE(SEQ)
    IF TEXT == '' THEN TEXT = ' ' /* BECAUSE OS2 PASSES BLNKS A NULL */
    IF POS('/*BEGTYPE',TEXT) ¬= 0 |,
       POS('ENDTYPE*/',TEXT) ¬= 0 THEN ITERATE
    SAY TEXT
    END
EXIT 000
SHWERREC:
IF EC = 199 THEN EC = 200
SAY 'RECSEQ='||QUESEQ||',' TAGX||COLON,
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
        QUEDATE QUETIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
QUEDROP = '*ERR'||EC
IF PCESEQ ¬= QUESEQ THEN QUETEXT = ','
CLKQUEUE = QUEDROP TAGX||COLON,
        QUEDATE QUETIME QUEFACT RUNDATE RUNTIME RUNCODE QUETEXT
CALL WRITEREC
RETURN
WRITEREC:
IF CMS THEN 'REXXWRTR' CTLQUE CTLTYP CTLMOD QUESEQ CLKQUEUE
IF TSO THEN 'REXXWRTR' CTLQUE CTLTYP 'A' QUESEQ CLKQUEUE
IF OS2 | DOS THEN DO 1
    'EXECIO * DISKR' CTLQUE'.'CTLTYP '1 ( STEM IO. FINIS'
    IF RC ¬= Ø THEN LEAVE
    IF IO.Ø < QUESEQ
       THEN DO
        X = IO.Ø + 1
        IO.X = CLKQUEUE
        IO.Ø = X   /* NEW NO OF RECS IN FILE */
        X = X + 1
        IO.X = ''  /* EOF */
        END
       ELSE DO
        IO.QUESEQ = CLKQUEUE
        END
    'ERASE' CTLQUE'.'CTLTYP
    IF RC ¬= Ø THEN LEAVE
    'EXECIO' IO.Ø 'DISKW' CTLQUE'.'CTLTYP 'Ø ( STEM IO. FINIS'
    DROP IO.
    END
IF RC ¬= Ø THEN SIGNAL ERRØ4Ø
WRTSEQ = QUESEQ
RETURN
ERRØ1Ø:
SAY 'CLKQUEUE - UNABLE TO FIND THE' CTLQUE CTLTYP 'FILE ON ANY DISK.'
EXIT Ø1Ø
ERRØ2Ø:
SAY "CLKQUEUE - EXECDATE CAN'T CALCULATE DAYS SINCE YEAR 19ØØ USING",
        NXTDATE||"."
EC = Ø2Ø
CALL SHWERREC
SIGNAL READQUE
ERRØ3Ø:
SAY 'CLKQUEUE - EXECDATE FAILED ON NEXT QUEUE DATE USING A FACTOR OF',
        QUEFACT||'.'
EC = Ø3Ø
CALL SHWERREC
SIGNAL READQUE
ERRØ4Ø:
SAY 'CLKQUEUE - REXXWRTR WAS UNABLE REWRITE CLKQUEUE RECORD NUMBER',
        QUESEQ||'.'
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
EXIT Ø4Ø
ERRØ45:
SAY 'CLKQUEUE - INPUT FILE IS NOT ON A READ/WRITE DISK.'
EXIT Ø45
ERRØ5Ø:
SAY 'CLKQUEUE - ENTERED SLEEP TIME IS NOT NUMERIC.  FOUND (' SLPMINS
').'
EC = Ø5Ø
CALL SHWERREC
EXIT Ø5Ø
ERRØ6Ø:
SAY 'CLKQUEUE - ENTERED NUMBER OF SLEEP CYCLES IS NOT NUMERIC.  FOUND
(',
        SLPCYCLS ').'
EC = Ø6Ø
CALL SHWERREC
EXIT Ø6Ø
ERRØ7Ø:
SAY 'CLKQUEUE - ENTERED SLEEP TIME IS GREATER THAN 99. FOUND (',
        ||SLPTIME||').'
EC = Ø7Ø
CALL SHWERREC
ERRØ8Ø:
SAY 'CLKQUEUE - SEE DOC! VALID TIME QUECODES ARE H, M, AND S.  FOUND (',
        QUECODE ').'
EC = Ø8Ø
CALL SHWERREC
SIGNAL READQUE
ERRØ9Ø:
SAY 'CLKQUEUE - QUEUE DAYS VALUE IS NOT NUMERIC.  FOUND (' QUEDAYS ').'
EC = Ø9Ø
CALL SHWERREC
SIGNAL READQUE
ERRØ95:
SAY 'CLKQUEUE - QUEUE SECS|MINS|HRS VALUE IS NOT NUMERIC.  FOUND (',
        QUECYCLE ').'
EC = Ø95
CALL SHWERREC
SIGNAL READQUE
ERR1ØØ:
SAY "CLKQUEUE - QUEUE SECONDS CAN'T BE LESS THAN 1.",
        " FOUND (" QUESECS ').'
SAY "CLKQUEUE - SECONDS WERE CALCULATED USING THE QUEUE CYCLE VALUE OF",
        QUECYCLE||'.'
EC = 1ØØ
CALL SHWERREC
SIGNAL READQUE
ERR11Ø:
SAY 'CLKQUEUE - TIMECALC EXEC FAILED.  EXECUTED: TIMECALC',
        QUESECS NXTTIME
```

 © 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
EC = 11Ø
CALL SHWERREC
SIGNAL READQUE
ERR12Ø:
SAY "CLKQUEUE - EXECDATE CAN'T CALC NEW DATE AFTER CYCLE ROLLOVER OF",
        NXTDYS  'DAY(S).'
SAY "CLKQUEUE - EXECDATE'S PROBLEM INPUT VALUE WAS" NEXTMATH||'.'
EC = 12Ø
CALL SHWERREC
SIGNAL READQUE
ERR13Ø:
SAY "CLKQUEUE - TIMECALC CAN'T CONVERT CURRENT TIME INTO CURRENT
SECONDS."
SAY "CLKQUEUE - TIMECALC'S PROBLEM INPUT VALUE WAS" RUNTIME||'.'
EC = 13Ø
CALL SHWERREC
SIGNAL READQUE
ERR14Ø:
SAY "CLKQUEUE - TIMECALC CAN'T CONVERT NEXT SECONDS",
        "INTEGER INTO NEXT TIME."
SAY "CLKQUEUE - TIMECALC'S PROBLEM INPUT VALUE WAS" NXTSECS||'.'
EC = 14Ø
CALL SHWERREC
SIGNAL READQUE
ERR15Ø:
SAY "CLKQUEUE - QUEUED COMMAND ISN'T VALID CP OR CMS COMMAND."
SAY "CLKQUEUE - THE PROBLEM COMMAND WAS (" SUBWORD(QUETEXT,1) ').'
EC = 15Ø
CALL SHWERREC
SIGNAL READQUE
ERR16Ø:
SAY "CLKQUEUE - SEE DOC. THE QUEUE REPEAT VALUE IS NOT NUMERIC."
SAY "CLKQUEUE - THE PROBLEM VALUE FOUND WAS (" QUELOOP ').'
EC = 16Ø
CALL SHWERREC
SIGNAL READQUE
ERR166:
SAY "CLKQUEUE - SEE DOC. THE QUEUE STOP TIME IS NOT IN 'HH:MM' FORMAT."
SAY "CLKQUEUE - THE PROBLEM VALUE FOUND WAS (" QUELOOP ').'
EC = 166
CALL SHWERREC
SIGNAL READQUE
ERR17Ø:
SAY "CLKQUEUE - TIMECALC CAN'T CONVERT QUEUE REPEAT SECONDS INTO START
TIME."
SAY "CLKQUEUE - TIMECALC'S PROBLEM INPUT VALUE WAS" QUEWK QUESTART'.'
EC = 17Ø
CALL SHWERREC
SIGNAL READQUE
ERR18Ø:
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
SAY 'CLKQUEUE - LOGICAL ERROR WHERE QUEUE START EXCEEDS THE QUEUE LOOP
TIME.'
SAY 'CLKQUEUE - THE INVALID START STOP TIMES WERE (' QUESTART QUESTOP
').'
EC = 18Ø
CALL SHWERREC
SIGNAL READQUE
ERR188:
SAY 'CLKQUEUE - LOGICAL ERROR WHERE QUEUE START EXCEEDS THE QUEUE STOP
TIME.'
SAY 'CLKQUEUE - THE INVALID ( START STOP ) TIMES WERE (',
        QUESTART QUESTOP ').'
EC = 188
CALL SHWERREC
SIGNAL READQUE
ERR19Ø:
SAY 'CLKQUEUE - QUEUE TAG VALUE IS NOT VALID, FOUND ('TAGX||COLON').'
EC = 19Ø
CALL SHWERREC
SIGNAL READQUE
ERR2ØØ:
SAY 'CLKQUEUE - INVALID LEADING CONTINUATION FORMAT. FOUND ('CLKPCE').'
SAY 'CLKQUEUE - ALL/ONLY CONTROL PARAMETERS GO ON FIRST LINE.'
SAY 'CLKQUEUE - QUETEXT COMMAND MUST FOLLOW ON SUBSEQUENT RECORD(S).'
EC = 199   /* THIS IS ONLY A FLAG TO INVOKE THE SHWERREC RTN LATER.*/
RETURN
ERR21Ø:
SAY 'CLKQUEUE - INVALID DATE VALUE ENTERED, FOUND ('QUEDATE').'
EC = 21Ø
CALL SHWERREC
SIGNAL READQUE
ERR22Ø:
SAY 'CLKQUEUE - INVALID TIME VALUE ENTERED, FOUND ('QUETIME').'
EC = 22Ø
CALL SHWERREC
SIGNAL READQUE
ERR23Ø:
SAY 'CLKQUEUE - ERROR CONVERTING NEXT TIME INTO SECONDS.'
EC = 23Ø
CALL SHWERREC
SIGNAL READQUE
ERR24Ø:
SAY 'CLKQUEUE - ERROR CONVERTING QUEUE TIME INTO SECONDS.'
EC = 24Ø
CALL SHWERREC
SIGNAL READQUE
ERR25Ø:
SAY 'CLKQUEUE - ERROR CALCULATING THE NEW NEXT TIME VALUE.'
EC = 25Ø
CALL SHWERREC
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
SIGNAL READQUE
ERR26Ø:
SAY 'CLKQUEUE - USER SET TIME LIMIT OF' TIMELMT 'REACHED AT' TIME()'.'
EXIT 26Ø
```

REXXDATE EXEC

```
/*UPPER/LOWER LGC NEEDED SO KEYS CAN BE USED WITH LOWER CASE LETTERS*/
SYS = ADDRESS()
IF SYS = 'CMS' | SYS = 'XEDIT' THEN CMS = 1; ELSE CMS = Ø
IF SYS = 'DOS' | SYS = 'KEDIT' | SYS = 'CMD' THEN DOS = 1; ELSE DOS = Ø
IF SYS = 'TSO' | SYS = 'MVS' | SYS = 'ISREDIT' THEN TSO = 1; ELSE TSO =
Ø
IF DOS THEN "@ECHO OFF"
PARSE ARG LOWSTRING; ARGSTRING = TRANSLATE(LOWSTRING)
DEBUG = ''; @X = FIND(ARGSTRING,'*DEBUG')
IF @X <> Ø THEN DO; LOWSTRING = (DELWORD(LOWSTRING,@X,1)); TRACE I
    ARGSTRING = TRANSLATE(LOWSTRING)
    DEBUG = '*DEBUG'; END
IF (FIND(ARGSTRING,'?')) = 1 THEN SIGNAL DOC
QUIET = Ø; X = FIND(ARGSTRING,'*QUIET')
IF X <> Ø THEN DO
    ARGSTRING = DELWORD(ARGSTRING,X,1)
    QUIET = 1
    XQUIET = '*QUIET'
    END
XQUIET = ''; XMSGONLY = ''
QUIET = Ø; X = FIND(ARGSTRING,'QUIET')
IF X <> Ø THEN DO
    ARGSTRING = DELWORD(ARGSTRING,X,1)
    QUIET = 1
    XQUIET = '*QUIET'
    END
MSGONLY = Ø; X = FIND(ARGSTRING,'*MSGONLY')
IF X = Ø THEN X = FIND(ARGSTRING,'*TEST')
IF X <> Ø THEN DO
    ARGSTRING = DELWORD(ARGSTRING,X,1)
    MSGONLY = 1
    XMSGONLY = '*MSGONLY'
    END
BEGIN:
/*************************************************************/
MDAYS.1 = 31; MDAYS.2 = 28; MDAYS.3 = 31; MDAYS.4 = 3Ø
MDAYS.5 = 31; MDAYS.6 = 3Ø; MDAYS.7 = 31; MDAYS.8 = 31
MDAYS.9 = 3Ø; MDAYS.1Ø= 31; MDAYS.11= 3Ø; MDAYS.12= 31
/*************************************************************/
DTEXT.1 = 'MONDAY';         DTEXT.2 = 'TUESDAY'
DTEXT.3 = 'WEDNESDAY';      DTEXT.4 = 'THURSDAY'
DTEXT.5 = 'FRIDAY';         DTEXT.6 = 'SATURDAY'
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
DTEXT.7 = 'SUNDAY'
/************************************************************/
MTEXT.1 = 'JANUARY';        MTEXT.2 = 'FEBRUARY'
MTEXT.3 = 'MARCH';          MTEXT.4 = 'APRIL'
MTEXT.5 = 'MAY';            MTEXT.6 = 'JUNE'
MTEXT.7 = 'JULY';           MTEXT.8 = 'AUGUST'
MTEXT.9 = 'SEPTEMBER';      MTEXT.10= 'OCTOBER'
MTEXT.11= 'NOVEMBER';       MTEXT.12= 'DECEMBER'
/************************************************************/
MATHBASE = 693595; DATEBASE = '01/01/1900'
CENTURY_DAYS = (100*365) + 24 /*DAYS IN 20TH CENTURY */
DAYSBEFORE80 = (080*365) + 19 /*DAYS BEFORE 1980 */
IF WORDS(ARGSTRING) > 2  THEN SIGNAL ERR010
PARSE VAR ARGSTRING IDATEX IDATEADJ .
IF IDATEADJ <> '' THEN DO
    IDATEADJ = STRIP(IDATEADJ); IADJTYPE = ''; IMATHADJ = ''
    IF POS(LEFT(IDATEADJ,1),'+-') ¬= 0
       THEN DO
        IADJTYPE = LEFT(IDATEADJ,1)
        IDATEADJ = SUBSTR(IDATEADJ,2)
        END
    IF IADJTYPE = '' THEN IADJTYPE = '+'
    IADJMODE = RIGHT(IDATEADJ,1)
    IF DATATYPE(IADJMODE) ¬= 'NUM'
       THEN DO
        IDATEADJ = LEFT(IDATEADJ,LENGTH(IDATEADJ)-1)
        SELECT
            WHEN IADJMODE = 'D'
               THEN NOP
            WHEN IADJMODE = 'W'
               THEN IDATEADJ = IDATEADJ * 7
            WHEN IADJMODE = 'M'
               THEN IDATEADJ = RIGHT('0'||(IDATEADJ+1),2)'/01/00'
            WHEN IADJMODE = 'Y'
               THEN IDATEADJ = RIGHT('0'IDATEADJ,2)'000'
            OTHERWISE NOP
            END
        END
    IF LENGTH(IDATEADJ) < 4
       THEN DO
        IDATEADJ = RIGHT('0000'IDATEADJ,5)
        END
    'DATEREXX' IDATEADJ XQUIET DEBUG
    IF RC <> 0 THEN SIGNAL ERR090
    PULL . '/' . '/' IYEARADJ IMATHADJ .
    END
```

Editor's note: this article will be concluded next month.

Marc Vincent Irvin
Move Immediate Software (USA)
© M V Irvin 1997

© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

XEDIT ALL macro extensions

These four macros – PLUS, LESS, SWAP, and SALT – are not intended to replace the XEDIT ALL macro, but to extend its flexibility and usefulness. They should also be compatible with other macros that use the SET SCOPE/SET SELECT commands.

PLUS as the name implies adds lines to the display set. LESS removes lines from the display set.

SALT displays the altered lines of a file. Any previous selection from ALL, PLUS, LESS, or other similar macros is cancelled.

SWAP reverses the display and non-display sets.

See help files for details on utilization.

LESS HELP

```
.cs Ø on
[]LESS[%(XEDIT macro)

 Use the LESS macro to exclude from the display a group of lines.

   EXAMPLE: LESS /stop/

      Do not display any lines containing the word "stop".

.cs Ø off
.cs 1 on
[]LESS[%(XEDIT macro)

[]Purpose[%

 Use the LESS macro to exclude from the display a group of lines.
.cs 1 off
.cs 2 on

 >>--LESS--target-----------------------------------------------------><

.cs 2 off
.cs 3 on
[]Operands[%

 target
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

defines which lines to exclude. It must be a valid XEDIT target.

```
.cs 3 off
.cs 4 on
.cs 4 off
.cs 5 on
[]Usage Notes[%
```

1. Unlike the ALL macro, LESS does not set the SCOPE to DISPLAY (the default). This could give unexpected - but predictable - results, if you set the SCOPE to ALL and use LESS to restrict the lines displayed.

2. LESS sets the selection level of all lines affected to zero if this level is not displayed. Otherwise, lines are set to one higher than the highest level being displayed.

3. In conjunction with ALL, LESS can be used to select lines based on the contents of different columns. For example,

   ```
   SET ZONE 2Ø 2Ø
   ALL /A/
   SET ZONE 3Ø 3Ø
   LESS /B/
   ```

   would display all lines that contain the character "A" in column 2Ø, but not "B" in 3Ø.

```
.cs 5 off
.cs 6 on
[]Messages and Return Codes[%

 DMS52ØE      Invalid operand: operand
 DMS545E      Missing operand(s)
 DMS546E      Target not found
 DMS58ØE      Invalid string: shift-out (SO) is not a valid delimiter
 DMS588E      Prefix subcommand waiting...

[]Return Codes[%

 Ø   Normal
 2   Target not found
 5   Missing or invalid operand
 8   Prefix subcommand waiting
.cs 6 off
.cs 7 on
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
 For RELATED information on "selective display", place the cursor under
the selected item and press ENTER or the PF1 key.


 XEDIT LESS   (BRIEF     LESS[%     - Exclude lines from display.
 XEDIT PLUS   (BRIEF     PLUS[%     - Select lines to display.
 XEDIT SALT   (BRIEF     SALT[%     - Show altered lines only.
 XEDIT SWAP   (BRIEF     SWAP[%     - Display excluded lines.

.cs 7 off
```

LESS XEDIT

```
/*********************************************************************/
 /*   Program    - LESS                                               */
 /*   Purpose    - Exclude selected lines from display                */
 /*   Format     - LESS target                                        */
 /*              -                                                    */
 /*              - target - group of lines to exclude from display    */
 /*                                                                   */
 /*********************************************************************/
  Parse arg parms
 Parse source . . . . . name .
 Address command 'MAKEBUF'
 If parms = '' Then Call Error 5 545        /* Target is missing      */
 Push parms
 'MACRO PARSE 1 Target'
 If rc  = 5 Then Call Error 5 580           /* Invalid delimiter      */
 If rc ¬= 0 Then Call Error 5 520 parms     /* Invalid operand        */
 Parse pull lines
 If lines ¬= 1 Then Call Error 5 520 parms  /* Invalid operand        */
 Parse pull start len
 target = substr(parms, start, len)
 rest   = delstr(parms, start, len)
 If rest ¬= '' Then Call Error 5 520 rest   /* Invalid operand        */

 'COMMAND EXTRACT %PENDING *%DISPLAY%'
 If pending.0 ¬= 0 Then Call Error 8 588    /* Pending subcommand     */

 'COMMAND PRESERVE'                    /* Save user's environment     */
 'COMMAND SET POINT .LSS_CURL'         /* Remember current line       */
 'COMMAND SET MSGMODE OFF'
 'COMMAND SET WRAP OFF'
 'COMMAND SET SCOPE DISPLAY'
 'COMMAND TOP'
 'COMMAND LOCATE' target
 If rc ¬= 0 Then                       /* Not found                   */
   Do
     'COMMAND RESTORE'                 /* Restore session settings    */
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
     'COMMAND LOCATE .LSS_CURL'        /* Restore current line        */
     'COMMAND SET POINT .LSS_CURL OFF'
     Call Error 2 546 parms
   End

 level = (display.1 = Ø) * (display.2 + 1)

 Do until rc ¬= Ø                      /* Change selection level      */
   'COMMAND SET SELECT' level
   'COMMAND UP'
   'COMMAND LOCATE' target             /* Locate all lines            */
 End

 'COMMAND SET SCOPE ALL'
 'COMMAND LOCATE .LSS_CURL'            /* Original current line       */
 'COMMAND SET SCOPE DISPLAY'           /* Adjust current line         */
 'COMMAND SET POINT .LSS_CURL OFF'
 'COMMAND RESTORE'                     /* Restore settings            */

 Address command 'DROPBUF'
 Exit

 Error:                                /* Display error messages      */
   Parse arg rcode msg subs, options   /* Uses CMS message repository */
   Address command 'XMITMSG' msg 'subs (CALLER LSS VAR' options
   Do i = 1 to message.Ø
     'COMMAND EMSG' substr(message.i, 4)
   End

   Address command 'DROPBUF'
   'COMMAND CMSG' name parms
 Exit rcode
```

PLUS HELP

```
.cs Ø on
[]PLUS[%(XEDIT macro)

 Use the PLUS macro to redisplay a group of lines previously excluded.

   EXAMPLE: PLUS /stop/

      Add to the display all lines containing the word "stop".

.cs Ø off
.cs 1 on
[]PLUS[%(XEDIT macro)

[]Purpose[%
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
 Use the PLUS macro to redisplay a group of lines previously excluded.
.cs 1 off
.cs 2 on

 >>--PLUS--target-----------------------------------------------------><

.cs 2 off
.cs 3 on
[]Operands[%

 target
     defines which lines to redisplay. It must be a valid XEDIT target.


.cs 3 off
.cs 4 on
.cs 4 off
.cs 5 on
[]Usage Notes[%


   1.   Unlike the ALL macro, PLUS does not set the SCOPE to DISPLAY
        (the default).  This could give unexpected - but predictable -
        results, if you set the SCOPE to ALL and use PLUS to control
        the lines displayed.

   2.   PLUS sets the selection level of all lines affected to the
        lowest level being displayed.

   3.   In conjunction with ALL, PLUS can be used to select lines based
        on the contents of different columns. For example,

             SET ZONE 2Ø 2Ø
             ALL /A/
             SET ZONE 3Ø 3Ø
             PLUS /B/

        would display all lines that contain the character "A" in column
        2Ø, or "B" in 3Ø.

.cs 5 off
.cs 6 on
[]Messages and Return Codes[%

 DMS52ØE      Invalid operand: operand
 DMS545E      Missing operand(s)
 DMS546E      Target not found
 DMS58ØE      Invalid string: shift-out (SO) is not a valid delimiter
 DMS588E      Prefix subcommand waiting...
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
[]Return Codes[%

 Ø   Normal
 2   Target not found
 5   Missing or invalid operand
 8   Prefix subcommand waiting
.cs 6 off
.cs 7 on

 For RELATED information on "selective display", place the cursor under
the selected item and press ENTER or the PF1 key.


 XEDIT LESS   (BRIEF     LESS[%     - Exclude lines from display.
 XEDIT PLUS   (BRIEF     PLUS[%     - Select lines to display.
 XEDIT SALT   (BRIEF     SALT[%     - Show altered lines only.
 XEDIT SWAP   (BRIEF     SWAP[%     - Display excluded lines.

.cs 7 off
```

PLUS XEDIT

```
/*********************************************************************/
 /*   Program    - PLUS                                              */
 /*   Purpose    - Redisplay selected lines                          */
 /*   Format     - PLUS target                                       */
 /*              -                                                   */
 /*              - target - group of lines to redisplay              */
 /*                                                                  */
 /*********************************************************************/

 Parse arg parms
 Parse source . . . . . name .

 Address command 'MAKEBUF'

 If parms = '' Then Call Error 5 545        /* Target is missing     */
 Push parms
 'MACRO PARSE 1 Target'
 If rc  = 5 Then Call Error 5 58Ø           /* Invalid delimiter     */
 If rc ¬= Ø Then Call Error 5 52Ø parms     /* Invalid operand       */

 Parse pull lines
 If lines ¬= 1 Then Call Error 5 52Ø parms  /* Invalid operand       */
 Parse pull start len
 target = substr(parms, start, len)
 rest   = delstr(parms, start, len)
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
If rest ¬= '' Then Call Error 5 52Ø rest   /* Invalid operand        */

'COMMAND EXTRACT %PENDING *%DISPLAY%SELECT%'
If pending.Ø ¬= Ø Then Call Error 8 588    /* Pending subcommand     */

'COMMAND PRESERVE'                     /* Save user's environment     */
'COMMAND SET POINT .PLS_CURL'          /* Remember current line       */
'COMMAND SET MSGMODE OFF'
'COMMAND SET WRAP OFF'
'COMMAND SET SCOPE DISPLAY'

found = Ø                              /* Control selected lines      */
check =                                /* Groups to check             */

If display.1 > Ø Then check = 'Ø' display.1 - 1
If display.2 < select.2 Then check = check display.2 + 1 select.2

Do while check ¬= ''
  Parse var check level_1 level_2 check
  'COMMAND TOP'
  'COMMAND SET DISPLAY' level_1 level_2
  'COMMAND LOCATE' target
  If rc = Ø Then
    Do
      found = 1                        /* We found at least one line  */
      Do until rc ¬= Ø                 /* Add all lines to redisplay  */
        'COMMAND SET SELECT' display.1
        'COMMAND UP'
        'COMMAND LOCATE' target
      End
    End
End

If ¬ found Then                        /* Not found                   */
  Do
    'COMMAND RESTORE'                  /* Restore session settings    */
    'COMMAND LOCATE .PLS_CURL'         /* Restore current line        */
    'COMMAND SET POINT .PLS_CURL OFF'
    Call Error 2 546 parms
  End

'COMMAND SET SCOPE ALL'
'COMMAND LOCATE .PLS_CURL'              /* Original current line       */
'COMMAND SET POINT .PLS_CURL OFF'
'COMMAND RESTORE'                      /* Restore other settings      */
'COMMAND SET DISPLAY' display.1 display.2  /* Reset selection levels */

Address command 'DROPBUF'
Exit
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
 Error:                                  /* Display error messages      */
   Parse arg rcode msg subs, options     /* Uses CMS message repository */

   Address command 'XMITMSG' msg 'subs (CALLER PLS VAR' options
   Do i = 1 to message.Ø
     'COMMAND EMSG' substr(message.i, 4)
   End

   Address command 'DROPBUF'
   'COMMAND CMSG' name parms
 Exit rcode
```

SALT HELP

```
.cs Ø on
[]SALT[%(XEDIT macro)

 Use the SALT macro to show altered lines only.

   EXAMPLE: SALT New

      Display all lines inserted during current session.

.cs Ø off
.cs 1 on
[]SALT[%(XEDIT macro)

[]Purpose[%

 Use the SALT show altered lines only.
.cs 1 off
.cs 2 on

           .-ALL-.
 >>--SALT--+-----+-------------------------------------------------><
           |-NEW-|
           '-OLD-'

.cs 2 off
.cs 3 on
[]Operands[%

 ALL
     displays all lines changed, or inserted, during current editing
session.

 NEW
     displays only new lines.
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
 OLD
     displays only old lines that were changed.


.cs 3 off
.cs 4 on
.cs 4 off
.cs 5 on
[]Usage Notes[%


   1.   Unlike the ALL macro, SALT does not set the SCOPE to DISPLAY
        (the default).  This could give unexpected - but predictable -
        results, if you set the SCOPE to ALL and use SALT to control
        the lines displayed.

   2.   SALT sets the selection level of the lines affected to one, and
        all the others to zero. The DISPLAY level is set to 1 1.


.cs 5 off
.cs 6 on
[]Messages and Return Codes[%

 DMS52ØE      Invalid operand: operand
 DMS588E      Prefix subcommand waiting...

[]Return Codes[%

 Ø   Normal
 5   Missing or invalid operand
 8   Prefix subcommand waiting
.cs 6 off
.cs 7 on

 For RELATED information on "selective display", place the cursor under
the selected item and press ENTER or the PF1 key.


 XEDIT LESS   (BRIEF     LESS[%     - Exclude lines from display.
 XEDIT PLUS   (BRIEF     PLUS[%     - Select lines to display.
 XEDIT SALT   (BRIEF     SALT[%     - Show altered lines only.
 XEDIT SWAP   (BRIEF     SWAP[%     - Display excluded lines.

.cs 7 off
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

SALT XEDIT

```
/*****************************************************************/
 /*   Program    - SALT                                          */
 /*   Purpose    - Show altered lines only                       */
 /*   Format     - SALT                                          */
 /*                                                              */
 /*****************************************************************/

 Parse arg parms
 Parse source . . . . . name .

 opt = translate(parms)
 If opt = '' Then opt = 'ALL'
 If find('ALL NEW OLD', opt) = 0 Then Call Error 5 520 parms

 'COMMAND EXTRACT %PENDING *%SCOPE%'
 If pending.0 ¬= 0 Then Call Error 8 588    /* Pending subcommand     */

 'COMMAND SET SCOPE ALL'               /* Check all records          */
 'COMMAND PRESERVE'                    /* Save user's environment    */
 'COMMAND SET POINT .SLT_CURL'         /* Remember current line      */
 'COMMAND TOP'
 'COMMAND SET SELECT 0 *'              /* Move all lines to level 0  */

 'COMMAND NEXT'
 Do while rc = 0
   'EXTRACT /CURLINE/'
   If curline.4 = 'ON' & (opt = 'ALL' | opt = word(curline.5, 1)) Then
     'COMMAND SET SELECT 1'
   'COMMAND NEXT'
 End

 'COMMAND LOCATE .SLT_CURL'
 'COMMAND SET POINT .SLT_CURL OFF'
 'COMMAND RESTORE'
 'COMMAND SET DISPLAY 1 1'
 'COMMAND SET SCOPE' scope.1
 Exit


 Error:                                /* Display error messages     */
   Parse arg rcode msg subs, options   /* Uses CMS message repository */

   Address command 'XMITMSG' msg 'subs (CALLER SLT VAR' options
   Do i = 1 to message.0
     'COMMAND EMSG' substr(message.i, 4)
   End

   'COMMAND CMSG' name parms
 Exit rcode
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

SWAP HELP

```
.cs Ø on
[]SWAP[%(XEDIT macro)

 Use the SWAP macro to switch between displayed and undisplayed lines.

   EXAMPLE: SWAP

      Display all excluded lines.

.cs Ø off
.cs 1 on
[]SWAP[%(XEDIT macro)

[]Purpose[%

 Use the SWAP macro to switch between displayed and undisplayed lines.
.cs 1 off
.cs 2 on

 >>--SWAP-----------------------------------------------------------><

.cs 2 off
.cs 3 on
.cs 3 off
.cs 4 on
.cs 4 off
.cs 5 on
[]Usage Notes[%


   1.   Unlike the ALL macro, SWAP does not set the SCOPE to DISPLAY
        (the default).  This could give unexpected - but predictable -
        results, if you set the SCOPE to ALL and use SWAP to control
        the lines displayed.

   2.   If there are lines not displayed with selection levels higher
        than the highest display level, and selection level zero is not
        displayed, SWAP will change the selection level of these lines
        to zero.

   3.   SWAP can be used to build a file with all lines displayed. For
        example,

           SWAP
           TOP
           DELETE *
           FILE newname newtype
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
        would create a file with only the displayed lines.

.cs 5 off
.cs 6 on
[]Messages and Return Codes[%

 DMS52ØE      Invalid operand: operand
 DMS588E      Prefix subcommand waiting...

[]Return Codes[%

 Ø   Normal
 5   Missing or invalid operand
 8   Prefix subcommand waiting
.cs 6 off
.cs 7 on

 For RELATED information on "selective display", place the cursor under
the selected item and press ENTER or the PF1 key.


 XEDIT LESS   (BRIEF     LESS[%     - Exclude lines from display.
 XEDIT PLUS   (BRIEF     PLUS[%     - Select lines to display.
 XEDIT SALT   (BRIEF     SALT[%     - Show altered lines only.
 XEDIT SWAP   (BRIEF     SWAP[%     - Display excluded lines.

.cs 7 off
```

SWAP XEDIT

```
/*********************************************************************/
 /*   Program    - SWAP                                              */
 /*   Purpose    - Switch displayed / non displayed lines            */
 /*   Format     - SWAP                                              */
 /*                                                                   */
 /*********************************************************************/

 Parse arg parms
 Parse source . . . . . name .

 If parms ¬= '' Then Call Error 5 52Ø parms /* Invalid operand       */

 'COMMAND EXTRACT %PENDING *%DISPLAY%SELECT%SCOPE%'
 If pending.Ø ¬= Ø Then Call Error 8 588    /* Pending subcommand    */

 If display.1 > Ø & display.2 < select.2 Then
   Do
     'COMMAND PRESERVE'               /* Save user's environment    */
     'COMMAND SET POINT .SWP_CURL'    /* Remember current line      */
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
    'COMMAND SET SCOPE DISPLAY'
    'COMMAND SET DISPLAY' display.2 + 1 select.2
    'COMMAND TOP'
    'COMMAND SET SELECT Ø *'          /* Move all lines to level Ø   */
    'COMMAND SET SCOPE ALL'
    'COMMAND LOCATE .SWP_CURL'
    'COMMAND SET POINT .SWP_CURL OFF'
    'COMMAND RESTORE'
  End

If display.1 > Ø Then
  Do
    'COMMAND SET DISPLAY Ø' display.1 - 1
    'COMMAND UP'                      /* Try stabilize current line  */
  End
Else
  'COMMAND SET DISPLAY' display.2 + 1 max(display.2 + 1, select.2)

'COMMAND SET SCOPE DISPLAY'           /* Adjust current line         */
'COMMAND SET SCOPE' scope.1
Exit

Error:                               /* Display error messages      */
  Parse arg rcode msg subs, options  /* Uses CMS message repository */

  Address command 'XMITMSG' msg 'subs (CALLER SWP VAR' options
  Do i = 1 to message.Ø
    'COMMAND EMSG' substr(message.i, 4)
  End

  'COMMAND CMSG' name parms
Exit rcode
```

Fernando Duarte
Analyst (Canada)

© F Duarte 1997

VM Update is always looking for new authors to share with other professionals their best EXECs, code, etc, and top performance hints and tips. You can send them to Trevor Eddolls at any of the addresses on page 2 or e-mail them to xephon@compuserve.com.

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

CMS back-up/restore – part 4

This month we continue with the code for the CMS back-up/restore system.

```
Get_Select_List:
 Parse Arg Split_File .
 Parse Value BRTAPCHK('LOAD') With Rc Vol1 Hdr1
 If Rc <> Ø Then Do
    Push 'Error reading from tape'
    Signal Abend
    End
 Parse Value Cp('REW 181') With .
 Return
Sort_Select_List:
 Push Translate(Arg(1))
'XEDIT CMSBR SELECTED A (NOMSG PROFILE BRSORT'
 If Rc > 99ØØØ Then Signal Abend
 Return
Read_Select_List:
 Selected. =
 Rc = Diskr('SELECTED','SELECTED.')
 Infoline   = Selected.Ø
 Selected.Ø = Selected.Ø - 1
 Parse Value Selected.Infoline With,
       Tdate Fullback Update_Table Packname,
       Sel_Disk Tot_Disk Sel_Cyls Tot_Cyls I_Inname I_Exname .
 Selected.Infoline =
 Parse Value Unpkdt(Tdate) With Date';'Time .
 Tapedate = Date 'at' Time
 Sel_Disk = C2D(Sel_Disk)
 Tot_Disk = C2D(Tot_Disk)
 Sel_Cyls = C2D(Sel_Cyls)
 Tot_Cyls = C2D(Tot_Cyls)
 If Fullback     = 'NO' Then;
    IText1 = 'Tape contains a limited back-up.'
 Else;
    IText1 = 'Tape contains a full back-up of' Packname'.'
 If Update_Table = 'NO' Then;
    IText2 = 'ØCompare table was NOT updated during the back-up.'
 Else IText2 = '+'
 Return
Split_Select_List:
 Do A=1 To Selected.Ø By 1
    Parse Value Selected.A With User Cuu Volid S_Cyl E_Cyl T_Cyl,
                                DasdType Format Label Volseq .
    Selected.A = ' 'Left(User,1Ø) Right(Cuu,5)'    'Volid,
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
                ' 'Format(S_Cyl,7) Format(E_Cyl,7) Format(T_Cyl,7),
                '  'Center(Format,5)'   'Left(Label,6) Right(Volseq,7)
   End
 Return
Print_Select_List:
 Pline.  =
 Pline.1 = '1*** CMS Back-up/Restore - Start of listing ***',
           '  Date:' Date() '  Time:' Time() '    VM:',
           Word(Cp('QUERY USERID'),4)
 Pline.2 = '-'
 Pline.3 = '-'Itext1 TotalTape
 Pline.4 = Itext2
 Pline.5 = 'ØTape created:' Tapedate
 If Fullback = 'NO' Then Do
    Pline.6 = 'Ø'Sel_Disk 'of' Tot_Disk 'mini-disks were selected.'
    Pline.7 = 'Ø'Sel_cyls 'of' Tot_cyls 'cylinders were backed up.'
    Pline.8 = 'ØInclude filename:' I_Inname
    Pline.9 = 'ØExclude filename:' I_Exname
    Pline.1Ø= 'ØList is sorted by:' Sort_By
    End
 If Fullback = 'YES' Then Do
    Pline.6 = 'Ø'Sel_Cyls 'of' Tot_Cyls 'cylinders were assigned.'
    Pline.7 = 'ØList is sorted by:' Sort_By
    End
 Parse Value Diag(8,'QUERY VIRTUAL ØØE') With,
       . . . Class Cont Hold . Copy . . Form '15'X,
       . For_To For_To_Id . Dest . . . Dist '15'X
 Parse Value Diag(8,'TAG QUERY DEV ØØE') With '15'X Tagtext '15'X
'EXEC BRPRINT START'
 If Rc <> Ø Then Signal No_Print
'EXECIO 1Ø PRINT (CC DATA STEM PLINE.'
 E_O_L   = 'N'
 Page    =  Ø
 Count   =  Ø
 Pline.  =
 Pline1  = '1Userid       Cuu    Volid     Start     End   Total   ',
           'Type   Label   Volseq     Page'
 Do While E_O_L = 'N'
    Page    = Page + 1
    Pline.1 = Pline1 Right(Page,4)
    Do A=3 To 48 By 1
       Count   = Count + 1
       Pline.A = Selected.Count
       If Pline.A = ' ' Then Do
          Pline.A = 'Ø*** CMS Back-up/Restore - End of listing ***'
          E_O_L   = 'Y'
          Leave
          End
       End
   'EXECIO' A 'PRINT (CC DATA STEM PLINE.'
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
    End
No_Print:
 Prt_Rc = Rc
 If Prt_Rc = Ø Then Do
   'EXEC BRPRINT END'
    Prt_Rc = Rc
    If Prt_Rc = Ø Then 'EMSG List has been created.'
    End
 If Prt_Rc <> Ø Then Do
   'CP CLOSE ØØE PURGE'
   'EMSG No list created - Rc='Prt_Rc
    End
'CP SPOOL ØØE' For_To For_To_Id 'CLASS' Class Cont Hold,
              'COPY' Copy 'FORM' Form 'DEST' Dest 'DIST' Dist
'CP TAG DEV ØØE' Tagtext
 Return
/*********************************************************************/
ALine:
 If Option = 'BATCH' Then Return ' '
 Parse Arg Text
 Return '"'Center("Ø-*Ø^"Text,8Ø)'"'
/*********************************************************************/
/** Makeline - Builds a "Line.linenum"                              **/
/** Input:     Linenum Text Varname Varlength Infotext              **/
/** Default for "Varname" is "Input.linenum"                        **/
/*********************************************************************/
Makeline:
 Parse Arg Lnr,Text,Vname,Vl,Info
 If Left(Info,1) <> 'Ø' Then Info.Lnr = 'Ø+'Info
                        Else Info.Lnr = Info
 If Text = ' ' Then Do
    Line.Lnr = "'"Copies(' ',42)"Ø+'Info."Lnr
    Return
    End
 If Vname = ' ' Then Vname = 'Input.'Lnr
 Line.Lnr = "'"Left(Text,3Ø,'.'),
            || ":'Pre."Lnr"''Left("Vname","Vl")'Ø+",
            || Copies(' ',9-Vl) "'Info."Lnr
 Pre.Lnr   = 'Ø('
 Input.Lnr =
 Return
/*********************************************************************/
/** Check the prefixes. If all OK then Accepted = 'YES' else 'NO'.  **/
/*********************************************************************/
Accept_Check:
 Parse Arg CheckLine
 Accepted = 'YES'
 Do X=6 To CheckLine By 2
    If Pre.X <> 'Ø%' Then Iterate
    Cur1 = X
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
    Accepted = 'NO'
    Leave
    End
 Line.21 = Accept.Accepted
 Return
/*********************************************************************/
/** A function has ended abnormally - First line stacked explains why
**/
/*********************************************************************/
Abend:
 Parse Pull Reason
 A_Rc = Rc
'DESBUF'
 If Option = 'BATCH' Then Do
    Call BatchLog Ø '*** ABEND *** ABEND *** ABEND *** ABEND ***'
    Call BatchLog Ø Function 'unsuccessful'
    Call BatchLog Ø Reason '(Rc=' A_Rc')'
    Call Exit A_Rc
    End
 Line. =
 Line.12 = "'Ø\'Copies('+',77)"
 Line.13 = "'Ø\++Ø/'Center(Function 'unsuccessful',71)'Ø\++'"
 Line.14 = "'Ø\++Ø/'Copies('-',71)'Ø\++'"
 Line.15 = "'Ø\++Ø/'Copies(' ',71)'Ø\++'"
 Line.16 = "'Ø\++Ø/'Center(Reason '(Rc= 'A_Rc')',71)'Ø\++'"
 Line.17 = "'Ø\++Ø/'Copies(' ',71)'Ø\++'"
 Line.18 = "'Ø\'Copies('+',77)"
 Signal MainMenu_X
/*********************************************************************/
DDR:
 ARG DDR_ARGS
 Address COMMAND 'EXECIO 4 DISKW INPUT DDR A 1 (FINIS STEM DDR.'
 ADDRESS COMMAND 'BRDDR' DDR_ARGS
 If Rc = Ø Then Return
 Rx = Rc
 Address COMMAND 'DESBUF'
 Rc = Rx
 Queue 'Error during DDR' Word(DDR.4,1)
 Signal Abend
/*********************************************************************
 ** Type       Checks if the input is                                **
 ** --------   ------------------------------------------------------ **
 ** YESNO      "YES" or "NO"                                         **
 ** FUNCTION   "BF" "BL", "RD", "RI", "LT" or "LU"                   **
 ** SORT       "VOLID" or "USERID"                                   **
 ** OUTPUT     "TERMINAL" or "PRINTER"                               **
 ** USERID     a valid userid                                        **
 ** CUU        a valid CUU specification                             **
 *********************************************************************/
Check:
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
Arg Check,PNum,Checktype
If Check = ' ' Then Do
   Help.Pnum = 'Must be specified.'
   Pre.Pnum = 'Ø%'
   Return ' '
   End
Check     = Strip(Check)
Checktype = Strip(Checktype)
Help.Pnum = 'No error in field.'
Pre.Pnum  = 'Ø('
Select
   When CheckType = 'YESNO' Then Do
        If Abbrev('YES',Check,1)       = 1 Then Return 'Yes'
        If Abbrev('NO',Check,1)        = 1 Then Return 'No'
        End
   When CheckType = 'FUNCTION' Then Do
        Check = Space(Check,Ø)
        If Length(Check) = 2 Then Do
           If Check = 'BF' Then Check = 'BACKUPFUL'
           If Check = 'BL' Then Check = 'BACKUPLIM'
           If Check = 'RD' Then Check = 'RESTOREDI'
           If Check = 'RI' Then Check = 'RESTOREIN'
           If Check = 'LT' Then Check = 'LISTTAPE'
           If Check = 'LU' Then Check = 'LISTUSER'
           End
        If Abbrev('BACKUPFUL',Check,7) Then Return,
                                       'Backup YES Ø&Full'
        If Abbrev('BACKUPLIM',Check,7) Then Return,
                                       'Backup NO Ø&Limited'
        If Abbrev('RESTOREDI',Check,8) Then Return,
                                       'Restore DIRECT Ø&Direct'
        If Abbrev('RESTOREIN',Check,8) Then Return,
                                       'Restore INDIRECT Ø&Indirect'
        If Abbrev('LISTTAPE',Check,5)  Then Return 'ListTape'
        If Abbrev('LISTUSER',Check,5)  Then Return 'ListUser'
        End
   When CheckType = 'SORT' Then Do
        If Check = '-' Then Return ' '
        If Check = 'VU' Then Check = 'VOLUSER'
        If Check = 'VS' Then Check = 'VOLSTART'
        If Abbrev('USER',Check,1)      = 1 Then Return 'User'
        If Abbrev('VOLUSER',Check,4)   = 1 Then Return 'VolUser'
        If Abbrev('VOLSTART',Check,4)  = 1 Then Return 'VolStart'
        End
   When CheckType = 'OUTPUT' Then Do
        If Abbrev('TERMINAL',Check,1)  = 1 Then Return 'Terminal'
        If Abbrev('PRINTER',Check,1)   = 1 Then Return 'Printer'
        End
   When CheckType = 'CUU' Then Do
        If Check <> Ø Then Do
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
            If Datatype(Check,'X') = 1 Then Return Right(Check,4,Ø)
            End
         End
    When CheckType = 'VOLUME' Then Do
         If Mdisk.Check <> '?' Then Return Check
         End
    When CheckType = 'USERID' Then Do
        'CP SPOOL CONS TO' Check
         Rx = Rc
        'CP SPOOL CONS TO *'
         If Rx = Ø Then Return Check
         End
    Otherwise Nop
    End
 Pre.Pnum = 'Ø%'
 Help.Pnum = Helptext.Checktype
 Return Check
/*********************************************************************/
/**                I N T E R N A L   R O U T I N E S                **/
/*********************************************************************/

/*********************************************************************/
/** DISKR - Read a file using EXECIO DISKR                          **/
/**                                                                 **/
/** Input: Fn Ft , Stemname , Find , Linenum , "NOFINIS"            **/
/** Linenum default is 1. "*" = no linenum specified                **/
/** "FINIS" is always asumed. Specify "NOFINIS" to override.        **/
/*********************************************************************/
Diskr:
 Arg Fileid,Stem,Find,Linenum,Finis
 Fileid = 'CMSBR' Fileid 'A'
 If Stem <> ' ' Then Stem = 'STEM' Stem
 If Find <> ' ' Then Find = 'FIND /'Find'/'
 If Linenum = ' ' Then Linenum = 1
 If Linenum = '*' Then Linenum =
 If Finis = 'NOFINIS' Then Finis = '('
                      Else Finis = '(FINIS'
 Address COMMAND 'EXECIO * DISKR' Fileid Linenum Finis Stem Find
 Return Rc
/*********************************************************************/
/** CP - Execute a CP command and strip the RC and the reply.       **/
/*********************************************************************/
CP:
 Parse Arg CpCommand
 Parse Value Diagrc(8,CpCommand) With Rc . Reply '15'X
 Return Strip(Rc) Strip(Reply)
/*********************************************************************/
/** PACKDT - Pack date and time                                     **/
/*********************************************************************/
Packdt:
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
 Parse Arg A1 A2
 If A1 = ' ' Then A1 = Date('S')
 If A2 = ' ' Then A2 = Time()
 Parse Value A2 With T1':'T2':'
 Return D2C(Date('S'),4) || D2C(T1) || D2C(T2)
/*********************************************************************/
/** UNPKDT - Unpack date and time                                   **/
/*********************************************************************/
Unpkdt:
 Parse Arg Pdate 5 T1 6 T2
 YMD  = C2D(Substr(Pdate,1,4))
 Date = Substr(YMD,7,2) Word(Months,Substr(YMD,5,2)) Left(YMD,4)
 TT   = Right(C2D(T1),2,Ø)
 MM   = Right(C2D(T2),2,Ø)
 Return Date';'TT':'MM YMD TT || MM
/*********************************************************************/
BatchLog:
 Procedure
 Parse Arg Prefix Text
 If Prefix = '%' Then Prefix = ' '
 TEXT = PREFIX || TEXT
 If Text = ' ' Then Return
'EXECIO 1 DISKW CMSBR BATCHLOG A Ø (FINIS VAR TEXT'
 Return
/*********************************************************************/
PrintLog:
'EXEC BRPRINT START'
 Address COMMAND 'PRINT CMSBR BATCHLOG A (CC'
'EXEC BRPRINT END'
 Return
/*********************************************************************/
/** Init - Initialize general variables and call BRSETUP          **/
/*********************************************************************/
Init:
 Helptext.FUNCTION = 'Invalid function selected.'
 Helptext.OUTPUT   = 'Must be "Terminal" or "Printer".'
 Helptext.SORT     = 'Must be "User", "VolUser" or "VolStart".'
 Helptext.YESNO    = 'Must be "Yes" or "No".'
 Helptext.USERID   = 'Userid not in CP diretory.'
 Helptext.CUU      = 'Invalid CUU specification.'
 Helptext.VOLUME   = 'Volume not available'
 DDR.3             = 'SYSPRINT CONS'
 Months = 'Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec'
 Disktype.C = ' CMS '
 Disktype.D = 'D(OS)'
 Disktype.U = '--?--'
'MACRO BRSETUP'
 If Rc < 24 Then Do
    Operinfo = 'Screen must have 24 lines or more'
    Call Exit Rc
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
    End
 Lscreen   = Rc
 Operinfo  =
 Line.     =
 Input.    =
 Save.4    =
 TotalTape =
 Return
/*********************************************************************/
/**                  E n d   o f   p r o g r a m                    **/
/*********************************************************************/
Syntax:
 Rx = Sigl
 Operinfo = 'Program error;' Errortext(Rc)
 Call Exit Rx
Exit:
 Arg Rx
 If Option = 'BATCH' Then Do
    Do B_Nr=B_Nr By 1 While B_Nr <= Batch.Ø
       If Word(Batch.B_Nr,1) <> 'PRINTLOG' Then Iterate
       Call Printlog
       Leave
       End
    End
 If Option <> 'NOCLEAN' Then Do
   'ERASE CMSBR    LOG      A'
   'ERASE CMSBR    SELECTED A'
   'ERASE CMSBR    SCANLIST A'
   'ERASE INPUT    DDR      A'
    End
 Parse Value BRTDISK('DETACH') With .
'DROPBUF'
 If Operinfo <> ' ' Then Queue Operinfo
'COMMAND QUIT' Rx
 Exit Rx
```

BRSETUP XEDIT

```
/*********************************************************************
 ** BRSETUP - CMS Back-up/Restore XEDIT SETUP                       **
 *********************************************************************/
'SET TOFEOF   OFF'
'SET CMDLINE  OFF'
'SET PREFIX   OFF'
'SET SCALE    OFF'
'SET LINEND   OFF'
'SET FULLREAD ON'
'SET REMOTE   ON'
'SET MSGLINE  ON -2 2 OVERLAY'
'SET MSGMODE ON'
'SET COLOR MSGLINE RED REV HIGH'
```

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

```
'SET CTLCHAR Ø   ESCAPE'                 /*****************************/
'SET CTLCHAR ( NOPROTECT WHI NON NOHIGH' /* Input field NOT in error  */
'SET CTLCHAR )   PROTECT WHI NON NOHIGH' /* Locked input field        */
'SET CTLCHAR % NOPROTECT RED REV   HIGH' /* Input field in error      */
'SET CTLCHAR +   PROTECT TUR NON NOHIGH' /* Normal info              */
'SET CTLCHAR &   PROTECT WHI NON   HIGH' /* Highlighted parts of info */
'SET CTLCHAR #   PROTECT BLU NON NOHIGH' /* Different use            */
'SET CTLCHAR ^   PROTECT YEL NON   HIGH' /* Different use            */
'SET CTLCHAR :   PROTECT YEL UND   HIGH' /* Headers                  */
'SET CTLCHAR -   PROTECT WHI BLI   HIGH' /* Reply/Action wait        */
'SET CTLCHAR /   PROTECT RED NON   HIGH' /* Used at ABEND            */
'SET CTLCHAR \   PROTECT PIN BLI   HIGH' /* Used at ABEND / Warnings  */
                                         /*****************************/
'SET RESERVED  2 GRE REV N' Copies(' ',8Ø)
'SET RESERVED -2 GRE REV N' COPIES(' ',8Ø)

'SET PA1'
'SET PA2'
'SET PA3'

'EXTRACT /LSCREEN/'
 Exit Lscreen.1
```

BRSORT XEDIT

```
/*********************************************************************
 ** BRSORT - Back-up/Restore SORT                                    **
 ** ---------------------------------------------------------------- **
 ** Sort a file according to the fileid and the specified function.  **
 *********************************************************************/
 If Queued() > Ø Then Pull Func .
 Sort.USER     = '1  8  1Ø 13'
 Sort.VOLUSER  = '15 2Ø  1  8 1Ø 13'
 Sort.VOLSTART = '15 2Ø 22 25'
'TOP'
'EXTRACT /FNAME/FTYPE/SIZE/'
 Id = Strip(Fname.1,'T')'-'Strip(Ftype.1,'T')
 ExitRc = Ø
 Select
    When Id = 'CMS-EXEC' Then Do
         If Func = 'NAME' Then;
           'SORT * A 8 25'
         If Func = 'DATE' Then;
           'SORT * D 7Ø 71 67 68 64 65 73 74 76 77 79 8Ø'
         Do Forever
           'DOWN 1'
            If Rc <> Ø Then Leave
           'EXTRACT /CURLINE/'
            Curline.3 = Substr(Curline.3,8,17)
```


© 1997. Xephon UK telephone 01635 33848, fax 01635 38345. USA telephone (940) 455 7050, fax (940) 455 2492.

```
            If Curline.3 = OldCurline Then Do
              'DELETE 1'
              'UP 1'
               END
            OldCurline = Curline.3
            End
         End
    When Id = 'CMSBR-SELECTED' Then Do
         ExitRc = Size.1
        'SORT' (Size.1-1) Sort.Func
         End
    When Id = 'CMSBR-SCANLIST' Then Do
         ExitRc = Size.1
        'SORT * 15 2Ø 22 25'
         End
    When Id = 'CMSBRNEW-COMPTBL' Then Do
        "SET ZONE 1Ø 13"
        ":2"
        "SET STAY ON"
        "CHANGE/A/"'FA'X"/* *"
        "CHANGE/B/"'FB'X"/* *"
        "CHANGE/C/"'FC'X"/* *"
        "CHANGE/D/"'FD'X"/* *"
        "CHANGE/E/"'FE'X"/* *"
        "CHANGE/F/"'FF'X"/* *"
        "SORT * 1 8 1Ø 13"
        "CHANGE/"'FA'X"/A/* *"
        "CHANGE/"'FB'X"/B/* *"
        "CHANGE/"'FC'X"/C/* *"
        "CHANGE/"'FD'X"/D/* *"
        "CHANGE/"'FE'X"/E/* *"
        "CHANGE/"'FF'X"/F/* *"
         End
    Otherwise
         Push 'BRSORT error. Invalid fileid: 'Id
         ExitRc = 99Ø99
    End
If ExitRc <= 99ØØØ Then Do
  'SAVE'
   If Rc <> Ø Then Do
      Exitrc = 99ØØØ + Rc
      Push 'BRSORT error. Unable to SAVE.'
      End
   End
'COMMAND QUIT' ExitRc
 Exit ExitRc
```

Editor’s note: this article will be concluded next month.

Michael Plannthin (Denmark)

© Xephon 1997

© 1997. Reproduction prohibited. Please inform Xephon of any infringement.

Microsoft has acquired Toronto-based LinkAge Software, which makes e-mail connectivity and directory synchronization software. This is relevant to VM users because end users with Microsoft Exchange Server can now use LinkAge technology to access messaging systems such as OfficeVision/VM. They can also access SNADS-based systems such as OfficeVision on the AS/400 and MVS, Fisher TAO, and Verimation Memo. The company's Notes Connector allows Microsoft Exchange and Lotus Notes users to share e-mail, directories, document links, and attachments.

For further information contact:
Microsoft, One Microsoft Way, Redmond, WA 98052-6399, USA.
Tel: (206) 882 8080.
Microsoft, Microsoft Place, Winnersh Triangle, Wokingham, Berks, RG11 5TP, UK.
Tel: (01734) 270001.

* * *

Compute (Bridgend) has announced Release 9.8 of SELCOPY, its year 2000-compliant file manipulation utility. SELCOPY/pc user licences are now automatically incorporated with mainframe SELCOPY licence agreements.

The company has also announced Release 9.8 of CBLVCAT, its VSAM catalog tuning/display utility, which, in addition to being year 2000 compliant, supports VRDS and local time-stamping reporting.

For further information contact:
Compute (Bridgend), 8 Merthyr Mawr Road, Bridgend, Mid-Glamorgan, CF31 3NH, UK.
Tel: (01656) 652222.
Compute (Bridgend), 38 Guided Court, Rexdale, ON, M9V 4K6, Canada.
Tel: (416) 746 4447.

* * *

IBM has announced product number ZP97-0117. This is its OfficeVision to Notes migration tool. It is aimed at VM sites that want to migrate from their 'old-fashioned' messaging system to the best groupware product available.

For further information contact your local IBM representative.

* * *